frontiers | Research Topics

Computational methods for microbiome analysis, volume 2

Edited by
Joao Carlos Setubal and Nikos Kyrpides

Published in
Frontiers in Bioinformatics

FRONTIERS EBOOK COPYRIGHT STATEMENT

The copyright in the text of individual articles in this ebook is the property of their respective authors or their respective institutions or funders. The copyright in graphics and images within each article may be subject to copyright of other parties. In both cases this is subject to a license granted to Frontiers.

The compilation of articles constituting this ebook is the property of Frontiers.

Each article within this ebook, and the ebook itself, are published under the most recent version of the Creative Commons CC-BY licence. The version current at the date of publication of this ebook is CC-BY 4.0. If the CC-BY licence is updated, the licence granted by Frontiers is automatically updated to the new version.

When exercising any right under the CC-BY licence, Frontiers must be attributed as the original publisher of the article or ebook, as applicable.

Authors have the responsibility of ensuring that any graphics or other materials which are the property of others may be included in the CC-BY licence, but this should be checked before relying on the CC-BY licence to reproduce those materials. Any copyright notices relating to those materials must be complied with.

Copyright and source acknowledgement notices may not be removed and must be displayed in any copy, derivative work or partial copy which includes the elements in question.

All copyright, and all rights therein, are protected by national and international copyright laws. The above represents a summary only. For further information please read Frontiers' Conditions for Website Use and Copyright Statement, and the applicable CC-BY licence.

ISSN 1664-8714
ISBN 978-2-83250-640-0
DOI 10.3389/978-2-83250-640-0

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

Computational methods for microbiome analysis, volume 2

Topic editors

Joao Carlos Setubal — University of São Paulo, Brazil

Nikos Kyrpides — Joint Genome Institute, Berkeley Lab (DOE), United States

Citation

Setubal, J. C., Kyrpides, N., eds. (2023). *Computational methods for microbiome analysis, volume 2*. Lausanne: Frontiers Media SA.

doi: 10.3389/978-2-83250-640-0

Table of contents

frontiers
in Bioinformatics

OPINION
published: 06 December 2021
doi: 10.3389/fbinf.2021.808003

Releasing the Kraken

Steven L. Salzberg[1,2,3,4]* *and Derrick E. Wood*[2,3]

[1]Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD, United States, [2]Center for Computational Biology, Johns Hopkins University, Baltimore, MD, United States, [3]Department of Computer Science, Johns Hopkins University, Baltimore, MD, United States, [4]Department of Biostatistics, Johns Hopkins University, Baltimore, MD, United States

Ten years ago, the dramatic rise in the number of microbial genomes led to an inflection point, when the approach of finding short, exact matches in a comprehensive database became just as accurate as older, slower approaches. The new idea led to a method that was hundreds of times times faster than those that came before. Today, exact k-mer matching is a standard technique at the heart of many microbiome analysis tools.

Keywords: metagenomics, sequence alignment, sequencing indexing, phylogenetic classification, k-mer matching, microbiome

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Nalvo F. Almeida,
Federal University of Mato Grosso do Sul, Brazil

****Correspondence:***
Steven L. Salzberg
salzberg@jhu.edu

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *02 November 2021*
Accepted: *18 November 2021*
Published: *06 December 2021*

Citation:
Salzberg SL and Wood DE (2021) Releasing the Kraken. Front. Bioinform. 1:808003. doi: 10.3389/fbinf.2021.808003

INTRODUCTION

The field of microbiome research began in the 2000s, at a time when sequencing technology was rapidly getting less costly, and it first became feasible to sequence an environmental sample containing an unknown mixture of organisms. The earliest studies (Venter et al., 2004; Gill et al., 2006) used Sanger sequencing, where sequence lengths were ~600–800 bp and the cost to sequence a bacterial genome was $50,000 or more. With the advent of Solexa (later Illumina) sequencing technology in 2007, read lengths dropped to just 25 bp, but sequencing costs dropped much faster. Read lengths crept up to 100 bp over the next few years, while costs continued to drop.

In one of the very first microbiome studies to use random shotgun sequencing, published in 2004 (Venter et al., 2004), just under two million reads were generated, averaging 818bp in length. The analysis began by assembling the reads into contigs, and then analyzing only those contigs with sufficient depth of coverage. This yielded 2,226 contigs spanning 30.9Mb, which the authors estimated to represent 1800 different species. The primary tool for identifying species was BLAST (Altschul et al., 1997), which they used to align all bacterial proteins in the NCBI database at the time (~627 thousand proteins) against the 6-frame translations of all contigs. This was relatively slow, but with just 2,226 contigs, it was feasible.

BLAST remains a powerful tool for determining the best match of any sequence to all known genomes. However, it is far too slow for analysis of modern shotgun sequencing (or even 16S sequencing) experiments. Microbiome experiments can easily generate tens of millions of reads, and it is not unusual to generate well over 100M reads in a single experiment. Any computational step that processes all these reads needs to be very fast.

How fast exactly? Well, in order to process 100M reads in 24 h, a program would have to process over 1,150 reads per second. That is far, far faster than BLAST.

MORE GENOMES = A NEW TYPE OF ALGORITHM

By 2009, there were over 500 complete bacterial genomes, with thousands more in progress (Brady and Salzberg, 2009). As the number of genomes grew, new computational methods were developed to assist with their analysis, and in particular with the core task of assigning a taxonomic label to each read. The label might be the name of a species, genus, family, order, class, or even phylum, depending on how much information was in the sequence. These early methods included: CARMA (Krause et al., 2008), which matched reads to known protein domains, a strategy that worked well when those domains were present, but that had very low sensitivity, only 6% in early experiments; Phylopythia (McHardy et al., 2007), a method that used support vector machines based on oligonucleotide frequencies, and worked best on sequences of 3000 bp or longer; MEGAN (Huson et al., 2007), which used BLAST plus a phylogenetic algorithm; and PhymmBL (Brady and Salzberg, 2009), a method that used interpolated Markov models (IMMs) trained on known species. PhymmBL could handle reads as short as 100 bp, unlike earlier methods, but running thousands of IMMs on each read made it relatively slow. None of these methods were truly superior to BLAST, but they included new ways to assign a read to a taxonomic category, ranging from species to phylum.

Once the number of sequenced species grew sufficiently large, though, it became likelier that most reads in a metagenomics sample would be similar to at least one of the previously-sequenced genomes. This is especially true for well-studied environments such as the human gut microbiome, which many sequencing projects have targeted. With complex environmental samples, more of the species in a sample might not have been seen before, but with over 360,000 prokaryotic genomes available today (of which 25,000 are complete and the rest are in various stages of assembly, as described at NCBI https://www.ncbi.nlm.nih.gov/genome/browse#!/prokaryotes/), the likelihood is far greater now, as compared to the 2000s, that at least one previously-sequenced species is very close to something in a sample.

This observation led us to the idea, back in 2012, that we could forego sequence alignment (e.g., BLAST) and instead identify reads by looking for exact matches of short sequences. Exact matching is far faster than alignment, because it requires a simple table lookup. In its optimal implementation, exact matching requires constant time, while alignment time is at least proportional to the length n of the query sequence (and optimal alignment requires $O(n^2)$ time).

For this approach to succeed, we need first to choose a value k for the length of our exact matches. K needs to be large enough that we can safely assume, in almost all cases, that a match of length k is not simply a random match, but rather that the two matching sequences came from the same species, or at least from very closely related species. Thus we can quickly rule out small values such as $k = 6$, because every one of the 4,096 possible 6-mers is likely to be present in most bacterial genomes. At larger values, e.g., $k = 20$, the vast majority of random k-mers will not be present in a given bacterial genome, since there are 4^{20} (just over one trillion) 20-mers, and a typical bacterial genome has just one to five million 20-mers.

Thus if we find a 20-base exact match between a read and a genome, there's a very good chance that the read comes from the same or a similar species. Why not increase the value of k even more, which will make this inference more precise (i.e., avoid false positives)? Clearly, for metagenomic analysis the value of k cannot be longer than a read. When Kraken first appeared it was not unusual to generate 75 bp reads, so 75 is an initial upper bound for k.

There are at least two reasons for reducing the value of the upper bound, though. The first reason is sequencing error: even if the species in a sample exactly matches a known genome, some of the reads will have errors. Illumina technology has a very low error rate, less than 0.5%, so it is reasonable to expect that most 75 bp reads will have one or 0 errors. If the single error is precisely in the middle of the read, then the reads must contain a 37-mer with no errors, suggesting that we might set $k = 37$. The second reason is the simple fact that the species in a microbiome will not be identical to previously-sequenced genomes. We cannot know in advance how similar they will be, but longer values of k will mean that we will fail to recognize some species. Thus we can probably choose a value of k somewhere between 20 and 37, with higher values yielding lower sensitivity but greater precision.

When we developed Kraken, we initially chose k = 31 for technical reasons: first because larger values of k reduce the number of queries to our data structure per sequence; and second because 31 is the largest value of k for which we could fit a k-mer into a 64-bit integer. In subsequent work, $k = 31$ worked well across a very wide range of databases and experiments, and therefore we kept it as the default value, although the user can adjust k when building the Kraken database.

SPEED MATTERS

When using exact matches instead of a full-blown alignment of reads to genomes, we know that we will never exceed the sensitivity of BLAST. Thus the usefulness of Kraken, and the many competitors that have emerged since, is dependent on its speed. Essentially, we need to find out whether or not a k-mer has ever been seen before, and identify where it appeared, as fast as possible. We decided early on that even a single k-mer match would be enough to label a read, but that we'd look at every k-mer in order to maximize sensitivity. Thus for 100 bp reads with $k = 31$, we would do exactly 70 lookups into our database.

Fortuitously, a very fast k-mer counter, called Jellyfish (Marçais and Kingsford, 2011) had recently been developed by our colleagues Guillaume Marçais and Carl Kingsford. Jellyfish counts k-mers in a set of DNA sequences (reads or genomes, of any length) and stores the k-mer counts in a specialized, highly optimized hash array. It can then query this array very rapidly to report, for any k-mer, how often it has occurred.

For metagenomic classification, we do not need to know how often a k-mer has appeared, but only what species it occurs in. Every species has a unique taxonomic identifier, available from NCBI, and taking advantage of this, we modified Jellyfish's output so that for each genome in the database, it would simply store that taxonomy ID next to every k-mer in the genome. The only question was what to do for k-mers that appear in more than one genome. To keep the data structure from growing too enormous, we wanted to store exactly one ID with each k-mer. We solved this problem by using the lowest common ancestor (LCA) of all the genomes in which a k-mer appeared. At the time it is building the database, if Kraken encounters a k-mer that it has seen before, it queries the NCBI taxonomy and finds the identifier of the LCA, which might be at the genus, family, or higher level.

Thus at the conclusion of the database construction step, Kraken has stored a single taxonomic identifier with every distinct k-mer across every genome. The database is stored in a file that is then used for metagenomic classification.

To classify a 100 bp read, Kraken simply walks through it, from position 1 to 70, and looks up all the 31-mers in its database. In most cases, all the k-mers are from the same genome and it can simply output that genome's identifier. If the k-mers yield multiple IDs, then Kraken computes the subtree of all the species that it found, and outputs a taxonomy label corresponding to the path in the tree with the most k-mers. (Our 2014 paper (Wood and Salzberg, 2014) contains more details.)

This strategy, simple as it is, turned out to be very accurate, with precision of >99% (meaning its false positive rate was <1%) and sensitivity of just over 90%. As expected, BLAST was slightly more sensitive, about 1% higher, and had slightly lower precision, less than 1% lower. (These results were on a simulated dataset in the original study; other results varied but the overall findings were consistent.) One benefit of Kraken's algorithm is that as the database of known genomes grows, Kraken's sensitivity has increased over time.

Kraken's big advantage was speed: in the original paper, we showed that it can classify 1.5 million 92 bp reads per minute (rpm) on a single 2.1 GHz CPU, while Megablast (the "fast" version of BLAST) achieved a rate of 7,143 rpm (Wood and Salzberg, 2014). The fast version of Kraken, Kraken-Q, was even faster, running at 3.9 million rpm, making it >500 times faster than Megablast. Other programs were much slower than Megablast. With slightly longer reads (156bp), Kraken clocked in at 892 K rpm, Kraken-Q ran at 2,842 K rpm, while Megablast processed 2,830 rpm. Thus for the longer reads, Kraken was about 315 times faster and Kraken-Q ran over 1,000 times faster than Megablast.

To illustrate the practical consequences of these speed differences, if we classified a relatively small run of 30 million Illumina reads, Kraken would take about 20 min. Megablast, in contrast, would take 70 h. Analyzing the output of a single run of a current-generation Illumina sequencer, which can generate three billion paired-end reads, would take 100 times longer, which would be less than a day and a half for Kraken, but 10 months with Megablast. This illustrates how the dramatic gains in DNA sequencing efficiency have driven the need for far faster computational methods, even when a solution such as BLAST might initially seem adequate.

CONCLUSION

Since we first released Kraken, many other methods have been developed for metagenomics analysis, some of them direct competitors and some that solved related but distinct problems. A recent benchmarking analysis (Ye et al., 2019) compared 20 different metagenomics classifiers on a variety of tasks, and Kraken (along with its successors, KrakenUniq (Breitwieser et al., 2018) and Kraken 2 (Wood et al., 2019)) remains one of the fastest and most accurate methods for identifying reads in a microbiome sample. That study concluded that methods using exact matching of long k-mers, the idea pioneered in Kraken, were among the best scoring methods, and that most of the k-mer based methods performed similarly to one another.

From an algorithmic perspective, classifying metagenomics reads is a straightforward alignment problem that can be solved by aligning each read to every genome known to science. Optimal solutions to this problem have been known for decades (Fickett, 1984), but they require time that is quadratic in the lengths of the sequences, which is far too slow. As a practical matter, very fast methods are required to keep pace with both the volume of sequence data and the number of sequenced genomes, both of which have been growing at an exponential rate for the past 2 decades. The success of Kraken demonstrates that exact matching of a relatively long subsequence delivers the requisite speed, and with a sufficiently large database of genomes, it also delivers similar accuracy as compared to other methods that are far slower.

AUTHOR CONTRIBUTIONS

SS conceived and wrote the manuscript. DW edited and wrote the manuscript.

FUNDING

This work was supported in part by the US. National Institutes of Health under grants R01-HG006677 and R35-GM130151.

REFERENCES

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., et al. (1997). Gapped BLAST and PSI-BLAST: a New Generation of Protein Database Search Programs. *Nucleic Acids Res.* 25 (17), 3389–3402. doi:10.1093/nar/25.17.3389

Brady, A., and Salzberg, S. L. (2009). Phymm and PhymmBL: Metagenomic Phylogenetic Classification with Interpolated Markov Models. *Nat. Methods* 6 (9), 673–676. doi:10.1038/nmeth.1358

Breitwieser, F. P., Baker, D. N., and Salzberg, S. L. (2018). KrakenUniq: Confident and Fast Metagenomics Classification Using Unique K-Mer Counts. *Genome Biol.* 19 (1), 198. doi:10.1186/s13059-018-1568-0

Fickett, J. W. (1984). Fast Optimal Alignment. *Nucleic Acids Res.* 12 (1 Pt 1), 175–179. doi:10.1093/nar/12.1part1.175

Gill, S. R., Pop, M., Deboy, R. T., Eckburg, P. B., Turnbaugh, P. J., Samuel, B. S., et al. (2006). Metagenomic Analysis of the Human Distal Gut Microbiome. *Science* 312 (5778), 1355–1359. doi:10.1126/science.1124234

Huson, D. H., Auch, A. F., Qi, J., and Schuster, S. C. (2007). MEGAN Analysis of Metagenomic Data. *Genome Res.* 17 (3), 377–386. doi:10.1101/gr.5969107

Krause, L., Diaz, N. N., Goesmann, A., Kelley, S., Nattkemper, T. W., Rohwer, F., et al. (2008). Phylogenetic Classification of Short Environmental DNA Fragments. *Nucleic Acids Res.* 36 (7), 2230–2239. doi:10.1093/nar/gkn038

Marçais, G., and Kingsford, C. (2011). A Fast, Lock-free Approach for Efficient Parallel Counting of Occurrences of K-Mers. *Bioinformatics* 27 (6), 764–770. doi:10.1093/bioinformatics/btr011

McHardy, A. C., Martín, H. G., Tsirigos, A., Hugenholtz, P., and Rigoutsos, I. (2007). Accurate Phylogenetic Classification of Variable-Length DNA Fragments. *Nat. Methods* 4 (1), 63–72. doi:10.1038/nmeth976

Venter, J. C., Remington, K., Heidelberg, J. F., Halpern, A. L., Rusch, D., Eisen, J. A., et al. (2004). Environmental Genome Shotgun Sequencing of the Sargasso Sea. *Science* 304 (5667), 66–74. doi:10.1126/science.1093857

Wood, D. E., Lu, J., and Langmead, B. (2019). Improved Metagenomic Analysis with Kraken 2. *Genome Biol.* 20 (1), 257. doi:10.1186/s13059-019-1891-0

Wood, D. E., and Salzberg, S. L. (2014). Kraken: Ultrafast Metagenomic Sequence Classification Using Exact Alignments. *Genome Biol.* 15 (3), R46. doi:10.1186/gb-2014-15-3-r46

Ye, S. H., Siddle, K. J., Park, D. J., and Sabeti, P. C. (2019). Benchmarking Metagenomics Tools for Taxonomic Classification. *Cell* 178 (4), 779–794. doi:10.1016/j.cell.2019.07.010

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2021 Salzberg and Wood. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

REVIEW
published: 17 January 2022
doi: 10.3389/fbinf.2021.826370

Challenges in Bioinformatics Workflows for Processing Microbiome Omics Data at Scale

Bin Hu[1]*, Shane Canon[2], Emiley A. Eloe-Fadrosh[2], Anubhav[3], Michal Babinski[1], Yuri Corilo[3], Karen Davenport[1], William D. Duncan[2], Kjiersten Fagnan[2], Mark Flynn[1], Brian Foster[2], David Hays[2], Marcel Huntemann[2], Elais K. Player Jackson[1], Julia Kelliher[1], Po-E. Li[1], Chien-Chi Lo[1], Douglas Mans[3], Lee Ann McCue[3], Nigel Mouncey[2], Christopher J. Mungall[2], Paul D. Piehowski[3], Samuel O. Purvine[3], Montana Smith[3], Neha Jacob Varghese[2], Donald Winston[4], Yan Xu[1] and Patrick S. G. Chain[1]*

[1]Bioscience Division, Los Alamos National Laboratory, Los Alamos, NM, United States, [2]Lawrence Berkeley National Laboratory, Berkeley, CA, United States, [3]Environmental Molecular Sciences Division, Pacific Northwest National Laboratory, Richland, WA, United States, [4]Polyneme LLC, New York, NY, United States

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Jianguo Xia,
McGill University, Canada
Sergio Gálvez Rojas,
University of Malaga, Spain

***Correspondence:**
Bin Hu
bhu@lanl.gov
Patrick S. G. Chain
pchain@lanl.gov

Specialty section:
This article was submitted to
Genomic Analysis,
a section of the journal
Frontiers in Bioinformatics

Received: 30 November 2021
Accepted: 28 December 2021
Published: 17 January 2022

Citation:
Hu B, Canon S, Eloe-Fadrosh EA, Anubhav, Babinski M, Corilo Y, Davenport K, Duncan WD, Fagnan K, Flynn M, Foster B, Hays D, Huntemann M, Jackson EKP, Kelliher J, Li P-E, Lo C-C, Mans D, McCue LA, Mouncey N, Mungall CJ, Piehowski PD, Purvine SO, Smith M, Varghese NJ, Winston D, Xu Y and Chain PSG (2022) Challenges in Bioinformatics Workflows for Processing Microbiome Omics Data at Scale.
Front. Bioinform. 1:826370.
doi: 10.3389/fbinf.2021.826370

The nascent field of microbiome science is transitioning from a descriptive approach of cataloging taxa and functions present in an environment to applying multi-omics methods to investigate microbiome dynamics and function. A large number of new tools and algorithms have been designed and used for very specific purposes on samples collected by individual investigators or groups. While these developments have been quite instructive, the ability to compare microbiome data generated by many groups of researchers is impeded by the lack of standardized application of bioinformatics methods. Additionally, there are few examples of broad bioinformatics workflows that can process metagenome, metatranscriptome, metaproteome and metabolomic data at scale, and no central hub that allows processing, or provides varied omics data that are findable, accessible, interoperable and reusable (FAIR). Here, we review some of the challenges that exist in analyzing omics data within the microbiome research sphere, and provide context on how the National Microbiome Data Collaborative has adopted a standardized and open access approach to address such challenges.

Keywords: microbiome, microbial ecology, omics, bioinformatics, infrastructure

1 INTRODUCTION

The microbiome is defined as a characteristic microbial community occupying a reasonably well-defined habitat which has distinct physio-chemical properties. It includes both the composition of the community (e.g., microbiota) and a theatre of activity, which can be measured with various forms of omics data (Berg et al., 2020). Microbiome research has greatly increased our understanding of the composition and distribution of microbial communities and has provided us with much insight into microbiome functioning, and clues into how best to perturb communities as potential solutions to improve our health and the health of our environment (Donohue and Cogdell 2006; Light et al., 2018; Lear et al., 2021).

While our increased knowledge of individual microbiomes has benefited from a growing number of individual microbiome investigations, the ability to compare data across projects is hampered by many challenges, due in part to the disparate nature of analysis methods employed to process omics data. The ongoing flux in software development and application of new methods to analyze these data have evolved from tackling low throughput technologies (e.g., microscopy) to increasingly high-throughput data, such as metagenomics (Tringe and Rubin 2005), metatranscriptomics (Carvalhais et al., 2012), metabolomics (Bundy et al., 2008), and metaproteomics (Lagier et al., 2018).

Several large-scale microbiome efforts have focused on generating reference genomic data and other valuable omics data (Human Microbiome Project Consortium 2012; Gilbert et al., 2014; Li et al., 2015; Proctor et al., 2019; Parks et al., 2020), yet the velocity at which microbiome data are generated has outpaced infrastructure resources for collection, processing, and distribution of these data in an effective, uniform, and reproducible manner. Given the magnitude of this challenge, there are limited efforts aimed at closing the analysis gap for metagenomic and community profiling data across diverse environments (Gonzalez et al., 2018; Mitchell et al., 2020; Chen et al., 2021). One such effort developed by the European Bioinformatics Institute, called MGnify, provides standardized taxonomic classification of small subunit ribosomal ribonucleic acid gene amplicon data, while for shotgun metagenomic and metatranscriptomics data, MGnify provides assembly, annotation, and contig binning. Importantly, programmatic access to the data for cross-database complex queries is also available via a RESTful application programming interface (API) (Mitchell et al., 2020), and a free service is available for users to submit raw metagenomics sequence data and associated metadata to the European Nucleotide Archive (ENA) followed by analysis using MGnify pipelines. While this platform does not yet support metabolomics and proteomics data analysis, it provides an intuitive way to enable cross-project sequence-based comparisons.

Comparisons across different microbiome studies are of great interest and would allow us to investigate cross-study patterns in a systematic manner to potentially enable generalizable principles to be uncovered. Further, most microbiome studies are underpowered (Kelly et al., 2015), and thus by combining data from different studies, one may find correlations or other associations that cannot be revealed by individual studies alone. For example, it may enable us to differentiate or find similarities in response to various environmental stressors among different microbiomes in different systems. However, several limitations, most notably the broad spectrum (or lack) of metadata standards that allow researchers to find the data they wish to compare, the heterogeneous nature of omics data generated from different labs, and the various data processing/bioinformatics methods, impede the further utilization of these data beyond the scope for which they were originally intended. For researchers interested in cross-study comparisons, it is thus a herculean effort to identify the relevant microbiome studies, to access both the raw omics data and analyzed results, and to re-analyze them in a standardized fashion with other datasets.

To minimize the effort required to identify reusable microbiome datasets, the National Microbiome Data Collaborative (NMDC) was established in 2019 to support microbiome data exploration and discovery through a collaborative, integrative data science ecosystem (Wood-Charlson et al., 2020; Eloe-Fadrosh et al., 2021; Vangay et al., 2021). The NMDC aims to both provide an interface that allows users to search for microbiome samples and omics data based on sample metadata and omics data results, and also provide exemplary open-source analytic workflows for processing petabyte level (10^{15} bytes) raw multi-omics data in microbiome research and producing FAIR compliant (Wilkinson et al., 2016) interoperable and reusable annotated data products. Compared to a typical microbiome study at gigabyte (10^{9}) scale, the scope of planned data processing in NMDC represents a 10^{6} fold increase.

Bioinformatics workflows have their own set of requirements compared to the more general and increasingly popular data science practices. For example, the coexistence of different file format standards, various upstream sample collection and preparation methods, and often incomplete sample metadata all require workflow developers to have a comprehensive understanding of both the biology underpinning the analyses, as well as the related statistical and computational methods.

In this paper, we provide a perspective and review some challenges faced since the inception of the NMDC and the implementation of solutions to support standardization and cross-study, cross-sample microbiome comparisons. We believe these challenges and the proposed solutions are applicable to any large-scale bioinformatics or scientific data portal development. We focus on challenges in 1) architecture considerations; 2) microbiome workflow selections; 3) Metadata to standardize and manage workflow data products.

2 ARCHITECTURE CONSIDERATIONS

There are two major architecture patterns for data portal design, namely data warehouse (Gardner 1998; Koh and Brusic 2005) and data federation (Haas et al., 2002). Though both patterns support multiple sources to submit data, the major difference is that with data commons all the data storage, analysis, and access are provided through a single location instead of from different participating sites. To avoid duplicating data from its submitters, the NMDC adopts the data federation pattern. The NMDC participating institutions can serve as satellite sites, which can be further categorized by its function as experimental site (where raw experimental data are generated), computing site (where bioinformatics workflows are executed), storage site (where raw and/or workflow output data are stored) or any combination. There is a separate central site that functions as the central registry to maintain a global catalog of metadata and data and to link a set of heterogeneous data sources. The central site implements an application programming interface (API) that allows search of the data and communication with satellite sites. It also hosts the web portal (**Figure 1**). A new institution can join the NMDC data federation by registering as a satellite site and implementing protocols that communicate with the NMDC API. Adopting the data federation pattern allows different sites to maintain their own computing environment setup indepently, e.g., using different job management solutions, such as SLURM

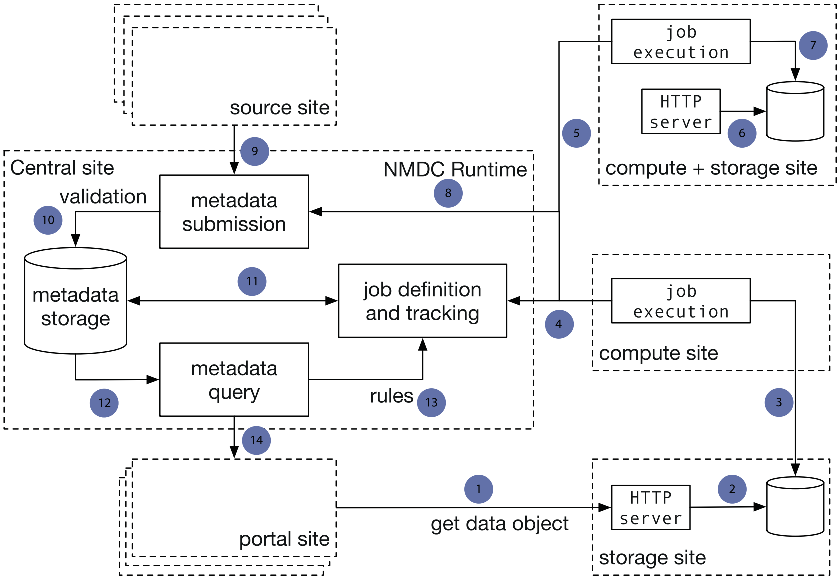


FIGURE 1 | Implementation of a data federation model in the NMDC pilot. The central site implements the NMDC Runtime API that orchestrates the data flow with a database that serves as the data registry. The Runtime validates submitted metadata against the NMDC schema and detects new jobs to be done based on submitted-data annotations. Source sites submit raw experimental data and sample metadata to the central site. Compute sites poll the Runtime for new workflow jobs to be done, claim jobs appropriate for their capabilities, and submit workflow job outputs to the central site. Storage sites store raw workflow outputs. The portal site provides a web-based interface. One site can serve as both a computing site and storage site. Arrows: 1): Portal site gets data object from HTTP server at a storage site; 2): The HTTP server retrieves data from a database; 3) A compute site deposits workflow run result data to a database at a separate storage site; 4) Compute sites claim computing jobs and provide job execution updates to the job tracking mechanism at the Central site; 5, 6, 7): A compute site can also serve as a storage site at the same time; 8) Compute jobs are associated with the sample metadata; 9) A source site submits sample metadata to the Central site; 10) Central site validates submitted sample metadata; 11) New jobs are created from the submitted samples metadata and become claimable by compute sites; 12) Sample metadata can be queried; 13) A set of rules define the type of computing jobs that can be claimed by every Compute site; 14) The Portal site queries metadata.

(Yoo et al., 2003) or Univa Grid Engine (https://www.altair.com/grid-engine/), which also brought us some additional considerations for workflow designs. It also provides the flexibility to bring bioinformatics workflows (gigabytes in size) to experimental and storage sites, instead of moving raw omics data (often terabytes or even larger in size) to a compute site. This model also allows experimental data generation sites to integrate with local data services used for tracking critical metadata and automatically submitting data into the central registry. The current NMDC sites are tightly coupled through the development of the NMDC as the original infrastructure developers, however future NMDC satellite sites can be more loosely coupled as they will not be responsible for maintaining the core infrastructure. Instead, these satellite sites will maintain data processing and exchange services based on their needs to connect with the NMDC project.

3 MICROBIOME OMICS WORKFLOW CONSIDERATIONS

As an increasingly varied array of omics data are being generated for more and more microbiomes, the NMDC team supports standardized workflows for the consistent analysis of metagenomics, metatranscriptomics, metaproteomics, and a suite of metabolomics data. Open-source bioinformatics workflows for processing raw multi-omics data have been developed based on production-quality workflows at the two Department of Energy User Facilities, the Joint Genome Institute (JGI) at Lawrence Berkeley National Laboratory (LBNL) and the Environmental Molecular Sciences Laboratory (EMSL) at Pacific Northwest National Laboratory (PNNL). For any given set of omics data processing or analysis, there exist many tools that typically undergo frequent updates as technologies advance. To accommodate the goals of providing an expanded search capability for NMDC users, the primary goal was to deliver a scalable, open source platform that could provide standardized results independent of the computing platform used, thereby accelerating and enabling future downstream comparative microbiome analytics. It is also worth noting that to help standardize the workflow outputs for cross-study comparisons, we have specified the parameters used in all the NMDC workflows. In other words, all the workflows are static to keep output consistency. Given the various experimental instruments for generating any of these omics data and the associated complexities of instrument-specific biases and error models, we decided to initially focus on the most popular methods applied to microbiome samples developed and maintained by the JGI and EMSL, including Illumina sequencing data, bottom-up proteomics using data-dependent acquisition (Stahl et al., 1996; Kalli et al., 2013), gas chromatography mass spectrometry (GC-MS) based untargeted metabolomics (Hiller et al., 2009;

Fiehn 2016) and Fourier transform ion cyclotron resonance mass spectrometry of complex mixtures (FT-ICR MS) (Kujawinski 2002; Ghaste et al., 2016; Corilo et al., 2021) data, in our initial workflow implementations and software package releases. A Liquid chromatography–mass spectrometry (LC-MS) based workflow is under development and will be available later this year.

3.1 Common Assumptions in Workflows

Many of the challenges in bioinformatics workflows relate to various assumptions made by the workflow software developers. Typically, a bioinformatics workflow tool is developed to solve a data analysis need for a specific experimental design, as well as specific data types and volumes generated for a specific project and to be run within a specific computational infrastructure. Adaptation of specific bioinformatics tools or workflows for a broader project such as that embarked upon by the NMDC requires a more thorough analysis of the workflow requirements and portability needs. The result is a solution that cannot readily be separated from the developers' computing environment with various explicit and implicit assumptions, such as the availability of specific job scheduler and compiler, Linux kernel module and even a specific distribution, instrumentation, file naming conversions, and storage location and formats of the input and output files. We have also investigated the memory usage requirements of various software components, particularly metagenome assemblers, which are known for their high-memory requirements (Kleftogiannis et al., 2013; Li et al., 2015). Another implicit but common assumption is that workflows are for scientists or humans to execute manually on a handful of datasets, instead of being automated for many thousands of datasets, and actively monitored by software, which is linked to workflow scaling.

3.2 Scaling Workflows

Scaling in bioinformatics workflows means the process of dynamically adjusting compute, storage, and network services to meet the data processing demands in an automated fashion in order to maintain availability and performance as utilization increases. Scaling is a common design requirement in cloud applications and has begun to attract attention from the bioinformatics community. Scaling is usually not a requirement for workflows designed to serve small to medium scale studies (with perhaps a few terabytes of raw data) since these workflows can be started manually and queued in a shared job environment. However, in large-scale studies, workflows are being used as a service and must be automatically triggered based on the detection of the availability of new experimental data and additional computing resources may need to be added without interruption to existing workflow executions (Clum et al., 2021). Also, large-scale studies often involve several experimental laboratories and data may be processed at different computing sites, which may run different job schedulers. Thus, it is important that job schedulers have to be separated from the workflow implementation and be configurable for each computing site. For example, based on the raw sequencing data size and complexity, the *de novo* assembly of metagenomic and metatranscriptomic data often requires access to high memory (>1 terabyte) computing nodes. An algorithm is needed to estimate the memory and time needed to process a given sequencing dataset and only allow a data processing site with available big memory nodes to claim such jobs. When cloud resources are used, the appropriate virtual machine instance with sufficient memory and storage must be instantiated. Within the NMDC, a runtime API (https://microbiomedata.github.io/nmdc-runtime/) was implemented that constantly monitors the raw data availability, raw data type (which decides which workflow needs to run), and the computing resources available at each computing site (**Figure 1**). The runtime API is based on the Global Alliance for Genomes and Health GA4GH Data Repository Service (DRS) standard (Rehm et al., 2021). Some other scaling related issues are listed in **Table 1**.

3.3 Selection of Workflows Based on Best Practices

Based on NMDC expertise and general knowledge of the bioinformatics landscape for varied omics data analysis software, no available workflows could accommodate our design needs, e.g., scalability, portability, and reproducibility. While there is no ultimate gold standard workflow for performing environmental microbiome omics analyses, the metagenomics, metatranscriptomics workflows developed at the JGI and the metabolomics and metaproteomics workflow developed at EMSL have been rigorously tested with hundreds and thousands of datasets in the past decade. These workflows, though developed with the assumptions about their local computing environments and not easily portable, do cover a variety of memory and parallelization requirements and follow some of the best practices, and were chosen as the foundation of the NDMC workflows (Piehowski et al., 2013; Li et al., 2017; Clum et al., 2021; Wu et al., 2021). We have introduced several enhancements on top of these foundations. Firstly, to make these workflows fully portable and scalable, we have removed or abstracted all computing environment dependencies by containerizing all the software components. Secondly, we implemented all the workflow logic using the workflow definition language (WDL) (Voss et al., 2017). We also added standardized workflow output file formats in a schema to verify workflow outputs are ready for data ingestion, described further below in the Workflow Metadata section. To help external users adopt these workflows and run them within their own computational environments, we have put all the workflow definitions with test datasets in the NMDC project Github organization (https://github.com/microbiomedata). In addition to this open access software, we further provide detailed documentation (https://nmdc-workflow-documentation.readthedocs.io/en/latest/). Additional training materials are also provided, including video instructions on using the NMDC portal site to examine processed data, and how to run the NMDC workflows in the NMDC EDGE web application (https://nmdc-edge.org), which provides access to all available NMDC omics workflows and is open for public use.

TABLE 1 | Scaling related considerations.

	Small scale studies (gigabytes to <10 terabytes)	Large scale studies and data portals (>10 terabytes)
Workflow management	Rarely used, typically job scheduler	Dedicated workflow manager program
Workflow reproducibility	Limited reproducibility within developers' specific computing environment, lack of long term support	Reproducible independent of the computing environment, better support
Metadata management	Usually at intra-study level and nonstandardized	Community standard based and enforced
Data Management	Spreadsheets	Databases with API access
Data Query	Manual lookup	Database queries and APIs

3.4 Workflow Manager and Workflow Definition

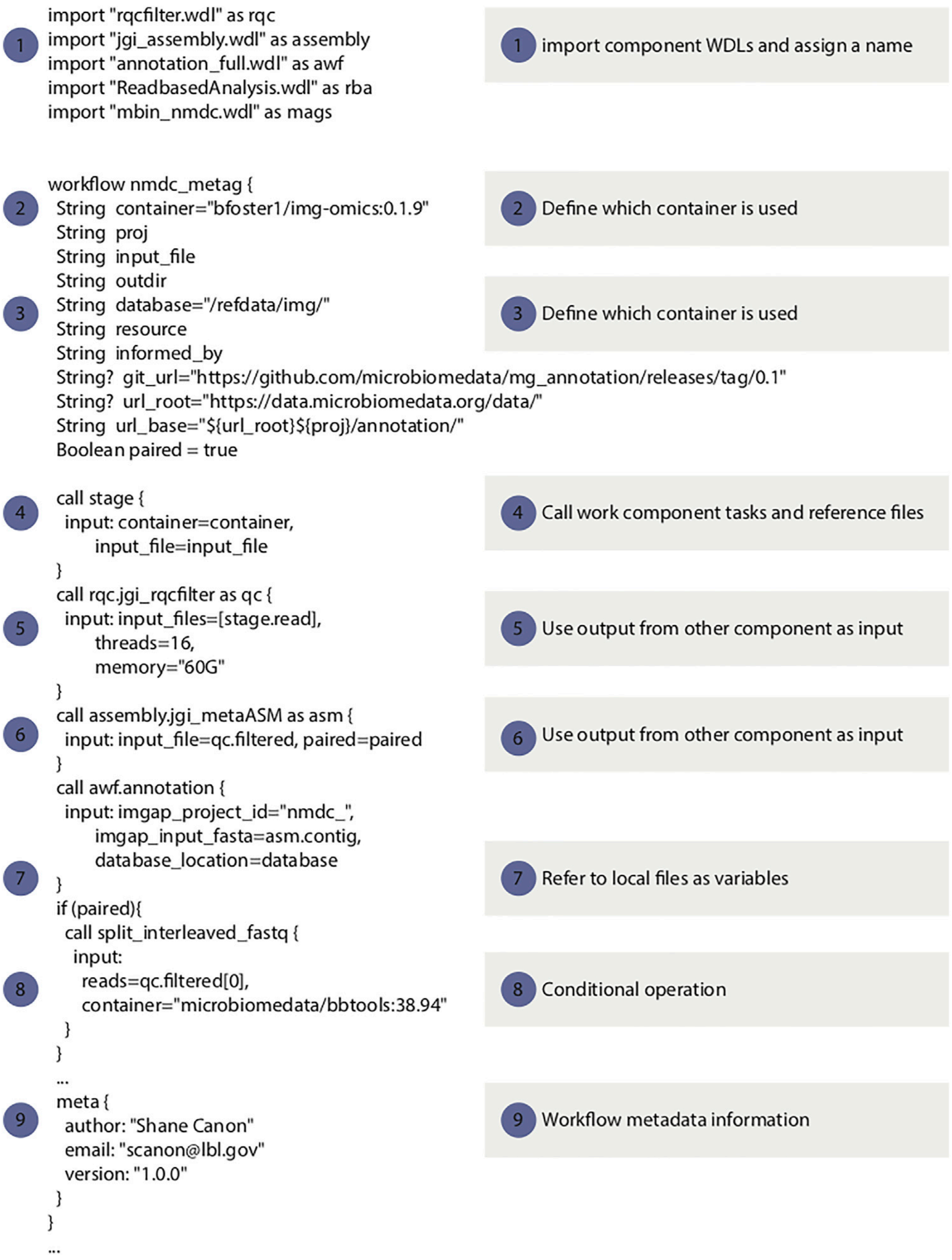
Containerizing workflow components and adopting a workflow definition language alone are not sufficient to separate the concerns of workflow logic and its execution environment. A workflow manager is still required to cleanly separate the concerns of workflow definition and workflow execution. Compared to traditional pipelines utilizing job schedulers or scripting languages, workflow managers excel at reproducibility, data provenance and portability (Di Tommaso et al., 2017; Wratten et al., 2021). Each data processing site only needs to install and configure its own data workflow manager instance based on its resources, such as memory, CPUs, job queues, and storage. With detailed information retrievable from the workflow manager's database, information about workflow execution status is no longer limited to the computing system's job queue itself (e.g., Slurm). This also provides support for resuming failed workflow executions from where the workflow stopped instead of at the beginning of the entire workflow.

For the NMDC, WDL was selected over other workflow languages primarily based on reusable workflow components and superior standardization, which has also been reported by others (Perkel 2019). The Cromwell workflow manager is used in the NMDC due to its native support for WDL (Voss et al., 2017). Cromwell also provides a rich set of features including existing support for a variety of batch systems, native support for containers, "call-caching" to reuse previously executed tasks and an API to facilitate automation. Several key best practices that were adopted by the NMDC for specifying workflows using WDL with component software packaged in containers are listed below. **Figure 2** displays a snippet of WDL code from the NMDC metagenomics workflow, to highlight several key considerations when developing WDL code.

(1) Utilize the WDL "import" function to break down the complexity in large workflows to smaller components. This makes the workflow maintenance easier and increases components reuse.
(2) All workflow tasks should use containers to improve portability, consistency and reproducibility.
(3) All container images should have published recipes (e.g. Dockerfiles). This makes it easier for others to understand how an image was generated and make modifications if needed.
(4) The WDL files should not include any site specific implementation. The Cromwell configuration file should be used to handle site integration. This ensures the WDL is as portable as possible.
(5) Workflows should avoid doing major pre-processing or post-processing outside the WDL. All of the major analysis should be captured in the WDL. This makes the analysis more transparent. For example, generating gene expression information has to be part of the WDL.
(6) Container images should be versioned and the version should be specified in the WDL. This makes the workflow more transparent and ensures that the tasks specified in the WDL are in sync with the image contents. For example, if a new tool version is used that has different command-line options, the WDL and image version can be changed in sync with one another.
(7) Reference data should be versioned and the workflow should specify which version of the reference data is to be used. This avoids potentially format mismatches and helps with reproducibility and transparency.
(8) Workflow WDL have to provide a metadata section that includes the workflow version and author information.

3.5 Workflow Deployment

One of the common challenges in complex bioinformatics workflows is how to best resolve the conflicting software dependencies, and managing the versioning of component software. For NMDC, we addressed this challenge by separating the workflow definition and its runtime by the adoption of workflow managers and WDL as described in the previous section. We also packaged all the runtime requirements for each workflow in Docker containers (Merkel 2014) and made them freely available to non-commercial users (https://hub.docker.com/u/microbiomedata). Some of the runtime components are developed by third parties and have restrictions for commercial users. However, researchers can still use the NMDC workflow WDL definitions by either acquiring appropriate component software licenses (free and open to non-profit organizations and universities). Currently, the NMDC workflows have been deployed to NMDC partner organizations: the National Energy Research Scientific Computing Center (NERSC), the Environmental Molecular



```
import "rqcfilter.wdl" as rqc
import "jgi_assembly.wdl" as assembly
import "annotation_full.wdl" as awf
import "ReadbasedAnalysis.wdl" as rba
import "mbin_nmdc.wdl" as mags


workflow nmdc_metag {
 String  container="bfoster1/img-omics:0.1.9"
 String  proj
 String  input_file
 String  outdir
 String  database="/refdata/img/"
 String  resource
 String  informed_by
 String?  git_url="https://github.com/microbiomedata/mg_annotation/releases/tag/0.1"
 String?  url_root="https://data.microbiomedata.org/data/"
 String  url_base="${url_root}${proj}/annotation/"
 Boolean paired = true

 call stage {
  input: container=container,
      input_file=input_file
 }
 call rqc.jgi_rqcfilter as qc {
  input: input_files=[stage.read],
      threads=16,
      memory="60G"
 }
 call assembly.jgi_metaASM as asm {
  input: input_file=qc.filtered, paired=paired
 }
 call awf.annotation {
  input: imgap_project_id="nmdc_",
      imgap_input_fasta=asm.contig,
      database_location=database
 }
 if (paired){
  call split_interleaved_fastq {
   input:
    reads=qc.filtered[0],
    container="microbiomedata/bbtools:38.94"
  }
 }
 ...
 meta {
  author: "Shane Canon"
  email: "scanon@lbl.gov"
  version: "1.0.0"
 }
}
...
```

1
2
3
4
5
6
7
8
9
1 import component WDLs and assign a name
2 Define which container is used
3 Define which container is used
4 Call work component tasks and reference files
5 Use output from other component as input
6 Use output from other component as input
7 Refer to local files as variables
8 Conditional operation
9 Workflow metadata information

FIGURE 2 | Code snippets of the metagenomic data workflow to illustrate the WDL best practices listed in this paper 1: example use of the "import" function (best practice point 1); 2–4: examples of using containers in WDL (best practice points 2-4 and 6); 5–8: examples of avoid site specific implementation (best practice point 4); 9: workflow metadata information (best practice point 8). The full workflow code is available from https://github.com/microbiomedata/metaG.

Sciences Laboratory (EMSL), the San Diego Super Computing Center (SDSC), and the Los Alamos National Laboratory (LANL). Adding another data processing site or running it in a local computing environment only requires installing and configuring Cromwell and Docker. In high performance computing (HPC) environments where elevated user privilege is a concern, the NMDC workflow containers have also been adapted to other software container solutions that are HPC-friendly, e.g., Singularity (Kurtzer et al., 2017), Shifter (Jacobsen and Canon 2015), and Charliecloud (Priedhorsky and Randles 2017).

Through our experience in packaging and testing of these workflows in different environments, we have learned a few lessons. One is user privilege management. For example, the Cromwell user account needs to have access to all the virtual storage volumes used by the workflow runtime containers. Also, when packaging tools into WDL, workflow developers should really avoid writing to the "/tmp" directory in containers since default settings of writing to "/tmp" is prohibited in Singularity and Charliecloud containers, while being allowed in Docker and Shifter. For testing purposes, we also suggest a minimum of including two testing data sets for each workflow. One smaller test data set can be used for rapid workflow logic validation. The second data set should be complex enough to test memory usage and allows for benchmarking and estimation of CPU and memory usages. In addition, when building software in the containers, chip specific instructions have to be avoided in order to maintain portability. This can mean trading off performance for portability. We have evaluated our workflow containers in both physical and virtual environments running on Intel and AMD processors. We have tested these workflows in various HPC facilities (NERSC/DOE, Expanse/San Diego Supercomputing Center, Texas Advanced Computing Center, Los Alamos National Laboratory, Environmental Molecular Sciences Laboratory/DOE). For some workflows that do not require large memories or databases to run, we have also tested on laptop computers. The NMDC continues to evaluate support for non-x86-64 architectures based on support of the underlying tools and the prevalence of these systems within the community. Presently, many of the underlying tools have not been tested or optimized for architectures such as ARM64 or PPC64 so supporting these is not in any near-term plans. Likewise, GPU support in most of the tools is limited or non-existent. We will continue to track any improvements and make adjustments in the NMDC workflow and images as these tools and community access evolves.

3.6 Workflow User Interface and Customization

The intended users include all microbiome researchers, including both bench scientists and bioinformaticians. To assist bench scientists to use the NMDC workflow, we have carefully engineered a web-based user interface (NMDC EDGE) to run the NMDC workflows interactively (https://nmdc-edge.org). Since we aim to provide a catalogue of the existing microbiome data based on unified analysis processes, we made a design decision to use static workflows with fixed parameters for all the data that feeds into the NMDC portal. Customized workflow runs, including changing the default workflow parameters and even modified the workflow components for internal analysis needs that are not submitted to the NMDC portal will be supported through the KBase (https://www.kbase.us) and future versions of the NMDC EDGE.

4 METADATA

Metadata in the NMDC includes both sample metadata that describes the origin and environmental attributes of the biological sample collection, as well as metadata related to the omics analysis processes employed. The NMDC schema controls which metadata elements are applicable or required for all data within the NMDC, whether it is sample data, or data generated from workflows. The NMDC schema is defined using Linked Data Modeling Language (LinkML, https://linkml.io/linkml/). LinkML is a rich modeling language that is used to create schemas that define the structure of data, allows for rich semantic description of data elements, as well as leveraging JSON-Schema for validation. For example, the relationship between studies, samples, workflows, and data objects is described using LinkML, and the metadata dictionary for samples is described using LinkML.

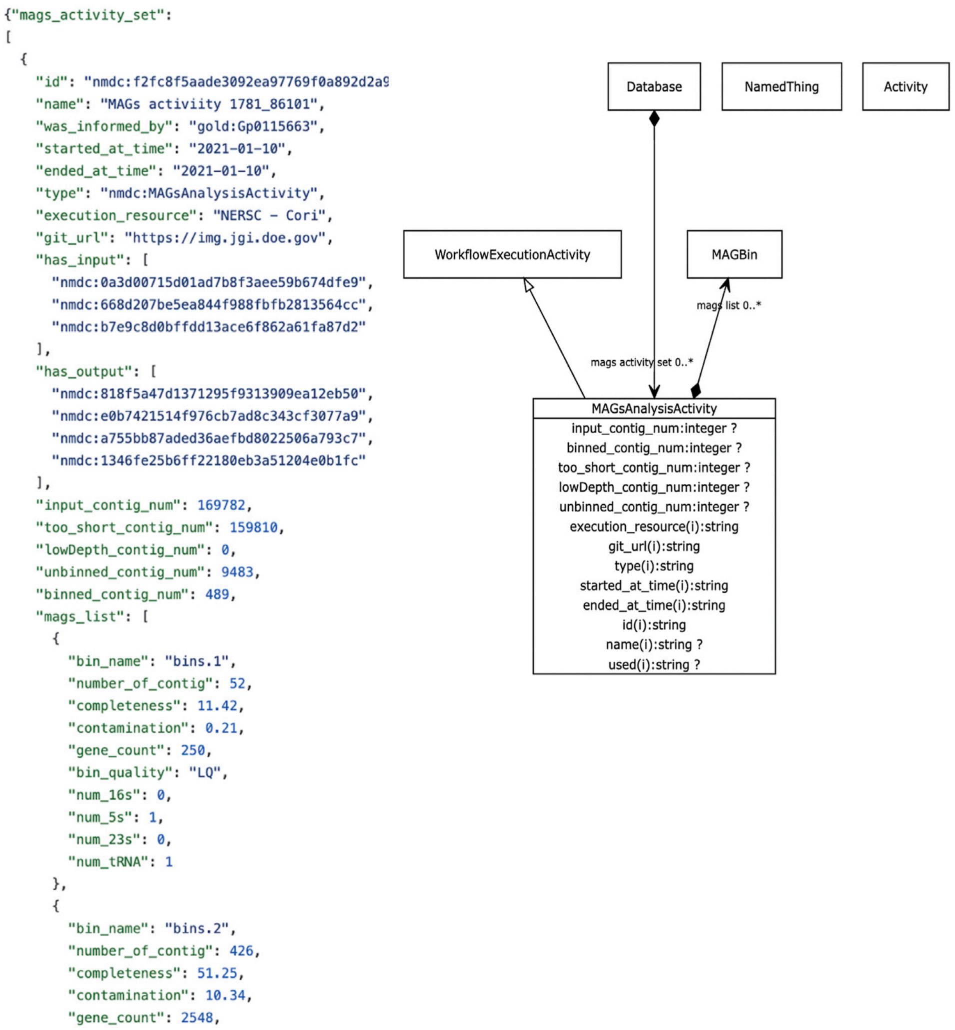
For sample metadata, our schema leverages the Minimal Information about any Sequence (MIxS) data dictionary provided by the (Genomics Standards Consortium (GSC, https://gensc.org/mixs/) (Yilmaz et al., 2011), as well as environmental descriptors taken from the Genomes Online Database (GOLD) (Mukherjee et al., 2021), and OBO Foundry's Environmental Ontology (EnvO) (Buttigieg et al., 2013).

The metadata we focus on in this review revolves around descriptive metadata on the procedures used to generate and process the data. The NMDC leverages the PROV ontology standard (https://www.w3.org/TR/prov-o/), which is a well-established practice in the semantic web community, to provide provenance information. Instances of workflow runs are represented as PROV *activities*. We include distinct schema classes for workflow executions such as Metagenome Assembly, Metabolomics Analysis Activity, Metagenome Annotation Activity, etc. Each of these has generic metadata associated such as time of execution, site of execution, inputs, outputs, etc., in addition to metadata specific to each type of workflow. For example, metabolomics activities have metadata such as calibrations, metabolite quantifications, instruments use. Where possible, these descriptors are mapped to existing standards and vocabularies. An example is provided in **Figure 3**. A Uniform Resource Identifier (URI) and associated workflow activities have been assigned to all the workflow output files that are ingested to the backend database. In the example for **Figure 3**, the URI "nmdc:MAGsAnalysisActivity" is assigned for the outputs of the metagenome binning workflow. This approach lays the foundation for checking workflow output integrity and it also helps to guide the user interface development decisions for the portal website (e.g., what types of searches will be allowed).

Metadata can also be used to steer the workflow execution. For example, nucleotide sequencing data generated from the Illumina platform can be either single-ended (SE) or paired-ended (PE) and PE reads can be stored either in two separate fastq files or interleaved in one fastq file, which makes three types of potential input formats for the NMDC metagenomic and metatranscriptomic workflows. The NMDC workflow supports both SE and PE reads. We plan to automate the detection of the SE/PE reads and leverage the sample preparation and instrumentation metadata to set parameters to the workflow and to trigger appropriate workflow component tasks.

5 SUMMARY AND FUTURE

Here, we have outlined some of the challenges and considerations in implementing disseminable standardized bioinformatics workflows for microbiome omics data, with the goal of



```
{"mags_activity_set":
[
  {
    "id": "nmdc:f2fc8f5aade3092ea97769f0a892d2a9
    "name": "MAGs activiity 1781_86101",
    "was_informed_by": "gold:Gp0115663",
    "started_at_time": "2021-01-10",
    "ended_at_time": "2021-01-10",
    "type": "nmdc:MAGsAnalysisActivity",
    "execution_resource": "NERSC - Cori",
    "git_url": "https://img.jgi.doe.gov",
    "has_input": [
      "nmdc:0a3d00715d01ad7b8f3aee59b674dfe9",
      "nmdc:668d207be5ea844f988fbfb2813564cc",
      "nmdc:b7e9c8d0bffdd13ace6f862a61fa87d2"
    ],
    "has_output": [
      "nmdc:818f5a47d1371295f9313909ea12eb50",
      "nmdc:e0b7421514f976cb7ad8c343cf3077a9",
      "nmdc:a755bb87aded36aefbd8022506a793c7",
      "nmdc:1346fe25b6ff22180eb3a51204e0b1fc"
    ],
    "input_contig_num": 169782,
    "too_short_contig_num": 159810,
    "lowDepth_contig_num": 0,
    "unbinned_contig_num": 9483,
    "binned_contig_num": 489,
    "mags_list": [
      {
        "bin_name": "bins.1",
        "number_of_contig": 52,
        "completeness": 11.42,
        "contamination": 0.21,
        "gene_count": 250,
        "bin_quality": "LQ",
        "num_16s": 0,
        "num_5s": 1,
        "num_23s": 0,
        "num_tRNA": 1
      },
      {
        "bin_name": "bins.2",
        "number_of_contig": 426,
        "completeness": 51.25,
        "contamination": 10.34,
        "gene_count": 2548,
```

Database
NamedThing
Activity
WorkflowExecutionActivity
MAGBin
mags list 0..*
mags activity set 0..*
MAGsAnalysisActivity
input_contig_num:integer ?
binned_contig_num:integer ?
too_short_contig_num:integer ?
lowDepth_contig_num:integer ?
unbinned_contig_num:integer ?
execution_resource(i):string
git_url(i):string
type(i):string
started_at_time(i):string
ended_at_time(i):string
id(i):string
name(i):string ?
used(i):string ?

FIGURE 3 | Example NMDC workflow metadata. Left panel shows an example JSON output snippet of a MAGSAnalysisActivity, which is a record of the metagenomic assembled genome (MAG) workflow execution. It includes generic workflow metadata (start/end time, execution resource) and MAG-specific metadata and workflow outputs. The full JSON example is available on-line (https://github.com/microbiomedata/nmdc-metadata/blob/master/examples/MAGs_activity.json). Right panel shows a visual depiction of the MAGSAnalysisActivity class in the NMDC LinkML schema (https://microbiomedata.github.io/nmdc-schema/MAGsAnalysisActivity/).

providing microbiome analyses that may be cross-compared across projects, regardless of the samples, or the computational environment used to generate the results. The initial efforts from the NMDC have shown how some of these challenges can be addressed by adopting workflow managers, workflow definition language, containerizing workflow runtimes, and developing a data schema for workflow files and their contents. The NMDC has also adopted a data federation model to allow multiple sites (as either data generation, computing, or data storage sites) to contribute to the NMDC while minimizing resource challenges on any single site.

While we document progress towards robust and standardized analyses of various microbiome omics data types, many challenges remain. For example, the data flow in the NMDC is not yet entirely automated which would significantly increase processing capacity. The NMDC is developing a runtime API to fully automate the processing of microbiome data and that supports continuous integration and continuous development. We are also actively developing and supporting a public-facing NMDC API. We have already implemented a set of APIs for satellite sites to register samples and submit workflow outputs in JSON to the portal. Currently, this API is used by the NDMC developers and will be open to external developers. We also plan to provide a set of APIs for programmatic data access, such as query and import data from the NMDC (e.g., the KBase project plans to provide this utility from within the KBase platform). Concomitantly, the optimization of workflow parameters based on sample metadata is also being undertaken, which would then further support automation.

The NMDC data that have been integrated thus far have been generated from JGI and EMSL, both of which serve as experimental data generation sites, compute sites, and storage sites. Separate omics data processing workflows have been developed, and the integration for these data happens through the harmonization of the sample metadata and the functional annotation information. Specifically, the metagenomics, metatranscriptomics, and metaproteomics workflows rely on the same underlying annotations to allow cross-comparisons. At this time, the integration of metabolomics data is only available through the sample metadata information. The workflows and infrastructure envisioned to process future microbiome data, including all microbiome data stored in the short read archive (SRA), are envisioned to be deployed as omics analysis platforms as a service (PaaS) in the cloud for prompt data processing. The NMDC team will coordinate with the user to decide the best strategy to process or deposit a large amount of data.

Lastly, one of the largest and likely ever-present challenges that remains surrounds the topics of sustainability and updating results. These must be considered given the constantly changing landscape of our knowledge of the biological world, and the tools and technology (both instruments and algorithms) used to interrogate microbiomes. Workflow extensibility and database version reliance regularly factor into workflow design considerations. Like almost all bioinformatic workflows, the NMDC workflows rely on several reference databases for genome annotation, metagenome-assembled genome binning, taxonomy classification, and protein and metabolome assignments. As a result, alternative databases, or updates to any of these databases can lead to differences in workflow outputs, which engender important considerations: when to reprocess sample data and how to control the versions of the workflows, databases, and their outputs. An open question is when it is necessary to rerun the analysis on all or a subset of microbiome omics data to update the analysis results. A rerun of the workflow may be triggered by a major update in either the database or the workflow itself. For example, a major change in the NCBI taxonomy database, which is used to identify taxa within metagenomic and metatranscriptomic data, would warrant reprocessing samples affected by these changes. Similarly, newly discovered genomes could enhance both taxonomy and annotation results, or new discoveries in protein structure and function or new metabolites may require reanalysis of metaproteome and metabolomic datasets as well. In addition, new or improvements in algorithms and software tools that significantly outperform existing tools will likely require rerunning relevant workflows.

The NMDC model, in terms of workflow development, implementation, and sharing has been to construct modular workflows from established, best practice tools and pipelines, and to make these open source (https://github.com/microbiomedata). While the NMDC plans to use these workflows to process currently available microbiome data, continued testing and evaluation of new tools, or new versions of existing tools is also underway and part of our internal processes and policies related to workflow management, updates, and adoption of new tools. Specifically, the current NMDC workflows are derived from established workflows from the NMDC participating organizations and will evolve over time. These upstream workflows will routinely undergo modifications in order to improve the quality and performance of the results and products, through the adoption of new tools, updates to the various software packages, updates to the reference databases and taxonomy, etc. The NMDC will synchronize with the upstream workflow changes to improve the NMDC workflows. The open source model also allows third party developers to substitute tools and examine how changes to the workflows may impact the results. As the NMDC team further develops support for curated metadata and production-quality bioinformatic workflows, we welcome contributions from the broader community of researchers to partner with us to make the NMDC a unique collaborative resource for microbiome researchers.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

This work is supported by the Genomic Science Program in the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research (BER) (DE-AC02-05CH11231 to L.B.N.L., 89233218CNA000001 to L.A.N.L. DE-AC05-76RL01830 to P.N.N.L.). Funding for open access charge: Department of Energy.

REFERENCES

Berg, G., Rybakova, D., Fischer, D., Cernava, T., Vergès, M. C., Charles, T., et al. (2020). Microbiome Definition Re-visited: Old Concepts and New Challenges. *Microbiome* 8 (1), 103. doi:10.1186/s40168-020-00875-0

Bundy, J. G., Davey, M. P., and Viant, M. R. (2008). Environmental Metabolomics: A Critical Review and Future Perspectives. *Metabolomics* 5 (1), 3–21. doi:10.1007/s11306-008-0152-0

Buttigieg, P., Morrison, N., Smith, B., Mungall, C. J., and Lewis, S. E. (2013). & the ENVO ConsortiumThe Environment Ontology: Contextualising Biological and Biomedical Entities. *J. Biomed. Sem* 4 (1), 43. doi:10.1186/2041-1480-4-43

Carvalhais, L. C., Dennis, P. G., Tyson, G. W., and Schenk, P. M. (2012). Application of Metatranscriptomics to Soil Environments. *J. Microbiol. Methods* 91 (2), 246–251. doi:10.1016/j.mimet.2012.08.011

Chen, I. A., Chu, K., Palaniappan, K., Ratner, A., Huang, J., Huntemann, M., et al. (2021). The IMG/M Data Management and Analysis System v.6.0: New Tools and Advanced Capabilities. *Nucleic Acids Res.* 49 (D1), D751–D763. doi:10.1093/nar/gkaa939

Clum, A., Huntemann, M., Bushnell, B., Foster, B., Foster, B., Roux, S., et al. (2021). DOE JGI Metagenome Workflow. *MSystems* 6 (3), e00804–20. doi:10.1128/mSystems.00804-20

Corilo, Y. E., Kew, W. R., and McCue, L. A. (2021). *EMSL-Computing/CoreMS: CoreMS 1.0.0. Zenodo*. doi:10.5281/zenodo.4641553

Di Tommaso, P., Chatzou, M., Floden, E. W., Barja, P. P., Palumbo, E., and Notredame, C. (2017). Nextflow Enables Reproducible Computational Workflows. *Nat. Biotechnol.* 35 (4), 316–319. doi:10.1038/nbt.3820

Donohue, T. J., and Cogdell, R. J. (2006). Microorganisms and Clean Energy. *Nat. Rev. Microbiol.* 4 (11), 800. doi:10.1038/nrmicro1534

Eloe-Fadrosh, E. A., Ahmed, F., Babinski, M., Baumes, J., Borkum, M., Bramer, L., et al. (2021). The National Microbiome Data Collaborative Data Portal: An Integrated Multi-Omics Microbiome Data Resource. *Nucleic Acids Res.* 1, gkab990. doi:10.1093/nar/gkab990

Fiehn, O. (2016). Metabolomics by Gas Chromatography-Mass Spectrometry: Combined Targeted and Untargeted Profiling. *Curr. Protoc. Mol. Biol.* 114, 1. doi:10.1002/0471142727.mb3004s114

Gardner, S. R. (1998). Building the Data Warehouse. *Commun. ACM* 41 (9), 52–60. doi:10.1145/285070.285080

Ghaste, M., Mistrik, R., and Shulaev, V. (2016). Applications of Fourier Transform Ion Cyclotron Resonance (FT-ICR) and Orbitrap Based High Resolution Mass Spectrometry in Metabolomics and Lipidomics. *Int. J. Mol. Sci.* 17 (6), 816. doi:10.3390/ijms17060816

Gilbert, J. A., Jansson, J. K., and Knight, R. (2014). The Earth Microbiome Project: Successes and Aspirations. *BMC Biol.* 12 (1), 69. doi:10.1186/s12915-014-0069-1

Gonzalez, A., Navas-Molina, J. A., Kosciolek, T., McDonald, D., Vázquez-Baeza, Y., Ackermann, G., et al. (2018). Qiita: Rapid, Web-Enabled Microbiome Meta-Analysis. *Nat. Methods* 15 (10), 796–798. doi:10.1038/s41592-018-0141-9

Haas, L. M., Lin, E. T., and Roth, M. A. (2002). Data Integration through Database Federation. *IBM Syst. J.* 41 (4), 578–596. doi:10.1147/sj.414.0578

Hiller, K., Hangebrauk, J., Jäger, C., Spura, J., Schreiber, K., and Schomburg, D. (2009). MetaboliteDetector: Comprehensive Analysis Tool for Targeted and Nontargeted GC/MS Based Metabolome Analysis. *Anal. Chem.* 81 (9), 3429–3439. doi:10.1021/ac802689c

Human Microbiome Project Consortium (2012). A Framework for Human Microbiome Research. *Nature* 486 (7402), 215–221. doi:10.1038/nature11209

Jacobsen, D. M., and Canon, R. S. (2015). Contain This, Unleashing Docker for HPC. *Proc. Cray User Group (2015)* 1, 33–49.

Kalli, A., Smith, G. T., Sweredoski, M. J., and Hess, S. (2013). Evaluation and Optimization of Mass Spectrometric Settings during Data-dependent Acquisition Mode: Focus on LTQ-Orbitrap Mass Analyzers. *J. Proteome Res.* 12 (7), 3071–3086. doi:10.1021/pr3011588

Kelly, B. J., Gross, R., Bittinger, K., Sherrill-Mix, S., Lewis, J. D., Collman, R. G., et al. (2015). Power and Sample-Size Estimation for Microbiome Studies Using Pairwise Distances and PERMANOVA. *Bioinformatics* 31 (15), 2461–2468. doi:10.1093/bioinformatics/btv183

Kleftogiannis, D., Kalnis, P., and Bajic, V. B. (2013). Comparing Memory-Efficient Genome Assemblers on Stand-Alone and Cloud Infrastructures. *PLOS ONE* 8 (9), e75505. doi:10.1371/journal.pone.0075505

Koh, J., and Brusic, V. (2005). Database Warehousing in Bioinformatics. *Bioinformatics Tech.* 1, 45–62. doi:10.1007/3-540-26888-X_3

Kujawinski, E. (2002). Electrospray Ionization Fourier Transform Ion Cyclotron Resonance Mass Spectrometry (ESI FT-ICR MS): Characterization of Complex Environmental Mixtures. *Environ. Forensics* 3 (3), 207–216. doi:10.1006/enfo.2002.0109

Kurtzer, G. M., Sochat, V., and Bauer, M. W. (2017). Singularity: Scientific Containers for Mobility of Compute. *PLoS ONE* 12, e0177459. doi:10.1371/journal.pone.0177459

Lagier, J. C., Dubourg, G., Million, M., Cadoret, F., Bilen, M., Fenollar, F., et al. (2018). Culturing the Human Microbiota and Culturomics. *Nat. Rev. Microbiol.* 16 (9), 540–550. doi:10.1038/s41579-018-0041-0

Lear, G., Kingsbury, J. M., Franchini, S., Gambarini, V., Maday, S. D. M., Wallbank, J. A., et al. (2021). Plastics and the Microbiome: Impacts and Solutions. *Environ. Microbiome* 16 (1), 2. doi:10.1186/s40793-020-00371-w

Li, D., Liu, C. M., Luo, R., Sadakane, K., and Lam, T. W. (2015). MEGAHIT: An ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* 31 (10), 1674–1676. doi:10.1093/bioinformatics/btv033

Li, P. E., Lo, C. C., Anderson, J. J., Davenport, K. W., Bishop-Lilly, K. A., Xu, Y., et al. (2017). Enabling the Democratization of the Genomics Revolution with a Fully Integrated Web-Based Bioinformatics Platform. *Nucleic Acids Res.* 45 (1), 67–80. doi:10.1093/nar/gkw1027

Light, S. H., Su, L., Rivera-Lugo, R., Cornejo, J. A., Louie, A., Iavarone, A. T., et al. (2018). A Flavin-Based Extracellular Electron Transfer Mechanism in Diverse Gram-Positive Bacteria. *Nature* 562 (7725), 140–144. doi:10.1038/s41586-018-0498-z

Merkel, D. (2014). Docker: Lightweight Linux Containers for Consistent Development and Deployment. *Linux J.* 2014 (239), 2.

Mitchell, A. L., Almeida, A., Beracochea, M., Boland, M., Burgin, J., Cochrane, G., et al. (2020). MGnify: The Microbiome Analysis Resource in 2020. *Nucleic Acids Res.* 48 (D1), D570–D578. doi:10.1093/nar/gkz1035

Mukherjee, S., Stamatis, D., Bertsch, J., Ovchinnikova, G., Sundaramurthi, J. C., Lee, J., et al. (2021). Genomes OnLine Database (GOLD) v.8: Overview and Updates. *Nucleic Acids Res.* 49 (D1), D723–D733. doi:10.1093/nar/gkaa983

Parks, D. H., Chuvochina, M., Chaumeil, P. A., Rinke, C., Mussig, A. J., and Hugenholtz, P. (2020). A Complete Domain-To-Species Taxonomy for Bacteria and Archaea. *Nat. Biotechnol.* 38 (9), 1079–1086. doi:10.1038/s41587-020-0501-8

Perkel, J. M. (2019). Workflow Systems Turn Raw Data into Scientific Knowledge. *Nature* 573 (7772), 149–150. doi:10.1038/d41586-019-02619-z

Piehowski, P. D., Petyuk, V. A., Sandoval, J. D., Burnum, K. E., Kiebel, G. R., Monroe, M. E., et al. (2013). STEPS: A Grid Search Methodology for Optimized Peptide Identification Filtering of MS/MS Database Search Results. *PROTEOMICS* 13 (5), 766–770. doi:10.1002/pmic.201200096

Priedhorsky, R., and Randles, T. (2017). "Charliecloud: Unprivileged Containers for User-Defined Software Stacks in HPC," in Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, November 2017, 1–10. doi:10.1145/3126908.3126925

Proctor, L. M., Creasy, H. H., Fettweis, J. M., Lloyd-Price, J., Mahurkar, A., Zhou, W., et al. (2019). (iHMP) Research Network ConsortiumThe Integrative Human Microbiome Project. *Nature* 569 (7758), 641–648. doi:10.1038/s41586-019-1238-8

Rehm, H. L., Page, A. J. H., Smith, L., Adams, J. B., Alterovitz, G., Babb, L. J., et al. (2021). GA4GH: International Policies and Standards for Data Sharing across Genomic Research and Healthcare. *Cell Genomics* 1 (2), 100029. doi:10.1016/j.xgen.2021.100029

Stahl, D. C., Swiderek, K. M., Davis, M. T., and Lee, T. D. (1996). Data-controlled Automation of Liquid Chromatography/tandem Mass Spectrometry Analysis of Peptide Mixtures. *J. Am. Soc. Mass. Spectrom.* 7 (6), 532–540. doi:10.1016/1044-0305(96)00057-8

Tringe, S. G., and Rubin, E. M. (2005). Metagenomics: DNA Sequencing of Environmental Samples. *Nat. Rev. Genet.* 6 (11), 805–814. doi:10.1038/nrg1709

Vangay, P., Burgin, J., Johnston, A., Beck, K. L., Berrios, D. C., Blumberg, K., et al. (2021). Microbiome Metadata Standards: Report of the National Microbiome Data Collaborative's Workshop and Follow-On Activities. *MSystems* 6 (1), e01194–20. doi:10.1128/mSystems.01194-20

Voss, K., Gentry, J., and Auwerader, G. V. (2017). *Full-stack Genomics Pipelining with GATK4 + WDL + Cromwell*. London: F1000 Research Ltd. 6. doi:10.7490/f1000research.1114631.1

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J. J., Appleton, G., Axton, M., Baak, A., et al. (2016). The FAIR Guiding Principles for Scientific Data Management and Stewardship. *Scientific Data* 3, 160018. doi:10.1038/sdata.2016.18

Wood-Charlson, E. M., Auberry, D., Blanco, H., Borkum, M. I., Corilo, Y. E., and Davenport, K. W. (2020). The National Microbiome Data Collaborative: Enabling Microbiome Science. Nature Reviews. *Microbiology* 18 (6), 313–314. doi:10.1038/s41579-020-0377-0

Wratten, L., Wilm, A., and Göke, J. (2021). Reproducible, Scalable, and Shareable Analysis Pipelines with Bioinformatics Workflow Managers. *Nat. Methods* 18 (10), 1161–1168. doi:10.1038/s41592-021-01254-9

Wu, R., Davison, M. R., Gao, Y., Nicora, C. D., Mcdermott, J. E., Burnum-Johnson, K. E., et al. (2021). Moisture Modulates Soil Reservoirs of Active DNA and RNA Viruses. *Commun. Biol.* 4 (1), 1–11. doi:10.1038/s42003-021-02514-2

Yilmaz, P., Kottmann, R., Field, D., Knight, R., Cole, J. R., Amaral-Zettler, L., et al. (2011). Minimum Information about a Marker Gene Sequence (MIMARKS) and Minimum Information about Any (X) Sequence (MIxS) Specifications. *Nat. Biotechnol.* 29 (5), 415–420. doi:10.1038/nbt.1823

Yoo, A. B., Jette, M. A., and Grondona, M. (2003). "SLURM: Simple Linux Utility for Resource Management," in *Job Scheduling Strategies for Parallel Processing*. Editors D. Feitelson, L. Rudolph, and U. Schwiegelshohn (Springer), 44–60. doi:10.1007/10968987_3

Conflict of Interest: Author DW is employed by Polyneme LLC.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Hu, Canon, Eloe-Fadrosh, Anubhav, Babinski, Corilo, Davenport, Duncan, Fagnan, Flynn, Foster, Hays, Huntemann, Jackson, Kelliher, Li, Lo, Mans, McCue, Mouncey, Mungall, Piehowski, Purvine, Smith, Varghese, Winston, Xu and Chain. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

TECHNOLOGY AND CODE
published: 27 January 2022
doi: 10.3389/fbinf.2021.826701

RecruitPlotEasy: An Advanced Read Recruitment Plot Tool for Assessing Metagenomic Population Abundance and Genetic Diversity

Kenji Gerhardt[1,2†], *Carlos A. Ruiz-Perez*[1,2†], *Luis M. Rodriguez-R*[3†], *Roth E. Conrad*[4] and *Konstantinos T. Konstantinidis*[1,2,4,5*]

[1]School of Biological Sciences, Atlanta, GA, United States, [2]Center for Bioinformatics and Computational Genomics, Atlanta, GA, United States, [3]Department of Microbiology and Digital Science Center (DiSC), University of Innsbruck, Innsbruck, Austria, [4]Ocean Science & Engineering, Atlanta, GA, United States, [5]School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA, United States

Mapping of short metagenomic (or metatranscriptomic) read data to reference isolate or single-cell genomes or metagenome-assembled genomes (MAGs) to assess microbial population relative abundance and/or structure represents an essential task of many studies across environmental and clinical settings. The filtering for the quality of the read match and assessment of read mapping results are frequently performed without visual aids or with the assistance of visualizations produced through ad-hoc, in-house approaches. Here, we introduce RecruitPlotEasy, a fully automated, user-friendly pipeline for these purposes that integrates statistical approaches to quantify intra-population sequence and gene-content diversity and identify co-occurring relative populations in the sample. Hence, RecruitPlotEasy should also greatly facilitate population genetics studies.

RecruitPlotEasy is implemented in Python and R languages and is freely available open source software under the Artistic License 2.0 from https://github.com/KGerhardt/RecruitPlotEasy.

Keywords: bioinformatics, software, metagenomics, MAG, population diversity

OPEN ACCESS

Edited by:
Nikos Kyrpides,
Joint Genome Institute, United States

Reviewed by:
Neha Jacob Varghese,
Joint Genome Institute, United States
Stephen Nayfach,
Lawrence Berkeley National Laboratory, United States

****Correspondence:***
Konstantinos T. Konstantinidis
kostas@ce.gatech.edu

†These authors have contributed equally to this work

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *01 December 2021*
Accepted: *28 December 2021*
Published: *27 January 2022*

Citation:
Gerhardt K, Ruiz-Perez CA, Rodriguez-R LM, Conrad RE and Konstantinidis KT (2022) RecruitPlotEasy: An Advanced Read Recruitment Plot Tool for Assessing Metagenomic Population Abundance and Genetic Diversity. Front. Bioinform. 1:826701. doi: 10.3389/fbinf.2021.826701

INTRODUCTION

Metagenomics studies of natural microbial populations have recently revealed that bacteria and archaea predominantly form sequence-discrete populations with intra-population genomic sequence relatedness typically ranging from ~95 to 100% genome-aggregate average nucleotide identity (or ANI) depending on the population considered (e.g., younger populations since the last population diversity sweep event show lower levels of intra-population diversity and thus, higher ANI). In contrast, ANI values between distinct populations are typically lower than 90% [**Figure 1** and reviewed in (Caro-Quintero and Konstantinidis, 2012)]. Such sequence-discrete populations were recovered from many different habitats, including marine, freshwater, soils, sediment, human gut and biofilms, and were typically persistent over time and space [e.g., (Konstantinidis and DeLong 2008; Meziti et al., 2019; Olm et al., 2020)]. Therefore, these populations appear to be "species-like" and may constitute important units of microbial communities. This discovery is also important in medical microbiology; for instance, in identifying which population is the causative agent of disease

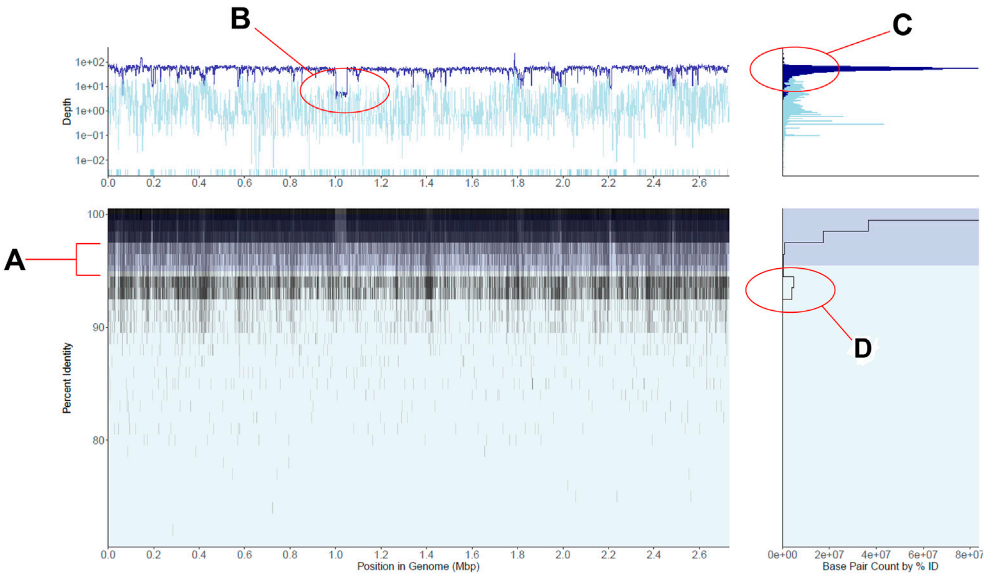


FIGURE 1 | Recruitment plot displaying gut metagenomic reads mapped to a single *Staphylococcus aureus* reference genome. **(A)** is a 2-D histogram displaying the percent identity of reads to the reference genome on the y-axis and the position in the genome on the x-axis. Cell fill color darkens as more reads fall within the cell, i.e., the region of the genome and percent identity window the cell represents, in a logarithmic scale. Shaded in dark blue is a region indicates the plot's current within-population percent nucleotide identity threshold, here shown at the tool's default 95%. **(B)** is a line plot of the average depth of coverage per genome region on the main panel. The dark blue line displays depth of coverage for reads mapping to regions of the genome within the population threshold from panel **(A)**, and the light blue line displays depth for reads outside this population. Note the logarithmic scale in the base pair counts axis as well as the highlighted area of lower coverage, representing a reference genomic region not shared by the majority of the metagenomic population. **(C)** is a histogram of depths of coverage across the entire genome, with colors corresponding to within and outside-population as in panel. **(D)** is a histogram of the number of bases displayed in panel **(A)** (x-axis) which fall into particular percent identity windows (y-axis), here displayed in log scale. Note the second peak in the number of bases mapping in the 94–95% identity range, representing a co-occurring *S. aureus*-like population. See also main text for further details.

(Pena-Gonzalez et al., 2019). More recent work has even shown that intermediate identity genotypes, for example, sharing 85–95% ANI, when present, are ecologically differentiated and thus, should probably be considered distinct species (Conrad et al., 2021; Rodriguez-R et al., 2021), rather than representing cultivation (or other sampling) biases (Murray et al., 2021).

Read recruitment plots are one of the most in-depth analyses to reveal and study sequence-discrete populations. In these plots, the reads of a metagenome are mapped against a genomic reference sequence that is representative of the population to be studied (e.g., an isolate genome or MAG). The mapping patterns that are revealed are informative about how well the metagenomic population matches the reference genome, gene content differences if any, the level of intra-population sequence diversity, and regions of sequence-discontinuity (Rusch et al., 2007; Konstantinidis and DeLong 2008) (**Figure 1**). Thus, read recruitment plots can provide a thorough and quantitative view of the natural population in a sample and its diversity, which represents highly useful information for several downstream analyses. Accordingly, several tools that can plot read mapping patterns have been developed for this purpose e.g., (Robinson et al., 2011; Zhu et al., 2013; Jaenicke et al., 2018). However, these tools typically provide no additional information or capabilities such as they do not include appropriate statistics to characterize the genome, gene allelic, and gene content diversity in spatial or time-series metagenomes and thus, do not allow targeted analyses of specific gene-based traits and exploration of selection pressure and population bottlenecks (Meziti et al., 2019).

Recently, we have developed bioinformatic scripts that can be applied to the read mapping output of a read recruitment plot to provide information based on read mapping that is not available by previous tools such as what is the average coverage of the genome by reads (a proxy for relative metagenome abundance), whether or not co-occurring populations exist in the dataset (sample) (Rodriguez and Konstantinidis, 2016), and which genes of the reference genome in the plot (isolate or MAG) are shared or not by the metagenomic population (Meziti, et al., 2019). Here, we present RecruitPlotEasy, a pipeline that integrates all these scripts into a single tool and represents a completely redesigned tool compared to that originally introduced as part of the enveomics script collection (Rodriguez and Konstantinidis, 2016) in order to scale-up with more data. RecruitPlotEasy also includes new, additional features such as the possibility to simultaneously view plots of multiple reference genomes and/or metagenomic read datasets and is interactive in that the user can browse over the plot to identify genes of interest and view their associated functional annotation (when provided) and relative abundance in the metagenome. Based on previous literature, we employed a (user-adjustable) 95% nucleotide identity threshold to identify reads that represent the same (target) population (a.k.a. *within* population diversity), while reads showing lower than 95% identity are considered to represent distinct, co-occurring populations (a.k.a. *outside* population). Using the RecruitPlotEasy tool requires no previous bioinformatics or coding skills.

INSTALLATION AND INPUTS

RecruitPlotEasy is operated entirely through a graphical user interface (GUI) which manages the selection of inputs, the manipulation of data, and the creation of plots through simple buttons and drop-down menus. Further, all menus and options are annotated with tooltips and reports that help the user easily navigate the workflow of the Recruitment Plot without prior experience using the tool. RecruitPlotEasy is written in a pair of scripts, one in R and one in Python 3. This two-script design takes advantage of multiple visualization libraries and the GUI of the Shiny library available in R, while operating with modest computational resources enabled by Python.

To use the tool, the user needs only to download and install R, Rstudio, and then run a single command from the R terminal. This command installs any missing R dependencies, installs Miniconda if it is absent (to ensure that the right version of Python is subsequently installed), retrieves the R and Python functions, and launches the GUI for the user. While installation only occurs on the first use of RecruitPlotEasy, this same command is used in subsequent sessions to activate the GUI again. RecruitPlotEasy requires a user to supply one (or more) genome file(s) in FASTA format and at least one set of reads mapped to that genome file in either tabular BLAST or SAM/BAM formats. Users may optionally supply gene functional annotation in GFF for gene-level analysis.

METHODS

The graphical interface of RecruitPlotEasy opens in the user's default web browser. This interface is organized into 4 tabs: Database Creation, Database Management, Recruitment Plot, and Interactive Plot (**Supplementary Figure S8**). The tabs organize the workflow of RecruitPlotEasy into smaller, manageable tasks where the options available on each page are directly relevant to the task that page supports. For instance, input selection occurs on the database creation tab, assessment of the contents of a database and the control of advanced options occurs on the database management tab, and the creation of plots occurs on the recruitment plot and interactive plot tabs. The workflow of the recruitment plot is further guided by multiple forms of user feedback. All buttons and input fields are annotated with tooltips that inform the user of the actions each button will perform upon hovering their cursor over it. This includes guidance on the type of file and the kind of data required in each input, and the consequences of changes made to plotting parameters. As inputs are selected, their formats are also checked for basic appropriateness.

The underlying data shown in a recruitment plot is a 2-dimensional matrix (or table) of counts. For a given genome, columns of this matrix correspond to successive regions of the genome and rows correspond to windows of percent nucleotide identity values. The width and height of each cell of the matrix are defined by the user, with defaults of 1000 base pairs for width and 0.5% identity for height. If viewing genes, percent identity windows are determined in the same way, but genome regions instead correspond exactly to the starts and ends of the gene sequences, with intergenic regions forming additional columns, as needed, to fill in the rest of the matrix. The cells of the matrix effectively form a 2-dimensional histogram of bins into which reads may fall.

To fill the matrix, reads aligning to the genome are assigned to their appropriate percent identity window and genome position bin. A user may choose to define percent identity to the reference as either the number of matches divided by the alignment length (local alignment) or matches divided by the entire length of the read, including unaligned sections (global alignment). After the percent identity row is determined, the read will increase the base pair (bp) count of the bin it falls into by its length. Should a read span two or more bins, each bin will receive its respective share of the read's length according to exactly where the read mapped to the genome and the boundaries of the bins. The count for each bin represents the sum of all bases of all reads that map within the corresponding percent identity and region of the genome. Once every read has been processed, the filled matrix is passed to the plotting component of RecruitPlotEasy.

Reads may also be filtered based on minimum alignment length, minimum percent of the read aligning to the reference (not available for tabular BLAST), and by selecting only those reads which map best to the currently viewed genome (i.e. allowing each read to map only once across the set of genomes available in the database). Reads are not processed in any way prior to their selection from the database, meaning that any plot and any statistics or data export based on a plot are created only from the reads which pass the filtering parameters selected by the user (or the defaults when user makes no choices). A record of the exact query used to select reads from the current RecruitPlotEasy database is exported alongside any saved plot or data export in order to ensure that the reads used to generate a plot can be recovered at later dates.

OUTPUT AND RESULTS

The recruitment plots from RecPlotEasy show four views of a single dataset; the main panel of a recruitment plot directly displays the underlying counts matrix, while the other three panels display useful summaries derived from the data in the main panel. **Figure 1** shows an example of the read recruitment plot obtained with a single gut metagenome mapped against a *Staphylococcus aureus* reference genome. We consider the subplot in the bottom left the main panel, which shows how metagenomic reads of sufficient nucleotide identity and alignment (user defined) thresholds map against the reference genome. Because read recruitment plots commonly recruit tens of thousands to millions of reads from a metagenome it is computationally intractable to directly plot each read individually. Instead, we plot the sum of all bases from the reads that fall within a specific region of the main panel defined as a grid with cells consisting of base pair width (x-axis; default = 1000) and percent sequence identity of the read alignment height (y-axis; default = 0.5%) (Methods section above).

The panel on the bottom right shows the number of nucleotide bases (i.e., density of reads) at each unit of nucleotide identity (y-axis) in the main panel. Note that the specific case in **Figure 1** shows an abundant *S. aureus* population in the sample, represented by a high density of high identity (>98% identity) reads mapping evenly across the reference genome along with a less abundant, closely related and co-occurring *S. aureus*-like population represented by the second lower peak in the number of mapped bases in the 94–95% identity range. The region between these two peaks shows the sequence discontinuity between the two sequence-discrete populations, one representing the reference genome and the other representing the sum of the remaining genomes.

Similarly, the upper left panel shows the coverage (i.e., how many times a base of the reference sequence is covered by mapped reads) of the genome at each unit of genomic position (x-axis) in the main panel. The darker blue color represents reads within the target population (default >95% nucleotide identity to the reference sequence) while the lighter blue represents lower identity reads considered outside the population (nucleotide threshold can be adjusted by the user). Note in **Figure 1** that the top left panel shows fairly even read coverage across the genome for the target population with the exception of a few lower coverage regions while the outside of population coverage is more variable. This is an expected result for a single, homogenous population that is a good match to the reference genome. When no reads map to a genomic region (i.e., the region shows zero coverage), the region is displayed at the bottom of the panel for its respective group (dark blue target or light blue off-target), discontinuous from the rest of the line chart. Such low- or zero-coverage windows typically occur when a reference gene(s) is absent in the sampled population; browsing over the windows in the interactive mode can reveal which genes are found in these windows and their functional annotation, when the latter information is provided at the input stage. Hence, RecruitPlotEasy can reveal the gene content differences between the reference genome and the metagenomic population.

Finally, the top-right panel shows the histogram of coverage depth values over regions of the genome, which should reveal a tight distribution around the mean in cases where the reference genome represents a single population and a not-chimeric genome, like in the *S. aureus* example in **Figure 1**. A wider distribution would have been expected in the case of a chimeric genome that represents two or more populations with distinct *in-situ* abundances. For further details on the panels, see also **Supplementary Figure S4**. **Supplementary Material** includes additional methodological details; **Supplementary Figures S5–S7** provide additional (less common) examples and use cases.

RecruitPlotEasy provides the user with the option to export their plots as high-quality figures and save the plotting data used to create them. Once the user has decided that they are satisfied with their plot as it appears in the GUI, they may provide a name and save the plotting data used to generate each recruitment plot sub-plot three in tab-delimited files and save a PDF of the plot image, laid out in a 16:9 aspect ratio to match the majority of modern screens. While RecruitPlotEasy uses the PDF format to save its plots, the graphics within each PDF are ultimately saved as an SVG that is both infinitely scalable without any loss in the figure's resolution (i.e. it can be zoomed in on as much as desired without losing any clarity) and can be easily imported into common figure editing software that support SVG manipulation such as Adobe Illustrator for further editing, labelling, or other manipulations desired by a user.

The interactive plots created by RecruitPlotEasy may also be exported, but the files produced are quite different from the PDFs that are generated by the normal plotting approach. The R Plotly library is used to generate interactive recruitment plots, and these interactive plots are saved as HTML files that contain an independent copy of the data used to create the plot and the graphical results. These can be subsequently opened by most internet browsers, and do not require RecruitPlotEasy, R at all, or any other external software to be shared and viewed.

To better support integration in workflows, RecruitPlotEasy also includes a built-in read filtering function. When a database is built, RecrutPlotEasy makes a note of the location of each input read file and indexes reads by their name and the genome each mapped to. When viewing plots, reads at or above a user-defined percent identity cutoff may be added to a cart. By returning to the database management tab of the GUI, reads in the cart can be exported to filtered outputs in the same format as the inputs (although BAM is converted to SAM format) that will contain only those reads matching the selection criteria currently chosen by the user. Depending on the users' chosen options, this can (and will, under default settings) include filtering reads to only their best matches, thus removing instances of reads mapping to multiple locations, filtering poorly aligned reads, selecting only reads mapping to specific genomes, and selecting only reads sufficiently similar to the reference sequence. This function is intended to be used in tandem with the plot saving component of RecruitPlotEasy: a user may inspect their read mapping results, tailor their filtering settings to match the properties of each genome as needed, and export reads alongside one or multiple recruitment plots that aid in justifying the choice of filtering parameters.

In summary, we have presented an advanced read recruitment plot tool that can provide a thorough and quantitative view of the natural population in a metagenome and its diversity, which represents highly useful information for several downstream analyses. Accordingly, we expect that RecruitPlotEasy will greatly facilitate microbiome research across the clinical and environmental fields and advance the toolbox for the analysis and summarization of large-scale genomic/metagenomic data and the communication of those results.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/KGerhardt/RecruitPlotEasy/tree/main/pub_data.

AUTHOR CONTRIBUTIONS

KTK and LMR-R conceived and designed research; LMR-R developed the first version of the code, which was substantially revised and improved by KG and CAR-P. REC provided useful suggestions regarding calling absent genes and identifying sequence gaps. Hence, all authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

This work has been supported by the US National Science Foundation (Award No 1759831 and 2129823)

ACKNOWLEDGMENTS

The authors would like to thank members of the Konstantinidis Lab for their comments and suggestions on read recruitment plots through the past decade.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2021.826701/full#supplementary-material

REFERENCES

Caro-Quintero, A., and Konstantinidis, K. T. (2012). Bacterial Species May Exist, Metagenomics Reveal. *Environ. Microbiol.* 14 (2), 347–355. doi:10.1111/j.1462-2920.2011.02668.x

Conrad, R., Viver, T., Gago, J. T., Hatt, J. K., Venter, F., Rosselló-Móra, R., et al. (2021). Toward Quantifying the Adaptive Role of Bacterial Pangenomes during Environmental Perturbations. *ISME J.* Accepted. doi:10.1038/s41396-021-01149-9

Jaenicke, S., Albaum, S. P., Blumenkamp, P., Linke, B., Stoye, J., and Goesmann, A. (2018). Flexible Metagenome Analysis Using the MGX Framework. *Microbiome* 6 (4), 76. doi:10.1186/s40168-018-0460-1

Konstantinidis, K. T., and DeLong, E. F. (2008). Genomic Patterns of Recombination, Clonal Divergence and Environment in marine Microbial Populations. *ISME J.* 2 (10), 1052–1065. doi:10.1038/ismej.2008.62

Meziti, A., Tsementzi, D., Rodriguez-R, L. M., Hatt, J. K., Karayanni, H., Kormas, K. A., et al. (2019). Quantifying the Changes in Genetic Diversity within Sequence-Discrete Bacterial Populations across a Spatial and Temporal Riverine Gradient. *ISME J.* 13 (3), 767–779. doi:10.1038/s41396-018-0307-6

Murray, C. S., Gao, Y., and Wu, M. (2021). Re-evaluating the Evidence for a Universal Genetic Boundary Among Microbial Species. *Nat. Commun.* 12 (1), 4059. doi:10.1038/s41467-021-24128-2

Olm, M. R., Crits-Christoph, A., Diamond, S., Lavy, A., Matheus Carnevali, P. B., and Banfield, J. F. (2020). Consistent Metagenome-Derived Metrics Verify and Delineate Bacterial Species Boundaries. *mSystems* 5 (1), e00731-19. doi:10.1128/mSystems.00731-19

Peña-Gonzalez, A., Soto-Girón, M. J., Smith, S., Sistrunk, J., Montero, L., Páez, M., et al. (2019). Metagenomic Signatures of Gut Infections Caused by Different *Escherichia coli* Pathotypes. *Appl. Environ. Microbiol.* 85 (24), e01820-19. doi:10.1128/AEM.01820-19

Robinson, J. T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G., et al. (2011). Integrative Genomics Viewer. *Nat. Biotechnol.* 29 (1), 24–26. doi:10.1038/nbt.1754

Rodriguez, -R., and Konstantinidis, K. T. (2016). The Enveomics Collection: a Toolbox for Specialized Analyses of Microbial Genomes and Metagenomes. *PeerJ Preprints*, e1900v1.

Rodriguez-R, L. M., Jain, C., Conrad, R. E., Aluru, S., and Konstantinidis, K. T. (2021). Re-Evaluating the Evidence for a Universal Genetic Boundary Among Microbial Species. *Narure Commun.* 12 (1), 4060.

Rusch, D. B., Halpern, A. L., Sutton, G., Heidelberg, K. B., Williamson, S., Yooseph, S., et al. (2007). The Sorcerer II Global Ocean Sampling Expedition: Northwest Atlantic through Eastern Tropical Pacific. *Plos Biol.* 5 (3), e77. doi:10.1371/journal.pbio.0050077

Zhu, Z., Niu, B., Chen, J., Wu, S., Sun, S., and Li, W. (2013). MGAviewer: a Desktop Visualization Tool for Analysis of Metagenomics Alignment Data. *Bioinformatics* 29 (1), 122–123. doi:10.1093/bioinformatics/bts567

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Gerhardt, Ruiz-Perez, Rodriguez-R, Conrad and Konstantinidis. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

frontiers in Bioinformatics

REVIEW
published: 24 February 2022
doi: 10.3389/fbinf.2022.821861

Applications and Comparison of Dimensionality Reduction Methods for Microbiome Data

*George Armstrong[1,2†], Gibraan Rahman[1,2†], Cameron Martino[1,2,3], Daniel McDonald[1], Antonio Gonzalez[1], Gal Mishne[4,5] and Rob Knight[1,5,6]**

[1]Department of Pediatrics, School of Medicine, University of California, San Diego, La Jolla, CA, United States, [2]Bioinformatics and Systems Biology Program, University of California, San Diego, La Jolla, CA, United States, [3]Center for Microbiome Innovation, Jacobs School of Engineering, University of California, San Diego, La Jolla, CA, United States, [4]Halıcıoğlu Data Science Institute, University of California, San Diego, La Jolla, CA, United States, [5]Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA, United States, [6]Department of Bioengineering, University of California, San Diego, La Jolla, CA, United States

OPEN ACCESS

Edited by:
Nikos Kyrpides,
Joint Genome Institute, United States

Reviewed by:
Rekha Seshadri,
Joint Genome Institute, United States
Travis Wheeler,
University of Montana, United States

****Correspondence:***
Rob Knight
robknight@ucsd.edu

†These authors have contributed equally to this work

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *24 November 2021*
Accepted: *08 February 2022*
Published: *24 February 2022*

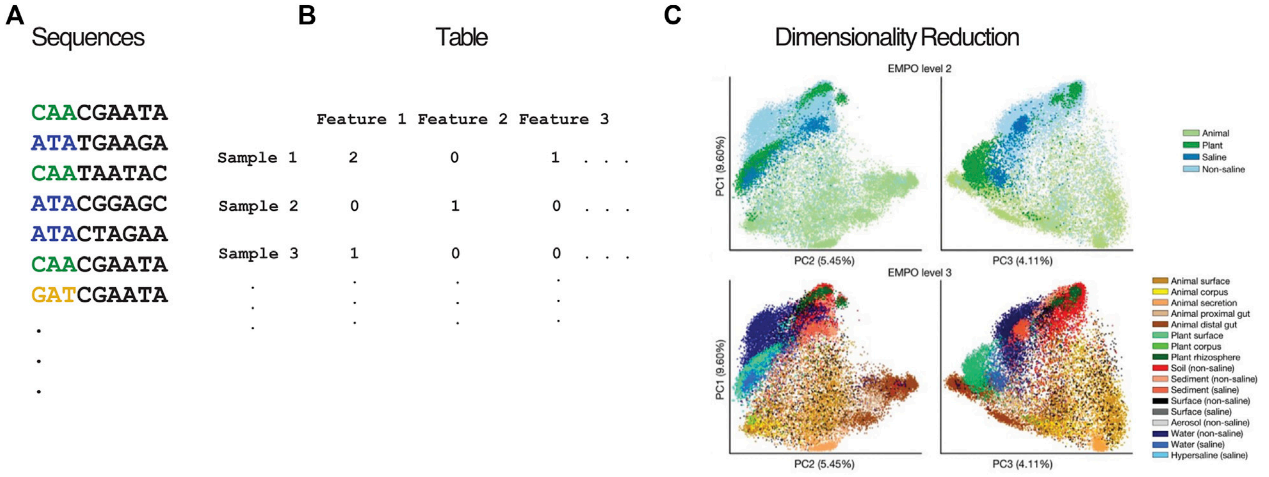
Citation:
Armstrong G, Rahman G, Martino C, McDonald D, Gonzalez A, Mishne G and Knight R (2022) Applications and Comparison of Dimensionality Reduction Methods for Microbiome Data. Front. Bioinform. 2:821861. doi: 10.3389/fbinf.2022.821861

Dimensionality reduction techniques are a key component of most microbiome studies, providing both the ability to tractably visualize complex microbiome datasets and the starting point for additional, more formal, statistical analyses. In this review, we discuss the motivation for applying dimensionality reduction techniques, the special characteristics of microbiome data such as sparsity and compositionality that make this difficult, the different categories of strategies that are available for dimensionality reduction, and examples from the literature of how they have been successfully applied (together with pitfalls to avoid). We conclude by describing the need for further development in the field, in particular combining the power of phylogenetic analysis with the ability to handle sparsity, compositionality, and non-normality, as well as discussing current techniques that should be applied more widely in future analyses.

Keywords: microbiome, dimensionality reduction, ordination, sequencing data, non-linear embeddings

INTRODUCTION: WHAT IS DIMENSIONALITY REDUCTION AND WHY DO WE DO IT?

To a first approximation, life on Earth consists of complex microbial communities, with “familiar” multicellular organisms such as plants and animals being rounding errors in terms of cell count and biomass. The genetic repertoire of such a community is called a “microbiome” (Turnbaugh et al., 2007), although the term “microbiome” is often also loosely applied to the collection of microbes that make up the community. In either sense, microbiomes are typically incredibly complex, containing vast numbers of species and genes, and how samples relate, even in well-studied contexts, are not predetermined. For example, in the Earth Microbiome Project (EMP) (Thompson et al., 2017) and the work leading up to it (Lozupone and Knight, 2007; Ley et al., 2008; Caporaso et al., 2011), an ontology constructed from the microbe’s perspective based on community similarities and differences revealed many surprises, such as a deep separation between free-living and host-associated samples, and between saline and non-saline samples. Accordingly, to truly understand the microbial perspective, we must get acquainted with the structure of the data in human-interpretable formats. This is especially important when we need to separate new biological discoveries from technical artifacts, such as distinguishing clusters related to different habitats



	Feature 1	Feature 2	Feature 3	
Sample 1	2	0	1	. . .
Sample 2	0	1	0	. . .
Sample 3	1	0	0	. . .
. . .	. . .	. . .	. . .	

FIGURE 1 | Overview of dimensionality reduction pipeline. Nucleotide sequences **(A)** from a biological experiment are organized in a feature table **(B)** containing the abundance of each feature (e.g., OTU, ASV, MAG) in each sample. **(C)** Beta diversity plots showing unweighted UniFrac coordinates of EMP annotated by EMPO levels 2 and 3. **(C)** is a derivative of **Figure 2C** from "A communal catalogue reveals Earth's multiscale microbial diversity" by Thompson et al. (2017) used under CC BY 4.0.

on the human body from artifacts caused by different sequencing methodologies such as PCR primers (The Human Microbiome Project Consortium, 2012).

When microbiome sequencing data (**Figure 1A**) are arranged into count tables (**Figure 1B**), such as those that count 16S amplicon sequence variants (ASVs) or the microbial genes present in a sample, the number of features being counted across all of the samples often vastly outnumbers the number of samples observed. This phenomenon of having many features, and particularly having far more features than samples, is a hallmark of high-dimensionality. For example, the EMP (Thompson et al., 2017) contained 23,828 samples and represented 307,572 ASVs, where each of these ASVs is considered a dimension of the resulting count table. This degree of high feature dimensionality creates difficulties for interpreting data and calculating meaningful statistics, since humans cannot visualize more than 3 dimensions, many of the features are noisy or redundant, the number of hypotheses that explain the data is far greater than the number of observations, and the number of features can cause run-time issues for downstream analysis. These are all common consequences of the "curse of dimensionality". Dimensionality reduction transforms a high-dimensional dataset into a representation with fewer dimensions, while retaining the key relationships among samples from the full dataset, making analysis tractable. Accordingly, dimensionality reduction is a core step in microbiome analyses, both for creating human-understandable visualizations of the data and as the basis for further analysis. The EMP used dimensionality reduction to produce plots of the 23,828 samples using 3 coordinates (in contrast to the 307,572 ASVs) that demonstrate the large difference between host-associated and non-host-associated microbiomes, and between saline and non-saline free-living microbiomes (**Figure 1C**). These differences in microbial communities were subsequently statistically validated. This example is particularly salient because it shows the value of preserving the structure of the data while using much less information to represent it. Owing to its importance, dimensionality reduction methods are included in many analysis packages, including QIIME 2 (Bolyen et al., 2019), mothur (Schloss et al., 2009), and phyloseq (McMurdie and Holmes, 2013), as well as online software such as Qiita (Gonzalez et al., 2018) and MG-RAST (Keegan et al., 2016).

In this review, we describe how the characteristics of microbiome data complicate dimensionality reduction. We then discuss common strategies for dimensionality reduction (**Table 1**), examining in detail whether and how they address each of the aspects that, in conjunction, confound microbiome analysis. Tried-and-true techniques, although useful, often have conceptual and practical problems that limit their utility in the microbiome, due to the inability to handle the data's most salient traits simultaneously (**Table 2**). In this light, we then focus on examples of how dimensionality reduction techniques have been used in the literature, highlighting biological findings that have been revealed by each, while also discussing what may have been obscured. We then discuss common artifacts of widely used dimensionality reduction techniques, including specific pitfalls that users of these techniques must avoid in order to draw conclusions that are robust, reproducible, and well-supported by their data. We end with guidance on how dimensionality reduction should be used responsibly by practitioners in the field, and with an outlook describing how additional techniques that are seldom used today might provide valuable advances.

Specific Features of Microbiome Data That Complicate Dimensionality Reduction

"Microbiome data" most often refers to sequencing results from two primary methodologies. The first class of microbiome sequencing is known as "amplicon sequencing" where a

TABLE 1 | Common characteristics of strategies for dimensionality reduction address different aspects of the data.

Table 1

Term	Definition
Compositionally aware	Transforms data to account for non-independence of features in sequence count data
Pseudo-counts or imputation	Requires no/minimal zeroes in the feature table due to numerical issues (such as logarithm transform being undefined on zeroes)
Able to incorporate phylogeny	Method is calculated with awareness of how each sampled microbial community is evolutionarily represented relative to other samples
Operates on beta-diversity dissimilarities	Dimensionality reduction step is performed on pairwise dissimilarities (arbitrary metric) between samples, rather than the feature table itself
Linear	Lower dimensional coordinates are computed via linear transform of features
Repeated measures	Subjects are sampled multiple times. Commonly sampled longitudinally
Feature relationships are interpretable	The method indicates the relevance of input microbial features with regard to its output coordinates
Supervised component	Method takes explanatory sample variables as an additional input

TABLE 2 | Dimensionality reduction methods each have their own characteristics. x indicates that the characteristic applies to the method. Examples of software capable of performing each method are included in the last column.

Table 2

	Compositionally aware	Avoids pseud-counts or imputation	Able to incorporate phylogeny	Operates on beta-diversity dissimilarities	Linear	Repeated measures	Feature relationships are interpretable	Supervised component	Software
PCoA	—	x	x	x	x	—	—	—	QIIME 2, CRAN phyloseq, mothur
PCA	—	x	—	—	x	—	x	—	scikit-learn, R built-in, mothur
UMAP	—	x	x	x	—	—	—	—	umap-learn, CRAN umap, QIIME 2
t-SNE	—	x	x	x	—	—	—	—	scikit-learn, CRAN tsne
nMDS	—	x	x	x	—	—	—	—	scikit-learn, CRAN vegan, mothur, CRAN phyloseq
CCA	—	—	—	—	x	—	x	x	scikit-bio, CRAN vegan, CRAN phyloseq
PLS-DA	—	—	—	—	x	—	x	x	CRAN mixOmics
Aitchison PCA	x	—	—	—	x	—	x	—	scikit-bio, QIIME 2
RPCA	x	x	—	—	x	—	x	—	gemelli, QIIME 2, vegan
CTF	x	x	—	—	x	x	x	—	gemelli, QIIME 2

specific gene or region of a gene is targeted in each sample. 16S, 18S, and ITS sequencing approaches all fall under this class of methods. Variants of the targeted nucleotide sequences are used as a proxy for discrete microbial taxa. These unique sequences can be clustered by sequence similarity into "operational taxonomic units" (OTUs) or used by themselves as individual units after denoisers, such as DADA2 and Deblur, resolve the individual sequence variants from error-prone sequences (Callahan et al., 2017; Amir et al., 2017). These filtered sequences are often called amplicon sequence variants (ASVs) (Callahan et al., 2017) or sub-OTUs (sOTUs). The second class of microbiome sequencing is shotgun or whole metagenome sequencing. In this method, the DNA from a sample is collected and sequenced broadly. The reads are then mapped to a reference database to determine the corresponding units, which can range from taxonomic identities to gene families or genes from a specific reference genome or metagenome-assembled genomes (MAG).

The result of these sequence analysis pipelines is typically a "feature table" that counts the microbial "units" or features

(OTU, ASV, MAG, etc., (**Figure 1B**)) associated with each sample. Additionally, information about the relationship between features, such as taxonomic identity or gene family, can optionally be used to "collapse" the feature table to a lower resolution sum of its units. At this point, the data are generally ready to pursue exploratory analysis with dimensionality reduction.

However, there are several features common to microbiome data that can make standard dimensionality reduction techniques difficult to apply or to interpret. Each method must therefore handle each of these key issues or be benchmarked carefully to determine that these issues do not strongly affect the results in ways that are problematic for biological interpretation. We demonstrate various dimension reduction techniques on two datasets: Lauber et al., 2009 (**Figures 2A–D**) and Shalapour et al., 2017 (**Figures 2E–H**) looking at soil pH and antibiotic-diet axis respectively.

High dimensionality. In this context, "dimensionality" refers to the number of features in a feature table. Microbiome data typically have far more features than samples. Across studies ranging from tens of samples to tens of thousands of samples, the number of features for taxonomic data typically exceeds the number of samples by 20-fold or more. With gene-oriented data, the number of genes represented in a metagenomic study typically exceeds samples by several orders of magnitude. This can lead many statistical methods to overfit or to produce artifactual results.

Sparsity. Most microbes are not found in most samples, even of the same biospecimen type, for example, most human stool specimens from the same population have relatively low shared taxa (Allaband et al., 2019). As a result, a feature table containing counts of each microbe in each sample often has many zeros corresponding to unobserved microbes. Most 16S microbiome datasets do not have even as many as 10% of the possible entries observed in most of the specimens. Feature tables with this over-abundance of unobserved counts are said to be "sparse", posing problems for statistical analysis. Moreover, the proportion of observed values tends to decrease as additional samples are sequenced, often leading to tables with density well below 1% (Hamady and Knight, 2009; McDonald et al., 2012).

Compositionality. In any high-throughput sequencing experiment, we impose an implicit limitation and randomness to the number of reads from a given sample due to many factors, including the random sub-sampling occurring in the process of collecting samples as well as uncontrolled variation in how efficiently each sample is amplified and incorporated into molecular libraries for sequencing. This limitation, termed "compositionality", should always be kept in mind when performing any microbiome analysis on abundance data (Gloor et al., 2017). The total number of sequences per sample can affect the distances between samples (Weiss et al., 2017). Strategies such as rarefaction and relative abundance normalization are common for normalizing differences in sequencing depth. However, the relative amount of one feature in the sample is not independent from the counts of the other features. A difference in just one feature of the original sample can induce an observation that many other features are also changing (Morton et al., 2019) and neither rarefaction nor relative abundance sampling solve this issue. Due to this effect, many dimensionality reduction methods, such as PCA, will emphasize false correlations in the data.

Repeated measures. One of the most challenging experimental aspects to account for in dimensionality reduction is repeated measures data, e.g., multiple timepoints from the same subject where the variation between subjects may be greater than the variation between timepoints (Wu et al., 2011). In the context of dimensionality reduction, subjects or sites with multiple samples represented (such as in longitudinal studies or replicate analysis) provide an additional source of variation that can inhibit interpretation of the experimental effect of interest; the samples from a single subject can be highly correlated, resulting in between-subject differences dominating the ordination [e.g., (Song et al., 2016)].

Feature interpretation. Analysis of high-dimensional microbiome data is often motivated to find microbial biomarkers associated with observed differences in sample communities (Fedarko et al., 2020). This line of inquiry is of interest for diagnosis and/or prognosis of disease status, dysbiosis, and a host of other biological questions. Although this task is often addressed with differential abundance methods, those methods make specific statistical assumptions and may not correspond to the group separation observed in an exploratory analysis performed with any dimensionality reduction method (Lin and Peddada, 2020). Thus, methods that offer a quantitative justification of their representation in terms of the microbial features are often desirable. However, methods with feature importance that are not specifically designed for the microbiome often fail to account for compositionality, which can include many false positives due to the induced correlation of features, and sparsity, where important but infrequently observed features will not be detected (false negatives).

Complex patterns. Microbiome data are often assumed to contain clusters or gradients (Kuczynski et al., 2010). For example, multiple samples swabbed from one's own keyboard are more likely to be similar to each other than samples from another individual's keyboard (Fierer et al., 2010), and the microbial composition of soils is expected to vary continuously with soil pH (Lauber et al., 2009). However, with larger and larger datasets with many covariates and metadata on these being collected, more complex patterns can be detected (Debelius et al., 2016), such as grouping by both biological and technical factors in the case of the Human Microbiome Project (The Human Microbiome Project Consortium, 2012). Furthermore, many conventional dimensionality reduction methods, such as principal component analysis (PCA), assume the data lie in a linear subspace, and this assumption is violated by microbiome data (Ginter and Thorndike, 1979; Greig-Smith, 1980; Potvin and Roff, 1993; Tabachnick and Fidell, 2013).

Strategies for Dimensionality Reduction in the Microbiome

The problems that complicate dimensionality reduction in microbiome data are scattered throughout the analysis pipeline. Difficulties can arise immediately from the raw sequence count data. Many can be corrected before the dimensionality reduction step, with careful preprocessing, though this can raise other issues. Furthermore, beta-diversity analysis, which seeks to quantify the pairwise differences in microbial communities among all samples with dissimilarity metrics (tailored to microbiome data), is often helpful for addressing many of the aforementioned circumstances (Pielou, 1966). Algorithms that are able to incorporate these metrics are particularly valuable, and this can be done in a variety of ways. Finally, additional constraints can be placed on dimensionality reduction algorithms to account for study design or provide additional information about the correspondence between the features and the reduced dimensions. In this section, we discuss each of these strategies in depth.

Compositionally Aware: Comparisons between and among samples must consider how sampling and sequencing depth can affect projection into low-dimensional space. Traditionally, compositionality has been addressed using logarithmic transformations of feature ratios. Transformations such as the additive log-ratio (ALR), centered log-ratio (CLR), and isometric log-ratio (ILR) can convert abundance data to the space of real numbers such that analysis and interpretation are less skewed by false positives (Aitchison and Greenacre, 2002; Pawlowsky-Glahn and Buccianti, 2011). After transformation, the Euclidean distance can be taken directly on the log-ratio transformed data (referred to as Aitchison distance) (Aitchison and Greenacre, 2002). Dimensionality reduction methods that incorporate log-ratio transformations attempt to preserve high-dimensional dissimilarities while taking into account the latent non-independence of microbial counts.

Pseudocounts and Imputation: High-dimensional microbiome data is almost always plagued by problems of "sparsity", or an overabundance of zeroes. The data transformations to address compositionality (as outlined above) are often based on logarithmic functions which are undefined at zero. The simplest solution is to add a small positive pseudocount to each entry of the feature table so that logarithmic functions can be applied. However, downstream analyses based on this approach are sensitive to the choice of pseudocount (Kumar et al., 2018) and there does not exist a standardized way to choose such a value. Other options include imputation of zeros (Martín-Fernández et al., 2003) through inference of the latent vector space. Fundamentally, zero handling is complicated by the inherent unknowability of the zero generating processes for each zero instance. In Silverman et al. (2020), they characterize the three different types of zero-generating processes (ZGP) as sampling, biological, and technical and demonstrate how the results of different zero-handling processes are affected by the (unknowable) mix of ZGPs in a given dataset. Recently Martino et al. (2019) introduced a version of the CLR transform that only computes the geometric mean on the non-zero components of a given sample. This avoids the problem of logarithms being undefined at 0 and thus dimensionality reduction through this method is robust to the high levels of sparsity in microbiome data.

Incorporating Phylogeny: Organisms identified using microbiome data can be related to one another through hierarchical structures that describe their evolutionary relationships. Typically, these structures take the form of either a taxonomy or a phylogeny. A taxonomy is a description of the organism relationships, generally derived subjectively using multiple biological criteria. A phylogeny, in contrast, is an inference of a tree, commonly with branch lengths, derived from quantitative algorithms that are typically applied to microbial, nucleic acid, or protein sequence data. Taxonomies have the advantage of being more directly interpretable because hierarchical structures correspond to a defined organization and classification pattern curated by experts in the field. However, these assignments and hierarchies are often putative and subject to change as more information about microbial taxa emerges. In contrast, phylogenies are derived from quantitative measures of sequence similarity from sample reads. These data structures are more easily incorporated into statistical analyses but often at the cost of less interpretability as the hierarchical structures do not necessarily map to pre-defined microbial relationships. These evolutionary relationships, particularly phylogenies, add information to microbiome analysis, because related organisms are more likely to exhibit similar phenotypes (although counterexamples do exist, especially closely related taxa such as *Escherichia* and *Shigella*, which are very similar genetically but produce different clinical phenotypes).

When comparing the similarity of pairs of microbial communities, it is possible to utilize these hierarchical structures, and derive a metric that computes a dissimilarity as a function of shared evolutionary history (Lozupone and Knight, 2005). Specifically, communities that are very similar will share most of their evolutionary history, whereas those that are very dissimilar will have relatively little in common. A popular form of phylogenetically-aware distances is the suite of UniFrac metrics, which includes both quantitative (Lozupone et al., 2007) and qualitative (Lozupone and Knight, 2005) forms. Numerous extensions to UniFrac have been developed (Chang et al., 2011; Chen et al., 2012), including variants that account explicitly for the compositional nature of microbiome data (Wong et al., 2016). Because these metrics all utilize not only exactly observed features, but also the relationships among features, they can better account for the sparsity of microbiome data which manifests at the tips of a phylogenetic tree (because most microbes are not observed in most environments). In contrast, a metric like the Euclidean distance is limited to only the information at the tips of these hierarchies, and, worse, assumes that all features at the tips are equally related to one another (so that in a tree consisting of a mouse, a rat, and a squid, there is no allowance for the fact that the two rodents are much more similar to each other than they are to the squid). Neither phylogenetic nor non-phylogenetic beta-diversity measures explicitly model differences in sequencing depth per sample, although these differences in depth can be standardized through rarefaction (Weiss et al., 2017).

Operates on Generalized Beta-Diversity Matrix: Many of the issues outlined above can be easily addressed at the sample dissimilarity level rather than directly through dimensionality reduction algorithms. A number of dissimilarity/distance metrics have been developed to account for factors such as phylogenetic data incorporation, compositionality, or sparsity that output a sample by sample matrix estimating high-dimensional dissimilarity. These dissimilarity matrices represent the overall community differences between pairwise samples calculated by a chosen beta-diversity metric. Dimensionality reduction methods that operate on arbitrary dissimilarity metrics are attractive options because the complex handling of the various feature table issues can be split into the choice of dissimilarity metric and the choice of dimensionality reduction algorithm. This adds a layer of flexibility for researchers to analyze their data depending on their needs. Methods based on multidimensional scaling approaches such as PCoA (Kruskal and Wish, 1978) and nMDS (Kruskal, 1964) attempt to preserve as much as possible the pairwise dissimilarities between subjects. Other methods such as t-distributed stochastic neighbor embedding (t-SNE) (van der Maaten and Hinton, 2008) and Uniform Manifold Approximation and Projection (UMAP) (McInnes et al., 2018) are non-linear dimensionality reduction techniques that aim to find a low-dimensional representation such that similar data points are placed closed together and dissimilar points are pushed apart. A caveat of these methods is that they can be very sensitive to the choice of dissimilarity used. Patterns that may appear from one measure of dissimilarity may not be as apparent in a different measure. As an example, phylogenetic metrics such as UniFrac may differ from non-phylogenetic metrics such as Bray-Curtis depending on the strength of phylogenetic contribution (Shankar et al., 2017). The choice of dissimilarity metric should therefore be considered carefully, as different dimensionality reduction techniques yield visually and statistically very different results on the same data (Kuczynski et al., 2011).

Linear vs Non-Linear Methods: Principal coordinates analysis (PCoA) and PCA are popular dimensionality reduction techniques that fall under the "linear" category. Linear techniques attempt to reduce or transform the data such that an approximation of the original data can be reconstructed by a weighted sum of the resulting coordinates. These methods typically involve computing decompositions/factorizations of the data that are highly computationally efficient and work well on data that is naturally linear. Various other techniques, such as robust Aitchison PCA (RPCA) (Martino et al., 2019), and nonnegative matrix factorization (NMF) (Lee and Seung, 1999) also fall under this class of techniques.

Other methods fall under the "non-linear" category, which perform more complex transformations that often excel at preserving different patterns that may not be linear. This category includes methods such as the non-metric multidimensional scaling (nMDS), t-SNE, and UMAP. These methods can more succinctly represent complex patterns, but possibly at the expense of additional computation. Furthermore, these models tend to have randomness (such as from initialization) and more hyperparameters that the output can be highly sensitive to, so it is usually necessary to run these algorithms multiple times to ensure the conclusions are reproducible. Other non-linear methods that have seen less frequent use in microbiome data (and bioinformatics generally) include kernel PCA (Scholkopf et al., 1999), locally linear embeddings (Roweis and Saul, 2000), Laplacian eigenmaps (Belkin and Niyogi, 2001), and ISOMAP (Tenenbaum et al., 2000).

Unlike its close, linear counterpart PCoA, nMDS performs the ordination onto a pre-specified number of dimensions and operates on the ranks of the dissimilarities, rather than the dissimilarities themselves. This rank-based approach can be beneficial for representing data that departs from the assumptions of linearity. Other non-linear methods, such as t-SNE and UMAP, also transform the data onto a pre-specified number of dimensions and operate by assuming the high-dimensional data follow a non-linear structure that can be represented with fewer dimensions.

Repeated Measures: If the biological variable of interest occurs at the subject level, repeated samples (such as through a longitudinal study design) can artificially inflate how tight a cluster appears in low-dimensional space. Dimensionality reduction methods for microbiome need to be designed for the purpose of handling this kind of data, with the intent to represent the relationships between explanatory variables while accounting for the inherent similarity between samples from the same subject. Methods to account for repeated measures can incorporate the relationship between individual samples and subjects by subject-aware decomposition of the data (Martino et al., 2021). There has also been discussion about incorporating prior sample relationship information into ordinations through Bayesian methods (Ren et al., 2017). Nevertheless, methods that incorporate repeated measures remain an underexplored area in dimensionality reduction literature.

Feature Importance: When the lower-dimensional representation of microbial communities shows separation between sample groups, a natural next question is what microbes or groups of microbes are driving such a separation. Dimensionality reduction methods that return a quantitative relationship between individual microbial features and the latent lower-dimensional space are a powerful class of methods that can demystify the construction of the lower-dimensional axes. However, certain methods that attempt to find high-dimensional patterns, such as non-linear methods, do not have an explicit interpretable correspondence between the output coordinates and the input features.

The most relevant category of methods for visualizing feature importance is the biplot ordination family of approaches. Biplots display both the samples and the driving variable vectors in reduced dimension space (**Figures 2A,D,E,H**). For example, PCA naturally quantifies the contribution of each microbe to the principal component axes through matrix factorization into linear combinations of features. RPCA modifies this approach to account for compositionality and sparsity while retaining interpretable feature loadings (Martino et al., 2019). Another set of ecologically motivated matrix factorization methods is the correspondence analysis (CA) family. The general CA method can be thought of as an implementation of PCA that operates on count data. It is also possible to explicitly

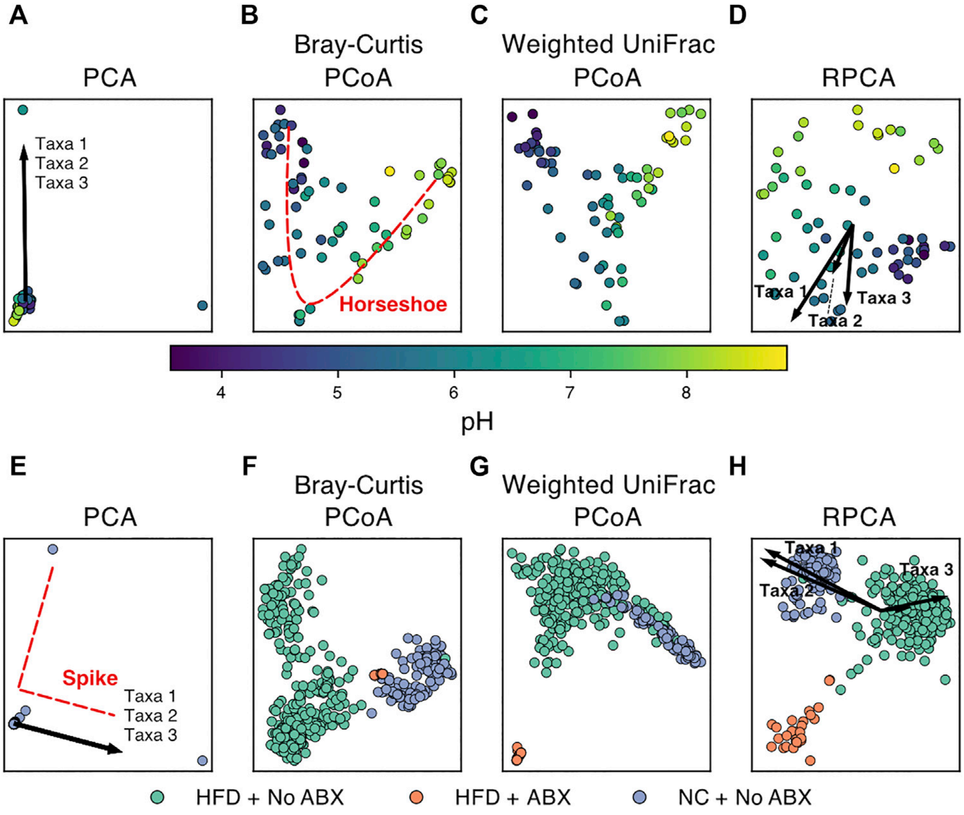


FIGURE 2 | Examples of dimensionality reduction techniques applied to publicly available microbiome data. (Top) Beta-diversity plots of soil samples colored by pH from (Lauber et al., 2009). (Bottom) Beta-diversity plots of murine fecal samples colored by diet and antibiotics usage from (Shalapour et al., 2017). (HFD = high-fat diet, NC = normal chow, ABX = antibiotics). PCA plots **(A,E)** show extremely high sample overlap due to outliers and characteristic "spike" artifacts. The top three taxa driving variation also overlap as shown by arrow superposition. **(B)** "Horseshoe" pattern emerges for samples following ecological gradients such as pH. RPCA plots **(D,H)** show the top three taxa driving separation of groups. **(F)** and **(G)** show strong overlap of HFD + ABX samples resolved by **(H)**.

incorporate sample metadata into these dimensionality reduction methods. Researchers are often interested in the explanatory power of their sample metadata (site, pH, subject, etc.). Certain dimensionality reduction methods can take as input both a feature table and a table of sample metadata to jointly estimate the low-dimensional representation of samples as well as the relative contribution of the provided metadata vectors. The general goal of these methods is to determine whether and/or which explanatory variables may be driving the differences in microbial communities among samples. Canonical correspondence analysis (CCA) is an extension of CA that incorporates sample variables of interest to determine which covariates are associated with the placement of samples and feature vectors in low-dimensional space (ter Braak, 1986). The results of CCA can be visualized as a "tri-plot" where samples are simultaneously visualized with the relative contribution of features and explanatory variables near related samples (Paliy and Shankar, 2016). Partial least squares discriminant analysis (PLS-DA) is a similar approach that uses only categorical sample metadata (classification) in the construction of lower-dimensional axes (Barker and Rayens, 2003; Ruiz-Perez et al., 2020). In each of these cases, the feature contributions can motivate subsequent statistical analysis of associations between sample metadata and specific microbial taxa.

Uses of Dimensionality Reduction for Microbiome Data

Over the past decade, PCoA has seen an increase in use in microbiome analyses, and it is the primary ordination method for beta-diversity included by default in workflows such as QIIME2 (Bolyen et al., 2019). It is typically used for exploratory visualization, as it excels at rendering biologically relevant patterns, such as clusters and gradients (Kuczynski et al., 2010). When used as an exploratory tool, observed patterns are often followed with statistical analysis on the original feature tables or dissimilarity matrices (Galloway-Peña and Hanson, 2020), such as ANOSIM (Clarke and Ainsworth, 1993), PERMANOVA (aka Adonis) (Anderson, 2017), ANCOM (Mandal et al., 2015), or bioenv (Clarke and Ainsworth, 1993). It should also be noted that some of these statistical techniques use the full table or dissimilarity matrix, not the reduced dimension matrix as visualized (at least by default) and may therefore introduce incongruent results between the statistics and the visualization.

Exploratory visualizations have revealed microbial-associated patterns in applications ranging from host-associated gut microbiomes to soil, ocean, and other environmental microbiome contexts. For example, studies have applied PCoA to demonstrate differences between host groups, such as differences between humans', chimpanzees', and gorillas' gut microbial taxa (Campbell et al., 2020), or the correspondence between human gut microbiomes and westernization (Yatsunenko et al., 2012; Campbell et al., 2020). Host microbiome-disease associations have also been identified using PCoA, such as in the case of colorectal cancer (Young et al., 2021) in humans and metritis in cows (Galvão et al., 2019). Uses also extend to host-environment relationships, such as demonstrating the differences between oyster digestive glands, oyster shells, and their surrounding soils (Arfken et al., 2017). The microbiome-shaping roles of environmental factors such as salinity in shaping free-living environments (Lozupone and Knight, 2007), pH in arctic soils (Malard et al., 2019) and depth in the ocean (Sunagawa et al., 2015) have also been elucidated with PCoA. In many of these cases, the PCoA visualizations demonstrated a separation between groups that was subsequently followed by statistical validation with PERMANOVA or ANOSIM.

In numerous other instances, PCoA has also been used to make claims that extend beyond exploratory group differences followed by statistical analysis. For example, Halfvarson et al. (2017) fit a plane to the healthy subjects in the first three coordinates of a PCoA and then used the distance to this plane to associate dissimilarities in the microbiome with the severity of irritable bowel disease (IBD) (Halfvarson et al., 2017); this approach has subsequently been replicated (Gonzalez et al., 2018). Others have used regression of participant and microbiome characteristics (e.g., age and alpha diversity, respectively) on PCoA coordinates to determine whether the given factors have a significant relationship with microbial community composition in the context of dietary interventions (Lang et al., 2018). In one case, while providing visualization with PCoA and statistical confirmation with ANOSIM, Vangay et al. (2018) additionally plotted ellipses for visualizing cluster centers/spread in their PCoA coordinates (Vangay et al., 2018). In another instance, Metcalf et al. (2017) showed the correspondence of dissimilarities between the 16S rRNA profiles and chloroplast marker profiles by performing a Procrustes analysis on the separate ordinations of the different data types (Metcalf et al., 2017).

We note that the choice of dissimilarity metric can have a significant impact on the low-rank embedding depending on the dataset. Shi et al. (2022) review the effect of high and low-abundance operational taxonomic units have on unsupervised clustering of Bray-Curtis and unweighted UniFrac (Shi et al., 2022). Marshall et al. (2019) compare Bray-Curtis ordination with weighted UniFrac on marine sediment samples and note that the most relevant clustering variable differed depending on the dissimilarity used (Marshall et al., 2019). These results imply that interpretation of low-dimensional embeddings and the putative driving variables must be performed in the context of the choice of dissimilarity. Metrics such as Bray-Curtis and weighted UniFrac take into consideration the abundance of individual microbes in each sample which can be important for datasets with many rare taxa. In contrast, some dissimilarity metrics such as Jaccard and unweighted UniFrac are only defined on binarized data, which may mask this property. Furthermore, phylogenetic metrics such as the UniFrac suite of metrics are best when the evolutionary relationships among microbial features is of interest in the context of sample communities. These metrics may also be more appropriate than other methods for datasets with particularly high sparsity.

PCA is arguably the most widely used and popular form of dimensionality reduction, which does not allow generalized beta-diversity dissimilarities (e.g., PCoA or UMAP), but does allow for the direct interpretation of feature importances relative to sample separations in the ordination. However, due to compositionality and sparsity, PCA often leads to spurious results on microbiome data (Hamady and Knight, 2009; Morton et al., 2017). Aitchison PCA attempts to fix these issues by using log transformation, but imputation is required (because the log of zero is undefined). Therefore, (Martino et al., 2019) proposed the adoption of RPCA for dimensionality reduction. This method has been shown to discriminate between sample groups in a wide array of biological contexts, including fecal microbiota transplants (Goloshchapov et al., 2019), cancer (Bali et al., 2021), and HIV (Parbie et al., 2021). Moreover, the generalized version of this technique accounts for repeated measures, allowing for large improvements in the ability to discriminate subjects by phenotypes across time or space (Martino et al., 2021). This advantage has been crucial in the statistical analysis of complicated longitudinal experimental designs such as early infant development models (Song et al., 2021). Feature loadings from these PCA-based methods can be used to inform selection of microbial features for log-ratio analysis (Morton et al., 2019; Fedarko et al., 2020), leading to novel biomarker discovery.

For feature interpretation, CCA is the most commonly used CA-based method for analyzing high dimensional microbiome data, due to its ability to incorporate sample metadata into the low-rank embeddings. This strategy has shown success in differentiating clinical outcomes following stem cell transplantation (Ingham et al., 2019) as well as diarrhea status in children (Dinleyici et al., 2018). CCA has also shown success in projecting environmental samples into lower-dimensional space such as in rhizosphere microbial communities (Benitez et al., 2017; Pérez-Jaramillo et al., 2017), and aerosol samples (Souza et al., 2021). Another approach designed for microbial feature interpretation has been posed by (Xu et al., 2021), explicitly modeling the ZGP through a zero-inflation model. This method attempts to optimize a statistical model for jointly estimating the "true" zero-generating probability as well as the Poisson rate of each microbial count.

Of non-linear methods, nMDS has historically been more widely used in microbiome data analysis, in part because it can incorporate an arbitrary dissimilarity measure. Furthermore, since nMDS is a rank-based approach, it is less likely than linear methods to be highly influenced by outliers in beta-diversity dissimilarities. Recent uses have involved using nMDS to show differences in the gastric microbiome between samples from patients with gastric cancer cases against the control of gastric dyspepsia (recurrent indigestion without apparent cause) (Castaño-Rodríguez et al., 2017) and demonstrating differences in the gut microbiome based on diabetes status (Das et al., 2021). In both of these cases, the visual distinction between groups was supported by PERMANOVA.

Other non-linear methods have been increasingly used for analyzing other types of sequencing data, especially in the single-cell genomics field, but have not yet been widely deployed in the microbiome. The most popular of these methods for visualization, t-SNE and UMAP, are starting to see more use in the microbiome field. (Xu et al., 2020) developed a method to classify microbiome samples using t-SNE embeddings. We recently reviewed the usage and provided recommendations for implementing UMAP for microbiome data (Armstrong et al., 2021). UMAP with an input beta-diversity dissimilarity matrix can reveal biological signals that may be difficult to see with traditional methods such as PCoA.

Artifacts and Cautionary Tales in Dimensionality Reduction

Dimensionality reduction is incredibly useful and has led to many interesting biological conclusions. However, when using dimensionality reduction techniques, one must be careful how results are interpreted. There are known examples of patterns that are induced by the properties of the data alone (rather than the relationships among specific samples or groups of samples), and others that are a product of the method itself. Here, we discuss several known issues, as well as insights into evaluating the degree to which an ordination represents the actual data.

One of the most well-known artifacts in microbial ecology is the horseshoe effect (Podani and Miklós, 2002), wherein the ordination has a curvilinear pattern along what otherwise appears to be a linear gradient. This pattern can occur when a variable, such as soil pH (Lauber et al., 2009) or length of time of corpse decay (Metcalf et al., 2016) corresponds with drastic changes in microbiome composition on a continuous scale. Since the characteristic "bend" in the horseshoe typically occurs along the second coordinate of a PCoA (**Figure 2B**), it can obfuscate additional gradients/associations along that axis. Recent research in the topic has also identified that indeed, it is unlikely the horseshoe appears from a real effect, and instead it is a product of the limitations of many dissimilarity metrics to capture distance along a gradient when no features are shared between many of the samples (i.e., saturation) (Morton et al., 2017), which can be an issue with many common metrics, such as Euclidean, Jaccard, and Bray-Curtis dissimilarities (Morton et al., 2017). As a result, a possible remedy for the artifact is to use a dissimilarity metric that considers the relationships between features, such that two samples that share no features do not necessarily have the same dissimilarity as two different samples that share no features, e. g, UniFrac or weighted UniFrac. If a change in metric does not resolve the issue, it may be possible to avoid the horseshoe artifact by using RPCA or a non-linear method (e.g., UMAP). "Spikes" are another artifact, more prevalent on cluster-structured data, where outliers dominate the embedding and it fails to separate into clusters in the visualization (Vázquez-Baeza et al., 2017). Spikes also appear to be mitigated with an appropriate choice in dissimilarity metric, such as UniFrac (Hamady and Knight, 2009). In both cases, since the issues are with representing the distances between distant or extreme samples, non-linear methods (such as UMAP or nMDS) that dampen the effect of outliers provide a potential workaround to reveal secondary gradients or the obfuscated cluster structures (Armstrong et al., 2021). Though it is possible that the benefits offered by non-linear methods for the horseshoe effect are limited by the aspect ratio of the gradient (Kohli et al., 2021), and potentially the parameters of the algorithms.

Dimensionality reduction is also commonly used in other bioinformatic disciplines. Particularly, single-cell transcriptomics has used dimensionality reduction prolifically, with many publications using PCA, t-SNE, or UMAP visualizations. Furthermore, single-cell RNA-seq data shares many properties with microbiome data, including sparsity/zero-inflation, sequencing depth differences, and even phylogenetic relationships (Lähnemann et al., 2020). This connection is further strengthened by the fact that researchers in both disciplines investigate similar types of questions, albeit with different underlying data. Microbiome researchers often ask whether there is a difference between different treatments or disease-statuses (David et al., 2013; Lloréns-Rico et al., 2021), and which microbes contribute to those differences (i.e., differential abundance analysis). Similarly, transcriptomics may investigate parallel scenarios (Ocasio et al., 2019; Taavitsainen et al., 2021), where the goal is to discover transcripts whose expression stratifies the desired groups (i.e., differential expression).

Despite these similarities, the most popular methods for dimensionality reduction in microbiome and single-cell publications differ significantly, with PCoA being more prevalent among microbiome publications, and t-SNE (or variants (Linderman et al., 2019)) and UMAP more prevalent in single-cell publications (Kobak and Berens, 2019). Given the similarities in hypotheses and the properties of the data, but use of different methods, it is reasonable to suppose that methods such as t-SNE and UMAP have potential utility in the microbiome. However, global distances are not necessarily preserved in these methods, therefore distances between different clusters should not be interpreted as demonstrating similarity or dissimilarity. Consequently, recent research concerning the representation of single-cell RNA-seq data should also be taken into account when applying these methods to microbiome data.

First, t-SNE and UMAP are fairly complex algorithms that have many hyperparameters that can be adjusted, so it is important to be able to evaluate the faithfulness of the embeddings they produce. The evaluation of dimensionality reduction has been performed with many different measures, each of which has its own characteristics. Some measures reward embeddings that adequately preserve the local-scale structures in the embedding but do not necessarily penalize inaccurate representations of large distances in the original high-dimensional data, like the KNN evaluation measure (Kobak and Berens, 2019), which takes the average accuracy of the k = 10 nearest neighbors in the reduced dimensions compared to the original space. Others, such as the correlation (either Pearson or Spearman) between distances in the original space and reduced dimensions have been used (Becht et al., 2019; Kobak and Berens, 2019; Kobak and Linderman, 2021). The correlation measure generalizes whether the two representations overall are similar, i.e. close points in the original space are close in the low-dimensional space, and similar for far points. However, high correlation does not guarantee that the fine-scale structures have been preserved. Additionally, measures that use sample metadata about known classes can be used, such as the KNC measure (Kobak and Berens, 2019), which measures whether the closest class/ category centers to a given category are preserved in the embedding. KNC emphasizes the preservation of relationships between classes, but not necessarily structures within the classes or between distant classes. These measures have been used to evaluate the quality of several dimensionality reduction methods across a variety of parameter settings on complex datasets. Notably, Kobak and Berens (2019) demonstrated on several single-cell transcriptomics datasets, that t-SNE with the default value for "perplexity" performed well at representing the relationships between nearby points (KNN), but poorly at representing the large-scale patterns (KNC and correlation). However, when they increased the perplexity parameter, they achieved improved KNC and correlation at the expense of a decreased KNN score. Kobak and Linderman (2021) observed with correlation that the best method (between t-SNE and UMAP) can vary by dataset. So, in practice, it may be necessary to compare multiple dimensionality reduction methods (and parameter settings) on a dataset using the measure that best suits the question, e.g., use the correlation measure when seeking a visualization of earth microbiomes by environment to show which environments are similar to each other.

Furthermore, since UMAP and t-SNE are algorithms that require configurable (possibly random) initializations, particular attention has been paid to their reproducibility. A metric to evaluate reproducibility comes from (Becht et al., 2019), which measures the preservation of pairwise distances in the embeddings by comparing an embedding on a subset of the points to the location of those points in the embedding of the entire dataset. In its original application, the reproducibility measure was used to demonstrate UMAP providing more reproducible results than t-SNE and variants of t-SNE. However, (Kobak and Linderman, 2021) showed that with appropriate (spectral) initialization, t-SNE can perform just as well by this metric as UMAP. While reproducibility is important, this metric should be applied carefully, because it fails to account for rotations in the embedding. Another important concern related to reproducibility is whether even random noise will yield apparent clusters. This phenomenon has been observed with t-SNE (Wattenberg et al., 2016), and whether other dimensionality reduction techniques are also susceptible to this effect warrants further systematic investigation. However, because these benchmarks are all performed within transcriptomics, further validation is needed to determine whether the conclusions generalize to microbiome data. These measures provide a starting point for evaluating the application of non-linear dimensionality reduction techniques on microbiome data.

Finally, literature from mathematics and computer science that has not been as widely applied to dimensionality reduction in bioinformatics may also be relevant. Of particular interest is the study of distortion, which is applicable when the goal of the embedding is to preserve distances, like one might expect for an exploratory analysis. Similar to the previously described correlation measure, distortion measures summarize the extent to which the distances in high dimensions match the distances in low-dimensions, however, distortion is defined in terms of the expansions and contractions of distances between points. Furthermore, there are many ways to summarize the expansions and contractions, including the worst-case, average-case and local-case, which are all detailed more in (Vankadara and von Luxburg, 2018).

DISCUSSION

The above examples illustrate that dimensionality reduction is an extremely powerful technique that has enhanced a wide range of microbiome studies. However, with great power comes great responsibility. It is unlikely that any one method will excel at representing all datasets, so responsible users of dimensionality reduction should try out several techniques, ideally guided by characteristics of the data rather than as a fishing expedition to see whether any one of many techniques produce results that "look good" (which may even happen in random data for some techniques and parameters) or that fulfill pre-conceived hypotheses and biases. We need standard protocols and software interfaces for choosing the algorithm that suits your data best, rather than the algorithm that shows what you want to see if you squint at it correctly. Methods are needed both for diagnosing the issues that may be most prevalent in your data and affecting your representation, and for rationally choosing among different methods that could be applied to a given dataset. Developing these methods is a key priority for the field.

Dimensionality reduction for the purposes of visualization has somewhat different goals from dimensionality reduction for other purposes and developing a better appreciation of this distinction is important for practice in the field. The goal of dimensionality reduction for visualization is primarily for exploratory overview by human observers (do groups differ from one another, is there overall structure such as gradients in the data). As such, visualization is usually done with three dimensions (more can be examined through parallel plots), while the intrinsic dimensionality of the data may be higher. Visualization is typically only the first step in the data analysis pipeline, and is followed by downstream analysis, such as multivariate analysis/ regression (PERMANOVA, ANOSIM, PERMDISP) either on the original distances or on a dimensionality-reduced version of the data (which can be higher than three dimensions). These results can also be used to motivate supervised differential abundance modeling, such as to determine which groups separate and then determine which microbes are driving these separations.

Dimensionality reduction is thus often an early step in a multi-step pipeline. What downstream analyses is dimensionality reduction a step towards, and how are these accomplished? Feature loadings (i.e. the importance of particular taxa or genes) can be interpreted using log ratios from tools such as DEICODE (Martino et al., 2019), which can then be visualized in Qurro (Fedarko et al., 2020). Classification can be accomplished using machine learning techniques such as random forests, allowing estimates of classifier accuracy and group stability, and also allowing tests of the reusability of these models, e.g. applying a model of human inflammatory bowel disease to dogs (Vázquez-Baeza et al., 2016) or models of aging between different human populations (Huang et al., 2020). A popular strategy is to use a lower-dimensional embedding for traditional statistical analysis, such as using PCA or PCoA coordinates as inputs for regression, classification, clustering, and other analyses. However, as we have seen, many dimensionality reduction methods induce various kinds of artifacts or distortions, and cannot generalize well beyond the data on which the model was initially optimized on, including PCoA, nMDS, RPCA/CTF, and UMAP/t-SNE. Consequently, analyses on these coordinates should be performed with caution. Furthermore, since the parameters and software versions used with these methods have the potential to be highly influential to their results, we recommend that these always be reported for dimensionality reduction methods.

Given the large number of publications that have used dimensionality reduction on microbiome data, we can start to draw conclusions about which dimensionality reduction strategies should be more widely used, and which less widely used. On larger, sparser, compositional datasets, we recommend against the use of conventional PCA, Bray-Curtis and Jaccard dissimilarities, and pseudocounts. Conventional PCA presents the clearest case of a method that should not be used on microbiome data due the sparsity and compositional nature of the data. UniFrac and weighted UniFrac are essentially phylogenetically informed versions of Jaccard and Bray-Curtis beta-diversity metrics respectively. Due to the current default generation of a phylogeny in most 16S and shotgun analyses, there is no reason not to use the phylogenetic counterparts, which have been shown to have better discriminatory power. Pseudocounts should not be used because the choice of pseudocount impacts the lower-dimensional embedding, and there is no clear method for determining which pseudocount value is best.

In contrast, CTF and non-linear methods should be used more in microbiome contexts. As the cost of acquiring microbiome data continues to decrease, experimental designs are getting increasingly complex, and include repeated measures, longitudinal studies, batch effects, etc. We therefore need methods that can determine which biological signals are relevant among all these confounding factors. Additionally, we are increasingly recognizing that many relationships between/ among samples are non-linear. Using non-linear methods can potentially explain more of such datasets with fewer dimensions, although additional benchmarking is required to understand the performance of these methods.

Our analyses suggest some important gaps in the field that could be important areas for future development. There are no dimensionality reduction methods yet that are both able to incorporate phylogeny and are compositionally aware. Several methods, such as Robust PCA and CTF, control for the sparsity, non-normality, compositionality, and are adaptable to specific study-designs of microbiome data but do not incorporate phylogenetic information. In contrast, phylogenetic techniques do not account for sparsity and compositionality, and some also perform poorly with non-normality. A unified method that is appropriate for any microbiome study is therefore still in the future, despite many important recent advances. The ability to perform this task using a generalizable dissimilarity measure would be particularly useful, because it would allow for full utilization of PCoA and non-linear methods including nMDS and UMAP.

Taken together, we conclude that dimensionality reduction is a key part of many, if not most, of the highest-impact microbiome studies performed to date. We can expect this situation to continue into the future, especially as larger study designs and datasets continue to accumulate, and additional method development advances increase the speed and range of applicability of these techniques.

AUTHOR CONTRIBUTIONS

GA, GR, CM, GM, RK contributed to conception of this review. GA, GR, CM, DM, GM, RK wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version. GA and GR contributed equally to this work.

FUNDING

This work was supported in part by grants NSF 2038509, NIH U24CA248454, NIH 1DP1AT010885, NIH U19AG063744, and by CRISP, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA. https://crisp.engineering.virginia.edu/.

REFERENCES

Aitchison, J., and Greenacre, M. (2002). Biplots of Compositional Data. *J. R. Stat. Soc C* 51, 375–392. doi:10.1111/1467-9876.00275

Allaband, C., McDonald, D., Vázquez-Baeza, Y., Minich, J. J., Tripathi, A., Brenner, D. A., et al. (2019). Microbiome 101: Studying, Analyzing, and Interpreting Gut Microbiome Data for Clinicians. *Clin. Gastroenterol. Hepatol.* 17, 218–230. doi:10.1016/j.cgh.2018.09.017

Amir, A., McDonald, D., Navas-Molina, J. A., Kopylova, E., Morton, J. T., Zech Xu, Z., et al. (2017). Deblur Rapidly Resolves Single-Nucleotide Community Sequence Patterns. *mSystems* 2.doi:10.1128/mSystems.00191-16

Anderson, M. J. (2017). *Permutational Multivariate Analysis of Variance (PERMANOVA).* Wiley StatsRef: Statistics Reference Online, 1–15. doi:10.1002/9781118445112.stat07841

Arfken, A., Song, B., Bowman, J. S., and Piehler, M. (2017). Denitrification Potential of the Eastern Oyster Microbiome Using a 16S rRNA Gene Based Metabolic Inference Approach. *PLoS One* 12, e0185071. doi:10.1371/journal.pone.0185071

Armstrong, G., Martino, C., Rahman, G., Gonzalez, A., Vázquez-Baeza, Y., Mishne, G., et al. (2021). Uniform Manifold Approximation and Projection (UMAP) Reveals Composite Patterns and Resolves Visualization Artifacts in Microbiome Data. *mSystems* 6, e0069121. doi:10.1128/mSystems.00691-21

Bali, P., Coker, J., Lozano-Pope, I., Zengler, K., and Obonyo, M. (2021). Microbiome Signatures in a Fast- and Slow-Progressing Gastric Cancer Murine Model and Their Contribution to Gastric Carcinogenesis. *Microorganisms* 9, 189. doi:10.3390/microorganisms9010189

Barker, M., and Rayens, W. (2003). Partial Least Squares for Discrimination. *J. Chemometrics* 17, 166–173. doi:10.1002/cem.785

Becht, E., McInnes, L., Healy, J., Dutertre, C.-A., Kwok, I. W. H., Ng, L. G., et al. (2019). Dimensionality Reduction for Visualizing Single-Cell Data Using UMAP. *Nat. Biotechnol.* 37, 38–44. doi:10.1038/nbt.4314

Belkin, M., and Niyogi, P. (2002). "Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering," in *NIPS'01: Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*. doi:10.7551/mitpress/1120.003.0080

Benitez, M. S., Osborne, S. L., and Lehman, R. M. (2017). Previous Crop and Rotation History Effects on maize Seedling Health and Associated Rhizosphere Microbiome. *Sci. Rep.* 7, 15709. doi:10.1038/s41598-017-15955-9

Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, C. C., Al-Ghalith, G. A., et al. (2019). Reproducible, Interactive, Scalable and Extensible Microbiome Data Science Using QIIME 2. *Nat. Biotechnol.* 37, 852–857. doi:10.1038/s41587-019-0209-9

Callahan, B. J., McMurdie, P. J., and Holmes, S. P. (2017). Exact Sequence Variants Should Replace Operational Taxonomic Units in Marker-Gene Data Analysis. *ISME J.* 11, 2639–2643. doi:10.1038/ismej.2017.119

Campbell, T. P., Sun, X., Patel, V. H., Sanz, C., Morgan, D., and Dantas, G. (2020). The Microbiome and Resistome of Chimpanzees, Gorillas, and Humans across Host Lifestyle and Geography. *ISME J.* 14, 1584–1599. doi:10.1038/s41396-020-0634-2

Caporaso, J. G., Lauber, C. L., Walters, W. A., Berg-Lyons, D., Lozupone, C. A., Turnbaugh, P. J., et al. (2011). Global Patterns of 16S rRNA Diversity at a Depth of Millions of Sequences Per Sample. *Proc. Natl. Acad. Sci. U. S. A.* 108 (Suppl. 1), 4516–4522. doi:10.1073/pnas.1000080107

Castaño-Rodríguez, N., Goh, K. L., Fock, K. M., Mitchell, H. M., and Kaakoush, N. O. (2017). Dysbiosis of the Microbiome in Gastric Carcinogenesis. *Sci. Rep.* 7, 15957. doi:10.1038/s41598-017-16289-2

Chang, Q., Luan, Y., and Sun, F. (2011). Variance Adjusted Weighted UniFrac: a Powerful Beta Diversity Measure for Comparing Communities Based on Phylogeny. *BMC Bioinformatics* 12, 118. doi:10.1186/1471-2105-12-118

Chen, J., Bittinger, K., Charlson, E. S., Hoffmann, C., Lewis, J., Wu, G. D., et al. (2012). Associating Microbiome Composition with Environmental Covariates Using Generalized UniFrac Distances. *Bioinformatics* 28, 2106–2113. doi:10.1093/bioinformatics/bts342

Clarke, K., and Ainsworth, M. (1993). A Method of Linking Multivariate Community Structure to Environmental Variables. *Mar. Ecol. Prog. Ser.* 92, 205–219. doi:10.3354/meps092205

Das, T., Jayasudha, R., Chakravarthy, S., Prashanthi, G. S., Bhargava, A., Tyagi, M., et al. (2021). Alterations in the Gut Bacterial Microbiome in People with Type 2 Diabetes Mellitus and Diabetic Retinopathy. *Sci. Rep.* 11, 2738. doi:10.1038/s41598-021-82538-0

David, L. A., Maurice, C. F., Carmody, R. N., Gootenberg, D. B., Button, J. E., Wolfe, B. E., et al. (2013). Diet Rapidly and Reproducibly Alters the Human Gut Microbiome. *Nature* 505, 559–563. doi:10.1038/nature12820

Debelius, J., Song, S. J., Vazquez-Baeza, Y., Xu, Z. Z., Gonzalez, A., and Knight, R. (2016). Tiny Microbes, Enormous Impacts: what Matters in Gut Microbiome Studies? *Genome Biol.* 17, 217. doi:10.1186/s13059-016-1086-x

Dinleyici, E. C., Martínez-Martínez, D., Kara, A., Karbuz, A., Dalgic, N., Metin, O., et al. (2018). Time Series Analysis of the Microbiota of Children Suffering from Acute Infectious Diarrhea and Their Recovery after Treatment. *Front. Microbiol.* 9, 1230. doi:10.3389/fmicb.2018.01230

Fedarko, M. W., Martino, C., Morton, J. T., González, A., Rahman, G., Marotz, C. A., et al. (2020). Visualizing 'omic Feature Rankings and Log-Ratios Using Qurro. *NAR Genom Bioinform* 2, lqaa023. doi:10.1093/nargab/lqaa023

Fierer, N., Lauber, C. L., Zhou, N., McDonald, D., Costello, E. K., and Knight, R. (2010). Forensic Identification Using Skin Bacterial Communities. *Proc. Natl. Acad. Sci. U. S. A.* 107, 6477–6481. doi:10.1073/pnas.1000162107

Galloway-Peña, J., and Hanson, B. (2020). Tools for Analysis of the Microbiome. *Dig. Dis. Sci.* 65, 674–685. doi:10.1007/s10620-020-06091-y

Galvão, K. N., Higgins, C. H., Zinicola, M., Jeon, S. J., Korzec, H., and Bicalho, R. C. (2019). Effect of Pegbovigrastim Administration on the Microbiome Found in the Vagina of Cows Postpartum. *J. Dairy Sci.* 102, 3439–3451. doi:10.3168/jds.2018-15783

Ginter, J. L., and Thorndike, R. M. (1979). Correlational Procedures for Research. *J. Marketing Res.* 16, 600. doi:10.2307/3150840

Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., and Egozcue, J. J. (2017). Microbiome Datasets Are Compositional: And This Is Not Optional. *Front. Microbiol.* 8. doi:10.3389/fmicb.2017.02224

Goloshchapov, O. V., Olekhnovich, E. I., Sidorenko, S. V., Moiseev, I. S., Kucher, M. A., Fedorov, D. E., et al. (2019). Long-term Impact of Fecal Transplantation in Healthy Volunteers. *BMC Microbiol.* 19, 312. doi:10.1186/s12866-019-1689-y

Gonzalez, A., Navas-Molina, J. A., Kosciolek, T., McDonald, D., Vázquez-Baeza, Y., Ackermann, G., et al. (2018). Qiita: Rapid, Web-Enabled Microbiome Meta-Analysis. *Nat. Methods* 15, 796–798. doi:10.1038/s41592-018-0141-9

Greig-Smith, P. (1980). The Development of Numerical Classification and Ordination. *Vegetatio* 42, 1–9. doi:10.1007/bf00048864

Halfvarson, J., Brislawn, C. J., Lamendella, R., Vázquez-Baeza, Y., Walters, W. A., Bramer, L. M., et al. (2017). Dynamics of the Human Gut Microbiome in Inflammatory Bowel Disease. *Nat. Microbiol.* 2, 17004. doi:10.1038/nmicrobiol.2017.4

Hamady, M., and Knight, R. (2009). Microbial Community Profiling for Human Microbiome Projects: Tools, Techniques, and Challenges. *Genome Res.* 19, 1141–1152. doi:10.1101/gr.085464.108

Huang, S., Haiminen, N., Carrieri, A. P., Hu, R., Jiang, L., Parida, L., et al. (2020). Human Skin, Oral, and Gut Microbiomes Predict Chronological Age. *mSystems* 5, e00630–19. doi:10.1128/mSystems.00630-19

Ingham, A. C., Kielsen, K., Cilieborg, M. S., Lund, O., Holmes, S., Aarestrup, F. M., et al. (2019). Specific Gut Microbiome Members Are Associated with Distinct Immune Markers in Pediatric Allogeneic Hematopoietic Stem Cell Transplantation. *Microbiome* 7, 131. doi:10.1186/s40168-019-0745-z

Keegan, K. P., Glass, E. M., and Meyer, F. (2016). MG-RAST, a Metagenomics Service for Analysis of Microbial Community Structure and Function. *Methods Mol. Biol.* 1399, 207–233. doi:10.1007/978-1-4939-3369-3_13

Kobak, D., and Berens, P. (2019). The Art of Using T-SNE for Single-Cell Transcriptomics. *Nat. Commun.* 10, 5416. doi:10.1038/s41467-019-13056-x

Kobak, D., and Linderman, G. C. (2021). Initialization Is Critical for Preserving Global Data Structure in Both T-SNE and UMAP. *Nat. Biotechnol.* 39, 156–157. doi:10.1038/s41587-020-00809-z

Kohli, D., Cloninger, A., and Mishne, G. (2021). LDLE: Low Distortion Local Eigenmaps. *J. Mach. Learn. Res.* 22, 1–64. Available at: https://arxiv.org/abs/2101.11055.

Kruskal, J. B. (1964). Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis. *Psychometrika* 29, 1–27. doi:10.1007/bf02289565

Kruskal, J., and Wish, M. (1978). *Multidimensional Scaling*. Thousand Oaks, CA: SAGE Publications, Inc.. doi:10.4135/9781412985130

Kuczynski, J., Lauber, C. L., Walters, W. A., Parfrey, L. W., Clemente, J. C., Gevers, D., et al. (2011). Experimental and Analytical Tools for Studying the Human Microbiome. *Nat. Rev. Genet.* 13, 47–58. doi:10.1038/nrg3129

Kuczynski, J., Liu, Z., Lozupone, C., McDonald, D., Fierer, N., and Knight, R. (2010). Microbial Community Resemblance Methods Differ in Their Ability to Detect Biologically Relevant Patterns. *Nat. Methods* 7, 813–819. doi:10.1038/nmeth.1499

Kumar, M. S., Slud, E. V., Okrah, K., Hicks, S. C., Hannenhalli, S., and Corrada Bravo, H. (2018). Analysis and Correction of Compositional Bias in Sparse Sequencing Count Data. *BMC Genomics* 19, 799. doi:10.1186/s12864-018-5160-5

Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D. J., Hicks, S. C., Robinson, M. D., et al. (2020). Eleven Grand Challenges in Single-Cell Data Science. *Genome Biol.* 21, 31–35. doi:10.1186/s13059-020-1926-6

Lang, J. M., Pan, C., Cantor, R. M., Tang, W. H. W., Garcia-Garcia, J. C., Kurtz, I., et al. (2018). Impact of Individual Traits, Saturated Fat, and Protein Source on the Gut Microbiome. *MBio* 9, e01604-18. doi:10.1128/mBio.01604-18

Lauber, C. L., Hamady, M., Knight, R., and Fierer, N. (2009). Pyrosequencing-based Assessment of Soil pH as a Predictor of Soil Bacterial Community Structure at the continental Scale. *Appl. Environ. Microbiol.* 75, 5111–5120. doi:10.1128/AEM.00335-09

Lee, D. D., and Seung, H. S. (1999). Learning the Parts of Objects by Non-negative Matrix Factorization. *Nature* 401, 788–791. doi:10.1038/44565

Ley, R. E., Lozupone, C. A., Hamady, M., Knight, R., and Gordon, J. I. (2008). Worlds within Worlds: Evolution of the Vertebrate Gut Microbiota. *Nat. Rev. Microbiol.* 6, 776–788. doi:10.1038/nrmicro1978

Lin, H., and Peddada, S. D. (2020). Analysis of Microbial Compositions: a Review of Normalization and Differential Abundance Analysis. *NPJ Biofilms Microbiomes* 6, 60. doi:10.1038/s41522-020-00160-w

Linderman, G. C., Rachh, M., Hoskins, J. G., Steinerberger, S., and Kluger, Y. (2019). Fast Interpolation-Based T-SNE for Improved Visualization of Single-Cell RNA-Seq Data. *Nat. Methods* 16, 243–245. doi:10.1038/s41592-018-0308-4

Lloréns-Rico, V., Gregory, A. C., Van Weyenbergh, J., Jansen, S., Van Buyten, T., Qian, J., et al. (2021). Clinical Practices Underlie COVID-19 Patient Respiratory Microbiome Composition and its Interactions with the Host. *Nat. Commun.* 12, 6243. doi:10.1038/s41467-021-26500-8

Lozupone, C., and Knight, R. (2005). UniFrac: a New Phylogenetic Method for Comparing Microbial Communities. *Appl. Environ. Microbiol.* 71, 8228–8235. doi:10.1128/AEM.71.12.8228-8235.2005

Lozupone, C. A., Hamady, M., Kelley, S. T., and Knight, R. (2007). Quantitative and Qualitative Beta Diversity Measures lead to Different Insights into Factors that Structure Microbial Communities. *Appl. Environ. Microbiol.* 73, 1576–1585. doi:10.1128/AEM.01996-06

Lozupone, C. A., and Knight, R. (2007). Global Patterns in Bacterial Diversity. *Proc. Natl. Acad. Sci. U. S. A.* 104, 11436–11440. doi:10.1073/pnas.0611525104

Malard, L. A., Anwar, M. Z., Jacobsen, C. S., and Pearce, D. A. (2019). Biogeographical Patterns in Soil Bacterial Communities across the Arctic Region. *FEMS Microbiol. Ecol.* 95, fiz128. doi:10.1093/femsec/fiz128

Mandal, S., Van Treuren, W., White, R. A., Eggesbø, M., Knight, R., and Peddada, S. D. (2015). Analysis of Composition of Microbiomes: a Novel Method for Studying Microbial Composition. *Microb. Ecol. Health Dis.* 26, 27663. doi:10.3402/mehd.v26.27663

Marshall, I. P. G., Ren, G., Jaussi, M., Lomstein, B. A., Jørgensen, B. B., Røy, H., et al. (2019). Environmental Filtering Determines Family-Level Structure of Sulfate-Reducing Microbial Communities in Subsurface marine Sediments. *ISME J.* 13, 1920–1932. doi:10.1038/s41396-019-0387-y

Martín-Fernández, J. A., Barceló-Vidal, C., and Pawlowsky-Glahn, V. (2003). Dealing with Zeros and Missing Values in Compositional Data Sets Using Nonparametric Imputation. *Math. Geol.* 35, 253–278. doi:10.1023/A:1023866030544

Martino, C., Morton, J. T., Marotz, C. A., Thompson, L. R., Tripathi, A., Knight, R., et al. (2019). A Novel Sparse Compositional Technique Reveals Microbial Perturbations. *mSystems* 4, e00016-19. doi:10.1128/mSystems.00016-19

Martino, C., Shenhav, L., Marotz, C. A., Armstrong, G., McDonald, D., Vázquez-Baeza, Y., et al. (2021). Context-aware Dimensionality Reduction Deconvolutes Gut Microbial Community Dynamics. *Nat. Biotechnol.* 39, 165–168. doi:10.1038/s41587-020-0660-7

McDonald, D., Clemente, J. C., Kuczynski, J., Rideout, J. R., Stombaugh, J., Wendel, D., et al. (2012). The Biological Observation Matrix (BIOM) Format or: How I Learned to Stop Worrying and Love the Ome-Ome. *Gigascience* 1, 7. doi:10.1186/2047-217X-1-7

McInnes, L., Healy, J., and Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. Available at: http://arxiv.org/abs/1802.03426 (Accessed November 21, 2021).

McMurdie, P. J., and Holmes, S. (2013). Phyloseq: an R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. *PLoS One* 8, e61217. doi:10.1371/journal.pone.0061217

Metcalf, J. L., Song, S. J., Morton, J. T., Weiss, S., Seguin-Orlando, A., Joly, F., et al. (2017). Evaluating the Impact of Domestication and Captivity on the Horse Gut Microbiome. *Sci. Rep.* 7, 15497. doi:10.1038/s41598-017-15375-9

Metcalf, J. L., Xu, Z. Z., Weiss, S., Lax, S., Van Treuren, W., Hyde, E. R., et al. (2016). Microbial Community Assembly and Metabolic Function during Mammalian Corpse Decomposition. *Science* 351, 158–162. doi:10.1126/science.aad2646

Morton, J. T., Marotz, C., Washburne, A., Silverman, J., Zaramela, L. S., Edlund, A., et al. (2019). Establishing Microbial Composition Measurement Standards with Reference Frames. *Nat. Commun.* 10, 2719. doi:10.1038/s41467-019-10656-5

Morton, J. T., Toran, L., Edlund, A., Metcalf, J. L., Lauber, C., and Knight, R. (2017). Uncovering the Horseshoe Effect in Microbial Analyses. *mSystems* 2, e00166-16. doi:10.1128/mSystems.00166-16

Ocasio, J., Babcock, B., Malawsky, D., Weir, S. J., Loo, L., Simon, J. M., et al. (2019). scRNA-Seq in Medulloblastoma Shows Cellular Heterogeneity and Lineage Expansion Support Resistance to SHH Inhibitor Therapy. *Nat. Commun.* 10, 5829. doi:10.1038/s41467-019-13657-6

Paliy, O., and Shankar, V. (2016). Application of Multivariate Statistical Techniques in Microbial Ecology. *Mol. Ecol.* 25, 1032–1057. doi:10.1111/mec.13536

Parbie, P. K., Mizutani, T., Ishizaka, A., Kawana-Tachikawa, A., Runtuwene, L. R., Seki, S., et al. (2021). Dysbiotic Fecal Microbiome in HIV-1 Infected Individuals in Ghana. *Front. Cel. Infect. Microbiol.* 11, 646467. doi:10.3389/fcimb.2021.646467

Pawlowsky-Glahn, V., and Buccianti, A. (2011). *Compositional Data Analysis: Theory and Applications*. Hoboken, NJ: John Wiley & Sons.

Pérez-Jaramillo, J. E., Carrión, V. J., Bosse, M., Ferrão, L. F. V., de Hollander, M., Garcia, A. A. F., et al. (2017). Linking Rhizosphere Microbiome Composition of Wild and Domesticated Phaseolus vulgaris to Genotypic and Root Phenotypic Traits. *ISME J.* 11, 2244–2257. doi:10.1038/ismej.2017.85

Pielou, E. C. (1966). The Measurement of Diversity in Different Types of Biological Collections. *J. Theor. Biol.* 13, 131–144. doi:10.1016/0022-5193(66)90013-0

Podani, J., and Miklós, I. (2002). Resemblance Coefficients and the Horseshoe Effect in Principal Coordinates Analysis. *Ecology* 83, 3331–3343. doi:10.1890/0012-9658(2002)083[3331:rcathe]2.0.co;2

Potvin, C., and Roff, D. A. (1993). Distribution-Free and Robust Statistical Methods: Viable Alternatives to Parametric Statistics. *Ecology* 74, 1617–1628. doi:10.2307/1939920

Ren, B., Bacallado, S., Favaro, S., Holmes, S., and Trippa, L. (2017). Bayesian Nonparametric Ordination for the Analysis of Microbial Communities. *J. Am. Stat. Assoc.* 112, 1430–1442. doi:10.1080/01621459.2017.1288631

Roweis, S. T., and Saul, L. K. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 290, 2323–2326. doi:10.1126/science.290.5500.2323

Ruiz-Perez, D., Guan, H., Madhivanan, P., Mathee, K., and Narasimhan, G. (2020). So You Think You Can PLS-DA? *BMC Bioinformatics* 21, 2–10. doi:10.1186/s12859-019-3310-7

Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, E. B., et al. (2009). Introducing Mothur: Open-Source, Platform-independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Appl. Environ. Microbiol.* 75, 7537–7541. doi:10.1128/AEM.01541-09

Scholkopf, B., Smola, A., and Müller, K.-R. (1999). "Kernel Principal Component Analysis," in *Advances in Kernel Methods - Support Vector Learning* (Cambridge, MA: MIT Press). Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.128.7613 (Accessed November 23, 2021).

Shalapour, S., Lin, X. J., Bastian, I. N., Brain, J., Burt, A. D., Aksenov, A. A., et al. (2017). Inflammation-induced IgA+ Cells Dismantle Anti-liver Cancer Immunity. *Nature* 551, 340–345. doi:10.1038/nature24302

Shankar, V., Agans, R., and Paliy, O. (2017). Advantages of Phylogenetic Distance Based Constrained Ordination Analyses for the Examination of Microbial Communities. *Sci. Rep.* 7, 6481. doi:10.1038/s41598-017-06693-z

Shi, Y., Zhang, L., Peterson, C., Do, K.-A., and Jenq, R. (2022). Performance Determinants of Unsupervised Clustering Methods for Microbiome Data. *Microbiome* 10, 25. doi:10.1186/s40168-021-01199-3

Silverman, J. D., Roche, K., Mukherjee, S., and David, L. A. (2020). Naught All Zeros in Sequence Count Data Are the Same. *Comput. Struct. Biotechnol. J.* 18, 2789–2798. doi:10.1016/j.csbj.2020.09.014

Song, S. J., Amir, A., Metcalf, J. L., Amato, K. R., Xu, Z. Z., Humphrey, G., et al. (2016). Preservation Methods Differ in Fecal Microbiome Stability, Affecting Suitability for Field Studies. *mSystems* 1. doi:10.1128/mSystems.00021-16

Song, S. J., Wang, J., Martino, C., Jiang, L., Thompson, W. K., Shenhav, L., et al. (2021). Naturalization of the Microbiota Developmental Trajectory of Cesarean-Born Neonates after Vaginal Seeding. *Med* 2, 951–964. e5. doi:10.1016/j.medj.2021.05.003

Souza, F. F. C., Mathai, P. P., Pauliquevis, T., Balsanelli, E., Pedrosa, F. O., Souza, E. M., et al. (2021). Influence of Seasonality on the Aerosol Microbiome of the Amazon Rainforest. *Sci. Total Environ.* 760, 144092. doi:10.1016/j.scitotenv.2020.144092

Sunagawa, S., Coelho, L. P., Chaffron, S., Kultima, J. R., Labadie, K., Salazar, G., et al. (2015). Ocean Plankton. Structure and Function of the Global Ocean Microbiome. *Science* 348, 1261359. doi:10.1126/science.1261359

Taavitsainen, S., Engedal, N., Cao, S., Handle, F., Erickson, A., Prekovic, S., et al. (2021). Single-cell ATAC and RNA Sequencing Reveal Pre-existing and Persistent Cells Associated with Prostate Cancer Relapse. *Nat. Commun.* 12, 5307–5316. doi:10.1038/s41467-021-25624-1

Tabachnick, B. G., and Fidell, L. S. (2013). *Using Multivariate Statistics*. Boston, MA: Pearson.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* 290, 2319–2323. doi:10.1126/science.290.5500.2319

ter Braak, C. J. F. (1986). Canonical Correspondence Analysis: A New Eigenvector Technique for Multivariate Direct Gradient Analysis. *Ecology* 67 (5), 1167–1179. doi:10.2307/1938672

The Human Microbiome Project Consortium (2012). Structure, Function and Diversity of the Healthy Human Microbiome. *Nature* 486, 207–214. doi:10.1038/nature11234

Thompson, L. R., Sanders, J. G., McDonald, D., Amir, A., Ladau, J., Locey, K. J., et al. (2017). A Communal Catalogue Reveals Earth's Multiscale Microbial Diversity. *Nature* 551, 457–463. doi:10.1038/nature24621

Turnbaugh, P. J., Ley, R. E., Hamady, M., Fraser-Liggett, C. M., Knight, R., and Gordon, J. I. (2007). The Human Microbiome Project. *Nature* 449, 804–810. doi:10.1038/nature06244

van der Maaten, L., and Hinton, G. (2008). Visualizing Data Using T-SNE. *J. Mach. Learn. Res.* 9, 2579–2605.

Vangay, P., Johnson, A. J., Ward, T. L., Al-Ghalith, G. A., Shields-Cutler, R. R., Hillmann, B. M., et al. (2018). US Immigration Westernizes the Human Gut Microbiome. *Cell* 175, 962–e10. doi:10.1016/j.cell.2018.10.029

Vankadara, L. C., and von Luxburg, U. (2018). Measures of Distortion for Machine Learning. *Adv. Neural Inf. Process. Syst.* 31. Available at: https://proceedings.neurips.cc/paper/2018/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf (Accessed November 20, 2021).

Vázquez-Baeza, Y., Gonzalez, A., Smarr, L., McDonald, D., Morton, J. T., Navas-Molina, J. A., et al. (2017). Bringing the Dynamic Microbiome to Life with Animations. *Cell Host Microbe* 21, 7–10. doi:10.1016/j.chom.2016.12.009

Vázquez-Baeza, Y., Hyde, E. R., Suchodolski, J. S., and Knight, R. (2016). Dog and Human Inflammatory Bowel Disease Rely on Overlapping yet Distinct Dysbiosis Networks. *Nat. Microbiol.* 1, 16177. doi:10.1038/nmicrobiol.2016.177

Wattenberg, M., Viégas, F., and Johnson, I. (2016). How to Use T-SNE Effectively. *Distill* 1, e2. doi:10.23915/distill.00002

Weiss, S., Xu, Z. Z., Peddada, S., Amir, A., Bittinger, K., Gonzalez, A., et al. (2017). Normalization and Microbial Differential Abundance Strategies Depend upon Data Characteristics. *Microbiome* 5, 27–18. doi:10.1186/s40168-017-0237-y

Wong, R. G., Wu, J. R., and Gloor, G. B. (2016). Expanding the UniFrac Toolbox. *PLoS One* 11, e0161196. doi:10.1371/journal.pone.0161196

Wu, G. D., Chen, J., Hoffmann, C., Bittinger, K., Chen, Y. Y., Keilbaugh, S. A., et al. (2011). Linking Long-Term Dietary Patterns with Gut Microbial Enterotypes. *Science* 334, 105–108. doi:10.1126/science.1208344

Xu, T., Demmer, R. T., and Li, G. (2021). Zero-inflated Poisson Factor Model with Application to Microbiome Read Counts. *Biometrics* 77, 91–101. doi:10.1111/biom.13272

Xu, X., Xie, Z., Yang, Z., Li, D., and Xu, X. (2020). A T-SNE Based Classification Approach to Compositional Microbiome Data. *Front. Genet.* 11, 620143. doi:10.3389/fgene.2020.620143

Yatsunenko, T., Rey, F. E., Manary, M. J., Trehan, I., Dominguez-Bello, M. G., Contreras, M., et al. (2012). Human Gut Microbiome Viewed across Age and Geography. *Nature* 486, 222–227. doi:10.1038/nature11053

Young, C., Wood, H. M., Seshadri, R. A., Van Nang, P., Vaccaro, C., Melendez, L. C., et al. (2021). The Colorectal Cancer-Associated Faecal Microbiome of Developing Countries Resembles that of Developed Countries. *Genome Med.* 13, 27–13. doi:10.1186/s13073-021-00844-8

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Armstrong, Rahman, Martino, McDonald, Gonzalez, Mishne and Knight. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

BRIEF RESEARCH REPORT
published: 11 April 2022
doi: 10.3389/fbinf.2022.866850

Improved Mobilome Delineation in Fragmented Genomes

Catherine M. Mageeney[1†], *Gareth Trubl*[2†] and *Kelly P. Williams*[1*]

[1]*Systems Biology Department, Sandia National Laboratories, Livermore, CA, United States,* [2]*Physical and Life Sciences Directorate, Lawrence Livermore National Laboratory, Livermore, CA, United States*

The mobilome of a microbe, i.e., its set of mobile elements, has major effects on its ecology, and is important to delineate properly in each genome. This becomes more challenging for incomplete genomes, and even more so for metagenome-assembled genomes (MAGs), where misbinning of scaffolds and other losses can occur. Genomic islands (GIs), which integrate into the host chromosome, are a major component of the mobilome. Our GI-detection software TIGER, unique in its precise mapping of GI termini, was applied to 74,561 genomes from 2,473 microbial species, each species containing at least one MAG and one isolate genome. A species-normalized deficit of ~1.6 GIs/genome was measured for MAGs relative to isolates. To test whether this undercount was due to the higher fragmentation of MAG genomes, TIGER was updated to enable detection of split GIs whose termini are on separate scaffolds or that wrap around the origin of a circular replicon. This doubled GI yields, and the new split GIs matched the quality of single-scaffold GIs, except that highly fragmented GIs may lack central portions. Cross-scaffold search is an important upgrade to GI detection as fragmented genomes increasingly dominate public databases. TIGER2 better captures MAG microdiversity, recovering niche-defining GIs and supporting microbiome research aims such as virus-host linking and ecological assessment.

Keywords: metagenome-assembled genome mobile genetic element, genomic island, prophage, metagenomics, metagenomeassembled genome

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Alessandro M. Varani,
São Paulo State University, Brazil
Antonio Camilo Da Silva Filho,
Federal University of Paraná, Brazil

****Correspondence:***
Kelly P. Williams
kpwilli@sandia.gov

†These authors have contributed equally to this work

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *31 January 2022*
Accepted: *17 March 2022*
Published: *11 April 2022*

Citation:
Mageeney CM, Trubl G and Williams KP (2022) Improved Mobilome Delineation in Fragmented Genomes. Front. Bioinform. 2:866850. doi: 10.3389/fbinf.2022.866850

INTRODUCTION

The mobilome is the collection of mobile genetic elements (MGEs), such as transposable elements, plasmids, and prophages, present in a genome. Aside from selfish genes for propagation, an MGE can carry cargo genes that benefit the host organism, for example by promoting catabolism of organic pollutants (van der Meer and Sentchilo, 2003), nitrogen fixation (Sullivan and Ronson, 1998) or biofilm formation (Drenkard and Ausubel, 2002). Acquisition of a new cargo-bearing MGE can quickly and profoundly alter the phenotype of the host microbe. Therefore to understand the evolution and ecological role of microbes, it is important to delineate their mobilomes. If the genome is complete and closed, plasmids are automatically identified as isolated replicons, but precise identification of those MGEs that lie integrated within the chromosome is more challenging. The fragmentation accompanying incomplete genomes, typical of metagenome-assembled genomes (MAGs), further increases the challenge of identifying MGEs.

Genomic islands (GIs) are a subclass of MGEs that integrate into microbial chromosomes, usually with high specificity for a particular chromosomal site (*attB*), determined by the GI-encoded integrase. They range from ~5 to hundreds of kbp and carry genes of diverse function. GIs can be horizontally transferred *via* conjugation, transformation or transduction, with mobility heavily influenced by other MGEs (Bertelli et al., 2019). Some GIs carry a gene set revealing the mode of transfer between microbes, either bearing conjugative genes that indicate an integrative and conjugative element (ICE), or viral genes that indicate a prophage, i.e., a temperate phage in the lysogenic phase of its life cycle. Other GIs are satellites, which do not carry their own transfer genes but require a helper, itself either an ICE or phage, to supply gene products promoting transfer (Fillol-Salom et al., 2018).

There are several computational GI prediction tools [reviewed in (Bertelli et al., 2019)] that exploit special GI features, such as sporadic occurrence within a species, differences from the nucleotide sequence composition of the chromosome, preference for tRNA genes, and gene content. Our methods Islander and TIGER are unique in their precise mapping of GIs (Hudson et al., 2015; Mageeney et al., 2020). Precise GI mapping improves genome annotation and allows discoveries of new *attB* site-specificity by integrases, site-promiscuous integrase clades, and cases where cells use GIs to regulate gene integrity.

The advent of metagenomics has reshaped our understanding of uncultured microbes and microbial communities. Early metagenomics provided mere gene catalogs of environmental samples, but the field has turned toward genome-centric characterization, as read-depth coverage and bioinformatic tools improved sufficiently to enable coverage-based binning of assembled scaffolds into population genomes or MAGs (Taş et al., 2021). Characterization of MAGs has revealed that high proportions of bacteria and archaea remain uncultured (Steen et al., 2019) and that most metagenomic reads do not map to any MAG or isolate genome (Nayfach et al., 2021).

MAGs are lower quality than same-species isolate genomes by every available metric (**Supplementary Table S1**). Some of the factors contributing to reduced MAG quality are similar to those that may plague any genome project: low coverage that can break or leave gaps in the assembly, and outright misassembly. The key feature distinguishing a metagenomic DNA sample from an isolate DNA sample is complexity. One way complexity manifests is through different levels of coverage for different microbes, exacerbating the low coverage problem for some MAGs in a metagenome. Complexity can also manifest as microdiversity, where a group of population-level variants exist in the sample. Resolution of multiple individual MAGs from the same microdiverse population is often impossible but has been achieved when species diversity is low (Tyson et al., 2004) or complexity is reduced (Sieradzki et al., 2020; Haro-Moreno et al., 2021; Nicolas et al., 2021). More often a single consensus MAG can be obtained for a population with moderate microdiversity, but high microdiversity can counteract assembly, perhaps leaving the more diverse genomic regions unassembled and reducing the completeness of the MAG. Finally, a problem unique to metagenomes can occur post-assembly, at the binning step (Evans and Denef, 2020). Shared nucleotide sequence composition of scaffolds is a major basis for binning, such that genomic regions departing from baseline composition can be misbinned, generating artifactual composite MAGs (Shaiber and Eren, 2019). We have observed cross-domain misbinning, where scaffolds with uniquely bacterial markers are mixed into archaeal MAGs (unpublished results).

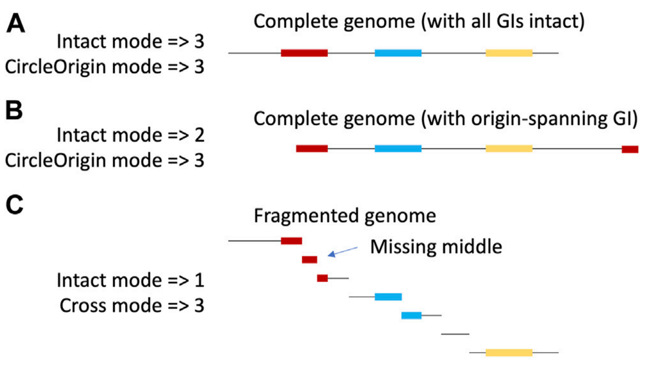


FIGURE 1 | New TIGER modes. The same circular chromosome with 3 (colored) GIs is shown with a complete **(A,B)** or fragmented assembly **(C)**. With complete assembly, if the origin of the linearized sequence of the circle is randomly chosen, it will occasionally fall within a GI, splitting the GI **(B)**. Yields are shown for the various TIGER modes. The original mode can only find intact GIs on a single scaffold, while the new modes, CircleOrigin (applied to complete assemblies) and Cross (applied to fragmented assemblies), can additionally find the split islands. Because TIGER focuses on GI-flanking sequences, the Cross-mode call for a multiply split GI (red in panel C) will only include the terminal fragments and exclude middle GI fragments.

There has been relatively little emphasis in the literature on the problems that metagenomic datasets pose for mobilome delineation. Scaffolds from within MGEs are more prone to misbinning because they can strongly differ in composition from their surrounding chromosomes (Carr et al., 2020; Maguire et al., 2020). MGEs tend to have higher microdiversity than chromosomal regions because MGE gene expression is largely repressed, reducing selective pressure to preserve MGE nucleotide sequence (Haro-Moreno et al., 2021). Finally, induction of a GI, i.e., its excision, circularization and possible replication in some cells within a MAG population, can confuse assemblers. We have observed such assembler confusion caused by inadvertent GI induction in isolate assemblies (unpublished results). Alternative GIs at the same genomic site is another formal possibility for a type of diversity that could affect assembly of MAGs. MGEs are not included in the assessment of MAG quality (Bowers et al., 2017); a MAG may thus be considered high quality, yet still be missing extensive portions of its mobilome.

Here, we present TIGER2, with new modes to identify GIs either across two contigs or around the circular origin of a chromosome (**Figure 1**), doubling average GI yields.

MATERIALS AND METHODS

Genomes. We collected a set of 74,561 genomes (for 7978 MAGS and 66,583 isolates) from 2,473 microbial (64 archaeal, 2,409 bacterial) species, where each species contained at least one MAG and one isolate genome (**Supplementary Table S2**). We downloaded 288,451 microbial genomes from GenBank in July 2019, after rejecting additional genomes with N50 < 10,000 or scaffold count >300. A script speciate. pl was developed employing MASH and fastANI that placed all but 1,656 of the GenBank genomes into a species defined by GTDB release 202 (Parks et al., 2022); for the 173,660 GenBank assembly IDs that had been treated by GTDB, which applies its own genome quality filters, the script mismatched the GTDB assignment in only 184 rejected cases, at least some due to major differences between versions of the assemblies. Among the 47,894 GTDB species, 2,487 were found to contain at least one MAG and one isolate genome. All remaining MAG genomes for these species, and many remaining isolate genomes (up to 200 total per species unless more were already available) were collected. Fourteen two-genome species were rejected in which the two genomes had identical scaffold size lists, suggesting duplicate entries.

TIGER version 2. TIGER was originally designed to map intact GIs present on a single scaffold. We re-wrote the core software to offer two new “split” modes that yield split GIs, in addition to the intact GIs (**Figure 1**). “CircleOrigin” mode finds split GIs that wrap around the origin of a circular replicon. “Cross” mode detects split GIs with termini on separate scaffolds. We applied CircleOrigin mode to the 9 008 genomes we considered complete (in five or fewer parts, to accommodate plasmids and secondary chromosomes), and applied Cross mode to the 65,553 remaining, fragmented genomes. To accommodate the new split GIs, the main TIGER wrapper and the merge. pl script that produces a tentative file of nonoverlapping GI calls were also revised, but we have not yet revised the orthogonal software Islander nor the resolve. pl script that compares Islander/TIGER calls and treats tandem GI arrays. New software is available at github/sandialabs/TIGER.

Genomic islands. TIGER is a comparative method, requiring a database of reference genomes. We prepared a tailored database for each species consisting of all genomes for that species, capping at 200. For species with ≥ 200 genomes, the most diverse 200 were chosen based on all vs. all MASH distance scores. TIGER2 was run in Intact and either Cross or CircleOrigin modes on all genomes through to the merge. pl script, and GIs were collected from the resulting genome. island.nonoverlap.gff files above a size cutoff of 5 kbp, containing a serine (S-Int) or tyrosine (Y-Int) integrase gene, and with crossover length <300 bp, allowing overlaps no larger than 100 bp. This yielded 223,323 GIs identified by both modes, 211,599 identified by split-scaffold mode only and 13,653 identified by same-scaffold mode only.

Typing of split GIs. TIGER typing software was adapted to handle split GIs. The two halves of the split GI are annotated with our Tater software (Mageeney et al., 2020) which uses Prodigal to call open reading frames, Prokka to assign gene names, and applies Pfam-A HMMs (v. 35) including subsets for phage and ICE proteins. Typing proceeds according to gene content of the entire split GI, as previously described (Mageeney et al., 2020). This yields seven output categories: Phage1, GI containing at least one structural and at least one non-structural phage Pfam; Phage2, GI containing at least one phage Pfam; PhageFil, GI less than 13 kb that contain the Pfam Zot, previously identified in many Inoviridae phages (Roux et al., 2019); ICE1, GI with ≥7 or ≥15% ICE Pfams; ICE2, GI under 10 kb with >2 or ≥12% ICE genes, or over 10 kb with >2 or >7% ICE genes; PhageICE, GI matching both Phage and ICE criteria (very rare and usually due to mistaken grouping of neighbors in a tandem array); Other, GI with none of the above calls.

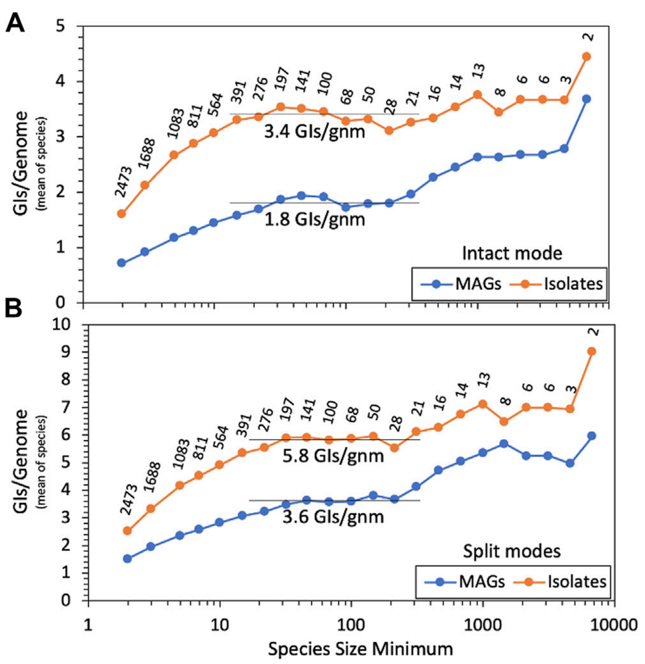


FIGURE 2 | GI yields for MAGs and isolate genomes. TIGER2 was run in **(A)** intact-only mode or **(B)** split modes on genomes from 2473 GTDB species containing at least one MAG and one isolate genome, measuring GIs/genome within each species; shown here is the mean of the GI/genome values for all species tested at each size (i.e., genome count) cutoff. Data labels show the numbers of species remaining with each size cutoff.

Testing large groups of islands. Four GI-abundant genomic loci, the *Escherichia icd*, tmRNA, and *ybhC/ybhB* loci and the *Mycobacterium* tRNA-Ser locus, were studied to examine the quality of the split GI calls. GI sequences were collected for the intact and split islands assigned to those sites, and the 600 bp *attL* and *attR* terminal GI-internal segments were taken as queries, except in cases where scaffold splitting left the terminal segment shorter than 600 bp, where the segment contained a transposase gene indicating sequence likely to be repeated throughout the genome, or where long blocks of ambiguous bases precluded even self-matching. Strong matches (≥500 bp and ≥95% identity) in all-vs. all BLASTN of the intact GI termini were clustered as connected components, combining *attL* and *attR* typing to produce the *attP* type for each intact GI.

RESULTS

GI yields for MAGs and isolates. As a null hypothesis, MAGs could be expected to contain numbers of GIs comparable to isolate genomes. Because some phylogenetic groups are more GI-rich than others (Mageeney et al., 2020), we reasoned that MAG/isolate comparisons would be most appropriate within a species, and that large numbers of such within-species comparisons could achieve statistical significance. The GTDB project has systematically treated most archaeal and bacterial genomes, applying a revised taxonomy that we employ here to improve genome comparison (Parks et al., 2022). Its most strictly defined rank is the species; each is seeded by a representative genome, and a genome must have 95–97% similarity to the representative for inclusion in the species. We analyzed 2 473 GTDB species containing at least one MAG and one isolate genome, totalling 74,561 genomes (7,978 MAGs and 66,583 isolates). Despite this overall bias toward isolates, 894 species had equal numbers of MAGs and isolates, and 549 had more MAGs than isolates.

We ran our GI discovery software TIGER on these genomes, counting GI yields for each. Average GI recovery over all MAGs or isolates would be misleading and dominated by relatively few overrepresented species due to the wide range of species sizes, from 2 to 9 114 genomes. MAG and isolate GI yields were averaged within each species, and we present (**Figure 2A**) averages over all species, using various cutoffs for species size. For both small and large species, there is a trend of increased GI yields with increasing species size. At the left of the figure, small species had small reference databases for TIGER, which likely explains their lower yields. The right of the figure suffers from noise due to low species numbers. The middle region is flatter and provides a species-normalized estimate of 3.4 GIs per isolate genome, with a large depression for MAGs, down to 1.8 GIs per genome. This depression is probably explained by the poorer quality of MAG genomes, worse than isolates by every available metric (**Supplementary Table S1**). Especially relevant is scaffold counts, averaging 95 for isolates and 152 for MAGs. TIGER was designed to search for GIs contained within a single scaffold, but in fragmented genomes, some GIs may also be fragmented, escaping detection.

TIGER2. TIGER employs a "ping-pong BLAST" method, first running a query sequence from the study genome (a candidate GI/chromosome boundary proximal to an integrase gene) against a reference genome database, then running a second query from each hit reference genome back to the original scaffold of the study genome, to find the distal end of the intact GI. In principle this second query can be applied to *all* scaffolds in the study genome to find GIs split among contigs. TIGER2 allows the original "Intact" mode that only finds within-scaffold GIs and two new split modes (either "Cross" for fragmented genomes or "CircleOrigin" for complete genomes) that can also find the termini of GIs when split onto different scaffold ends. We also prepared new species-focused reference databases (Materials and Methods, "Genomic islands"). Running the split modes on the genomes produced many more GI calls. There were 223,323 GIs for which intact and split modes agreed, 13,653 found by intact mode only, and a surprisingly large number, 211,599, found by split modes only. All GI calls from TIGER2 are reported in **Supplementary Table S2**. Repeating the yield analysis (**Figure 2B**), the split modes improved GI yields 1.7-fold for isolates and 2.0-fold for MAGs, elevating the MAGs:isolates ratio from 0.52 (intact mode) to 0.62 (split modes).

The split GIs are generally better supported than competing intact GI calls. A support value is computed for each GI call equal to the number of reference database genomes found to be precisely deleted for (and thereby mapping) the GI. For the 11,152 contests where a split-only GI overlapped an intact-only GI, 806 were tied for support, 543 of the contests were won by higher support for the intact-only GI, and 9,419 were won by the split-only GI.

TABLE 1 | Validation of split GI calls at four commonly used integration loci. Analysis of the tRNA-Ser locus was from 6,283 *Mycobacterium* genomes and of the *icd*, tmRNA and lambda loci from 15,111 *Escherichia* genomes.

Locus	Icd	tmRNA	Lambda	tRNA-ser
Total GIs	3,905	4,882	4,651	6,155
Found by intact and split modes	1,379	2,246	1,248	6,088
Found by intact mode only	10	53	0	6
Found by split modes only	2,516	2,583	3,403	61
Split mode, novel intact	3	4	0	11
Circular origin spanning	7	2	4	0
Cross-scaffold	2,506	2,577	3,399	50
Intact GIs typed	1,361	2,193	1,110	6,099
Intact GI *attL* types	31	100	137	7
Intact GI *attR* types	29	154	8	7
Intact GI *attP* types	66	278	140	7
Split GIs typed	2,361	2,357	2,856	46
Split GIs, known *attP* type	2,286	2075	2,726	46
Split GIs, novel *attP* type	75	282	130	0

Assessing GIs at four common genomic integration sites. To further assess the quality of TIGER2 calls, four large groups of GIs integrating into the same genomic site in the same large genus were examined, at the *icd* and tmRNA genes and the phage lambda locus (the *ybhC/ybhB* intergenic site) of *Escherichia*, and the tRNA-Ser gene in *Mycobacterium* (**Table 1**). The *Mycobacterium* tRNA-Ser gene (and other loci in the genus) have far fewer split-only GIs than the *Escherichia* loci. This may be simply explained by the much larger scaffold:GI length ratio, 11.0, for tRNA-Ser GIs in *Mycobacterium*; this ratio is only 1.6–1.8 for the *Escherichia* GIs. Databases were prepared from the genomes containing an intact GI at the site for the genus. For each locus, the GI-internal terminal DNA sequences were used to type intact and split GIs, and a split GI was considered validated when its termini matched those of an intact GI. This test of GI-*internal* sequences is orthogonal to the TIGER method itself, which finds GIs based only on their *flanking* sequences. The sequences from the *attL* region were independently typed, as were the *attR* sequences, and together these produced an *attP* type for each GI. Although the goal of this typing was to assess the new split GIs, we first characterized *attP* types among intact GIs only.

Intact GIs at the four integration sites. At the *Mycobacterium* tRNA-Ser locus, only seven *attP* types were observed, that do not mix *attL* and *attR* types, and are strictly segregated by species, for example, the largest type (6,075 GIs) is restricted to *M. tuberculosis* (Mtu) and is the only *attP* type in that species (**Table 1**). At the *Escherichia* loci there is much greater *attP* type diversity, strong but imperfect species segregation, and each shows mixing of the half-*attP*s. For example, between two abundant *attP* types at *icd*, L1-R2 (this designation indicates its composition from *attL* type 1 and *attR* type 2) and L10-R4, both mixtures are observed, L1-R4 and L10-R2. Such swapping of unrelated *attP* halves is probably an example of the mosaicism

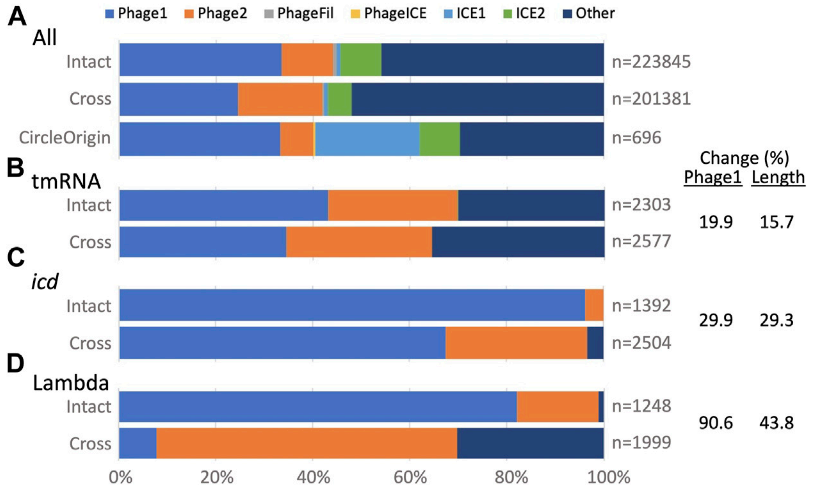


FIGURE 3 | TIGER2 GI type breakdowns for composition categories (Intact, Cross and CircleOrigin). **(A)** All GIs, or **(B–D)** GIs at three *Escherichia* loci. Percent change is given for Intact vs. Cross GIs; change for Phage1 counts correlates with change in GI length across the three loci.

that is pervasive among GIs, but in some cases could be due to unresolved tandem GI arrays. At the lambda site, we observe lopsided mosaicism: one main *attR* type and many different *attL* types. The tmRNA gene has the highest occupancy and the highest diversity of *attP* types, perhaps related to its known targeting by multiple independent integrase clades (Williams, 2003).

The Mtu tRNA-Ser GI reveals a problem with using small, single-species reference databases; this GI is so widespread in the species that only one of the 200 genomes in the Mtu reference database was lacking the GI and therefore able to identify, map, and support it. With a support value of only one, a false positive GI call with support values as low as two might overlap the tRNA-Ser GI and eliminate it during the merging step. Six false negative intact Mtu GIs were identified through matches to the split GI queries; all had been identified by the TIGER core module, but rejected during merging due to overlapping false positives. In the future we will prepare reference databases that include some genomes from outside the species.

Split GIs at the four loci. Results for the tRNA-Ser locus can be succinctly summarized. The 31 split GIs from Mtu all had the same *attP* type as all intact Mtu GIs. The remaining 15 split GIs, from *M. immunogenum*, had an *attP* type of intact *M. immunogenum* GIs. For the *Escherichia* loci there were more split GIs than intact GIs, and some new *attP* types. Altogether 93.6% of the tested split GIs were validated, matching *attP* types known from intact GIs. Some of the mismatches may reflect additional mosaicism (**Table 1**).

GI typing. The TIGER typing module determines whether a GI is a credibly complete prophage (Phage1) or contains less than a full complement of phage genes (Phage2), and likewise assigns a category one and two for ICEs, otherwise leaving the type undetermined (Other). This module was updated to accommodate split GIs. Examining all GIs (**Figure 3A**), the type breakdown for intact GIs is similar to that observed before (Mageeney et al., 2020), with almost half labeled Other and the next largest fraction labeled Phage1. For cross-scaffold GIs, the Phage1 fraction is appreciably smaller, while Phage2 and Other fractions are larger than for intact GIs. This "downward" typing shift may be due to "missing middles," that is, if a GI is split onto more than two scaffolds, its central fragments would remain unidentified because TIGER2 finds only the terminal fragments of split GIs (**Figure 1**). Circle Origin GIs, which should not suffer from missing middles, have the same fraction of Phage1 as the intact GIs, with a notable expansion of the ICE1 category. ICEs tend to range to larger sizes than prophages; the arbitrary origin point of complete circular chromosomes may land more frequently on these larger GIs.

We also examined typing for GIs at the above three *Escherichia* sites, which all had large numbers of both intact and split GIs. For intact GIs, each site showed a different balance between Phage1, Phage2 and Other calls (**Figures 3B–D**). All had a downward typing shift for split GIs. According to our "missing middle" hypothesis, this downward typing shift might correlate with shorter split GI calls that omit central fragments. Extents of downward typing did indeed correlate with reductions in GI length (**Figures 3B–D**). For the split GIs at the tmRNA gene, the drops in Phage1 type and average GI lengths were small (20 and 16%). At the other extreme, the Phage1 fraction for the lambda site GIs dropped by 91% and the average GI length concomitantly dropped by 44%. Some features in many lambda site GIs may especially antagonize assembly, leaving more missing middle segments than for the tmRNA and *icd* GIs.

DISCUSSION

Our original GI detection software, operating only on single scaffolds, yielded substantially fewer GIs for MAGs than for species-matched isolate genomes. Suspicion that this was due to higher fragmentation of MAGs than isolates motivated a software update enabling cross-scaffold search. TIGER2 doubled GI yields for MAGs. This surprisingly large improvement shows that fragmentation levels in current microbial genomes substantially impact GI detection. Even with this new approach, MAG yields are still not equal to same-species isolate yields. A possible biological reason for this remaining discrepancy might be sought in the "domestication" of isolates through many generations of passage in the lab (Barreto et al., 2020); however we expect the opposite trend from domestication, that GIs could only be lost by excision events in isolates. Other aspects of quality such as completeness may depress yields in MAGs, when high microdiversity within a GI prevents its full assembly into a scaffold (Haro-Moreno et al., 2021). A third explanation is that only very small databases of related genomes may be available for many MAG-rich species, insufficient for TIGER (or any comparative method) to find all GIs.

The quality of the new split GIs is high by several criteria. GI support values outscore those of competing calls by intact-mode TIGER. At frequently-used genomic loci of integration, the split GIs share the *attP* compositions of the intact GIs. Split GIs have type profiles (phage:non-phage) comparable to intact GIs, although with a shift downward explainable by missing middle segments; TIGER2 finds only the terminal fragments of a GI such that its call will omit any additional internal fragments that might exist for the GI.

CONCLUSION

Cross-scaffold search by TIGER2 doubles GI yields across diverse microbial species, linking more scaffolds and improving the quality of fragmented genomes such as MAGs. This will aid detection of viruses in metagenomic datasets, offer insights into population microdiversity and its phenotypic and ecological consequences, and help address questions such as the balance of temperate phages between the lysogenic state and free virions. We will apply TIGER2 to our larger genome database to produce an atlas of MGEs with precisely mapped termini in microbial genomes; although applied here only to GIs, the TIGER principle also discovers and maps other MGE classes, such as transposable elements (Mageeney et al., 2020). The rise of long-read sequencing is a welcome trend that will improve mobilome representation in MAGs; lengths can now be attained sufficient to contain an entire GI within a single read (Warwick-Dugdale et al., 2019; Nicolas et al., 2021; Zablocki et al., 2021).

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/**Supplementary Material**.

AUTHOR CONTRIBUTIONS

KW conceived the project and drafted the figures and tables. CM, GT, and KW provided manuscript writing and edits, analyzed the data, made a substantial, direct, and intellectual contribution to the work, and approved it for publication.

FUNDING

CM and KW were supported by the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research under the Secure Biosystems Design Initiative and by the Laboratory Directed Research and Development (LDRD) program of Sandia National Laboratories, which is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. G.T. was supported by the US Department of Energy (DOE) Office of Science, Office of Biological and Environmental Research Genomic Science program award SCW1632, LLNL LDRD 21-LW-060, and under the auspices of the DOE under contract DE-AC52-07NA27344.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.866850/full#supplementary-material

REFERENCES

Barreto, H. C., Cordeiro, T. N., Henriques, A. O., and Gordo, I. (2020). Rampant Loss of Social Traits during Domestication of a *Bacillus Subtilis* Natural Isolate. *Sci. Rep.* 10 (1), 18886. doi:10.1038/s41598-020-76017-1

Bertelli, C., Tilley, K. E., and Brinkman, F. S. L. (2019). Microbial Genomic Island Discovery, Visualization and Analysis. *Brief Bioinform.* 20 (5), 1685–1698. doi:10.1093/bib/bby042

Bowers, R. M., Kyrpides, N. C., Stepanauskas, R., Harmon-Smith, M., Doud, D., Reddy, T. B. K., et al. (2017). Minimum Information about a Single Amplified Genome (MISAG) and a Metagenome-Assembled Genome (MIMAG) of Bacteria and Archaea. *Nat. Biotechnol.* 35 (8), 725–731. doi:10.1038/nbt.3893

Carr, V. R., Witherden, E. A., Lee, S., Shoaie, S., Mullany, P., Proctor, G. B., et al. (2020). Abundance and Diversity of Resistomes Differ between Healthy Human Oral Cavities and Gut. *Nat. Commun.* 11 (1), 693. doi:10.1038/s41467-020-14422-w

Drenkard, E., and Ausubel, F. M. (2002). *Pseudomonas* Biofilm Formation and Antibiotic Resistance Are Linked to Phenotypic Variation. *Nature* 416 (6882), 740–743. doi:10.1038/416740a

Evans, J. T., and Denef, V. J. (2020). To Dereplicate or Not to Dereplicate? *mSphere* 5 (3), e00971. doi:10.1128/mSphere.00971-19

Fillol-Salom, A., Martínez-Rubio, R., Abdulrahman, R. F., Chen, J., Davies, R., and Penadés, J. R. (2018). Phage-inducible Chromosomal Islands Are Ubiquitous within the Bacterial Universe. *ISME J* 12 (9), 2114–2128. doi:10.1038/s41396-018-0156-3

Haro-Moreno, J. M., López-Pérez, M., and Rodriguez-Valera, F. (2021). Enhanced Recovery of Microbial Genes and Genomes from a Marine Water Column Using Long-Read Metagenomics. *Front. Microbiol.* 12, 708782. doi:10.3389/fmicb.2021.708782

Hudson, C. M., Lau, B. Y., and Williams, K. P. (2015). Islander: a Database of Precisely Mapped Genomic Islands in tRNA and tmRNA Genes. *Nucleic Acids Res.* 43 (Database issue), D48–D53. doi:10.1093/nar/gku1072

Mageeney, C. M., Lau, B. Y., Wagner, J. M., Hudson, C. M., Schoeniger, J. S., Krishnakumar, R., et al. (2020). New Candidates for Regulated Gene Integrity Revealed through Precise Mapping of Integrative Genetic Elements. *Nucleic Acids Res.* 48 (8), 4052–4065. doi:10.1093/nar/gkaa156

Maguire, F., Jia, B., Gray, K. L., Lau, W. Y. V., Beiko, R. G., and Brinkman, F. S. L. (2020). Metagenome-assembled Genome Binning Methods with Short Reads Disproportionately Fail for Plasmids and Genomic Islands. *Microb. Genom* 6 (10), mgen000436. doi:10.1099/mgen.0.000436

Nayfach, S., Roux, S., Seshadri, R., Udwary, D., Varghese, N., Schulz, F., et al. (2021). A Genomic Catalog of Earth's Microbiomes. *Nat. Biotechnol.* 39 (4), 499–509. doi:10.1038/s41587-020-0718-6

Nicolas, A. M., Jaffe, A. L., Nuccio, E. E., Taga, M. E., Firestone, M. K., and Banfield, J. F. (2021). Soil Candidate Phyla Radiation Bacteria Encode Components of Aerobic Metabolism and Co-occur with Nanoarchaea in the Rare Biosphere of Rhizosphere Grassland Communities. *mSystems* 6 (4), e0120520. doi:10.1128/mSystems.01205-20

Parks, D. H., Chuvochina, M., Rinke, C., Mussig, A. J., Chaumeil, P. A., and Hugenholtz, P. (2022). GTDB: an Ongoing Census of Bacterial and Archaeal Diversity through a Phylogenetically Consistent, Rank Normalized and Complete Genome-Based Taxonomy. *Nucleic Acids Res.* 50 (D1), D785–d794. doi:10.1093/nar/gkab776

Roux, S., Krupovic, M., Daly, R. A., Borges, A. L., Nayfach, S., Schulz, F., et al. (2019). Cryptic Inoviruses Revealed as Pervasive in Bacteria and Archaea across Earth's Biomes. *Nat. Microbiol.* 4 (11), 1895–1906. doi:10.1038/s41564-019-0510-x

Shaiber, A., and Eren, A. M. (2019). Composite Metagenome-Assembled Genomes Reduce the Quality of Public Genome Repositories. *mBio* 10 (3), e00725. doi:10.1128/mBio.00725-19

Sieradzki, E. T., Koch, B. J., Greenlon, A., Sachdeva, R., Malmstrom, R. R., Mau, R. L., et al. (2020). Measurement Error and Resolution in Quantitative Stable Isotope Probing: Implications for Experimental Design. *mSystems* 5, e00151. doi:10.1128/mSystems.00151-20

Steen, A. D., Crits-Christoph, A., Carini, P., DeAngelis, K. M., Fierer, N., Lloyd, K. G., et al. (2019). High Proportions of Bacteria and Archaea across Most Biomes Remain Uncultured. *ISME J* 13 (12), 3126–3130. doi:10.1038/s41396-019-0484-y

Sullivan, J. T., and Ronson, C. W. (1998). Evolution of Rhizobia by Acquisition of a 500-kb Symbiosis Island that Integrates into a Phe-tRNA Gene. *Proc. Natl. Acad. Sci. U S A.* 95 (9), 5145–5149. doi:10.1073/pnas.95.9.5145

Taş, N., de Jong, A. E., Li, Y., Trubl, G., Xue, Y., and Dove, N. C. (2021). Metagenomic Tools in Microbial Ecology Research. *Curr. Opin. Biotechnol.* 67, 184–191. doi:10.1016/j.copbio.2021.01.019

Tyson, G. W., Chapman, J., Hugenholtz, P., Allen, E. E., Ram, R. J., Richardson, P. M., et al. (2004). Community Structure and Metabolism through Reconstruction of Microbial Genomes from the Environment. *Nature* 428 (6978), 37–43. doi:10.1038/nature02340

van der Meer, J. R., and Sentchilo, V. (2003). Genomic Islands and the Evolution of Catabolic Pathways in Bacteria. *Curr. Opin. Biotechnol.* 14 (3), 248–254. doi:10.1016/s0958-1669(03)00058-2

Warwick-Dugdale, J., Solonenko, N., Moore, K., Chittick, L., Gregory, A. C., Allen, M. J., et al. (2019). Long-read Viral Metagenomics Captures Abundant and Microdiverse Viral Populations and Their Niche-Defining Genomic Islands. *PeerJ* 7, e6800. doi:10.7717/peerj.6800

Williams, K. P. (2003). Traffic at the tmRNA Gene. *J. Bacteriol.* 185 (3), 1059–1070. doi:10.1128/jb.185.3.1059-1070.2003

Zablocki, O., Michelsen, M., Burris, M., Solonenko, N., Warwick-Dugdale, J., Ghosh, R., et al. (2021). VirION2: a Short- and Long-Read Sequencing and Informatics Workflow to Study the Genomic Diversity of Viruses in Nature. *PeerJ* 9, e11088. doi:10.7717/peerj.11088

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Mageeney, Trubl and Williams. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

frontiers | Frontiers in Bioinformatics

ORIGINAL RESEARCH
published: 26 April 2022
doi: 10.3389/fbinf.2022.869150

Pathway Tools Management of Pathway/Genome Data for Microbial Communities

Peter D. Karp[1]*, *Suzanne Paley*[1], *Markus Krummenacker*[1], *Anamika Kothari*[1], *Michael J. Wannemuehler*[2] and *Gregory J. Phillips*[2]

[1]Bioinformatics Research Group, Artificial Intelligence Center, SRI International, Menlo Park, CA, United States, [2]Department of Veterinary Microbiology, Iowa State University, Ames, IA, United States

The Pathway Tools (PTools) software provides a suite of capabilities for storing and analyzing integrated collections of genomic and metabolic information in the form of organism-specific Pathway/Genome Databases (PGDBs). A microbial community is represented in PTools by generating a PGDB from each metagenome-assembled genome (MAG). PTools computes a metabolic reconstruction for each organism, and predicts its operons. The properties of individual MAGs can be investigated using the many search and visualization operations within PTools. PTools also enables the user to investigate the properties of the microbial community by issuing searches across the full community, and by performing comparative operations across genome and pathway information. The software can generate a metabolic network diagram for the community, and it can overlay community omics datasets on that network diagram. PTools also provides a tool for searching for metabolic transformation routes across an organism community.

Keywords: microbiome, data management, genome database, metabolic pathways, metabolic routes

OPEN ACCESS

Edited by:
Nikos Kyrpides,
Joint Genome Institute, United States

Reviewed by:
Kazuhiro Takemoto,
Kyushu Institute of Technology, Japan
Natalia Ivanova,
Berkeley Lab (DOE), United States

****Correspondence:***
Peter D. Karp
pkarp@ai.sri.com

Specialty section:
This article was submitted to
Genomic Analysis,
a section of the journal
Frontiers in Bioinformatics

Received: *03 February 2022*
Accepted: *05 April 2022*
Published: *26 April 2022*

Citation:
Karp PD, Paley S, Krummenacker M, Kothari A, Wannemuehler MJ and Phillips GJ (2022) Pathway Tools Management of Pathway/Genome Data for Microbial Communities.
Front. Bioinform. 2:869150.
doi: 10.3389/fbinf.2022.869150

1 INTRODUCTION

The Pathway Tools (PTools) software Karp et al. (2019), Karp et al. (2020) was originally developed to facilitate functional analysis of individual genomes. The software has a range of capabilities including genome informatics, metabolic pathway informatics, regulatory informatics, omics data analysis, and comparative analysis. A typical workflow is to import a genome into PTools, compute a metabolic reconstruction, infer operons of the organism, and then apply the search, visualization, and comparative analysis tools to investigate the functional properties of the organism.

The software has been extended in recent years to support functional analysis of microbiomes (Prakash and Taylor, 2012; Krishnan et al., 2015; Sung et al., 2017; Eng and Borenstein, 2019; Visconti et al., 2019) that provides causal insights regarding the interactions of organisms within a microbiome. Whereas many microbiome-related informatics tools aim to quantify and compare properties of an overall community, PTools is more focused on enabling detailed reconstructions of community members and their interactions. For example, PTools does not perform taxonomic analysis of metagenome samples, nor does it compute case-control studies such as statistical comparison of healthy and diseased individuals.

Questions that can be addressed using the software include the following: What metabolic reactions and pathways are present in a metagenome, or in each organism in a community? How do their metabolic capabilities complement one another? What pathways are unique to a given community member? What metabolic transformations can be accomplished by the community, for example, via what metabolic route might the community convert a starting metabolite into an ending metabolite? The PTools software does perform metabolic modeling of individual microbes and of microbial communities via flux-balance analysis Latendresse et al. (2022) (see also Greenblum et al., 2013; Levy and Borenstein, 2014; Krishnan et al., 2015; Esvap and Ulgen, 2021; Heinken et al., 2021), although that topic is beyond the scope of this article.

The first step in a typical workflow is to import a set of metagenome-assembled genomes (MAGs) into PTools. Each MAG is converted to a PTools Pathway/Genome Database. A number of PTools computational inference tools are next applied to each MAG to infer its metabolic reactions and pathways, its transport reactions, and its operons. The community members are now captured within a set of PGDBs that comprehensively encode their genomes and metabolic networks.

Next the user can apply a set of search and comparative analysis tools to assess and compare the functional capabilities of community members. For example, the user can search across all community PGDBs for the presence of a gene, a metabolite, or a pathway; the software can produce comparisons of the entire metabolic networks of the community.

If meta-transcriptomics and/or meta-metabolomics data are available for the community, then PTools provides an analysis tool for visualizing such data on a multi-organism metabolic map diagram.

PTools provides a community route-search tool that requires as user inputs a set of PGDBs as well as a starting metabolite and ending metabolite. The tool generates minimal-cost metabolic routes (linear reaction paths) from the starting to the ending metabolite that show how the community might accomplish that transformation.

The remainder of the article describes these tools in more detail and illustrates their use on the Altered Schaedler Flora (ASF), a community of eight microorganisms from the mouse gut microbiome Wannemuehler et al. (2014). The ASF were selected by experimentalists as a model microbiome for their dominance and persistence in the mouse gut, and for their ability to be grown in the laboratory.

2 METHODS

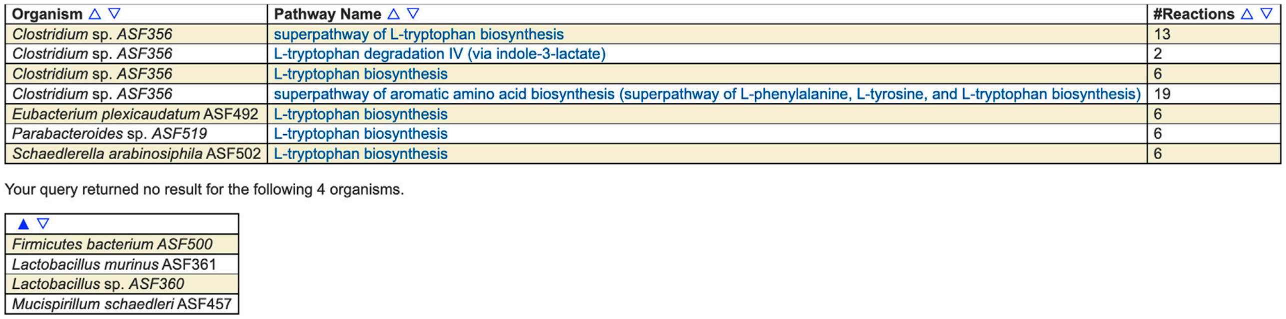
2.1 Importing a Microbial Community Into Pathway Tools

To import a microbial community into PTools, the metagenomic sequencing data must have been binned by a separate program into separate groups, one for each detected member of the community. Each such MAG consists of a collection of sequenced contigs covering a subset of the genome of each organism. The contigs must be annotated by a tool such as MetaPathways Konwar et al. (2013), MetaErg Dong and Strous (2019), MG-RAST Keegan et al. (2016), MEGAN Huson et al. (2016), Prokka Seemann (2014), or the National Center for Biotechnology Information (NCBI) Prokaryotic Genome Annotation Pipeline Tatusova et al. (2016), meaning that an ORF-finding program has been run on each organism, and protein function-prediction tools have been run on each identified gene to assign protein names such as "pyruvate kinase," as well as to assign Enzyme Commission (EC) numbers (optional).

The resulting sequence data, gene locations, and protein functions can be provided as inputs to PTools in either GFF3 format or GenBank format, preferably as one file per MAG. The files can be provided within a directory structure containing one directory per genome that is processed by invoking the PathoLogic component of PTools from the command line, as described in the Pathway Tools User's Guide SRI International (2021).

PathoLogic applies a series of processing steps to each input MAG to obtain a comprehensive PGDB for each organism. Those steps are as follows.

1. The input files are parsed.
2. The input sequence, gene locations, and annotations are converted to PGDB format. PGDBs are encoded using the Ocelot object-oriented database system. A database object is created for each replicon, each gene, and each gene product described by the input files.
3. The reactome of the organism is predicted from the annotated gene functions using a previously published algorithm Karp et al. (2011). Enzyme names and EC numbers are associated



Organism △ ▽	Pathway Name △ ▽	#Reactions △ ▽
Clostridium sp. *ASF356*	superpathway of L-tryptophan biosynthesis	13
Clostridium sp. *ASF356*	L-tryptophan degradation IV (via indole-3-lactate)	2
Clostridium sp. *ASF356*	L-tryptophan biosynthesis	6
Clostridium sp. *ASF356*	superpathway of aromatic amino acid biosynthesis (superpathway of L-phenylalanine, L-tyrosine, and L-tryptophan biosynthesis)	19
Eubacterium plexicaudatum ASF492	L-tryptophan biosynthesis	6
Parabacteroides sp. *ASF519*	L-tryptophan biosynthesis	6
Schaedlerella arabinosiphila ASF502	L-tryptophan biosynthesis	6

Your query returned no result for the following 4 organisms.

▲ ▽
Firmicutes bacterium ASF500
Lactobacillus murinus ASF361
Lactobacillus sp. *ASF360*
Mucispirillum schaedleri ASF457

FIGURE 1 | Results of searching across the ASF for pathways whose name contains "tryptophan." Four of the organisms contain such a pathway and four do not. The pathways include biosynthesis and degradation, as well as super-pathways and base pathways.

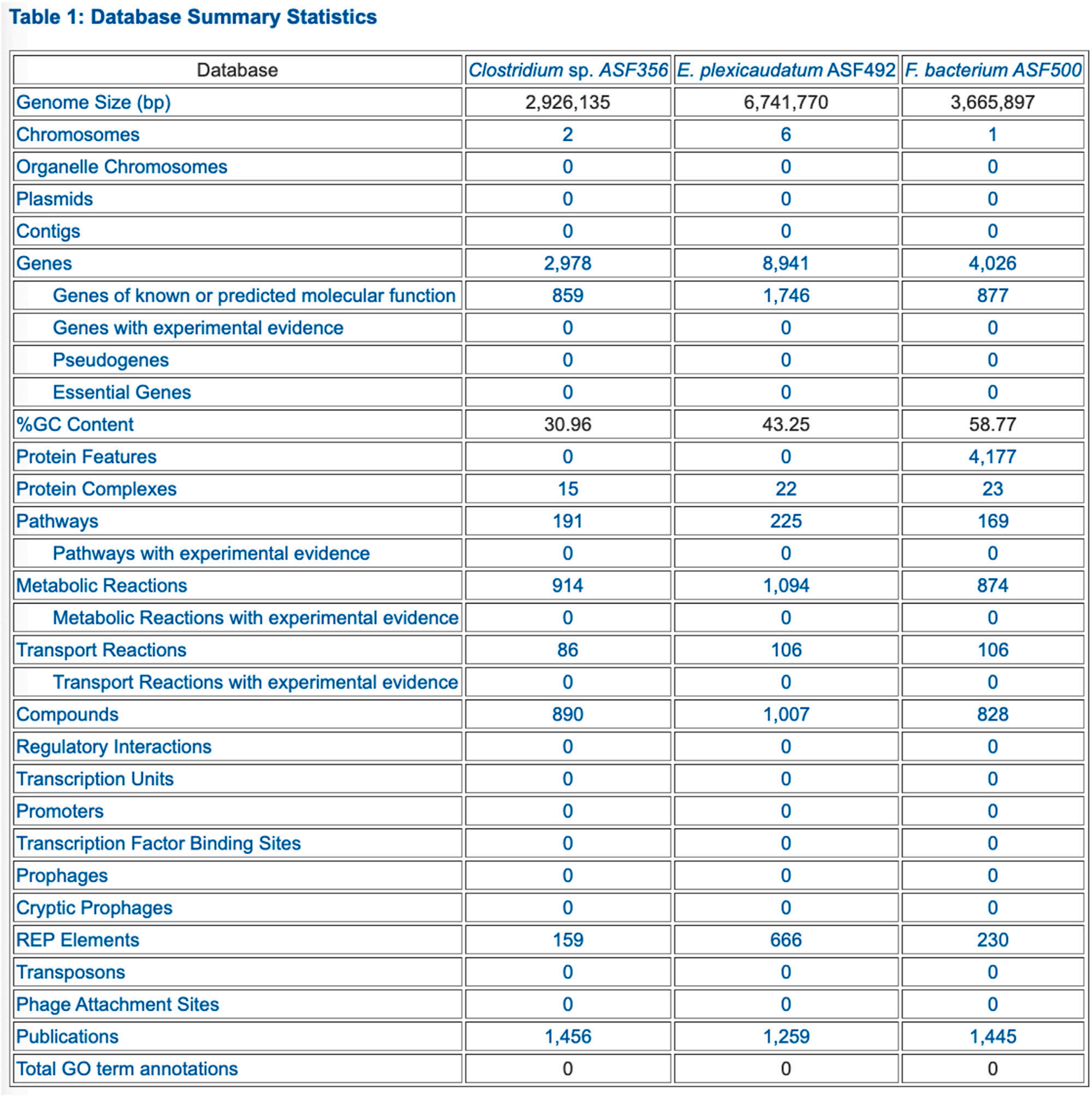


Table 1: Database Summary Statistics

Database	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
Genome Size (bp)	2,926,135	6,741,770	3,665,897
Chromosomes	2	6	1
Organelle Chromosomes	0	0	0
Plasmids	0	0	0
Contigs	0	0	0
Genes	2,978	8,941	4,026
Genes of known or predicted molecular function	859	1,746	877
Genes with experimental evidence	0	0	0
Pseudogenes	0	0	0
Essential Genes	0	0	0
%GC Content	30.96	43.25	58.77
Protein Features	0	0	4,177
Protein Complexes	15	22	23
Pathways	191	225	169
Pathways with experimental evidence	0	0	0
Metabolic Reactions	914	1,094	874
Metabolic Reactions with experimental evidence	0	0	0
Transport Reactions	86	106	106
Transport Reactions with experimental evidence	0	0	0
Compounds	890	1,007	828
Regulatory Interactions	0	0	0
Transcription Units	0	0	0
Promoters	0	0	0
Transcription Factor Binding Sites	0	0	0
Prophages	0	0	0
Cryptic Prophages	0	0	0
REP Elements	159	666	230
Transposons	0	0	0
Phage Attachment Sites	0	0	0
Publications	1,456	1,259	1,445
Total GO term annotations	0	0	0

FIGURE 2 | PTools generated table that summarizes database contents for three selected ASF organisms.

with biochemical reactions via queries to the MetaCyc metabolic database (DB) Caspi et al. (2020). Those reactions are imported into the new PGDB from MetaCyc.

4. The metabolic pathways of the organism are predicted from the predicted reactome Karp et al. (2011). For each pathway in MetaCyc the prediction algorithm considers which of its component reactions are catalyzed by an enzyme in the PGDB, and computes a score expressing the likelihood that the pathway is present. Pathways that exceed a threshold are imported into the PGDB.
5. The Transport Inference Parser Lee et al. (2008) is executed to predict the transport reactions of the organism from annotated transporter names.
6. The PTools operon predictor Romero and Karp (2004) is executed to predict the operons of the organism.
7. PathoLogic executes an automatic layout algorithm that creates an organism-specific metabolic network diagram for the organism based on its complement of pathways, metabolic reactions, and transport reactions Paley et al. (2021).

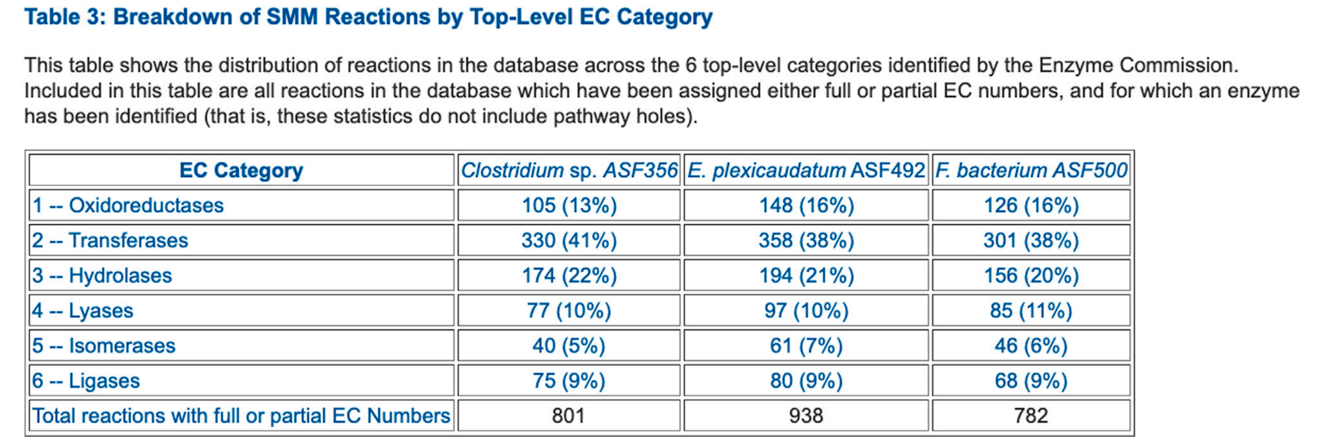
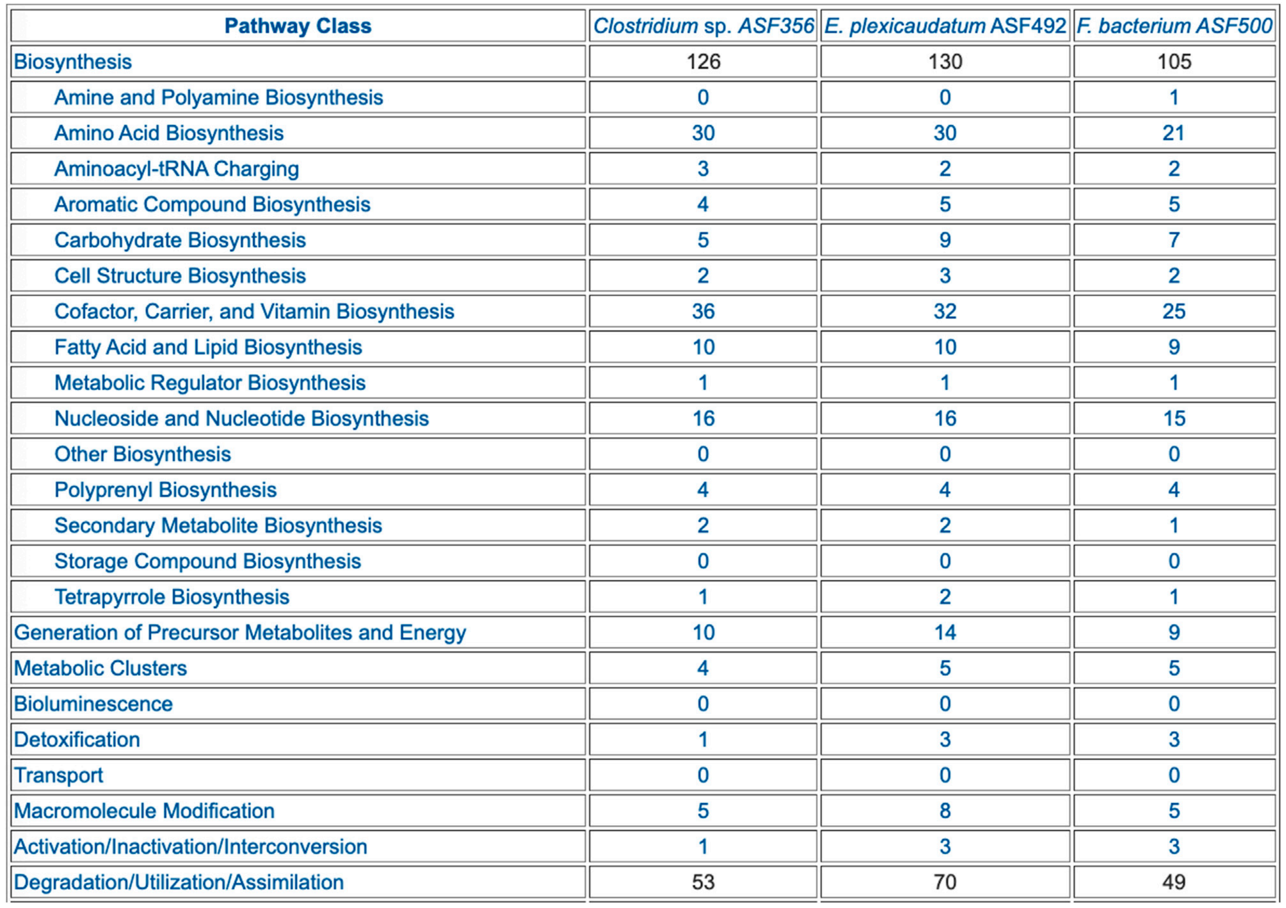


Table 3: Breakdown of SMM Reactions by Top-Level EC Category

This table shows the distribution of reactions in the database across the 6 top-level categories identified by the Enzyme Commission. Included in this table are all reactions in the database which have been assigned either full or partial EC numbers, and for which an enzyme has been identified (that is, these statistics do not include pathway holes).

EC Category	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
1 -- Oxidoreductases	105 (13%)	148 (16%)	126 (16%)
2 -- Transferases	330 (41%)	358 (38%)	301 (38%)
3 -- Hydrolases	174 (22%)	194 (21%)	156 (20%)
4 -- Lyases	77 (10%)	97 (10%)	85 (11%)
5 -- Isomerases	40 (5%)	61 (7%)	46 (6%)
6 -- Ligases	75 (9%)	80 (9%)	68 (9%)
Total reactions with full or partial EC Numbers	801	938	782

FIGURE 3 | Table that summarizes the number of enzymes in each Enzyme Commission top-level category for selected ASF organisms.



Pathway Class	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
Biosynthesis	126	130	105
Amine and Polyamine Biosynthesis	0	0	1
Amino Acid Biosynthesis	30	30	21
Aminoacyl-tRNA Charging	3	2	2
Aromatic Compound Biosynthesis	4	5	5
Carbohydrate Biosynthesis	5	9	7
Cell Structure Biosynthesis	2	3	2
Cofactor, Carrier, and Vitamin Biosynthesis	36	32	25
Fatty Acid and Lipid Biosynthesis	10	10	9
Metabolic Regulator Biosynthesis	1	1	1
Nucleoside and Nucleotide Biosynthesis	16	16	15
Other Biosynthesis	0	0	0
Polyprenyl Biosynthesis	4	4	4
Secondary Metabolite Biosynthesis	2	2	1
Storage Compound Biosynthesis	0	0	0
Tetrapyrrole Biosynthesis	1	2	1
Generation of Precursor Metabolites and Energy	10	14	9
Metabolic Clusters	4	5	5
Bioluminescence	0	0	0
Detoxification	1	3	3
Transport	0	0	0
Macromolecule Modification	5	8	5
Activation/Inactivation/Interconversion	1	3	3
Degradation/Utilization/Assimilation	53	70	49

FIGURE 4 | Table summarizing the pathway composition of selected ASF organisms, organized by the MetaCyc pathway ontology. The table is truncated for space considerations.

The result of this process is a community of PGDBs—one for each binned organism—describing its genome, proteome, reactome, metabolic pathways, and operons. For example, we have created PGDBs for each of the eight members of the ASF, all of which are available through the BioCyc.org website (which is powered by PTools). Enter "ASF" into

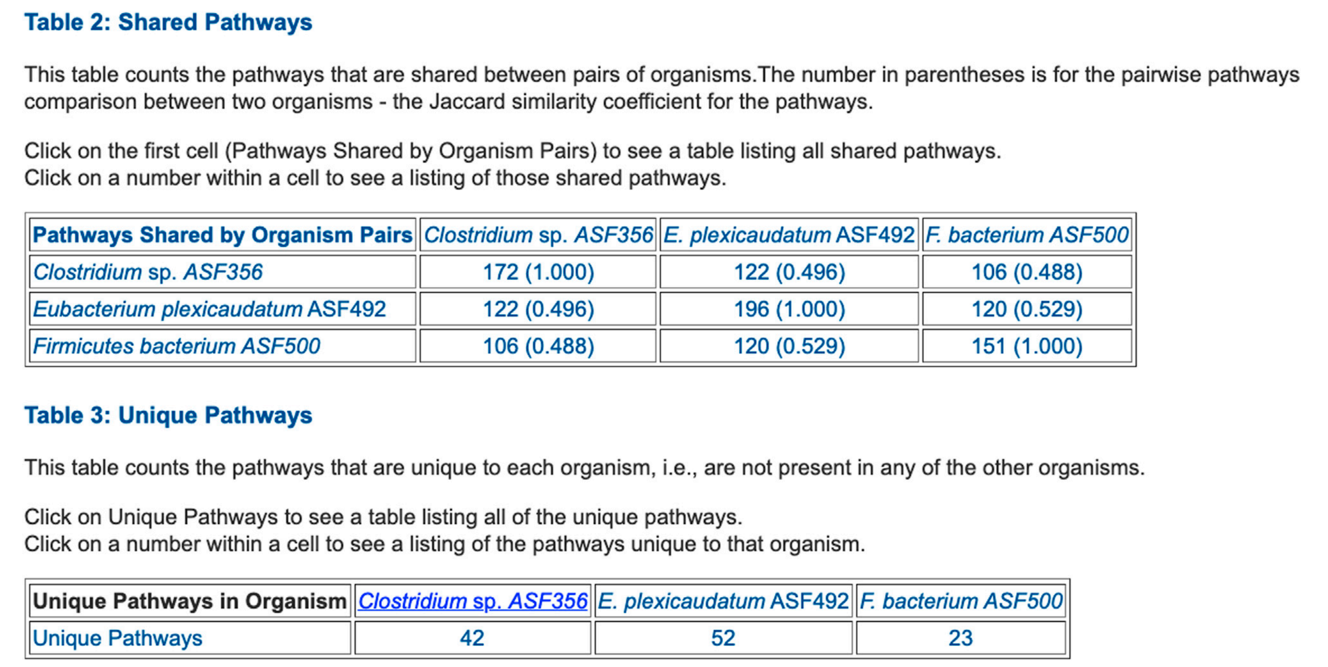


Table 2: Shared Pathways

This table counts the pathways that are shared between pairs of organisms.The number in parentheses is for the pairwise pathways comparison between two organisms - the Jaccard similarity coefficient for the pathways.

Click on the first cell (Pathways Shared by Organism Pairs) to see a table listing all shared pathways.
Click on a number within a cell to see a listing of those shared pathways.

Pathways Shared by Organism Pairs	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
Clostridium sp. *ASF356*	172 (1.000)	122 (0.496)	106 (0.488)
Eubacterium plexicaudatum ASF492	122 (0.496)	196 (1.000)	120 (0.529)
Firmicutes bacterium ASF500	106 (0.488)	120 (0.529)	151 (1.000)

Table 3: Unique Pathways

This table counts the pathways that are unique to each organism, i.e., are not present in any of the other organisms.

Click on Unique Pathways to see a table listing all of the unique pathways.
Click on a number within a cell to see a listing of the pathways unique to that organism.

Unique Pathways in Organism	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
Unique Pathways	42	52	23

FIGURE 5 | Table summarizing the number of metabolic pathways shared between pairs of selected ASF organisms, and the number of pathways unique to each of the three organisms.

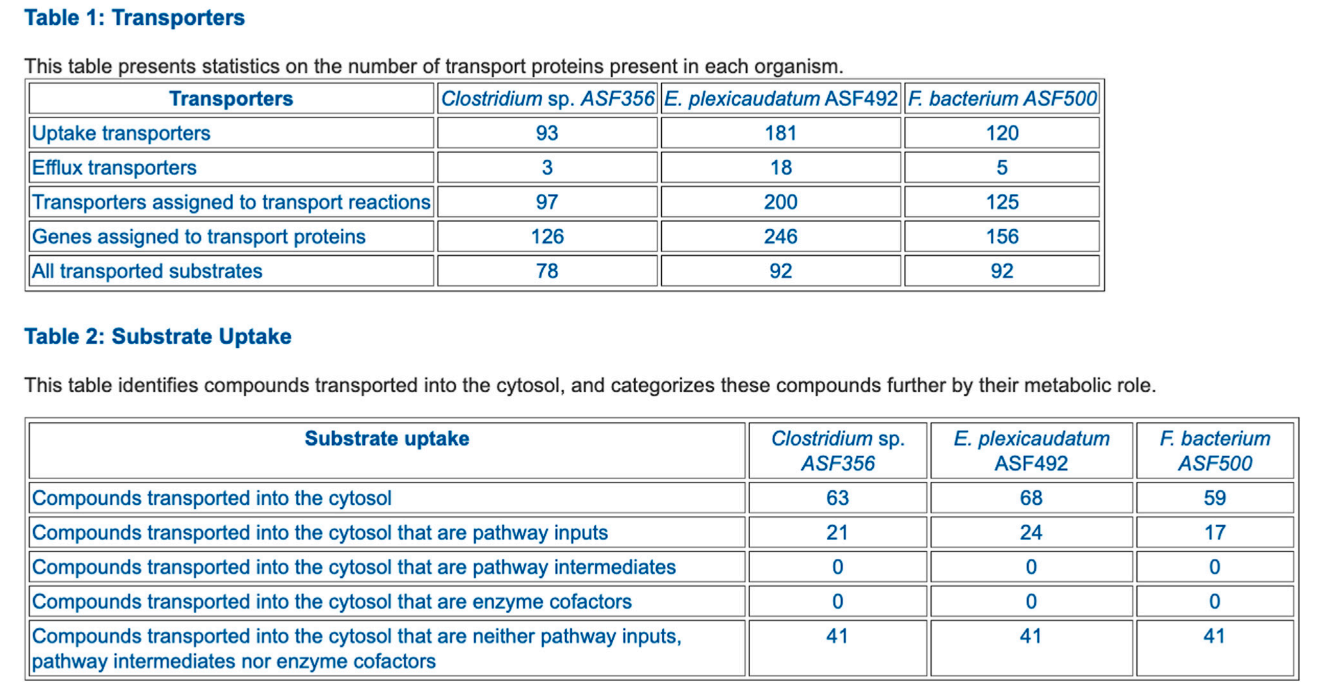


Table 1: Transporters

This table presents statistics on the number of transport proteins present in each organism.

Transporters	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
Uptake transporters	93	181	120
Efflux transporters	3	18	5
Transporters assigned to transport reactions	97	200	125
Genes assigned to transport proteins	126	246	156
All transported substrates	78	92	92

Table 2: Substrate Uptake

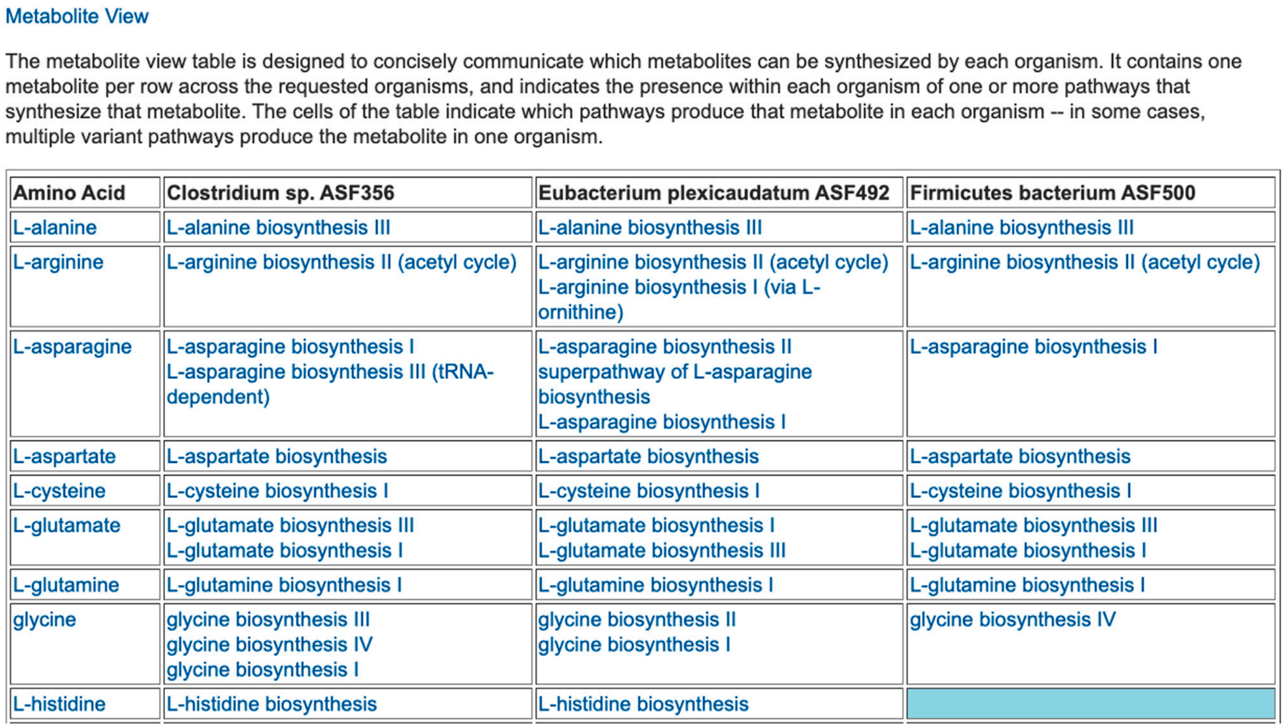
This table identifies compounds transported into the cytosol, and categorizes these compounds further by their metabolic role.

Substrate uptake	*Clostridium* sp. *ASF356*	*E. plexicaudatum* ASF492	*F. bacterium ASF500*
Compounds transported into the cytosol	63	68	59
Compounds transported into the cytosol that are pathway inputs	21	24	17
Compounds transported into the cytosol that are pathway intermediates	0	0	0
Compounds transported into the cytosol that are enzyme cofactors	0	0	0
Compounds transported into the cytosol that are neither pathway inputs, pathway intermediates nor enzyme cofactors	41	41	41

FIGURE 6 | Table comparing the transporter complements of selected ASF organisms.

the BioCyc organism selection tool to search for these databases.

We are not aware of other metagenome-analysis software that performs operon prediction or transport-reaction prediction. A number of other software tools Prakash and Taylor (2012); Huson et al. (2016) perform metabolic reaction and pathway prediction, often based on KEGG Kanehisa et al. (2021). The metabolic reconstruction approaches of KEGG



Metabolite View

The metabolite view table is designed to concisely communicate which metabolites can be synthesized by each organism. It contains one metabolite per row across the requested organisms, and indicates the presence within each organism of one or more pathways that synthesize that metabolite. The cells of the table indicate which pathways produce that metabolite in each organism -- in some cases, multiple variant pathways produce the metabolite in one organism.

Amino Acid	Clostridium sp. ASF356	Eubacterium plexicaudatum ASF492	Firmicutes bacterium ASF500
L-alanine	L-alanine biosynthesis III	L-alanine biosynthesis III	L-alanine biosynthesis III
L-arginine	L-arginine biosynthesis II (acetyl cycle)	L-arginine biosynthesis II (acetyl cycle) L-arginine biosynthesis I (via L-ornithine)	L-arginine biosynthesis II (acetyl cycle)
L-asparagine	L-asparagine biosynthesis I L-asparagine biosynthesis III (tRNA-dependent)	L-asparagine biosynthesis II superpathway of L-asparagine biosynthesis L-asparagine biosynthesis I	L-asparagine biosynthesis I
L-aspartate	L-aspartate biosynthesis	L-aspartate biosynthesis	L-aspartate biosynthesis
L-cysteine	L-cysteine biosynthesis I	L-cysteine biosynthesis I	L-cysteine biosynthesis I
L-glutamate	L-glutamate biosynthesis III L-glutamate biosynthesis I	L-glutamate biosynthesis I L-glutamate biosynthesis III	L-glutamate biosynthesis III L-glutamate biosynthesis I
L-glutamine	L-glutamine biosynthesis I	L-glutamine biosynthesis I	L-glutamine biosynthesis I
glycine	glycine biosynthesis III glycine biosynthesis IV glycine biosynthesis I	glycine biosynthesis II glycine biosynthesis I	glycine biosynthesis IV
L-histidine	L-histidine biosynthesis	L-histidine biosynthesis	🟦

FIGURE 7 | For each of three selected ASF organisms, this figure lists the biosynthetic pathways it contains for each amino acid. Multiple variants of amino-acid biosynthetic pathways are often known, as designated with roman numerals. The blue cell indicates that Firmicutes bacterium ASF500 does not contain a pathway for biosynthesis of L-histidine. The table is truncated for space considerations.

and PTools differ in the following respects. They use different reference databases of pathways and reactions: as of August 2021, KEGG contained 400 metabolic pathway modules versus 2,969 metabolic pathways in the MetaCyc DB; KEGG contained 11,603 reactions versus 17,412 in MetaCyc. Thus, MetaCyc has far wider coverage of metabolism (7.4 times as many pathways, 1.5 times as many reactions). MetaCyc pathways were derived from and cite 69,000 literature citations and 9,739 textbook-equivalent pages of mini-reviews that explain the role of each pathway; KEGG contains very few citations or mini-reviews. The KEGG algorithm for reactome and pathway prediction has never been published to our knowledge, therefore its processing steps are unknown, whereas the PTools pathway prediction algorithm has been published Karp et al. (2011). KEGG does not produce organism-specific metabolic network diagrams, but it does have a series of global overview maps that span all KEGG pathways, thereby showing many pathways that are not present in a particular organism.

MAPLE Takami et al. (2016) also uses KEGG for metagenome pathway analysis. Its pathway prediction method is based on the "module completion ratio," that is, assessing the evidence for pathway presence based solely on the fraction of reactions within a pathway that have an enzyme present. This simple method causes many false-positive predictions—particularly for larger pathway DBs such as MetaCyc—which is why we developed a more elaborate prediction method that considers factors such as pathway taxonomic range and key reactions Karp et al. (2011).

2.2 Searching Across an Organism Community

A suite of search tools enables scientists to perform basic searches across a set of microbiome-derived PGDBs, such as to determine which organisms in the community contain a given gene, protein, metabolite, or pathway. Such searches enable a researcher to quickly determine the functional roles played by different organisms in the community. In addition, more advanced searches are supported to find the organisms in the community containing genes, proteins, metabolites, or pathways matching specified conditions.

These searches are available in both the web and desktop modes of PTools, with somewhat different user interfaces available in the two modes. In web mode, the multi-organism search tools are present under the Tools > Search menu. For example, the Search Pathways command enables multi-organism pathway searches, the Search Genes, Proteins, or RNAs command enables multi-organism searches against genes and gene products, and the Search Compounds command enables multi-organism metabolite searches. By default these tools perform single-organism searches; to enable multi-organism searches, click the box next to "Search across Multiple Organisms/Databases."

For example, **Figure 1** shows the result of searching across the ASF PGDBs on BioCyc.org for all pathways whose name contains "tryptophan." Pathway searches can also search by ontology (such as for all detoxification pathways in the organism), pathway length, substrate(s) contained within the pathways, evidence code, and publication.

Gene/protein searches can search by sequence length, molecular weight, genome map position, pI, evidence code, cellular location, Gene Ontology (GO) term, publication, and by protein features.

Metabolite searches can search by ontology, monoisotopic mass, molecular weight, chemical formula, SMILES Anderson et al. (1988) substructure, and InChI Stein et al. (2003).

We are not aware of other tools that provide these types of multi-MAG search capabilities.

2.3 Comparative Analysis Operations on a Microbiome

PTools provides an extensive set of comparative operations that can be run across a set of PGDBs for a microbial community. Each comparative operation generates a series of pre-defined tables. The comparative operations are available at BioCyc.org under Tools > Comparative Analysis. The comparison tables (some of which are appropriate for genomes, but not for MAGs) span these aspects of the selected PGDBs (table numbers refer to tables within the web pages):

- Organism comparison
  - Table 1: Database Summary Statistics (example in **Figure 2**)
  - Table 2: Phenotype Metadata
  - Table 3: Collection Metadata
  - Table 4: Annotation Metadata
- Reaction comparison
  - Table 1: Breakdown of Reactions by Type
  - Table 2: Reactions of Small Molecule Metabolism (SMM)
  - Table 3: Breakdown of SMM Reactions by Top-Level EC Category (example in **Figure 3**)
  - Table 4: Distribution of Isozymes across SMM Reactions
  - Table 5: Shared Reactions
  - Table 6: Unique Reactions
- Pathway comparison
  - Table 1: Breakdown of Pathways by Pathway Class (example in **Figure 4**)
  - Table 2: Shared Pathways (example in **Figure 5**)
  - Table 3: Unique Pathways (example in **Figure 5**)
  - Table 4: Pathway Holes
- Metabolite comparison
  - Table 1: All Compounds
  - Table 2: Shared Compounds
  - Table 3: Unique Compounds
  - Table 4: Statistics on the Frequency with which Different Compounds Appear in Different Metabolic Roles
- Gene/protein comparison
  - Table 1: Selected Gene/Protein Statistics
  - Table 2: Gene Annotation
  - Table 3: Frequency Distribution of Heteromultimers by Number of Unique Gene Products
  - Table 4: Enzymes
  - Table 5: Multifunctional Enzymes
  - Table 6: Gene Ontology
- Transporter comparison
  - Table 1: Transporters (example in **Figure 6**)
  - Table 2: Substrate Uptake (example in **Figure 6**)
  - Table 3: Substrate Efflux
  - Table 4: Multiple Transporters and Substrates
  - Table 5: Transcription
- Transcription unit and regulation comparison
  - Table 1: Number of Genes per Transcription Unit
  - Table 2: Number of Operons per Pathway
  - Table 3: Regulation

The preceding tables are computationally generated such that clicking hyperlinks within the tables will produce a new table with an expanded level of information. For example, clicking on the row name "Amino Acid Biosynthesis" in **Figure 4** will generate the table shown in **Figure 7**, which shows the biosynthetic pathways present in each organism for each amino acid.

A number of other tools (e.g., MEGAN) present summaries of pathway abundances across different metagenome samples. In contrast, PTools reports differences in pathway compositions of different MAGs; we are not aware of other tools that perform such comparisons.

2.4 Analysis of Meta-Transcriptomics and Meta-Metabolomics Data

In a PGDB for a single organism, the PTools-generated cellular overview diagram provides a visual summary of all the metabolic and transport capabilities of the organism. A rectangular outer border represents the cell membrane. For Gram-negative bacteria, this consists of a double membrane with an intervening periplasmic space. Transporters and other membrane proteins are drawn on the appropriate membrane. Within the interior, representing the cytosol, metabolic pathways are shown to the left, and a grid containing all reactions not assigned to any pathway appears to the right. Within the pathway section, pathways are organized according to the MetaCyc pathway ontology, with biosynthetic pathways to the left, energy metabolism pathways in the middle, and catabolic pathways to the right. These sections are further subdivided into functionally-based blocks. For example, within the biosynthetic section are separate blocks for Carbohydrate Biosynthesis and Secondary Metabolite Biosynthesis. Pathways generally flow downwards, and connections between pathways are mostly omitted. As the user zooms in on the diagram, more detail is shown. At the highest level of detail, pathway, metabolite, enzyme and gene names all become visible. Users can overlay omics data for an organism onto the cellular overview diagram to visualize experimental results in a metabolic context Paley et al. (2021).

For a community of organisms, the user can create a community overview diagram (within the desktop version

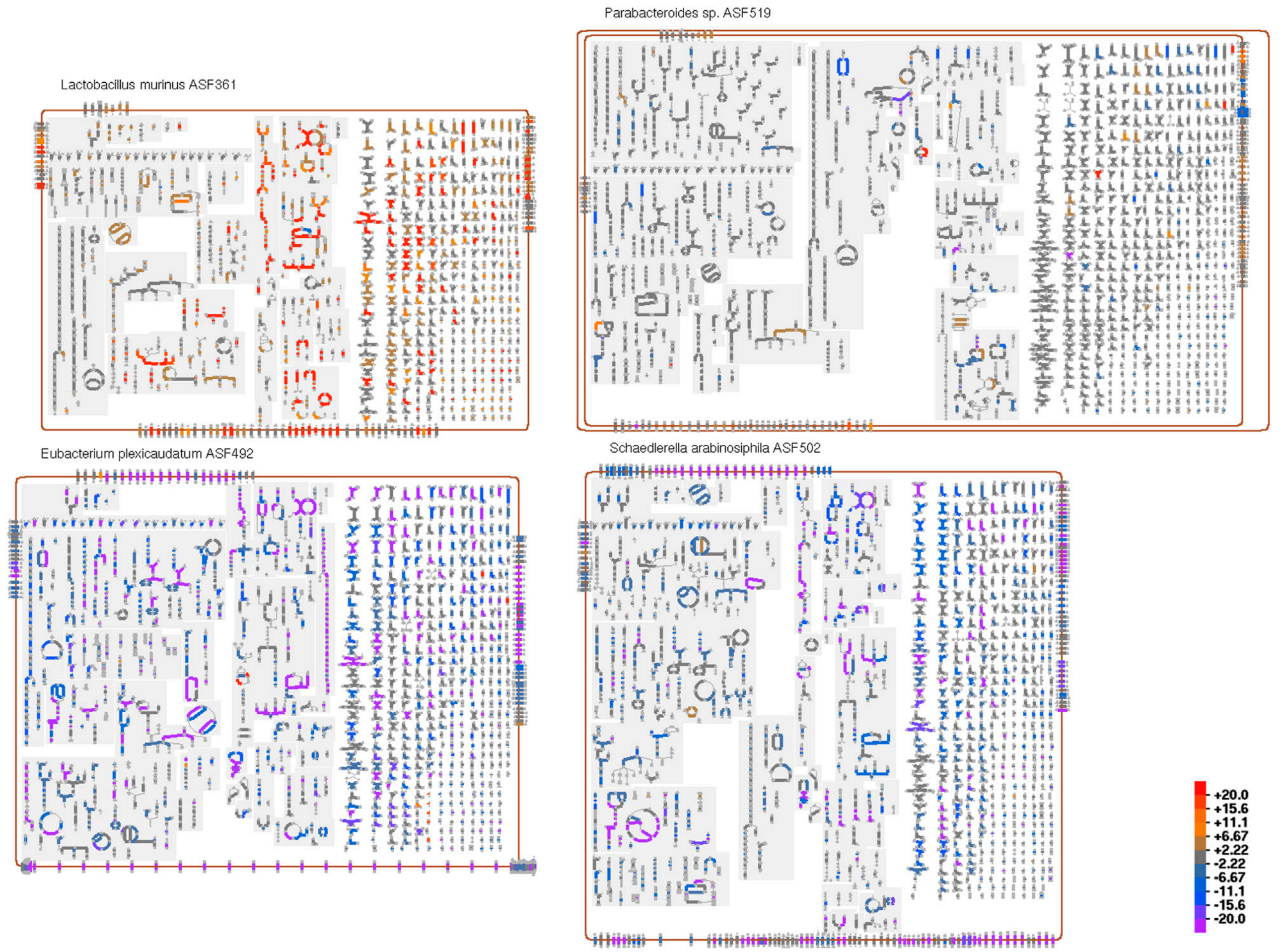
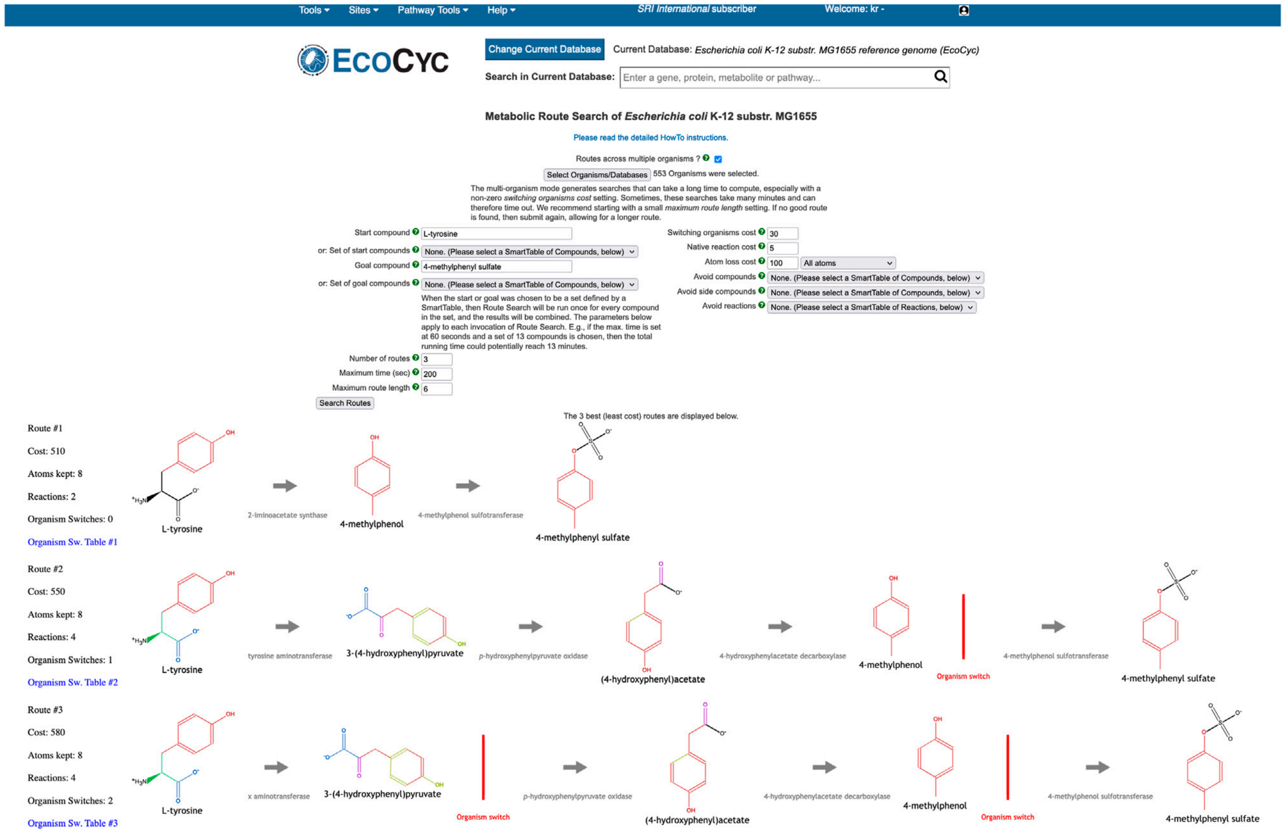

Lactobacillus murinus ASF361
Parabacteroides sp. ASF519
Eubacterium plexicaudatum ASF492
Schaedlerella arabinosiphila ASF502
+20.0
+15.6
+11.1
+6.67
+2.22
-2.22
-6.67
-11.1
-15.6
-20.0

FIGURE 8 | A community overview diagram for four of the bacterial species that make up the Altered Schaedler Flora model gut microbiome, overlaid with data from an example transcriptomics dataset. Reactions colored orange or red indicate genes with increased expression levels, whereas reactions colored blue or purple indicate genes with reduced expression levels.

of PTools only) that condenses and combines the overview diagrams from multiple organisms into a grid, forming a single large diagram. While initially shown at a low level of detail, users can interrogate the diagram via mouse-overs, zoom in to show more detail, or apply a range of highlight operations. Meta-transcriptomics or meta-metabolomics data can then be mapped onto this community overview diagram to visualize how experimental conditions affect the metabolism of the entire community. Omics data are supplied as a set of tab-delimited files, one per organism in the community, each with the first column containing gene or metabolite identifiers, and a single numeric data column (any additional columns in the file will be ignored), which can contain either absolute data (e.g., counts, intensities, concentrations) or relative data (e.g., ratios or log ratios of two experimental conditions or experiment *vs.* control).

Figure 8 shows a community overview diagram consisting of four organisms from the ASF microbial community overlaid with an example transcriptomics dataset. To identify the metabolic pathways that showed differential activity in response to altered gut environmental conditions, we conducted global transcriptome analysis (RNA-seq) of the ASF community recovered directly from wild type mice (129Sv6 background) along with IL-$10^{-/-}$ knockout mice on the same genetic background. IL-10 is a well-characterized immunomodulatory cytokine and IL-$10^{-/-}$ knockout conventional (i.e., complex microbiota) mice are known to exhibit an altered microbiota composition Overstreet et al. (2021). **Figure 8** shows the functional changes in the microbiome as the ASF responded to the altered immune status of the host as determined by identifying differentially expressed genes associated with specific metabolic pathways. The transcriptome dataset used for this analysis was generated by DeSeq2 Love et al. (2014).

In addition to visually drawing attention to particular metabolic reactions and pathways that undergo significant change, organism-wide effects also become apparent. For example, in this dataset we immediately notice that the metabolism of one organism, *Ligilactobacillus murinus*



Current Database: *Escherichia coli K-12 substr. MG1655 reference genome (EcoCyc)*

Search in Current Database: Enter a gene, protein, metabolite or pathway...

Metabolic Route Search of *Escherichia coli* K-12 substr. MG1655

Please read the detailed HowTo instructions.

Routes across multiple organisms ? ☑

Select Organisms/Databases 553 Organisms were selected.

The multi-organism mode generates searches that can take a long time to compute, especially with a non-zero *switching organisms cost* setting. Sometimes, these searches take many minutes and can therefore time out. We recommend starting with a small *maximum route length* setting. If no good route is found, then submit again, allowing for a longer route.

Start compound: L-tyrosine

or: Set of start compounds: None. (Please select a SmartTable of Compounds, below)

Goal compound: 4-methylphenyl sulfate

or: Set of goal compounds: None. (Please select a SmartTable of Compounds, below)

When the start or goal was chosen to be a set defined by a SmartTable, then Route Search will be run once for every compound in the set, and the results will be combined. The parameters below apply to each invocation of Route Search. E.g., if the max. time is set at 60 seconds and a set of 13 compounds is chosen, then the total running time could potentially reach 13 minutes.

Switching organisms cost: 30

Native reaction cost: 5

Atom loss cost: 100 All atoms

Avoid compounds: None. (Please select a SmartTable of Compounds, below)

Avoid side compounds: None. (Please select a SmartTable of Compounds, below)

Avoid reactions: None. (Please select a SmartTable of Reactions, below)

Number of routes: 3

Maximum time (sec): 200

Maximum route length: 6

Search Routes

The 3 best (least cost) routes are displayed below.

Route #1
Cost: 510
Atoms kept: 8
Reactions: 2
Organism Switches: 0
Organism Sw. Table #1

Route #2
Cost: 550
Atoms kept: 8
Reactions: 4
Organism Switches: 1
Organism Sw. Table #2

Route #3
Cost: 580
Atoms kept: 8
Reactions: 4
Organism Switches: 2
Organism Sw. Table #3

L-tyrosine
2-iminoacetate synthase
4-methylphenol
4-methylphenol sulfotransferase
4-methylphenyl sulfate
L-tyrosine
tyrosine aminotransferase
3-(4-hydroxyphenyl)pyruvate
p-hydroxyphenylpyruvate oxidase
(4-hydroxyphenyl)acetate
4-hydroxyphenylacetate decarboxylase
4-methylphenol
Organism switch
4-methylphenol sulfotransferase
4-methylphenyl sulfate
L-tyrosine
x aminotransferase
3-(4-hydroxyphenyl)pyruvate
Organism switch
p-hydroxyphenylpyruvate oxidase
(4-hydroxyphenyl)acetate
4-hydroxyphenylacetate decarboxylase
4-methylphenol
Organism switch
4-methylphenol sulfotransferase
4-methylphenyl sulfate

FIGURE 9 | MORS computed routes from L-tyrosine to 4-methylphenyl sulfate.

(i.e., *Lactobacillus murinus*) is generally increased (red/orange reactions), with the increases concentrated in certain pathway classes; the metabolism of two other organisms, *Eubacterium plexicaudatum* and *Schaedlerella arabinosiphila*, generally decreases (blue/purple). Mousing over any reaction will show a tooltip that includes the omics data values for all genes associated with that reaction. The user can zoom in on the diagram for a more detailed view of regions of interest.

We are not aware of other tools that can display metabolic network diagrams from multiple organisms simultaneously and paint these diagrams with meta-omics data.

2.5 Community Metabolic Route Search

Single-organism metabolic route search enables the discovery of the most optimal series of reactions (called routes), that will transform a starting compound into a goal compound, within the organism's reaction network. Optimal means that the reaction series has the lowest cost. The cost of a route is computed by a weighted combination of atom conservation, route length in terms of sequential reactions, and other parameters. To compute the number of conserved atoms, our RouteSearch algorithm Latendresse et al. (2014) uses pre-computed atom mappings Latendresse et al. (2012) of reactions that are available in MetaCyc. An atom mapping of a reaction gives a one-to-one correspondence of each non-hydrogen atom, from reactants to products. The more atoms are conserved, the more efficient the transformation from start to goal becomes, thus resulting in a lower cost.

The Multi-Organism RouteSearch (MORS) algorithm Krummenacker et al. (2019) is a recent extension of single-organism RouteSearch that enables route discovery across arbitrary sets of organisms, simultaneously searching across the union of reactions in their PGDBs. MORS enables dissecting the metabolic contributions originating from specific organisms, within the overall transformation performed by the microbial community. A typical use case is searching HumanCyc as well as the organisms in a microbiome body site, such as the gastrointestinal-tract, to investigate how a combination of organisms might synthesize a compound that is toxic to the host organism. To perform MORS searches at BioCyc.org, invoke Tools > Metabolism > Metabolic Route Search, and check the box next to "Routes across Multiple Organisms."

The MORS algorithm requires an additional input beyond the inputs to RouteSearch, namely the set of PGDBs to be searched. The reaction network searched by MORS will be the union of all reactions from that organism set. Additionally, the user may alter a new MORS parameter, the cost for "organism switching." A switch occurs when the two organism sets of two consecutive reactions in a route have no overlap. In other words, if the first reaction is known to occur in one set of organisms and the second reaction is occurring in a different organism set, but there is no organism that contains both reactions simultaneously, then the route must switch organisms by transferring the compound connecting both reactions from one organism to another (by unspecified transport mechanisms). Each discovered route is displayed horizontally across the web page, with the start compound on the left and the goal compound on the right. An organism switch is depicted in a route by a red vertical line. A SmartTable of the route can be generated, which lists the organism sets that provide the enzymes that catalyze each reaction along the route.

As an example, let us use BioCyc.org to examine how dietary L-tyrosine is transformed into toxic 4-methylphenyl sulfate, which is a protein fermentation product that has been modified in the liver and is implicated in kidney disease. As it is known that this toxin originates from L-tyrosine Selmer and Andrei (2001), the MORS start compound was set to L-tyrosine and the goal compound to 4-methylphenyl sulfate. We selected all organisms in the human microbiome body site called "gastrointestinal-tract" plus *Homo sapiens*. The total count of organisms was 553. The resulting top three routes are shown in **Figure 9**. All routes retain eight atoms. The first route consists of two reactions, and the other two routes consist of four reactions. The first route does not need an organism switch, because one microbe was found that can perform both reactions of this route. In the other two routes, the last reaction after the organism switch is found only in *Homo sapiens*. However, the reaction immediately before the switch occurs in 26 organisms in both routes. The third route found a different choice for the first reaction, which incurs the cost of an additional organism switch.

We are not aware of other tools that can perform multi-organism metabolic route search.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: biocyc.org.

AUTHOR CONTRIBUTIONS

PK supervised the work and authored much of the article. SP implemented the community search tools, the community overview and omics analysis tools, and some of the comparative analysis tools, and authored a portion of the manuscript. MK worked on the implementation of Multi-Organism Route Search and wrote the corresponding article section. AK worked on comparative analysis operations. GP and MW contributed data for analysis; GP contributed to manuscript revision.

FUNDING

Research reported in this publication was supported by the National Institute of General Medical Sciences of the National Institutes of Health (NIH) under award number R01GM080746 PK, by the National Institute of Allergy and Infectious Diseases of the NIH under award number R01AI160719 to PK; and by NIH/NCI grant R03 CA195305 and the DOD-Defense Threat Reduction Agency (DTRA) award HDTRA12110015 to GP.

ACKNOWLEDGMENTS

We thank Lawrance Chandra for his contributions.

REFERENCES

Anderson, E., Veith, G. D., and Weininger, D. (1988). Smiles: A Chemical Language and Information System. *J. Chem. Inf. Comput. Sci.* 28, 31–36.

Caspi, R., Billington, R., Keseler, I. M., Kothari, A., Krummenacker, M., Midford, P. E., et al. (2020). The MetaCyc Database of Metabolic Pathways and Enzymes - a 2019 Update. *Nucleic Acids Res.* 48, D445–D453. doi:10.1093/nar/gkz862

Dong, X., and Strous, M. (2019). An Integrated Pipeline for Annotation and Visualization of Metagenomic Contigs. *Front. Genet.* 10, 999. doi:10.3389/fgene.2019.00999

Eng, A., and Borenstein, E. (2019). Microbial Community Design: Methods, Applications, and Opportunities. *Curr. Opin. Biotechnol.* 58, 117–128. doi:10.1016/j.copbio.2019.03.002

Esvap, E., and Ulgen, K. O. (2021). Advances in Genome-Scale Metabolic Modeling toward Microbial Community Analysis of the Human Microbiome. *ACS Synth. Biol.* 10, 2121–2137. doi:10.1021/acssynbio.1c00140

Greenblum, S., Chiu, H. C., Levy, R., Carr, R., and Borenstein, E. (2013). Towards a Predictive Systems-Level Model of the Human Microbiome: Progress, Challenges, and Opportunities. *Curr. Opin. Biotechnol.* 24, 810–820. doi:10.1016/j.copbio.2013.04.001

Heinken, A., Basile, A., Hertel, J., Thinnes, C., and Thiele, I. (2021). Genome-scale Metabolic Modeling of the Human Microbiome in the Era of Personalized Medicine. *Annu. Rev. Microbiol.* 75, 199–222. doi:10.1146/annurev-micro-060221-012134

Huson, D. H., Beier, S., Flade, I., Górska, A., El-Hadidi, M., Mitra, S., et al. (2016). MEGAN Community Edition - Interactive Exploration and Analysis of Large-Scale Microbiome Sequencing Data. *Plos Comput. Biol.* 12, e1004957. doi:10.1371/journal.pcbi.1004957

Kanehisa, M., Furumichi, M., Sato, Y., Ishiguro-Watanabe, M., and Tanabe, M. (2021). KEGG: Integrating Viruses and Cellular Organisms. *Nucleic Acids Res.* 49, D545–D551. doi:10.1093/nar/gkaa970

Karp, P. D., Latendresse, M., and Caspi, R. (2011). The Pathway Tools Pathway Prediction Algorithm. *Stand. Genomic Sci.* 5, 424–429. doi:10.4056/sigs.1794338

Karp, P. D., Midford, P. E., Billington, R., Kothari, A., Krummenacker, M., Latendresse, M., et al. (2019). Pathway Tools Version 23.0 Update: Software for Pathway/genome Informatics and Systems Biology. *Brief Bioinform* 22, 109–126. https://academic.oup.com/bib/article-abstract/22/1/109/5669859?redirectedFrom=fulltext. doi:10.1093/bib/bbz104

Karp, P. D., Midford, P., Paley, S., Krummenacker, M., Billington, R., Kothari, A., et al. (2020). Pathway Tools Version 24.0: Integrated Software for Pathway/genome Informatics and Systems Biology. [v4]. *arXiv* , 1–98. http://arxiv.org/abs/1510.03964v4.

Keegan, K. P., Glass, E. M., and Meyer, F. (2016). MG-RAST, a Metagenomics Service for Analysis of Microbial Community Structure and Function. *Methods Mol. Biol.* 1399, 207–233. doi:10.1007/978-1-4939-3369-3_13

Konwar, K. M., Hanson, N. W., Pagé, A. P., and Hallam, S. J. (2013). MetaPathways: a Modular Pipeline for Constructing Pathway/genome Databases from Environmental Sequence Information. *BMC Bioinformatics* 14, 202. doi:10.1186/1471-2105-14-202

Krishnan, S., Alden, N., and Lee, K. (2015). Pathways and Functions of Gut Microbiota Metabolism Impacting Host Physiology. *Curr. Opin. Biotechnol.* 36, 137–145. doi:10.1016/j.copbio.2015.08.015

Krummenacker, M., Latendresse, M., and Karp, P. D. (2019). Metabolic Route Computation in Organism Communities. *Microbiome* 7, 89–96. doi:10.1186/s40168-019-0706-6

Latendresse, M., Krummenacker, M., and Karp, P. D. (2014). Optimal Metabolic Route Search Based on Atom Mappings. *Bioinformatics* 30, 2043–2050. doi:10.1093/bioinformatics/btu150

Latendresse, M., Ong, W. K., and Karp, P. D. (2022). Metabolic Modeling with MetaFlux. *Methods Mol. Biol.* 2349, 259–289. doi:10.1007/978-1-0716-1585-0_12

Latendresse, M., Malerich, J., Travers, M., and Karp, P. D. (2012). Accurate Atom-Mapping Computation for Biochemical Reactions. *J. Chem. Inf. Model.* 52 (11), 2970–2982. doi:10.1021/ci3002217

Lee, T. J., Paulsen, I., and Karp, P. (2008). Annotation-based Inference of Transporter Function. *Bioinformatics* 24, i259–67. http://bioinformatics.oxfordjournals.org/cgi/content/full/24/13/i259. doi:10.1093/bioinformatics/btn180

Levy, R., and Borenstein, E. (2014). Metagenomic Systems Biology and Metabolic Modeling of the Human Microbiome: from Species Composition to Community Assembly Rules. *Gut Microbes* 5, 265–270. doi:10.4161/gmic.28261

Love, M. I., Huber, W., and Anders, S. (2014). Moderated Estimation of Fold Change and Dispersion for RNA-Seq Data with DESeq2. *Genome Biol.* 15, 550. doi:10.1186/s13059-014-0550-8

Overstreet, A. C., Ramer-Tait, A. E., Suchodolski, J. S., Hostetter, J. M., Wang, C., Jergens, A. E., et al. (2020). Temporal Dynamics of Chronic Inflammation on the Cecal Microbiota in IL-10-/- Mice. *Front. Immunol.* 11, 585431. doi:10.3389/fimmu.2020.585431

Paley, S., Billington, R., Herson, J., Krummenacker, M., and Karp, P. D. (2021). Pathway Tools Visualization of Organism-Scale Metabolic Networks. *Metabolites* 11, 64. doi:10.3390/metabo11020064

Prakash, T., and Taylor, T. D. (2012). Functional Assignment of Metagenomic Data: Challenges and Applications. *Brief Bioinform* 13, 711–727. doi:10.1093/bib/bbs033

Romero, P. R., and Karp, P. D. (2004). Using Functional and Organizational Information to Improve Genome-wide Computational Prediction of Transcription Units on Pathway-Genome Databases. *Bioinformatics* 20, 709–717. doi:10.1093/bioinformatics/btg471

Seemann, T. (2014). Prokka: Rapid Prokaryotic Genome Annotation. *Bioinformatics* 30 (14), 2068–2069. doi:10.1093/bioinformatics/btu153

Selmer, T., and Andrei, P. I. (2001). p-Hydroxyphenylacetate Decarboxylase from *Clostridium difficile*. A Novel Glycyl Radical Enzyme Catalysing the Formation of P-Cresol. *Eur. J. Biochem.* 268, 1363–1372. doi:10.1046/j.1432-1327.2001.02001.x

SRI International (2021). Pathway Tools User's Guide Version 25.5. Available from SRI International.

Stein, S. E., Heller, S. R., and Tchekhovskoi, D. (2003). "An Open Standard for Chemical Structure Representation: The IUPAC Chemical Identifier," in *Proc. 2003 International Chemical Information Conference (Nimes)*, 131–143.

Sung, J., Kim, S., Cabatbat, J. J. T., Jang, S., Jin, Y. S., Jung, G. Y., et al. (2017). Global Metabolic Interaction Network of the Human Gut Microbiota for Context-specific Community-Scale Analysis. *Nat. Commun.* 8, 15393. doi:10.1038/ncomms15393

Takami, H., Taniguchi, T., Arai, W., Takemoto, K., Moriya, Y., and Goto, S. (2016). An Automated System for Evaluation of the Potential Functionome: MAPLE Version 2.1.0. *DNA Res.* 23, 467–475. doi:10.1093/dnares/dsw030

Tatusova, T., DiCuccio, M., Badretdin, A., Chetvernin, V., Nawrocki, E. P., Zaslavsky, L., et al. (2016). NCBI Prokaryotic Genome Annotation Pipeline. *Nucleic Acids Res.* 44, 6614–6624. doi:10.1093/nar/gkw569

Visconti, A., Le Roy, C. I., Rosa, F., Rossi, N., Martin, T. C., Mohney, R. P., et al. (2019). Interplay between the Human Gut Microbiome and Host Metabolism. *Nat. Commun.* 10, 4505. doi:10.1038/s41467-019-12476-z

Wannemuehler, M. J., Overstreet, A. M., Ward, D. V., and Phillips, G. J. (2014). Draft Genome Sequences of the Altered Schaedler flora, a Defined Bacterial Community from Gnotobiotic Mice. *Genome Announc* 2, e00287–14. doi:10.1128/genomeA.00287-14

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Karp, Paley, Krummenacker, Kothari, Wannemuehler and Phillips. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

METHODS
published: 26 April 2022
doi: 10.3389/fbinf.2022.828703

GMEmbeddings: An R Package to Apply Embedding Techniques to Microbiome Data

***Christine Tataru*[1]*, *Austin Eaton*[1] and *Maude M. David*[1,2]**

[1]Department of Microbiology, College of Science, Oregon State University, Corvallis, OR, United States, [2]Department of Pharmaceutical Sciences, College of Pharmacy, Oregon State University, Corvallis, OR, United States

Large-scale microbiome studies investigating disease-inducing microbial roles base their findings on differences between microbial count data in contrasting environments (e.g., stool samples between cases and controls). These microbiome survey studies are often impeded by small sample sizes and database bias. Combining data from multiple survey studies often results in obvious batch effects, even when DNA preparation and sequencing methods are identical. Relatedly, predictive models trained on one microbial DNA dataset often do not generalize to outside datasets. In this study, we address these limitations by applying word embedding algorithms (GloVe) and PCA transformation to ASV data from the American Gut Project and generating translation matrices that can be applied to any 16S rRNA V4 region gut microbiome sequencing study. Because these approaches contextualize microbial occurrences in a larger dataset while reducing dimensionality of the feature space, they can improve generalization of predictive models that predict host phenotype from stool associated gut microbiota. The GMEmbeddings R package contains GloVe and PCA embedding transformation matrices at 50, 100 and 250 dimensions, each learned using ~15,000 samples from the American Gut Project. It currently supports the alignment, matching, and matrix multiplication to allow users to transform their V4 16S rRNA data into these embedding spaces. We show how to correlate the properties in the new embedding space to KEGG functional pathways for biological interpretation of results. Lastly, we provide benchmarking on six gut microbiome datasets describing three phenotypes to demonstrate the ability of embedding-based microbiome classifiers to generalize to independent datasets. Future iterations of GMEmbeddings will include embedding transformation matrices for other biological systems. Available at: https://github.com/MaudeDavidLab/GMEmbeddings.

Keywords: microbiome, embedding, deep learning, machine learning, 16s sequencing

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Gavin Douglas,
Dalhousie University, Canada
Leandro de Mattos Pereira,
Fiocruz Pernambuco, Brazil

****Correspondence:***
Christine Tataru
tataruc@oregonstate.edu

Specialty section:
This article was submitted to
Genomic Analysis,
a section of the journal
Frontiers in Bioinformatics

Received: *03 December 2021*
Accepted: *11 February 2022*
Published: *26 April 2022*

Citation:
Tataru C, Eaton A and David MM
(2022) GMEmbeddings: An R Package
to Apply Embedding Techniques to
Microbiome Data.
Front. Bioinform. 2:828703.
doi: 10.3389/fbinf.2022.828703

1 INTRODUCTION

Gut microbiomes can impact the physiology of their human hosts by modifying the availability of molecules in the environment or through direct interactions with host cells Ruff et al. (2020). The most commonly used and cost-effective method to observe microbiomes is 16S rRNA amplicon sequencing, which allows researchers to observe which bacterial species are present in an environment, their relative quantities, and their relative evolutionary distances to one another Johnson et al. (2019).

While 16S rRNA amplicon sequencing has many strengths and provides insight into general microbiome compositions, analysis of 16S data is often impeded by small sample sizes paired with massive feature spaces. This can lead to underpowered studies and spurious associations being detected Schloss (2018), Ioannidis (2005), Fan et al. (2012). While meta-analyses of microbiome datasets generally support associations between microbiome community structure and disease, these interactions are often relatively weak and confounded by inter-study and individual microbiome variation Sharpton et al. (2021), Sze and Schloss (2016), Holman and Gzyl (2019), Duvallet et al. (2017), Wirbel et al. (2021). In addition, 16S analysis generally treats amplicon sequence variants (ASVs) or operational taxonomic units (OTUs), also generally called taxa, as independent features, despite the complex network of known relationships between bacterial species that influence their function Albright et al. (2021), Shoaie et al. (2013). By reducing dimensionality while simultaneously analyzing 16S sequences in the context of co-occurrence and co-abundance patterns across studies, we can increase the generalizability of classifiers and gain insight into microbiome community function.

Embedding has emerged as a method in natural language processing to both decrease the dimensionality of the feature space as well as consider co-occurrence relationships between entities across corpuses of documents. Embedding algorithms produce a numerical vector representation of every feature, then datasets can be projected into this newly defined numerical space. Vector representations can be learned in multiple ways–here we use both GloVe and Principal Component Analysis (PCA) algorithms on American Gut Project (AGP) data to produce two sets of embedding vectors. GloVe is an algorithm designed for natural language processing which learns numerical representation of features by projecting a co-occurrence matrix between features into a lower dimensional space. In the case of natural language, these numerical vector representations of words can then be used to cluster words by their shared meanings and relationships (e.g., king–male = queen) Pennington et al. (2014). PCA is a method used frequently in ecology which learns numerical representation of features such that samples fall along the axes of highest variation across the dataset Karl (1901). To some extent, this method takes into account co-abundances between taxa across samples to learn a representation.

We used both of these algorithms to create embedding transformation matrices. Numerical representations of 48,279 ASVs found in 15,709 samples were learned, and representations were created in 50, 100, and 250 dimensions.

We present GMEmbeddings, an open-source R package that transforms 16s microbiome data (ASV counts) into an embedding space that captures information about taxa co-occurrence or co-abundance patterns. GMEmbeddings currently contains embedding vectors to enable embedding of 16S V4 reads from the human gut microbiome. While the presented embedding matrices are not meant to be used to transform counts from other 16S regions or other biomes, future iterations will include other sets of embedding vectors. The package also enables the ability to interpret the learned numerical representations in the context of microbial metabolic pathways.

Previous iterations of this work used only forward reads from the American Gut Project that were ~ 150 bp long, resulting in less specificity and less coverage during the transformation into embedding space Tataru and David (2020). This iteration contains full length V4 reads (~ 250 bp) to improve performance, and is additionally more accessible through use of a complete R package.

2 METHODS

2.1 Making Embedding Transformation Matrix

2.1.1 Data Collection

Fastq files were downloaded from ftp://ftp.microbio.me/AmericanGut/20nov2020-demultiplexed-data/. Only sequences from stool samples were kept. Each folder represents a study, and studies with less than 50 samples were removed. Of the 72 folders originally associated with the AGP, 50 folders were kept. All gzipped FASTQ files were then collected from each folder, totalling 43,256 individual files sharing a combined size of 113 Gigabytes of space. The files were then filtered using Cutadapt in order to remove primers from the sequences Martin (2011). We removed the 515F-806R primer pairs: GTGYCAGCMG CCGCGGTAA (Fwd V4), GGACTACNVGGGTWTCTAAT (Rev V4), GTGCCAGCMGCCGCGGTAA (Fwd V4), GGACTACHVGGGTWTCTAAT (Rev V4) McDonald et al. (2018). In an effort to keep only the most accurate samples available, further filtration was performed to retain only files containing over 5,000 sequence reads.

2.1.2 Process Into ASVs

Fastq files were then processed using the DADA2 pipeline Callahan et al. (2016). In short, forward and reverse reads were trimmed to 140 base pairs, and maxEE and truncQ were set to 2. Reads that matched the phiX contamination database were removed Mukherjee et al. (2015). The error rates were then learned from the data, and later the core sample inference algorithm was applied to the filtered and trimmed sequence data. We then merged the forward and reverse reads together to obtain the full denoised sequences and removed any chimeras from the data. Lastly, bloom sequences obtained from the following link were removed: https://github.com/knightlab-analyses/bloom-analyses/blob/master/data/newbloom.all.fna.

2.1.3 Filter for Prevalence

After completing the quality filter and trimming steps in the DADA2 pipeline, we created a sequence table. The entries in the sequence table represented counts of the number of times the sequence read was detected in each of the samples. In total, there were unique 898,853 ASVs and 15,706 samples (merged forward and reverse reads). However, many of these ASVs had low rates of occurrence among the samples, so further filtering was done to remove reads that were detected in 10 or fewer samples. Filtering for blooms and prevalence reduced the size of our sequence table from 898,855 ASVs to 48,279 ASVs.

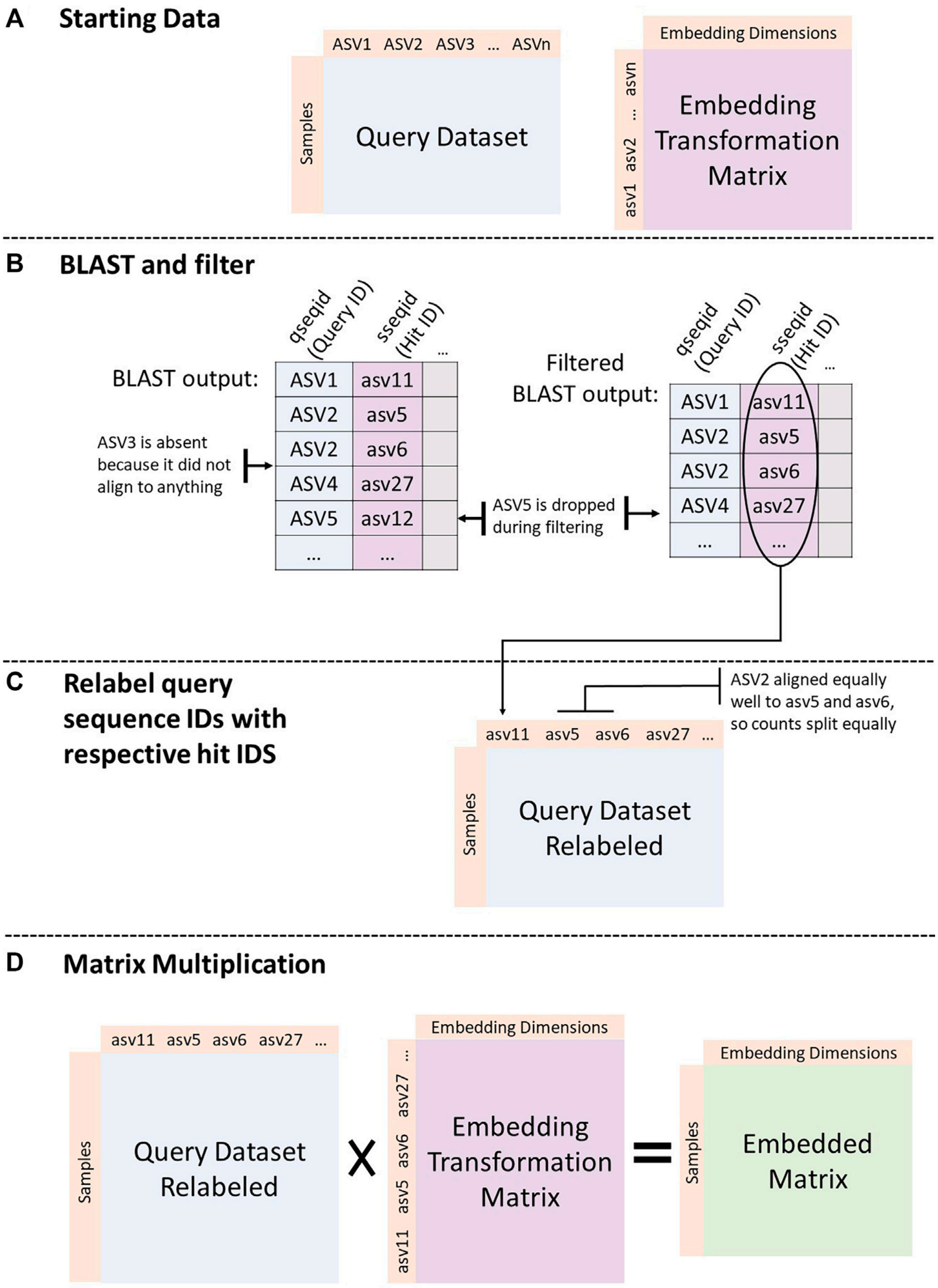


FIGURE 1 | Embedding a dataset. **(A)** Start with query dataset (sample by ASV counts) and an embedding transformation matrix (either from GloVe or PCA run on the American Gut Project data). **(B)** BLAST ASV sequences from the query dataset against the sequences in the transformation matrix. Filter the BLAST output to include only top hits (min E-value, max percent identity, and max alignment length) per query sequence. **(C)** Assign ids from BLAST hit column to query sequences. If there are more than one best hit, split counts from original query sequence between all best hits equally. **(D)** Matrix multiply the query count table with the embedding transformation matrix.

2.1.4 Calculate Co-Occurrence

Next, we created an ASV co-occurrence text file. Each line in the file contained the full length ASV sequence of all ASVs in one sample. The final file contained 15,706 lines, with one for every sample. In essence, we could think of each sample as having a specific sentence/catchphrase, and each line in this file contained the catchphrase of one sample. However, instead of words, each catchphrase was composed of the genetic sequences observed in each sample. This is the format for input files to the GloVe software (version 0.2). Pennington et al. (2014).

TABLE 1 | Maximum E-value and minimum length of alignment that were accepted in aligning sequences from each dataset to embedding transformation sequences.

Dataset	Max E-value accepted	Min. Length of alignment accepted
Halfvarson	3.72e-47	89
Schirmer	2.41e-86	133
M3	4.91e-125	129
Pilot	9.04e-122	110
Baxter	2.11e-133	110
Zeller	2.11e-133	121

2.1.5 GloVe Algorithm

The GloVe algorithm was then applied to our co-occurrence file in order to create an embedding transformation matrix of a set size Pennington et al. (2014). GloVe stands for global vectors for word representation and is an unsupervised learning algorithm to generate word embeddings from aggregated global word-word co-occurrence data. When the algorithm is applied to ASV sequences instead of words, the result is a vector representation for each ASV in the set that represents co-occurrence patterns. Cosine distances between vectors represents probability of co-occurrence of the corresponding ASVs. Short distances between sequences represent higher probabilities of ASVs sharing co-occurrence patterns, while greater distances represent lower probabilities of sequences sharing co-occurrence patterns. Distances are normalized by how frequently ASVs occur overall. The algorithm was run three times with output vector sizes of 50, 100, and 250.

2.1.6 PCA Algorithm

The PCA transformation matrices were obtained using SVD decomposition on the sample by ASV count table, and taking the (V^T) matrix. Prior to decomposition, ASV count vectors were mean centered and scaled to have a variance of 1, to avoid issues with heteroskedasticity. R/making_embedding_transformation_matrix/make_PCA_transformation_matrix.py.

2.1.7 Creating BLAST Database for ASVs in Embedding Database

The ASV sequences from the GloVe output were then used to create a FASTA formatted file. We then used the makeblastdb functionality of BLAST (Basic Local Alignment Search Tool) to generate a database based on the nucleotide sequences in our FASTA file. The database is used to check nucleotide sequences from other

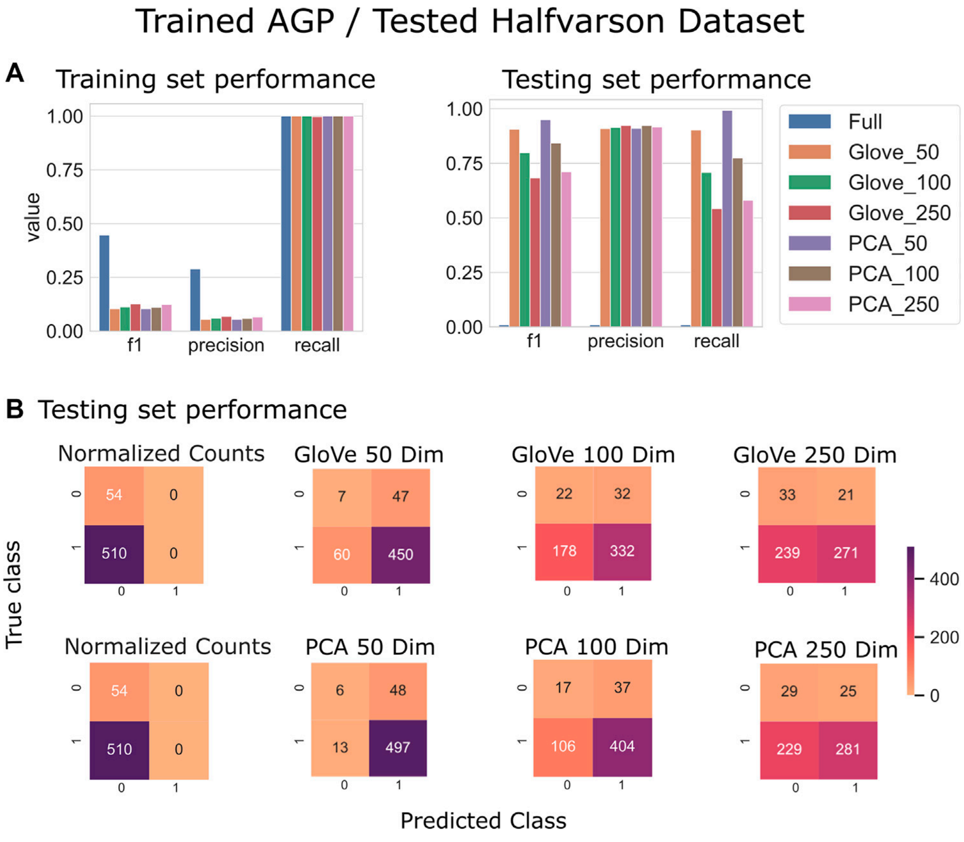
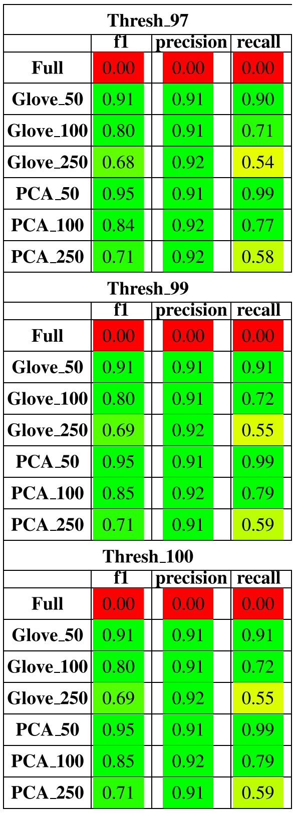


FIGURE 2 | Predicting IBD: Model trained on AGP data and tested on Halfvarson data. **(A)**: Models built using GloVe embedded data, PCA embedded data (50, 100, or 250 dimensions), or normalized ASV counts performance on training and testing sets. **(B)**: Confusion matrices showing the distribution of correct to predicted classes on the testing dataset.

TABLE 2 | Performance metrics of models trained on AGP data and tested on Halfvarson data using a 97, 99 and 100% sequence similarity threshold.



Thresh_97			
	f1	precision	recall
Full	0.00	0.00	0.00
Glove_50	0.91	0.91	0.90
Glove_100	0.80	0.91	0.71
Glove_250	0.68	0.92	0.54
PCA_50	0.95	0.91	0.99
PCA_100	0.84	0.92	0.77
PCA_250	0.71	0.92	0.58
Thresh_99			
	f1	precision	recall
Full	0.00	0.00	0.00
Glove_50	0.91	0.91	0.91
Glove_100	0.80	0.91	0.72
Glove_250	0.69	0.92	0.55
PCA_50	0.95	0.91	0.99
PCA_100	0.85	0.92	0.79
PCA_250	0.71	0.91	0.59
Thresh_100			
	f1	precision	recall
Full	0.00	0.00	0.00
Glove_50	0.91	0.91	0.91
Glove_100	0.80	0.91	0.72
Glove_250	0.69	0.92	0.55
PCA_50	0.95	0.91	0.99
PCA_100	0.85	0.92	0.79
PCA_250	0.71	0.91	0.59

studies against our “embedding database” sequences. BLAST database files can be found here: https://files.cgrb.oregonstate.edu/David_Lab/microbiome_embeddings/blastdb_fullseq/.

2.2 Transforming Query Data Into Embedding Space

Figure 1 shows the process implemented by the GMEmbeddings package to transform any query ASV count table into embedding space.

2.2.1 BLAST Alignment

To embed a query sequence table, we first created a corresponding FASTA file for the ASVs in that study. We then used BLAST to obtain all hits for each query sequence against the sequences in the embedding database:

```
blastn -db path_to_blastdb_dir/GloVe_emb_fullseq
-query path_to_fasta_file -out output_file_name
-outfmt "6 qseqid sseqid qseq sseq evalue bitscore length pident"
```

2.2.2 Filter BLAST Output

We filtered the BLAST hits to include only the top match per query sequence (lowest E-value, highest percent identity, and highest length of alignment). We kept matches with a maximum E-value threshold of $1*10^{-40}$. Using a 97% similarity threshold, the maximum E-value and minimum length of alignment observed and accepted are available in **Table 1**. See R/making embedding transformation matrix/scripts/filter blast hits.sh.

2.2.3 Relabel Query Sequence Ids With Respective Hit IDs

We relabeled query sequences with their respective hits in the embedding database. If the query sequence had only one top hit, we replaced its label with the label from the embedding database. If the sequence had multiple hits, we split its counts evenly among all of the top hits. If the sequence had over 100 hits that are all tied, it was dropped in an effort to increase the specificity of the method. If a query sequence had no hits, it was dropped. We also removed any sequences from the embedding transformation matrix that were never included as a top hit for any query sequence.

2.2.4 Matrix Multiplication

After the above processing, the column space of the query count table matched the rowspace of the embedding transformation matrix. We then took the dot product between the two matrices to obtain the embedded form of the query count table. In the final embedded table, rows were samples and columns were dimensions in embedding space. Ultimately, the embedded form of a matrix represents the original samples transformed into a mathematical space, taking into account the co-occurrence patterns of ASVs across a population.

2.3 Machine Learning Process

We trained seven random forest models per dataset to predict phenotype, one using normalized read counts, three using GloVe embedded data at 50, 100, and 250 dimensions, and three using PCA embedded data at 50, 100, and 250 dimensions.

Model feature spaces had to match between training and testing sets, so some modification of feature spaces was required:

1) For the model on normalized read counts, we included only the ASVs that were present in both datasets. We performed a BLAST alignment between the query dataset and AGP sequences using a 100% sequence similarity cutoff, and assigned the ASV full length sequences from AGP to the secondary dataset (similar to the process of embedding without matrix multiplication). Only the best hits were considered from the resulting BLAST alignment after imposing the 100% similarity cutoff. Read counts from the secondary dataset were split equally among all the tied best hits in the AGP data.
2) For the models based on embedded data, we followed the procedure outlined above in “Transforming Query Data into Embedding Space”.

Prior to being fed to a machine learning model, all data was normalized using an inverse hyperbolic sin function, $(\sin^{-1}(x) = \log(x + (x^2 + 1)^{1/2}))$, which mimics the function $\log(2x)$ almost exactly, except for behavior near 0. Below 1, the log function returns a negative value, and is undefined at 0. In contrast, inverse hyperbolic sin does not fall below 0 when the argument is low, and is defined as 0 at 0 Burbidge et al. (1988),

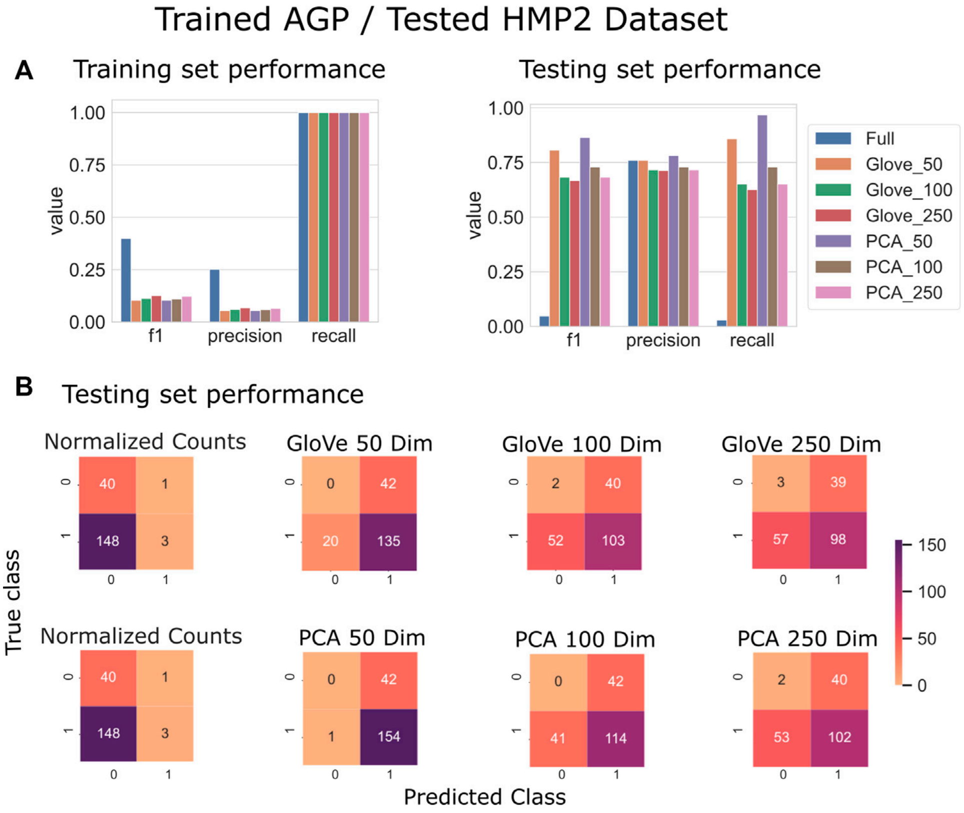


FIGURE 3 | Predicting IBD: Model trained on AGP data and tested on HMP2 data. **(A)**: Models built using GloVe embedded data, PCA embedded data (50, 100, or 250 dimensions), or normalized ASV counts performance on training and testing sets. **(B)**: Confusion matrices showing the distribution of correct to predicted classes on the testing dataset. (full).

Sankaran and Holmes (2019). This function allows log transformation of counts without the addition of pseudocounts.

All models were trained entirely on one large dataset and tested on an independent dataset, paired as follows (AGP: Halfvarson, AGP: HMP2, M3: Pilot, Baxter: Zeller). Datasets are described below. We used random forest predictive models, and set maximum tree depth to the square root of the number of input features and the number of trees to 100. Classes were weighted inversely to their a priori probabilities in the training dataset ($pos\_weight = \frac{N}{N_{pos}}$). For example, if the positive class is represented by 5% of the training samples, the weight on the positive class for the training classifier is 20, and the weight on the negative class is 1.

2.4 Metabolic Pathway Correlation

In order to interpret the dimensions that define embedding spaces, we correlated each dimension in embedding space to all prokaryotic metabolic pathways available in the KEGG database Kanehisa et al. (2015). An infographic describing the process is available in **Supplementary File S1**. First, we created a binary pathway (ko id) by gene (KO id) table describing which genes are present in which metabolic pathways using the KEGGREST API in R (A). Then, we created a matrix of gene (KO id) by ASVs by using PICRUSt Langille et al. (2013) (B). We multiplied the pathway by gene table (A) with the gene by ASV table (B) to obtain an ASV by pathway table (C), where higher values suggest a higher presence of a pathway in that organism. We then calculated the Spearman correlation between all columns of these two matrices to obtain a pathway by dimension correlation matrix. These values can be used to interpret dimensions in a biological context.

2.5 Test Dataset Descriptions

2.5.1 American Gut Project

In the American Gut Project (AGP) dataset, the majority of samples come from participants residing in the United States (n = 6,634) and the United Kingdom (n = 2,071), with a small number of samples generated from people living in other countries and territories. Participants in the United States inhabit largely urban areas (n = 7,317), with rural (n = 29) and mixed (n = 98) communities (2010 U.S. Census data based on participant ZIP codes) contributing in much smaller numbers. These participants also span a wide range of ages, race, and ethnicity. The read length of each sequence was around 150 base pairs which, when merged, resulted in a read length of 250 base pairs.

In the present study, we used a subset of 15,709 samples that were part of cohorts with > 50 samples in the consortium. These

TABLE 3 | Performance metrics of models trained on AGP data and tested on HMP2 data using a 97, 99 and 100% sequence similarity threshold.

	Thresh_97		
	f1	precision	recall
Full	0.05	0.76	0.03
Glove_50	0.82	0.76	0.88
Glove_100	0.69	0.72	0.66
Glove_250	0.67	0.71	0.63
PCA_50	0.87	0.78	0.99
PCA_100	0.74	0.73	0.75
PCA_250	0.69	0.72	0.66
	Thresh_99		
	f1	precision	recall
Full	0.05	0.76	0.03
Glove_50	0.79	0.97	0.67
Glove_100	0.32	0.92	0.19
Glove_250	0.19	1.00	0.11
PCA_50	0.97	0.98	0.96
PCA_100	0.47	0.95	0.32
PCA_250	0.25	1.00	0.14
	Thresh_100		
	f1	precision	recall
full	0.05	0.76	0.03
Glove_50	0.77	0.97	0.63
Glove_100	0.29	0.91	0.18
Glove_250	0.22	1.00	0.12
pca_50	0.98	0.98	0.98
pca_100	0.60	0.96	0.44
pca_250	0.35	1.00	0.21

samples contained a collective 898,855 ASVs. We removed ASVs present in less than 10 samples, and 50,425 ASVs remained, each 253 base pairs in length.

2.5.2 Halfvarson

The Halfvarson dataset Halfvarson et al. (2017) consists of 683 samples taken from 118 patients at multiple timepoints. Microbiome composition for each sample is ascertained by sequencing the V4 region of the 16S rRNA gene for a total of 248 million 16S rRNA gene amplicons and a total of 38,513 unique amplicon sequence variants (ASVs) at a read length of 253 bp.

In the present study, we filtered to include only samples with the diagnoses Crohn's disease (CD), ulcerative colitis (UC), and healthy control (HC). We used 608 of these samples from 118 patients (220 CD, 290 UC, and 54 HC samples). When embedding using a 97% similarity threshold, 15,998 ASVs (61%) from the Halfvarson dataset aligned to some read in the embedding dataset (**Supplementary File S2**). Each query sequence was aligned to a mean of 10.4 and a median of 2 embedding sequences (**Supplementary File S3**). These same statistics applied to the PCA transformed data.

2.5.3 HMP2

The HMP2 study Lloyd-Price et al. (2019) follows 132 subjscts for 1 year to generate longitudinal molecular profiles.

In the present study, we used only the 16S samples from the HMP2 study consisting of 197 samples taken from 83 individuals sampled at multiple timepoints (111 CD, 44 UC, 42 HC). This subsetted dataset contained a total of 5,869 unique ASVs at a length of 253 bp. When embedding using a 97% similarity threshold, 4,977 ASVs (85%) from the HMP2 dataset aligned to some read in the embedding dataset (**Supplementary File S2**). Each query sequence was aligned to a mean of 7.8 and a median of 2 embedding sequences (**Supplementary File S3**). These same statistics applied to the PCA transformed data.

2.5.4 M3

The M3 dataset Tataru et al. (2021) consists of 432 total samples from 72 age-matched sibling pairs. The pairs included one sibling diagnosed with ASD and the other who is developing typically (TD). The participants were between the ages of 2 and 8 years old. Researchers recorded 331 diet and lifestyle variables for each individual participating in the study. For each sample collected there were an additional 100 variables detailing lifestyle and dietary variables recorded. Samples were collected across the United States. Before filtration, the average depth of reads per sample measured 157,103 nucleotides (with a minimum of 23,321 and maximum of 996,530). The dataset contains a total of 5,265 ASVs (16S V4) at a length of 233 bp.

In the present study, all samples from the M3 dataset were used. When embedding using a 97% similarity threshold, 4,555 ASVs (87%) from the M3 dataset aligned to some read in the embedding dataset (**Supplementary File S2**). Each query sequence was aligned to a mean of 2 and a median of 1 embedding sequences (**Supplementary File S3**).

2.5.5 Pilot

The dataset obtained from the Pilot study David et al. (2021) contained 117 samples, of which, 60 were considered autism spectrum disorder (ASD) and 57 were controls. The population in the study consisted of age-matched sibling pairs between the ages of 2 and 7 years old, where the siblings needed to be within 2 years of each other. Of the 117 child subjects, there were 55 sibling pairs, two sibling pairs accompanied by a third sibling with autism, and 5 singleton samples. Samples were collected from 24 states: California, Colorado, Florida, Georgia, Hawaii, Illinois, Indiana, Massachusetts, Maryland, Michigan, Minnesota, Missouri, North Carolina, Nebraska, New Jersey, Nevada, New York, Ohio, Pennsylvania, Tennessee, Texas, Utah, Washington, and Wisconsin. The dataset contains a total of 1,664 ASVs (16S V4) at a length of 233 bp.

In the present study, all samples from the Pilot dataset were used. When embedding using a 97% similarity threshold, 1,500 ASVs (90%) from the pilot dataset aligned to some read in the embedding dataset (**Supplementary File S2**). Each query sequence was aligned to a mean of 1.8 and a median of 1 embedding sequences (**Supplementary File S3**).

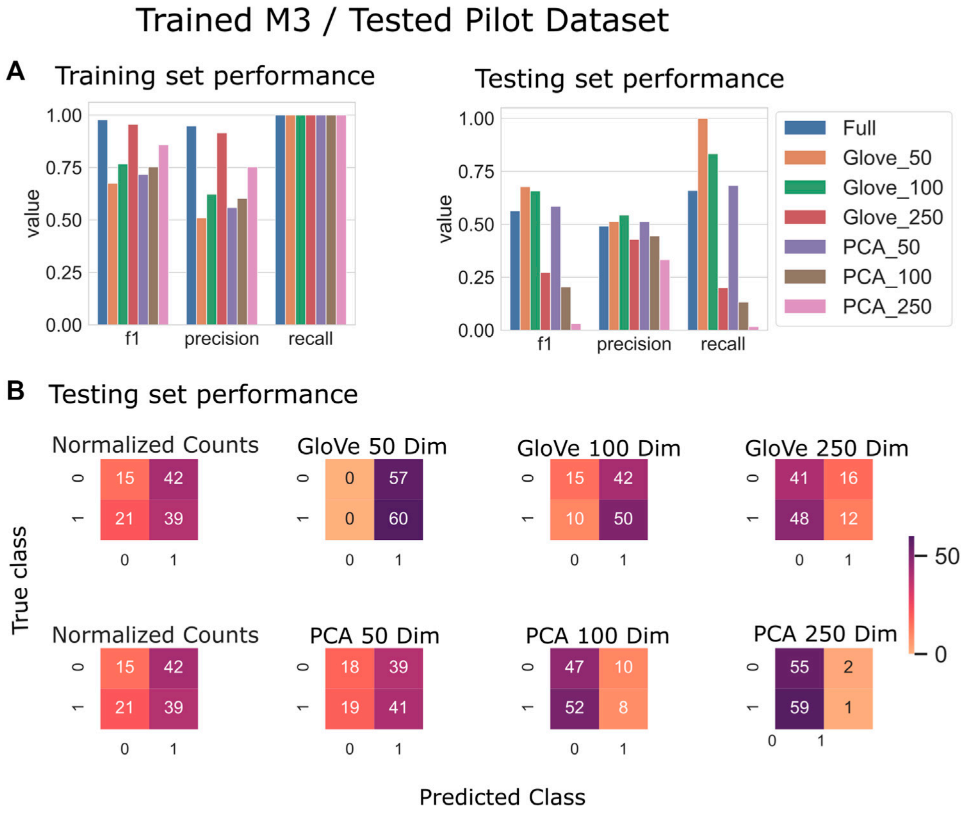


FIGURE 4 | Predicting autism spectrum disorder: Model trained on the M3 dataset and tested on the Pilot dataset. **(A)**: Models built using GloVe embedded data, PCA embedded data (50, 100, or 250 dimensions), or normalized ASV counts performance on training and testing sets. **(B)**: Confusion matrices showing the distribution of correct to predicted classes on the testing dataset.

2.5.6 Zeller

The Zeller dataset Zeller et al. (2014) consists of three populations of participants: 129 colonoscopy patients from a French hospital (53 CRC, 42 adenoma, and 61 controls), 38 colorectal cancer patients from a German hospital, and 5 healthy individuals living in Germany. A subset of these participants were chosen for fecal sample 16s sequencing by the original authors for stool 16S sequencing.

The present study used 75 control and 41 CRC samples, and this set of samples contained a total of 6,968 unique ASVs at a length of 253 bp. When embedding using a 97% similarity threshold, 5,581 ASVs (80%) from the Zeller dataset aligned to some read in the embedding dataset (**Supplementary File S2**). Each query sequence was aligned to a mean of 1.2 and a median of 1 embedding sequences (**Supplementary File S3**).

2.5.7 Baxter

The Baxter dataset Baxter et al. (2016) contains participants of ages 29–89 years with a median of 60 years. All patients were asymptomatic and were excluded if they had undergone surgery, radiation, or chemotherapy for current CRC prior to baseline samples or had inflammatory bowel disease, known hereditary non-polyposis CRC, or familial adenomatous polyposis. Colonoscopies were performed and fecal samples were collected from participants in four locations: Toronto (ON, Canada), Boston (MA, United States), Houston (TX, United States), and Ann Arbor (MI, United States).

The present study used 314 samples, (187 control and 127 CRC).

When embedding using a 97% similarity threshold, 7,879 ASVs (88%) from the Baxter dataset aligned to some read in the embedding dataset (**Supplementary File S2**). Each query sequence was aligned to a mean of 1.33 and a median of 1 embedding sequences (**Supplementary File S3**).

2.6 Metrics

2.6.1 Precision

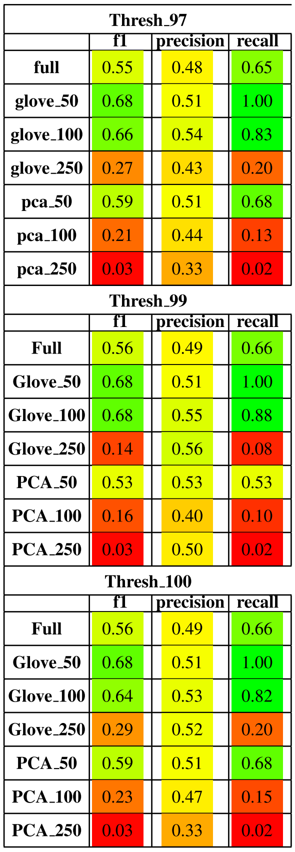
Precision is an indicator of a model's performance and refers to the number of true positives divided by the total number of positive predictions. Total number of positive predictions can be found by summing the number of true positives with the number of false positives.

$$precision = \frac{(true positives)}{(true positives)+(false positives)}$$

2.6.2 Recall

Recall gives indication of positive samples that the model has missed. It is calculated by dividing the number of true positives found by the model by the total number of positive samples that could have been made. The number of possible positive samples is the sum of true positives and false negatives.

TABLE 4 | Performance metrics of models trained on M3 data and tested on Pilot data using a 97, 99 and 100% sequence similarity threshold.



Thresh_97			
	f1	precision	recall
full	0.55	0.48	0.65
glove_50	0.68	0.51	1.00
glove_100	0.66	0.54	0.83
glove_250	0.27	0.43	0.20
pca_50	0.59	0.51	0.68
pca_100	0.21	0.44	0.13
pca_250	0.03	0.33	0.02
Thresh_99			
	f1	precision	recall
Full	0.56	0.49	0.66
Glove_50	0.68	0.51	1.00
Glove_100	0.68	0.55	0.88
Glove_250	0.14	0.56	0.08
PCA_50	0.53	0.53	0.53
PCA_100	0.16	0.40	0.10
PCA_250	0.03	0.50	0.02
Thresh_100			
	f1	precision	recall
Full	0.56	0.49	0.66
Glove_50	0.68	0.51	1.00
Glove_100	0.64	0.53	0.82
Glove_250	0.29	0.52	0.20
PCA_50	0.59	0.51	0.68
PCA_100	0.23	0.47	0.15
PCA_250	0.03	0.33	0.02

$$recall = \frac{(truepositives)}{(truepositives)+(falsenegatives)}$$

2.6.3 F1

The F1 score is the weighted average of precision and recall. It takes both false positives and false negatives into account and tells us a model's performance on a dataset. A perfect model would have an F1 score of 1.

$$F1Score = \frac{2^{*}(recall)^{*}(precision)}{(recall)+(precision)}$$

3 RESULTS

From the sequence counts from the American Gut Project (AGP), we created GloVe and PCA based embedding transformation matrices at 50, 100, and 250 dimensions. We then projected the sequence tables from six independent datasets, as well as that from the AGP, into both GloVe and PCA spaces. We then trained random forest predictive models to predict host phenotype using microbiome data in one of seven forms (GloVe embedded at 50, 100, and 250 dimensions, PCA embedded at 50, 100, and 250 dimensions, and normalized ASV counts). For each phenotype of inflammatory bowel disease (IBD), autism spectrum disorder (ASD), and colorectal cancer (CRC), models were trained on one dataset and tested on an independent set with no fine-tuning. No other metadata about samples was included in addition to microbiome data.

3.1 Inflammatory Bowel Disease Prediction

Random forest models were trained on the American Gut Project data, then tested on both the Halfvarson and HMP2 datasets Halfvarson et al. (2017), Lloyd-Price et al. (2019) to predict host phenotype of "healthy control" *vs.* "inflammatory bowel disease" which included Crohn's disease and ulcerative colitis. On the Halfvarson test dataset, models that used normalized ASV counts (full) had a higher training performance but much lower testing performance than any of the other methods, implying an overfit model (**Figure 2**; **Table 2**). Similarly, while larger models using 250 dimensions generalized to a testing set less well (f1 = 0.68–0.71), small models using only 50 dimensions were able to generalize much more effectively (f1 = 0.9–0.95). GloVe and PCA embedding methods exhibited largely similar performance, regardless of the choice of sequence alignment threshold (**Table 2**, **Supplementary File S4**).

On the HMP2 test dataset, a similar phenomenon emerged. The full model trained well but failed to generalize well to the testing dataset, and the larger embedding-based models performed less well than smaller embedding-based models (**Figure 3**). Increasing sequence similarity threshold resulted in removing more original sequences (**Supplementary File S3**), and in this case, decreased overall performance considerably (**Supplementary File S5**, **Table 3**). There was similar performance between GloVe and PCA embedding-based models when using a 97% sequence similarity threshold, but PCA based methods maintained a higher performance as similarity threshold increased, as compared to GloVe based models (**Supplementary File S5**, **Table 3**).

3.2 Austism Spectrum Disorder Prediction

Random forest models were trained on the M3 dataset and tested on the Pilot dataset (see Test Dataset Descriptions) Tataru et al. (2021), David et al. (2021) to classify the phenotype of participants with autism spectrum disorder and their typically developing siblings. While the full model outperformed other models during training, it obtained an F1 score of 0.56 in testing, while the GloVe_50, GloVe_100 models obtained higher F1 scores of 0.67, 0.66 respectively (**Figure 4**; **Table 4**). Increasing sequence similarity threshold improved the performance of GloVe_250 and PCA_100 models, and did not significantly effect other models (**Supplementary File S6**).

3.3 Colorectal Cancer Prediction

Random forest models were trained on the Baxter dataset and tested on the Zeller dataset (see Test Dataset Descriptions) Baxter et al. (2016), Zeller et al. (2014) to classify the phenotype of participants with colorectal cancer *vs.* healthy controls. The full model had higher training performance but failed to generalize to the test set, and this trend repeated in the models built on more features in both GloVe and PCA based models. The highest performing models were PCA_50 and GloVe_50 with F1 scores of 0.45 and 0.4 respectively (**Figure 5**; **Table 5**). Sequence similarity threshold had little effect on final performance (**Supplementary File S7**, **Table 5**).

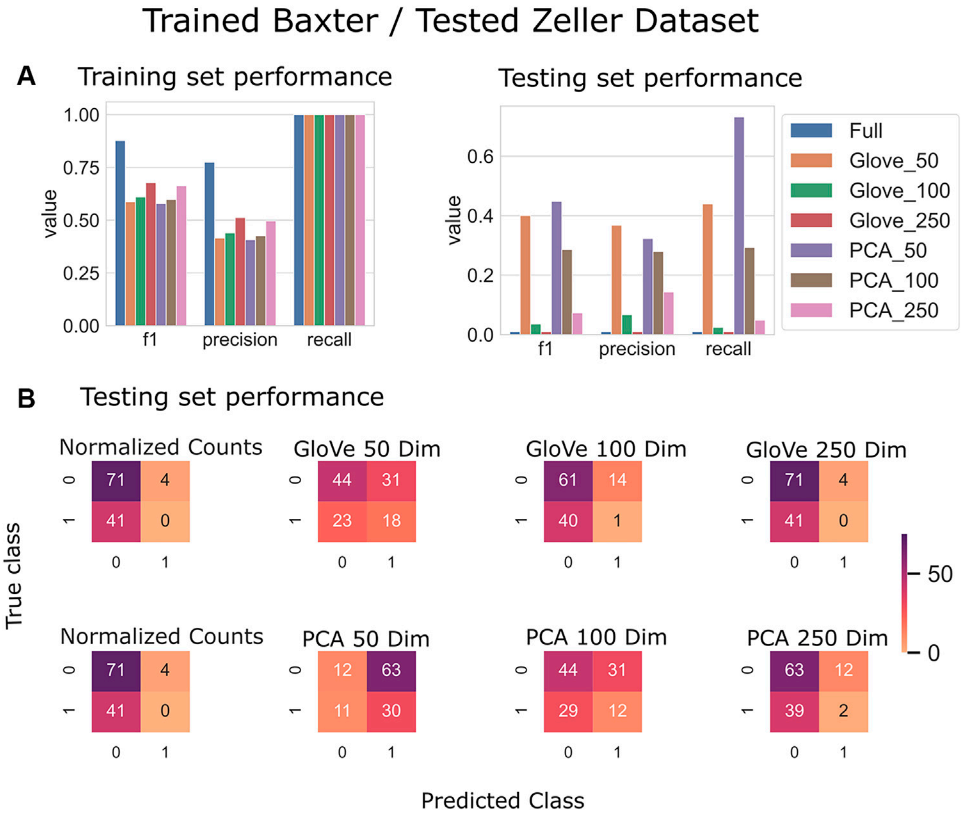


FIGURE 5 | Predicting Colorectal Cancer: Models trained on the Baxter dataset and tested on the Zeller dataset. **(A)**: Models built using GloVe embedded data, PCA embedded data (50, 100, or 250 dimensions), or normalized ASV counts performance on training and testing sets. **(B)**: Confusion matrices showing the distribution of correct to predicted classes on the testing dataset.

3.4 Metabolic Pathway Correlation

We correlated each embedding dimension with metabolic pathway genetic potential obtained from KEGG and PiCrust (See Methods). From this, we saw that dimensions all correlate to some groupings of metabolic pathways but not others (**Supplementary Files S8–S13**). This serves as a starting point in interpreting the biological functions of the otherwise mathematically defined dimensions in embedding space.

4 DISCUSSION

16S studies often result in spurious associations between specific ASVs and host phenotype due to necessarily small sample sizes in comparison to feature spaces and the treatment of ASVs as independent features Schloss (2018), Ioannidis (2005), Fan et al. (2012). Embedding methods can address these issues by defining a new feature space, which can be thought of as combinations of ASVs, where ASVs are considered similar if they share co-occurrence or co-abundance patterns across a large dataset Pennington et al. (2014). Applying embedding methods to smaller datasets can increase the generalization of predictive classifiers that use gut microbiome data, and may lead to new insights about overarching microbial properties that independent ASV counts do not otherwise reflect Tataru and David (2020).

The embedding methods presented here are aimed to address the curse of dimensionality caused by a large number of variables (ASVs) measured across a relatively small number of samples. Machine learning models with too many input variables can easily overfit the training data, as observed with the normalized count data in this study. In addition, having too many input variables can saturate distance metrics, giving datapoints unique feature subsets that cause them to all appear equidistant Bai (2014). By reducing the dimensionality of the input data, we show that models are able to learn generalizable microbial patterns of disease and avoid overfitting on biomarkers specific to single datasets.

In the datasets tested, 50 dimensions offered the best, most consistently high performance on test set predictions. PCA-based transformation obtained higher recall without significant drop in precision as compared to GloVe-based transformation, but, in these datasets, both obtained considerably improved performance over the method of

TABLE 5 | Performance metrics of models trained on Baxter data and tested on Zeller data using a 97, 99 and 100% sequence similarity threshold.

Thresh_97			
	f1	precision	recall
Full	0.00	0.00	0.00
Glove_50	0.40	0.37	0.44
Glove_100	0.04	0.07	0.02
Glove_250	0.00	0.00	0.00
PCA_50	0.45	0.32	0.73
PCA_100	0.29	0.28	0.29
PCA_250	0.07	0.14	0.05
Thresh_99			
	f1	precision	recall
Full	0.00	0.00	0.00
Glove_50	0.33	0.31	0.34
Glove_100	0.10	0.17	0.07
Glove_250	0.00	0.00	0.00
PCA_50	0.44	0.32	0.71
PCA_100	0.29	0.27	0.32
PCA_250	0.07	0.13	0.05
Thresh_100			
	f1	precision	recall
Full	0.00	0.00	0.00
Glove_50	0.40	0.36	0.44
Glove_100	0.03	0.06	0.02
Glove_250	0.00	0.00	0.00
PCA_50	0.47	0.33	0.78
PCA_100	0.33	0.30	0.37
PCA_250	0.07	0.14	0.05

using normalized ASV counts. In most datasets, increasing the sequence similarity threshold did not affect generalizable performance significantly, with the exception of the HMP2 dataset where increasing threshold decreased recall significantly. This may be due to the relatively low number of original sequences utilized in embedding under the more stringent threshold.

4.1 Comparison to Other Work

Kubinski et al. tested machine learning predictive models using a leave one study out cross-validation across 15 IBD datasets that performed 16s sequencing on stool samples. Their random forest models obtained average F1 scores of 0.72 across studies when using species level annotations Kubinski et al. (2021). A study from Manandhar et al. also obtained a similar F1 score of 0.74 on a hold-on test portion of the American Gut Projet dataset Manandhar et al. (2021). These performances are just below the IBD testing results from this study using the Embed 50 and PCA50 models (F1 = 0.82–0.95) on HMP2 and Halfvarson datasets respectively. Interestingly, a study from Hassouneh et al. that combined metagenomic features (as opposed to 16s) and included 3 independent datasets obtained an F1 score 0.87, suggesting that perhaps the integration of multiple datasets into the training data combined with the use of non-amplicon microbiome features may lead to increased accuracy Hassouneh et al. (2021).

Wu et al. tested the predictive power of 16s microbiome features in predictive autism by annotating OTUs from five studies at the genus level, then applying a random forest model. When training on one dataset and testing on another, the models' performance ranged from an F1 of 0.17–0.73 Wu et al. (2020). In comparison, the best performing model in the present study, GloVe_50, obtained an F1 of 0.68 on the testing data. Though they did not report F1 scores, other studies have reported surprisingly high values for area under the receiver operating curve when predicting autism (AUC = 0.93 and 0.98)Ding et al. (2020), Dan et al. (2020). This exceedingly high performance may be attributable to the sampling strategy, where ASD participants were recruited from the local hospital and typically developing participants from local kindergardens.

Wu et al. created a classifier that used fecal microbiome 16s sequences as well as age, sex, and BMI to distinguish patients with adenomas from colorectal cancer patients, and obtained an F1 score of 0.72. Models with equivalent hyperparameters and feature inputs trained on additional datasets also obtained F1 scores of 0.77 and 0.72) Wu et al. (2021). This is in line with the training F1 score obtained from the full model in this study from the Baxter data (F1 = 0.86) but higher than the training scores obtained from embedding methods (F1 = 0.58–0.68). Zhou et al. trained a random forest classifier to differentiate CRC from healthy controls using the same Baxter dataset presented in this study, and obtained an F1 score of 0.41, which is in the range of the F1 scores obtained here when testing the PCA50 model on an independent dataset (F1 = 0.43) Zhou et al. (2021). Neither of these studies tested their pre-trained models on independent datasets, so their true generalization capacity remains untested.

4.2 Limitations

This study used only the American Gut Project data to form the embedding transformation matrices. Integration of other, independent datasets would likely make the transformation process even more generalizable, especially to populations outside the United States.

In addition, Dada2 processing of reads and error model learning was performed on all the sequencing runs from the American Gut Project simultaneously in order to obtain one set of ASVs for all samples. This resulted in over 800,000 ASVs, most of which were not present in more than 10 samples. Learning an error model per sequencing run may have resulted in a lower rate of chimeric ASVs, which may have seen higher presence across samples Callahan et al. (2016).

While data transformed with either PCA or GloVe did provide grounds for more generalizable models, the interpretation of the learned representation remains a challenge. We find that correlations between learned taxa vector representations and metabolic pathway potential exist, however, each dimension correlates to a mixture of pathways, making direct implications difficult to conclude. In previous work, we found that mixtures of phylogenetic signal are also captured by learned dimensions Tataru and David (2020).

Utilizing other natural language processing methods for dimensionality reduction like deep learning networks may allow us to take advantage of other interpretation methods like attention, saliency maps, or explanation generation to obtain a more complete understanding of the system Sun et al. (2021).

Lastly, the embedding matrices provided are specific to human gut microbiomes as measured from stool–embedding matrices for other biomes will be provided in future iterations.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: Halfvarson dataset: accession number PRJEB18471: https://www.ncbi.nlm.nih.gov/bioproject/PRJEB18471, HMP2 dataset: accession number PRJNA389280: https://www.ncbi.nlm.nih.gov/bioproject/PRJNA389280, Zeller dataset: accession number PRJEB18471: https://www.ncbi.nlm.nih.gov/bioproject/PRJEB18471, Baxter dataset: accession number PRJNA290926: https://www.ncbi.nlm.nih.gov/bioproject/PRJNA290926, M3 dataset: https://files.cgrb.oregonstate.edu/David_Lab/M3/, Pilot dataset: https://files.cgrb.oregonstate.edu/David_Lab/M3_longitudinal_16s.

AUTHOR CONTRIBUTIONS

CT: Conception, design, analysis, interpretation, manuscript preparation. AE: Data collection, data processing, manuscript preparation. MD: Conception, design.

FUNDING

This work was supported in part with funds from NSF Award ID 2025457, NIH Award NIDA R44DA04395402, and the Larry W. Martin Joyce B. O'Neill Endowed Fellowship. The funding bodies did not play any role in the design, collection, analysis, interpretation or writing of the study.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.828703/full#supplementary-material

REFERENCES

Albright, M. B. N., Louca, S., Winkler, D. E., Feeser, K. L., Haig, S. J., Whiteson, K. L., et al. (2021). Solutions in Microbiome Engineering: Prioritizing Barriers to Organism Establishment. *ISME J.* doi:10.1038/s41396-021-01088-5

Bai, E.-w. (2014). "Big Data: The Curse of Dimensionality in Modeling," in Proceedings of the 33rd Chinese Control Conference, 6–13. doi:10.1109/chicc.2014.6896586

Baxter, N. T., Ruffin, M. T., Rogers, M. A., and Schloss, P. D. (2016). Microbiota-based Model Improves the Sensitivity of Fecal Immunochemical Test for Detecting Colonic Lesions. *Genome Med.* 8, 37. doi:10.1186/s13073-016-0290-3

Burbidge, J. B., Magee, L., and Robb, A. L. (1988). Alternative Transformations to Handle Extreme Values of the Dependent Variable. *J. Am. Stat. Assoc.* 83, 123–127. doi:10.1080/01621459.1988.10478575

Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J., and Holmes, S. P. (2016). DADA2: High-Resolution Sample Inference from Illumina Amplicon Data. *Nat. Methods* 13, 581–583. doi:10.1038/nmeth.3869

Dan, Z., Mao, X., Liu, Q., Guo, M., Zhuang, Y., Liu, Z., et al. (2020). Altered Gut Microbial Profile Is Associated with Abnormal Metabolism Activity of Autism Spectrum Disorder. *Gut Microbes* 11, 1246–1267. doi:10.1080/19490976.2020.1747329

David, M. M., Tataru, C., Daniels, J., Schwartz, J., Keating, J., Hampton-Marcell, J., et al. (2021). *Children with Autism and Their Typically Developing Siblings Differ in Amplicon Sequence Variants and Predicted Functions of Stool-Associated Microbes*. mSystems 6.

Ding, X., Xu, Y., Zhang, X., Zhang, L., Duan, G., Song, C., et al. (2020). Gut Microbiota Changes in Patients with Autism Spectrum Disorders. *J. Psychiatr. Res.* 129, 149–159. doi:10.1016/j.jpsychires.2020.06.032

Duvallet, C., Gibbons, S. M., Gurry, T., Irizarry, R. A., and Alm, E. J. (2017). Meta-analysis of Gut Microbiome Studies Identifies Disease-specific and Shared Responses. *Nat. Commun.* 8, 1784. doi:10.1038/s41467-017-01973-8

Fan, J., Guo, S., and Hao, N. (2012). Variance Estimation Using Refitted Cross-Validation in Ultrahigh Dimensional Regression. *J. R. Stat. Soc. Ser. B Stat Methodol* 74, 37–65. doi:10.1111/j.1467-9868.2011.01005.x

Halfvarson, J., Brislawn, C. J., Lamendella, R., Vázquez-Baeza, Y., Walters, W. A., Bramer, L. M., et al. (2017). Dynamics of the Human Gut Microbiome in Inflammatory Bowel Disease. *Nat. Microbiol.* 2, 17004. doi:10.1038/nmicrobiol.2017.4

Hassouneh, S. A., Loftus, M., and Yooseph, S. (2021). Linking Inflammatory Bowel Disease Symptoms to Changes in the Gut Microbiome Structure and Function. *Front. Microbiol.* 12, 673632. doi:10.3389/fmicb.2021.673632

Holman, D. B., and Gzyl, K. E. (2019). A Meta-Analysis of the Bovine Gastrointestinal Tract Microbiota. *FEMS Microbiol. Ecol. 95*. doi:10.1093/femsec/fiz072

Ioannidis, J. P. (2005). Why Most Published Research Findings Are False. *Plos Med.* 2, e124. doi:10.1371/journal.pmed.0020124

Johnson, J. S., Spakowicz, D. J., Hong, B. Y., Petersen, L. M., Demkowicz, P., Chen, L., et al. (2019). Evaluation of 16S rRNA Gene Sequencing for Species and Strain-Level Microbiome Analysis. *Nat. Commun.* 10, 5029. doi:10.1038/s41467-019-13036-1

Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. (2015). KEGG as a Reference Resource for Gene and Protein Annotation. *Nucleic Acids Res.* 44, D457–D462. doi:10.1093/nar/gkv1070

Kubinski, R., Djamen-Kepaou, J.-Y., Zhanabaev, T., Hernandez-Garcia, A., Bauer, S., Hildebrand, F., et al. (2021). *Benchmark of Data Processing Methods and Machine Learning Models for Gut Microbiome-Based Diagnosis of Inflammatory Bowel Disease*. bioRxiv. doi:10.1101/2021.05.03.442488

Langille, M. G., Zaneveld, J., Caporaso, J. G., McDonald, D., Knights, D., Reyes, J. A., et al. (2013). Predictive Functional Profiling of Microbial Communities Using 16S rRNA Marker Gene Sequences. *Nat. Biotechnol.* 31, 814–821. doi:10.1038/nbt.2676

Lloyd-Price, J., Arze, C., Ananthakrishnan, A. N., Schirmer, M., Avila-Pacheco, J., Poon, T. W., et al. (2019). Multi-omics of the Gut Microbial Ecosystem in Inflammatory Bowel Diseases. *Nature* 569, 655–662. doi:10.1038/s41586-019-1237-9

Manandhar, I., Alimadadi, A., Aryal, S., Munroe, P. B., Joe, B., and Cheng, X. (2021). Gut Microbiome-Based Supervised Machine Learning for Clinical Diagnosis of Inflammatory Bowel Diseases. *Am. J. Physiol. Gastrointest. Liver Physiol.* 320, G328–G337. doi:10.1152/ajpgi.00360.2020

Martin, M. (2011). Cutadapt Removes Adapter Sequences from High-Throughput Sequencing Reads. *EMBnet j.* 17, 10–12. doi:10.14806/ej.17.1.200

McDonald, D., Hyde, E., Debelius, J. W., Morton, J. T., Gonzalez, A., Ackermann, G., et al. (2018). American Gut: an Open Platform for Citizen Science Microbiome Research. mSystems 3

Mukherjee, S., Huntemann, M., Ivanova, N., Kyrpides, N. C., and Pati, A. (2015). Large-scale Contamination of Microbial Isolate Genomes by Illumina PhiX Control. *Stand. Genomic Sci.* 10, 18. doi:10.1186/1944-3277-10-18

Pearson, K. (1901). Liii. On Lines and Planes of Closest Fit to Systems of Points in Space. *Lond. Edinb. Dublin Phil. Mag. J. Sci.* 2, 559–572. doi:10.1080/14786440109462720

Pennington, J., Socher, R., and Manning, C. D. (2014). "Glove: Global Vectors for Word Representation," in Empirical Methods in Natural Language Processing (EMNLP), 1532–1543. doi:10.3115/v1/d14-1162

Ruff, W. E., Greiling, T. M., and Kriegel, M. A. (2020). Host-microbiota Interactions in Immune-Mediated Diseases. *Nat. Rev. Microbiol.* 18, 521–538. doi:10.1038/s41579-020-0367-2

Sankaran, K., and Holmes, S. P. (2019). Latent Variable Modeling for the Microbiome. *Biostatistics* 20, 599–614. doi:10.1093/biostatistics/kxy018

Schloss, P. D. (2018). *Identifying and Overcoming Threats to Reproducibility, Replicability, Robustness, and Generalizability in Microbiome Research*. mBio 9.

Sharpton, T. J., Stagaman, K., Sieler, M. J., Arnold, H. K., and Davis, E. W. (2021). Phylogenetic Integration Reveals the Zebrafish Core Microbiome and its Sensitivity to Environmental Exposures. *Toxics* 9. doi:10.3390/toxics9010010

Shoaie, S., Karlsson, F., Mardinoglu, A., Nookaew, I., Bordel, S., and Nielsen, J. (2013). Understanding the Interactions between Bacteria in the Human Gut through Metabolic Modeling. *Sci. Rep.* 3, 2532. doi:10.1038/srep02532

Sun, X., Yang, D., Li, X., Zhang, T., Meng, Y., Qiu, H., et al. (2021). *Interpreting Deep Learning Models in Natural Language Processing: A Review*. CoRR abs/2110.10470.

Sze, M. A., and Schloss, P. D. (2016). *Looking for a Signal in the Noise: Revisiting Obesity and the Microbiome*. mBio 7.

Tataru, C. A., and David, M. M. (2020). Correction: Decoding the Language of Microbiomes Using Word-Embedding Techniques, and Applications in Inflammatory Bowel Disease. *Plos Comput. Biol.* 16, e1008423. doi:10.1371/journal.pcbi.1008423

Tataru, C., Martin, A., Dunlap, K., Peras, M., Chrisman, B., Rutherford, E., et al. (2021). Longitudinal Study of Stool-Associated Microbial Taxa in Sibling Pairs with and without Autism Spectrum Disorder. *ISME Commun. Accepted*. doi:10.1038/s43705-021-00080-6

Wirbel, J., Zych, K., Essex, M., Karcher, N., Kartal, E., Salazar, G., et al. (2021). Microbiome Meta-Analysis and Cross-Disease Comparison Enabled by the SIAMCAT Machine Learning Toolbox. *Genome Biol.* 22, 93. doi:10.1186/s13059-021-02306-1

Wu, T., Wang, H., Lu, W., Zhai, Q., Zhang, Q., Yuan, W., et al. (2020). Potential of Gut Microbiome for Detection of Autism Spectrum Disorder. *Microb. Pathog.* 149, 104568. doi:10.1016/j.micpath.2020.104568

Wu, Y., Jiao, N., Zhu, R., Zhang, Y., Wu, D., Wang, A. J., et al. (2021). Identification of Microbial Markers across Populations in Early Detection of Colorectal Cancer. *Nat. Commun.* 12, 3063. doi:10.1038/s41467-021-23265-y

Zeller, G., Tap, J., Voigt, A. Y., Sunagawa, S., Kultima, J. R., Costea, P. I., et al. (2014). Potential of Fecal Microbiota for Early-Stage Detection of Colorectal Cancer. *Mol. Syst. Biol.* 10, 766. doi:10.15252/msb.20145645

Zhou, J., Ye, Y., and Jiang, J. (2021). Kernel Principal Components Based cascade forest towards Disease Identification with Human Microbiota. *BMC Med. Inform. Decis. Mak* 21, 360. doi:10.1186/s12911-021-01705-5

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Tataru, Eaton and David. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 27 April 2022
doi: 10.3389/fbinf.2022.864597

Taxonomy Informed Clustering, an Optimized Method for Purer and More Informative Clusters in Diversity Analysis and Microbiome Profiling

Antonios Kioukis[1], *Mohsen Pourjam*[2], *Klaus Neuhaus*[2] *and Ilias Lagkouvardos*[2,3]*

[1]Medical School, University of Crete, Heraklion, Greece, [2]Core Facility Microbiome, ZIEL – Institute for Food & Health, Technical University Munich, Freising, Germany, [3]Institute of Marine Biology, Biotechnology and Aquaculture, Hellenic Centre for Marine Research, Heraklion, Greece

Bacterial diversity is often analyzed using 16S rRNA gene amplicon sequencing. Commonly, sequences are clustered based on similarity cutoffs to obtain groups reflecting molecular species, genera, or families. Due to the amount of the generated sequencing data, greedy algorithms are preferred for their time efficiency. Such algorithms rely only on pairwise sequence similarities. Thus, sometimes sequences with diverse phylogenetic background are clustered together. In contrast, taxonomic classifiers use position specific taxonomic information in assigning a probable taxonomy to a given sequence. Here we introduce Taxonomy Informed Clustering (TIC), a novel approach that utilizes classifier-assigned taxonomy to restrict clustering to only those sequences that share the same taxonomic path. Based on this concept, we offer a complete and automated pipeline for processing of 16S rRNA amplicon datasets in diversity analyses. First, raw reads are processed to form denoised amplicons. Next, the denoised amplicons are taxonomically classified. Finally, the TIC algorithm progressively assigning clusters at molecular species, genus and family levels. TIC outperforms greedy clustering algorithms like USEARCH and VSEARCH in terms of clusters' purity and entropy, when using data from the Living Tree Project as test samples. Furthermore, we applied TIC on a dataset containing all *Bifidobacteriaceae*-classified sequences from the IMNGS database. Here, TIC identified evidence for 1000s of novel molecular genera and species. These results highlight the straightforward application of the TIC pipeline and superior results compared to former methods in diversity studies. The pipeline is freely available at: https://github.com/Lagkouvardos/TIC.

Keywords: taxonomic classification, microbial diversity, clustering, microbiome analysis, amplicon sequencing, NGS processing pipeline

OPEN ACCESS

Edited by:
Nikos Kyrpides,
Lawrence Berkeley National Laboratory, United States

Reviewed by:
Thorsten Stoeck,
University of Kaiserslautern, Germany
George Tsiamis,
University of Patras, Greece

****Correspondence:***
Ilias Lagkouvardos
ilias.lagkouvardos@tum.de

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *28 January 2022*
Accepted: *31 March 2022*
Published: *27 April 2022*

Citation:
Kioukis A, Pourjam M, Neuhaus K and Lagkouvardos I (2022) Taxonomy Informed Clustering, an Optimized Method for Purer and More Informative Clusters in Diversity Analysis and Microbiome Profiling. Front. Bioinform. 2:864597. doi: 10.3389/fbinf.2022.864597

1 INTRODUCTION

Today, profiling of microbial communities is often conducted by inexpensive and high throughput DNA-sequencing (i.e., next generation sequencing, NGS). These profiling techniques often rely on amplifying target marker genes by using the polymerase chain reaction (PCR) and subsequent parallel sequencing (Nocker et al., 2007). The obtained sequences are then compared to gene databases for probable taxonomic assignment. All assigned sequences of a sample result in a microbial profile. Since many years, the 16S rRNA gene is the primary target for most microbiome and diversity studies due to its versatility and phylogenetic information density (Woese et al., 1980). This technique can even resolve the microbial profile down to strain level, as shown in a study of Johnson et al. (2019).

In common approaches, sequence reads are usually *de novo* clustered into groups based on their sequence similarity (Blaxter et al., 2005), (Porter and Hajibabaei, 2018). Subsequently, the centroids of these similarity groups are classified to the closest known taxonomic level, obtaining so called Operational Taxonomic Units (OTUs). To form these clustered groups, multiple methods have been proposed. Several are based on calculating pairwise sequence similarities from multiple sequence alignments using UPGMA or neighbor-joining algorithms (Liu et al., 2009), (Li, 2015). However, these algorithms are computationally demanding processes and not the fastest in finding similar sequences in multiple sequence alignments, especially when using large similarity matrices as needed in microbiome studies. Thus, heuristic distance-based greedy clustering (DGC) and abundance-based greedy clustering (AGC) algorithms have been developed that produce the required clustering with a single pass through the data and are much faster (Edgar, 2010), (Rognes et al., 2016). Taken together, compromises must be taken between accurate and thorough methods on one side and fast analysis methods on the other side. The shortcomings of the DGC and AGC algorithms follow from their single pass through the data. For instance, these algorithms choose the first amplicon from the sequence pool and take it as the first OTU centroid. The next sequence is compared to the first based solely on similarity. If sufficiently similar, the sequence is added to the centroid. In case the sequence is too different, a second centroid (second OTU) is initiated. Thus, an OTU is formed by adding sequences being similar to the centroid above a defined threshold. This step is repeated with the remaining, not yet clustered sequences until all are assigned to OTUs (Edgar, 2010), (Rognes et al., 2016). Hence, the order of sequences in each data set strongly influences the resulting clustering output. The sequential addition of new sequences to existing OTUs might even sort sequences into different OTUs even though they have a significant similarity. However, these sequences are never evaluated together due to the sequential nature of the process. Ultimately, this causes random variation in microbial community assignments (Koeppel and Wu, 2013). While preordering the sequences based on their abundance in the dataset increases the reproducibility of the clustering process (Edgar, 2013) this does not eliminate the possible misplacements of sequences in different OTUs (Edgar, 2013).

More recent approaches argue against the process of clustering and rather support the processing of sequences only by removal of chimeras and sequencing errors down to what is referred to as denoised sequences. Two algorithms are the most common used for denoising, DADA2 (Callahan et al., 2016) and UNOISE3 (Edgar, 2016). The results are, as said, denoised sequences in both cases, while the creators of DADA2 call their result amplicon sequence variant (ASV) and the author of USEARCH names them zero-radius OTU (zOTU).

In any case, after having processed all sequences to a list of OTUs representatives or denoised sequences they are classified to their closest taxonomy possible. The outcome of this process is dependent on initial primer choice (i.e., the variable region of the 16S rRNA gene used), the software chosen to perform each task and reference databases used (Abellan-Schneyder et al., 2021), including RDP (Lan et al., 2012) or SILVA (Quast et al., 2012). Unfortunately, reference databases have partially different taxonomic nomenclature, differ in update frequency, and unavoidable errors in such reference databases are affecting the quality and comparability of the results (Sierra et al., 2020). For example, the database GreenGenes DeSantis et al. (2006) has not been updated since 2013 and should not be used anymore. Through the years, SILVA and RDP have distinguished themselves and are currently the most frequently used by classifiers.

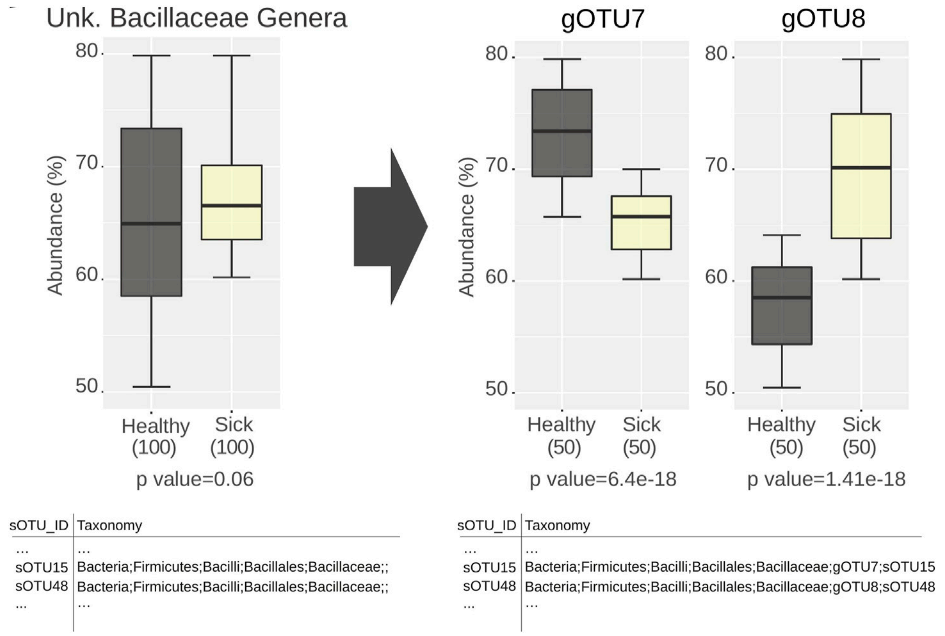
Taxonomic classification performs well on sequences from characterized bacteria and archaea, correctly assigning them up to their genus level. However, unknown sequences not represented in the reference databases result in incomplete taxonomic paths. In every sample, there will be sequences from yet undescribed taxa. For instance, in gut samples, the proportion of OTUs that can be assigned to fully described species ranges from 35 to 65%. For environmental samples, this ratio is even lower (Lagkouvardos et al., 2017). For analysis of ecological patterns in higher taxonomic levels (e.g., family), sequences with incomplete taxonomic classification are collectively binned intro groups of "Unknown taxon" or simply discarded. Obviously, these problems limit the resolution of the biological signal that could have been extracted from available sequence data (**Figure 1**).

Here we present "Taxonomy Informed Clustering" (TIC), a novel tool that flips the above paradigm, i.e., classifying after clustering. Here, we first taxonomically classify each sequence before any clustering is conducted. The taxonomic information acquired and now attached to each sequence acts both as a guide and as a limit in an incremental clustering process (**Figure 2**). Thus, the dataset is divided into subsets following the assigned taxonomies and, working within each subset, we avoid merging sequences from diverse lineages together. As a result, the created clusters have a higher purity and their number resembles more that of the intrinsic community structure. The incremental clustering procedure also allows sequences with incomplete taxonomic classification to be positioned in the taxonomic tree, allowing for higher resolution in compositional comparisons of microbiome studies. Our novel tool, TIC, is offered as a complete set of scripts, allowing researchers to perform a thorough analysis from raw reads to compositional tables for subsequent comparisons (e.g., in alpha- and beta-diversity, *etc.*) within a single pipeline.

2 MATERIALS AND METHODS

2.1 Overview

The TIC-Pipeline consists of a setup (i.e., installation) and four processing steps: *1*) Processing raw reads from a study's FASTQ files, *2*) Extraction of the consensus 16S region, creation of zOTUs, and taxonomic classification up to the genus level, *3*) De-novo clustering based on taxonomic information (TIC) of the used zOTUs, *4*) Reporting the results from all the previous steps.



sOTU_ID	Taxonomy
...	...
sOTU15	Bacteria;Firmicutes;Bacilli;Bacillales;Bacillaceae;;
sOTU48	Bacteria;Firmicutes;Bacilli;Bacillales;Bacillaceae;;
...	...

sOTU_ID	Taxonomy
...	...
sOTU15	Bacteria;Firmicutes;Bacilli;Bacillales;Bacillaceae;gOTU7;sOTU15
sOTU48	Bacteria;Firmicutes;Bacilli;Bacillales;Bacillaceae;gOTU8;sOTU48
...	...

FIGURE 1 | Schematic representation of the shortcomings of missing detailed taxonomic assignments in microbiome analysis. OTUs missing taxonomic classification for a certain level (e.g., genus) are analyzed together under the unknown label. The resulting conclusions can be deceiving when the constituting natural divisions are present nonuniformly across conditions.

2.1.1 Pipeline Installation

The TIC-Pipeline is a mixture of bash commands, python, and R scripts connected by a main python script. An installer script handles the installation of the command-line tools and their dependencies. The installer also downloads the reference databases (SILVA v.138), and the necessary programs, which includes KronaTools v.2.8 (Ondov et al., 2011), rapidNJ v.2.3.2 (Simonsen et al., 2010), SINA v.1.7.2 (Pruesse et al., 2012), SortMeRNA v.2.1 (Kopylova et al., 2012), USEARCH v.10 (Edgar, 2010), and VSEARCH v.2.13.4 (Rognes et al., 2016). In addition, the installer uses a dedicated file-server hosting the tools and the databases to set up dependencies for the pipeline, guaranteeing availability without breakages. Taken together, users just run the installer script, which installs R libraries and the Python packages needed automatically. After installation, we suggest a test run to ensure that all dependencies are met.

The structure of the pipeline is modular. An easy to modify text file "config options.txt" contains the configuration options controlling the pipeline's flow. Configuration options include the number of threads to use, the current active mode (production or testing), or the input files' location. The user may also execute each pipeline's step independently, given that they provide correctly formatted data. For example, RDP classifications (Wang et al., 2007) could be used instead of the default SINA classifier. Detailed documentation of each option for all steps is given at the tool repository. Illustrations shown in the present manuscript directly correspond to generated outputs from the TIC-Pipeline.

2.1.2 Sample-wise Processing

At this step, raw sequencing data are processed to obtain unique amplicon sequences. Those are the basis for any downstream analysis. This process includes sequence trimming to remove primers, merging paired reads, and filtering sequences based on expected error thresholds. Default options are indicative only and users are expected to fine-tune the parameters according to their needs.

2.1.3 Overlapping Regions Detection and Taxonomic Classification

Sequences from different studies cannot always be directly compared as usage of different V-regions of the 16S rRNA genes results in sequences of different lengths and sometimes non-overlapping V-regions, which cannot be integrated. Matters are further complicated, even for sequences originating from the same method, due to the usage of diverse primers (and despite using the same V-regions) among studies. Reference Based Alignment (RBA), like SINA, has been used in the past to tackle this problem, effectively detecting any overlapping region among sequences from various studies and focusing the analysis on regions, which are represented most often (Lagkouvardos et al., 2014). Every sequence in the input dataset is aligned to the reference database (SILVA) producing a global alignment of 50,000 positions. By summing the number of aligned bases in each position of the multiple sequence alignment, the user can identify the most representative region, enabling the extraction of this region. The TIC pipeline provides an automatic calculation of this vector and plots the result, so the user can confidently identify the most informative region and set proper limits for the extraction of that region (**Figure 3**). The chosen region is used for taxonomic classification. The classification sub-process uses the Last

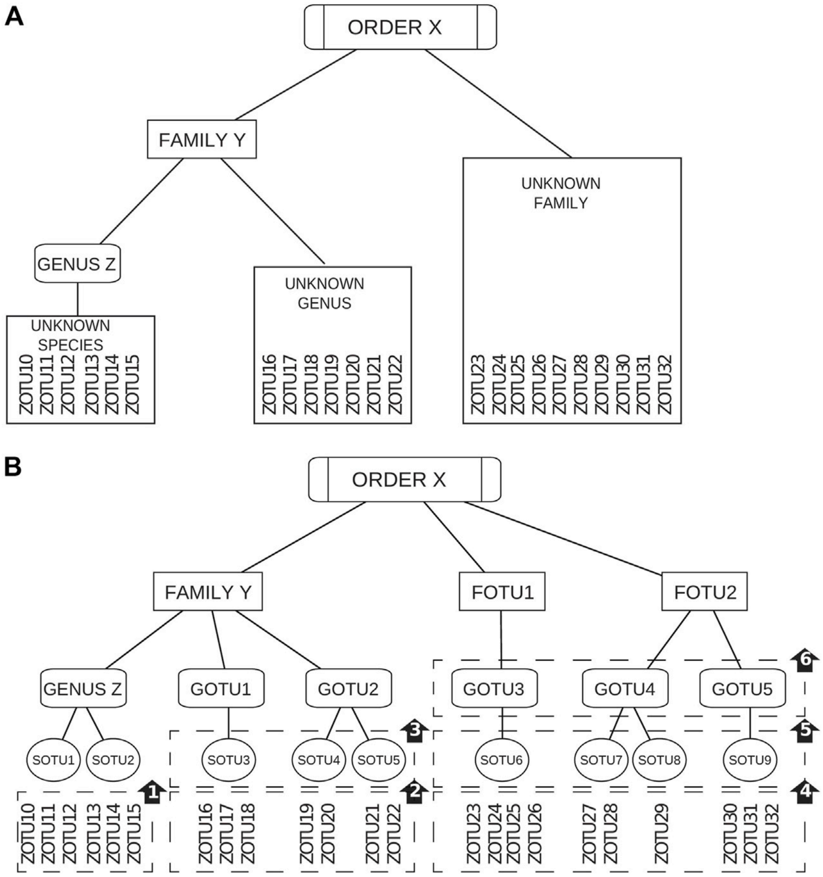

A
ORDER X
FAMILY Y
UNKNOWN FAMILY
GENUS Z
UNKNOWN GENUS
UNKNOWN SPECIES
ZOTU10 ZOTU11 ZOTU12 ZOTU13 ZOTU14 ZOTU15
ZOTU16 ZOTU17 ZOTU18 ZOTU19 ZOTU20 ZOTU21 ZOTU22
ZOTU23 ZOTU24 ZOTU25 ZOTU26 ZOTU27 ZOTU28 ZOTU29 ZOTU30 ZOTU31 ZOTU32
B
ORDER X
FAMILY Y
FOTU1
FOTU2
GENUS Z
GOTU1
GOTU2
GOTU3
GOTU4
GOTU5
SOTU1 SOTU2 SOTU3 SOTU4 SOTU5 SOTU6 SOTU7 SOTU8 SOTU9
1 2 3 4 5 6
ZOTU10 ZOTU11 ZOTU12 ZOTU13 ZOTU14 ZOTU15
ZOTU16 ZOTU17 ZOTU18 ZOTU19 ZOTU20 ZOTU21 ZOTU22
ZOTU23 ZOTU24 ZOTU25 ZOTU26 ZOTU27 ZOTU28 ZOTU29 ZOTU30 ZOTU31 ZOTU32

FIGURE 2 | Simplified representation of the TIC algorithm. **(A)** Sequences are divided based on their identified taxonomic level. **(B)** *1*) Denoised sequences within the same genus are clustered to produce sOTUs. *2*) Sequences of unknown genera within the same family are clustered into sOTUs. *3*) Produced sOTUs are further clustered into novel gOTUs. *4*) Sequences of the unknown family are first clustered into sOTUs. *5*) Those sOTUs produce gOTUs. *6*) fOTUs are formed from the gOTUs of an unknown family.

Common Ancestor (LCA) shared by at least 7 of the 10 closest sequences in the database to place a sequence.

2.1.4 Region Extraction and Denoising

Based on the user's evaluation of the region with the highest coverage among samples (recorded as parameter in the configuration file), each sequence is trimmed for positions outside this defined region. Trimmed sequences are pooled, dereplicated, and denoised using the UNOISE3 algorithm (Edgar, 2016) to create zOTUs. All denoised sequences are checked for valid 16S rRNA sequences by SortMeRNA using the SILVA bacteria and archaea databases. The taxonomic information derived from the previous step is added automatically to the header of the zOTU FASTA file and is used in the next step to guide the clustering.

2.1.5 Taxonomy Informed Clustering

A step-wise taxonomy-guided clustering was implemented to utilize position-specific taxonomic information for purer clusters. TIC's starting point is the pool of unique denoised sequences (zOTUs) with a recognized genus name (Gseqs). Gseqs are clustered within each genus to produce molecular species (sOTUs) within the identified genera. Afterwards, sequences that have been classified only up to the family level (Fseqs) are processed. In order to account for limitations in the taxonomic classification (i.e., missing levels), such Fseqs are first searched if they match any existing sOTU from Gseqs within the current family. In case a match is found, the taxonomy of the zOTU in question is updated if within a designated species cutoff level. However, sequences matching existing sOTUs above the designated genus cutoff level, but below the species level, are assumed to be novel sOTUs within the existing genera. Finally, Fseqs without a match, even at the genus level to existing sOTUs, are used to produce novel sOTUs that are next clustered again to novel gOTUs. Sequences with an unidentified taxonomic family follow the same procedure as before, but with the added layer of fOTUs. For instance, they are first matched against sOTUs, gOTUs, and known families of the same order. If no matches are found according to corresponding cutoffs, the unidentified sequences are designated as novel fOTUs (**Figure 4**).

Since there is no consensus on sequence similarity values between orders, classes, and phyla across all bacteria, TIC

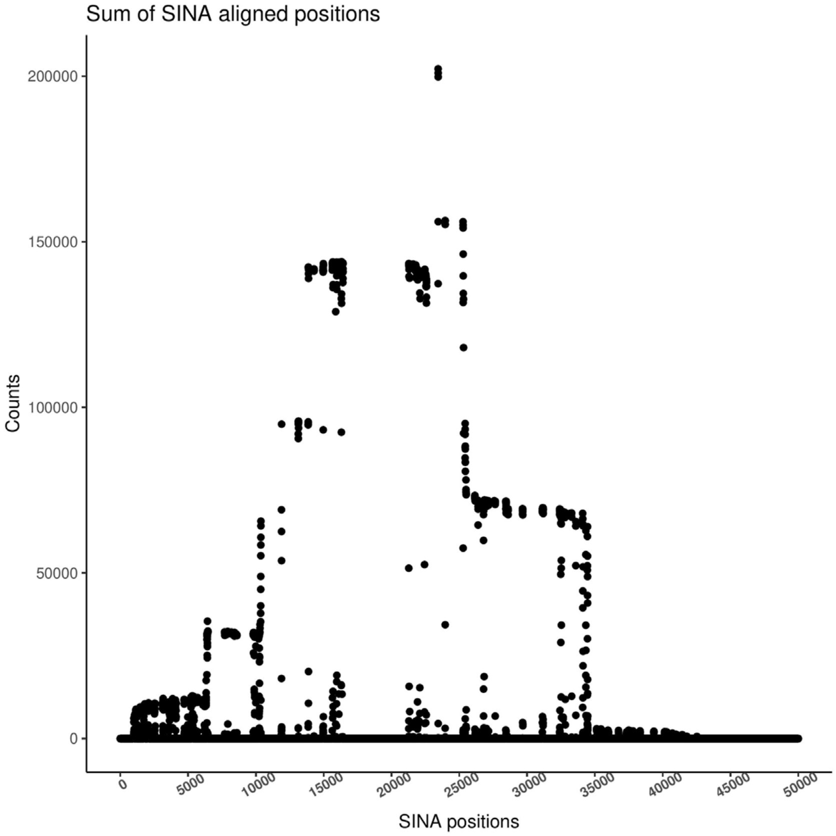


FIGURE 3 | Sum of bases on each SINA alignment position. The height identifies the region with the most coverage in the coverage plot. Guided by this plot, users should select the target region for their analysis. All sequences will be trimmed around those positions, and only those containing a sufficient number of bases will be passed to the next step.

produces only novel fOTUs, gOTUs and sOTUs, while filling the other missing taxonomic ranks (i.e., phylum, class and order) with a placeholder, i.e., UNKPHYLUM, UNKCLASS, and UNKORDER, respectively. Since no universal cutoffs for 16S rRNA gene fragments (i.e., amplicons) exists for delineating species, genera, and families, we adopt the popular cutoffs normally used for the whole 16S rRNA molecule (97, 95, and 90%, respectively). However, we recommend that those cutoffs be tailored to each analysis to reflect the variance captured within the selected fragment (i.e., V-region used).

2.1.6 Results Reporting and Graphs

The produced zOTUs are outputted in FASTA format with the full taxonomic path up to sOTU level, incorporating any novel families and genera in the header of each zOTU. Furthermore, the taxonomic tree (**Figure 5A**) indicates novelty (i.e., unknown bacteria and archaea) within the given study by color-coding each branch (Asnicar et al., 2015). Towards this end, the microbial novelty and diversity in the examined samples are displayed by uniting the taxonomic tree and the quantification information contained within the Krona plot in a combined figure (**Figure 5B**). Finally, the zOTU table produced shows how many reads in each sample constitute the respective sOTU together with the sOTUs' taxonomy. Additional mapping files produced as output reflect the relations between sOTUs to gOTUs and gOTUs to fOTUs.

2.2 Benchmarking

2.2.1 Naive Classifiers *vs* TIC

Comparisons between naive classifiers (USEARCH and VSEARCH) and TIC require a dataset for which the complete taxonomic information is available. We created a dataset fulfilling this requirement by using the sequences from the Living Tree Project (Yarza et al., 2008) and their corresponding similarity matches at a threshold of 98% of the non-redundant SILVA v128 database. This dataset was designated LTP. All sequences included were classified with SINA and, in order to simulate real-life scenarios, we pruned the produced taxonomies with two strategies, designated "hard" and "soft." Hard pruning corresponds to the removal of whole clades from the taxonomic tree at random levels. We removed about 10, 5, 2.5 and 1% from the tree at the level of genera, family, order, and class, respectively. This hard pruned dataset was used to test the performance for cases where completely unknown taxonomic groups are present within the actual data. The chosen percentages were based on empirical observations on missing taxonomic

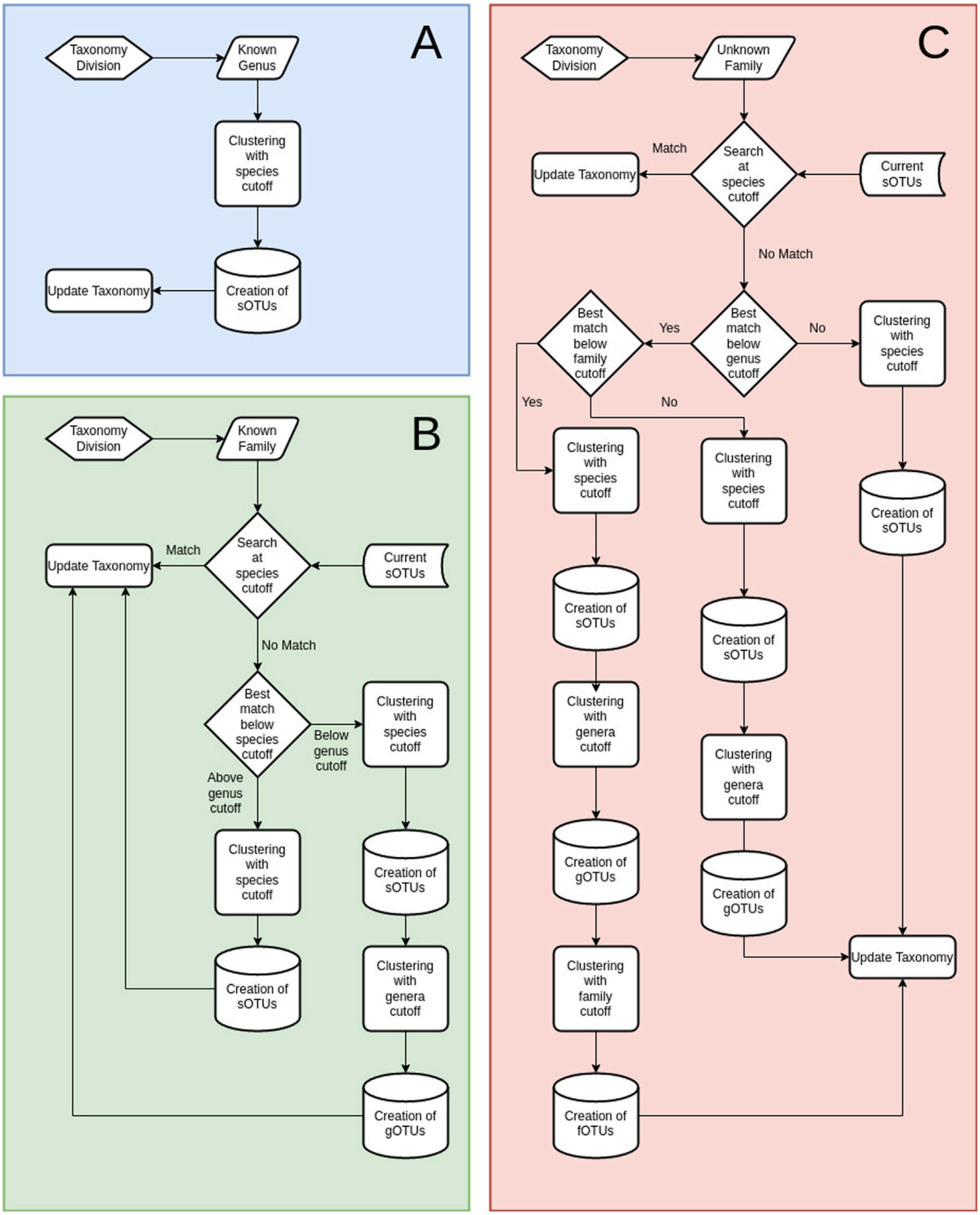


FIGURE 4 | Overview of the TIC processes. **(A)** Diagram of the TIC process for sequences with identified genus-level taxonomy. All sequences within each genus are used to create sOTUs. **(B)** Diagram of the TIC process for sequences with identified taxonomy up to the family. First step is searching for matches among those sequences and sOTUs contained within genera in the current family. Not matched sequences create novel sOTUs, which are searched for matches at the genera cutoff level (default 95%), as specified at the configuration file; if not matched again, they produce novel gOTUs. Any matched sequence gets the taxonomy of its match.
(C) Diagram of the TIC process for sequences without family classification. Searching for matches among existing sOTUs at the order level. Not matched sequences create novel sOTUs, which are searched for matches at the genera cutoff level; if not matched again, they produce novel gOTUs. Another search is conducted afterwards at the user-specified family similarity percentage (default 90%), afterward, and if not matched again, novel fOTUs are created. Any matched sequence gets the taxonomy of its match.

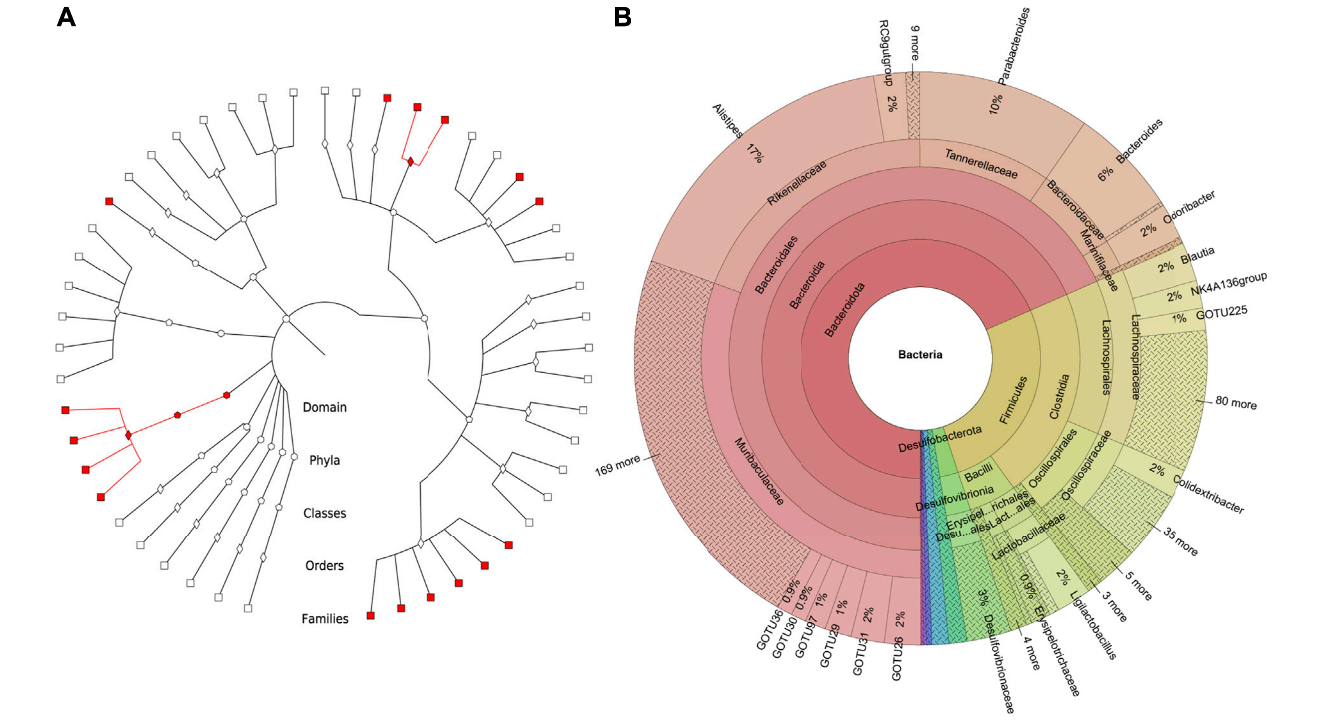


FIGURE 5 | Plots produced from the TIC-Pipeline from the mouse dataset of Muller et al. **(A)** Graphlan plot depicting the taxonomic tree of the denoised sequences after TIC incorporated both novel (red) and known (white) clades up to the family level. **(A)** Krona plot quantifies the size of each taxonomy in the merged study samples. Contains novel and known taxonomies as produced by the SINA classifier and TIC.

classification at each level when using SILVA on real data. The second pruning strategy "soft" is the stochastic removal of taxonomic information, simulating shortcomings of the classification process in assigning taxonomies to every leaf of each clade correctly, also following the above percentages. Those strategies are needed because LTP consists only of taxonomically known sequences on which classifiers have an advantage. Our pruning strategies allow a fair comparison between the taxonomy-aware TIC and the naive clustering algorithms. For testing, the clustering was performed 100 times for each strategy and tool.

2.2.2 Clustering Metrics

The following metrics were calculated for every trial: cluster purity, Adjusted Rand Index (ARI), and Normalized Mutual Information (NMI). Concerning cluster purity, this value ranges from 0 to 1. It shows the mean fraction of sequences across all clusters that are correctly pooled together according to the genus taxonomic information included in LTP. Next, the ARI gives a value about how often a randomly chosen sequence from the dataset was found in the same cluster as in the original LTP data set, when producing the same clustering (Steinley, 2004). Finally, the Normalized Mutual Information (NMI) quantifies the amount of information we obtain from clustering A by observing the clustering B; thus, it is a measure on how similar two different clustering runs (i.e., A and B) are (Vinh et al., 2010). A higher NMI score indicates that the information we got by clustering reflects the original taxonomic assignments closer. In turn, this allows us to approximate the entropy of the produced clustering.

We compared execution time for TIC with USEARCH and VSEARCH, allocating eight threads on the same machine and with Debian Linux as the host operating system. Each tool was evaluated further based on the number of produced families, genera, and species. This evaluation allowed us to determine the inflation for each type of clustering in each's diversity measures.

2.2.3 Template Data

Bifidobacteriaceae are a group of bacteria, which are responsible for oligosaccharide metabolism in mammals. They are one of the dominant families present in the human gastrointestinal tract during infancy (Pham et al., 2016). There is a growing interest in their role as probiotics. Therefore, illuminating the microbial diversity within this family will help us evaluate the range within which we operate and potential sources of hidden diversity. The template data include 227,418 *Bifidobacteriaceae* sequences, which were classified as such by RDP classifier. These sequences have been originally detected across 11,074 samples of diverse environmental origin within IMNGS. IMNGS is a database containing currently more than 500 k samples analyzed by 16S rRNA gene amplicon sequencing. All data are preprocessed and IMNGS offers, next to other means, automated export of all sequences belonging to a selected taxonomy at once (Lagkouvardos et al., 2016). In addition to

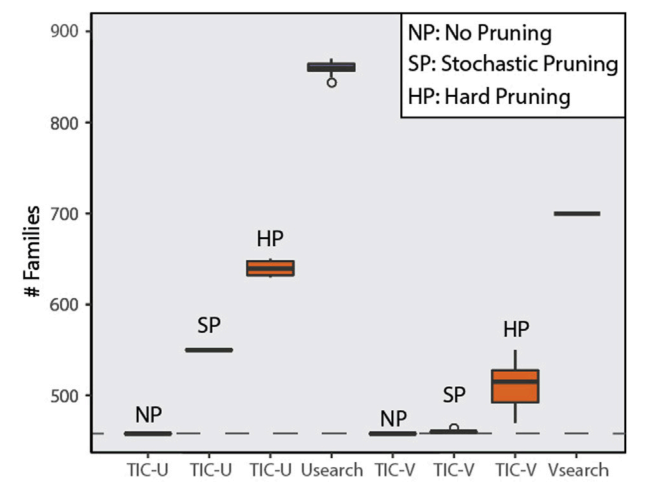


FIGURE 6 | Comparison of the three tools in regards to predicted family number on the LTP dataset under different configurations. TIC was executed containing USEARCH (TIC-U) and VSEARCH (TIC-V) as the integrated clustering tool, with different modes of taxonomic pruning of the input sequences. These tools were also executed as standalone. The dashed line represents the actual number of families in the dataset. USEARCH performs worse in terms of inflation of predicted family level clusters, with VSEARCH resulting in only moderate inflation. TIC reflects this trend in its operation depending on the tool utilized, especially with hard pruning (compete for removal of assignments for whole taxonomic branches). For soft (stochastic) pruning (only removing taxonomic information for random sequences), the TIC performs significantly better than the naïve usage of the corresponding clustering tool. In cases where the classifier can successfully assign family level taxonomy to all sequences, as for the sequences in the LTP dataset, the TIC mirrors this information resulting in a perfect grouping of the sequences as expected.

the above, to illustrate the use of TIC in microbial profiling studies based on amplicon data, we processed the dataset from Müller et al. (2016). In this study, the role of nutrition and hygiene concerning mice's gut microbiomes was investigated. The original results demonstrated that diet and the hygiene level of the mouse facility affect the mice's gut microbial profiles. The raw sequencing data of the study are available in ENA under accession PRJEB13041.

3 RESULTS

3.1 Benchmarking Results

3.1.1 Number of Created Taxa

The LTP dataset used contains sequences with known taxonomic assignment up to the species level, with 458 families, 1,590 genera, and 13,903 species. The TIC pipeline identified 508, 460, and 458 clusters at family level when using hard- and soft-pruned, and the complete taxonomy, respectively. In contrast, both USEARCH and VSEARCH resulted in estimations almost twice the size of the actual family numbers (**Figure 6**). Of note, when the complete SINA classification is available (no pruning), TIC, as expected, successfully mirrors the underlying family structure. Thus, overall, values produced by TIC are the closest reflection of the ground truth we could get.

The results for the genus level reconstruction showed that, when taxonomic information is missing (e.g., due to novel taxa or incomplete classifications), TIC and VSEARCH perform similarly. In contrast, USEARCH inflates genus numbers (**Supplementary Figure S1**). Since TIC is the only tool utilizing taxonomy knowledge, the results match the initial genus composition as expected in the no pruning scenario. All three tools fail to recapture the species diversity contained within the dataset (**Supplementary Figure S2**). We suggest that this is because taxonomic species definitions are not solely based on 16S rRNA gene sequence similarity, and none of the tools can account for this external information. However, TIC calling USEARCH produces species cluster closer to the ground truth regardless of the pruning scenario, while the TIC with VSEARCH improves its performance significantly when compared with default running of VSEARCH.

3.1.2 Quality of Created Taxa

Clusters produced by TIC are purer than those produced by USEARCH and VSEARCH (**Tables 1**, **2**). Since TIC uses taxonomic information, unwanted merging of sequences originating from distant taxonomies is less likely, while other tools are blind to taxonomy and, thus, combine unrelated sequences solely based on similarity thresholds. Although VSEARCH produces a higher ARI score, this stems from the inflation of the number of produced species, genera, and families in combination with the rigorous approach taken when calculating pairwise similarity scores, resulting in many one-member clusters that should have been merged otherwise. The NMI score calculated for all tools is almost identical. Therefore,

TABLE 1 | Clustering quality comparison among tools.

Level	Scenario	Purity	ARI	NMI
Species	TIC_Stohastic_VSEARCH	**0.99**	0.93	0.97
	TIC_Stohastic_USEARCH	**0.99**	0.93	0.97
	TIC_Hard_VSEARCH	**0.99**	0.93	**0.98**
	TIC_Hard_USEARCH	**0.99**	0.93	0.97
	USEARCH	0.98	0.88	0.97
	VSEARCH	0.97	**0.97**	0.97
Genera	TIC_Stohastic_VSEARCH	**1**	0.93	0.97
	TIC_Stohastic_USEARCH	**1**	0.93	**0.98**
	TIC_Hard_VSEARCH	**1**	0.93	**0.98**
	TIC_Hard_USEARCH	**1**	0.93	0.97
	USEARCH	0.87	0.88	0.97
	VSEARCH	0.93	**0.97**	0.97
Families	TIC_Stohastic_VSEARCH	**1**	0.93	0.97
	TIC_Stohastic_USEARCH	**1**	0.93	0.97
	TIC_Hard_VSEARCH	**1**	0.93	**0.98**
	TIC_Hard_USEARCH	**1**	0.93	0.97
	USEARCH	0.97	0.88	0.97
	VSEARCH	0.88	**0.97**	0.97

Regardless of the pruning method, taxonomic level, and the underlying tool, the TIC creates better clusters in terms of purity and the NMI statistic. VSEARCH inflates the number of clusters and, in conjunction with its no-heuristic approach when calculating the sequence pairwise identity score, results in higher ARI scores. Although USEARCH uses heuristics for this calculation, the TIC restrains it, thus keeping the ARI score high. Maximum values are highlighted (bold) for each column for each level.

TABLE 2 | Level of impurity for genus and family level clusters created by USEARCH and VSEARCH compared with the TIC approach for the LTP dataset.

Tool	Species	Mixed genera (percentage)	Mixed families (percentage)
USEARCH	6,668	299 (08.20)	115 (11.50)
VSEARCH	5,817	179 (05.84)	83 (08.80)
TIC_Soft_VSEARCH	5,824	0	0
TIC_Soft_USEARCH	6,315	0	0
TIC_Hard_VSEARCH	5,839	0	0
TIC_Hard_USEARCH	6,371	0	0

Impurity was calculated as the number of genera/families containing LTP sequences with conflicting taxonomic backgrounds. Both naive clustering tools result in more than 5% of genera and 8% of families having impure composition.

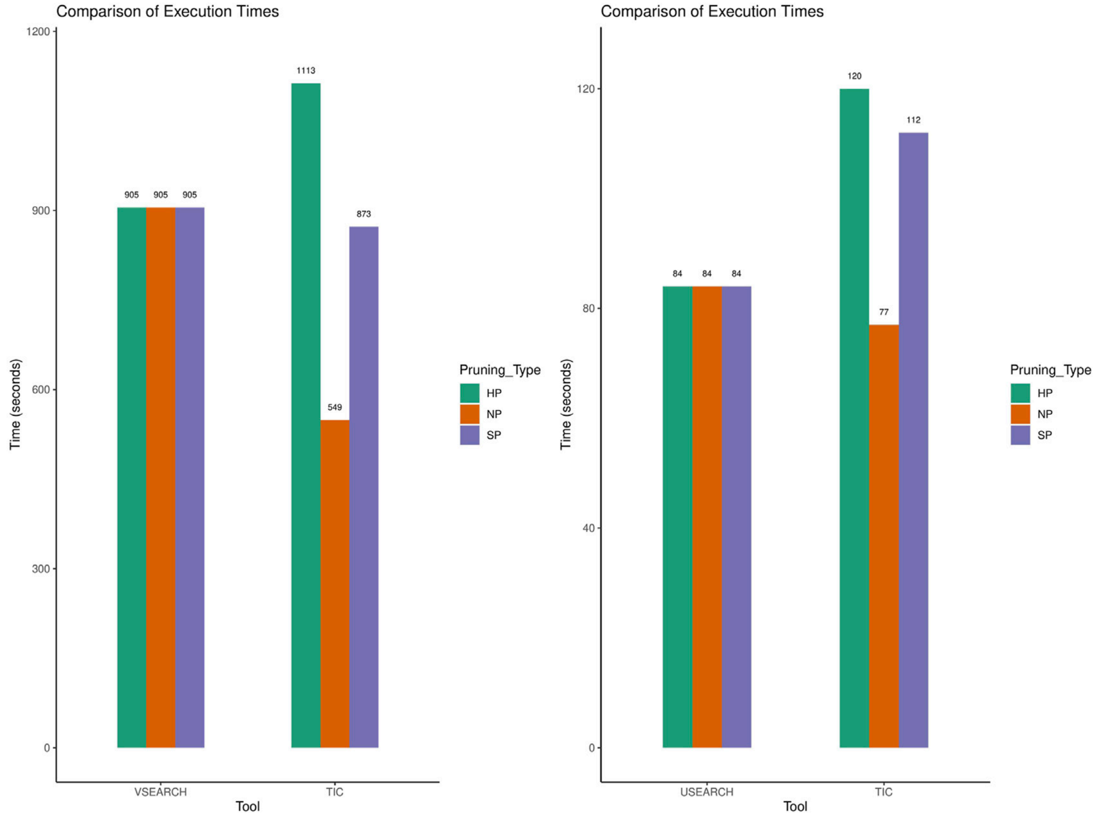


FIGURE 7 | Comparison of execution times for VSEARCH, USEARCH, and TIC running with each as an underlying tool respectively. Although slower, TIC is comparable with either tool regardless of the pruning method.

this value should not be viewed in isolation. Taken together, across all metrics tested here, TIC is the better choice.

Performance

The computational speed for TIC is primarily dependent on the underlying tool. TIC manages to offset the required time to handle taxonomic information by clustering smaller subsets of data created from the taxonomy classification (**Figure 7**). Performance is further affected by the rate of available taxonomic information and no-pruning run times are always shorter than those from simulations, including partial classifications.

3.2 Template Results

3.2.1 Amplicon Showcase

The Müller dataset (Müller et al., 2016) contains 238,936 raw sequences produced from 24 samples. This dataset contains 6,580 unique sequences after extraction of the representative region (i.e., SINA alignment positions: 6,500–22,500, number of bases: 384), trimming around it, and denoising to zOTUs. Taxonomic classification using the integrated SINA classifier with SILVA as the reference database resulted in 319 and 2,412 unclassified zOTUs for family and genus level, respectively. Clustering those sequences to form molecular species (sOTUs) using TIC or the two naive clustering tools (i.e., USEARCH, VSEARCH) resulted

TABLE 3 | Comparison of TIC with naïve clustering approaches on microbial amplicon data from mice.

Tool	Predicted SOTUs	No genus assigned	No family assigned
USEARCH	1,378	656	153
VSEARCH	1,380	694	150
TIC-Pipeline	1,279	—	—

Nearly 700 sOTUs produced by naïve de novo *clustering have the missing genus-level classification, and around 150 of those could not be assigned to a known family. The TIC organizes those sequences to 356 novel gOTUs and introduces 16 novel fOTUs.*

in similar sOTU numbers (≈ 1380; **Table 3**). However, 78 and 83 out of the predicted sOTUs created by USEARCH and VSEARCH, respectively, contain zOTUs with non-matching taxonomic assignment, strongly suggesting impure clusters. Moreover, 656 and 694 sOTUs created by USEARCH and VSEARCH respectively, have incomplete taxonomic assignments when we follow the old paradigm of clustering first and assign taxonomy later. For instance, 153 and 150 sOTUs produced by USEARCH and VSEARCH, respectively, have not been assigned to any family, while 503 and 544 sOTUs, respectively, have family classification only, but were not assigned to a genus (**Table 3**).

In contrast, TIC organized unclassified sOTUs in many cases within distinct gOTUs of a given family. Such unclassified sOTUs would otherwise be collectively treated as unknowns or even discarded. Similarly, for four taxonomic orders containing sOTUs with unknown family assignments, TIC stratified the sequences in appropriate fOTUs, further enhancing the insights into the community's structure of this dataset.

3.2.2 Diversity Showcase

For testing about diversity outcomes when applying TIC, the used dataset contains only sequences within the *Bifidobacteriaceae* family as identified by the RDP classifier (v.2.11 with training dataset 16) included in the IMNGS database. We re-classified the retrieved sequences using SINA and the latest online RDP classifiers (training dataset 18), removing all sequences not classified as *Bifidobacteriaceae*. After identifying the most representative region in this dataset (i.e., SINA alignment positions: 12,000–25,300, number of bases: 288), trimming and dereplication, almost 75,000 unique denoised sequences remained. The produced dataset was processed with TIC, resulting in about 72,000 molecular species organized in about 1,100 gOTUs. The known genus *Bifidobacterium* has about 69,000 sOTUs, reflecting the total molecular species diversity. The rest of the nine described genera from the *Bifidobacteriaceae* have 2,876 sOTUs, with an average of 320 sOTUs per genus (**Figure 8A**). The 1,134 remaining novel gOTUs contain only a single sOTU (**Figure 8C**). Comparing TIC to the other naive clustering algorithms shows again an inflation of the numbers of species and genera cluster formed (**Table 4**). Furthermore, both similarity-based tools separated the *Bifidobacteriaceae* sequences into 1000s of new families, while TIC kept them as one family.

About half of the discovered gOTUS (1.1 k) incorporate sequences originating solely from bovine samples, with only 13 gOTUs (which include most of the already described genera) containing sequences from diverse origins. The other half of the gOTUs consist exclusively of sequences of non-bovine origin (**Figure 8B**), including the genera *Bombiscardovia*, *Scardovia*, and *Gardnerella* that were not found in any of the bovine samples used in our analysis.

4 DISCUSSION

4.1 Amplicon Studies Integration Is Problematic due to Partially Overlapping Targeted Regions

Selection of different hypervariable regions for each amplicon-based experiment inevitably results in different primer sets used in different studies (Schloss et al., 2011), (Liu et al., 2008). The absence of a consensus (Abellan-Schneyder et al., 2021) of the scientific community on which region should be targeted for a given purpose further complicates this issue (Li et al., 2014), (Dassi et al., 2014). Such diverse primer designs prohibit the effortless integration of amplicon studies even in the absence of other experimental differences. In such cases, the suggested procedure is to identify a common region across studies, when such a region exists, and trim all sequences accordingly (Lagkouvardos et al., 2016). The proposed TIC pipeline follows this idea by using the SINA aligner. Extracting the region of overlap for different studies and collapsing gaps (which are inserted otherwise for better alignments) makes the sequences compatible and allows us to analyze samples processed with different, but overlapping V-regions together. Currently, the selection of the common region is performed by manual inspection, but an automated procedure is in development.

4.2 Naive Classification Tends to Produce Impure Clusters

Naive clustering tools are based solely on sequence similarity in creating groups. In contrast, TIC enhances the clustering process by utilizing the taxonomic information of each sequence acquired beforehand. The metrics tested here, clustering purity, ARI, and NMI, show that TIC outperforms both USEARCH and VSEACH (**Table 4**).

Fixed similarity levels cutoffs used for clustering will not always produce clusters that correspond to valid taxonomic paths (Edgar, 2018), (Schloss and Westcott, 2011), (White et al., 2010), (Huse et al., 2010). New approaches to clustering have been proposed, based on machine learning and other methods, but they have not yet seen widespread adaptations

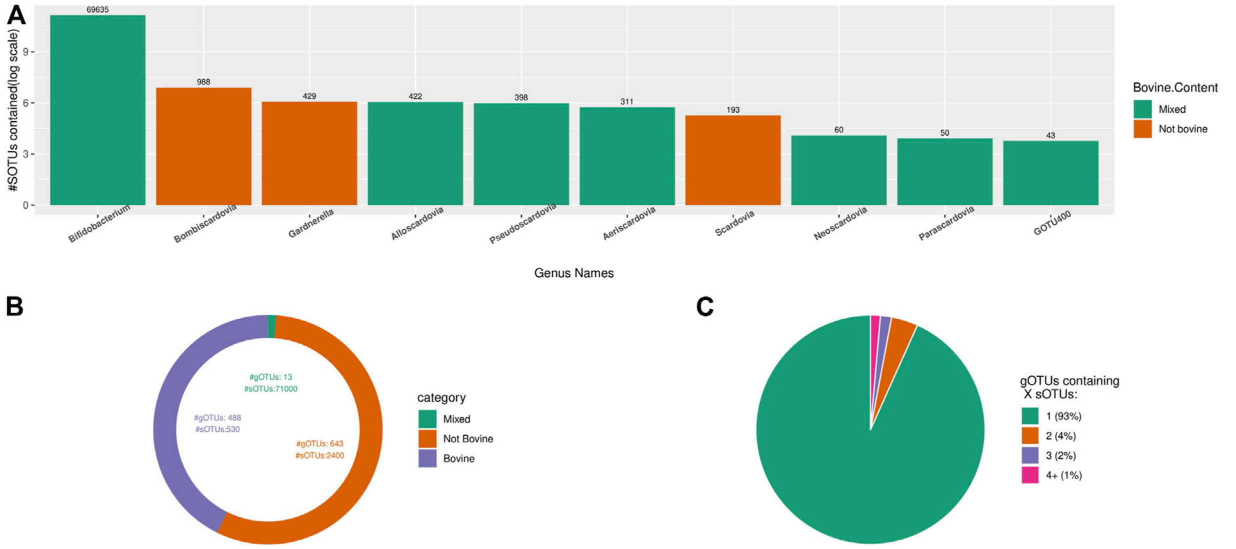


FIGURE 8 | Overview of the environmental origins of the *Bifidobacteriaceae* sequences grouped in gOTUs. **(A)** Rank order of 10 most diverse gOTUs, differentiated by the origins of their constituent sOTUs. *Bifidobacterium* is by far the most diverse genus of this family. **(B)** High niche specificity of *Bifidobacteriaceae* gOTUs contained within the bovine samples. **(C)** Pie chart indicating the size of gOTUs created by TIC from all available sequences classified as belonging to the *Bifidobacteriaceae* family extracted from IMNGS.

TABLE 4 | Diversity estimations among the three tools for the *Bifidobacteriaceae* sequences extracted from IMNGS.

Tool	Species number (k)	Genera number (k)	Families number
USEARCH	62	35	2.8 k
VSEARCH	52	28	3.5 k
TIC-Pipeline	52	1.1	1

Denoised sequences were clustered with the three tools. Using VSEARCH for within branch clustering, The TIC produces the most conservative results and should be used as a baseline.

(James et al., 2018) (Eren et al., 2015), (Navlakha et al., 2010), (Preheim et al., 2013), (Mahé et al., 2014). The accuracy of similarity-based tools can be improved by introducing clade specific similarity cutoffs. For such an approach, phylogenetic distances of all described taxa could be used to generate clade-specific similarity limits reflecting the average distance of taxonomic units (e.g., average distances among sequences from all genera within a family to set the genus similarity cutoff for that family). These limits should be further refined based on the selected region of the 16S rRNA gene used in each study.

In any case, all tools tested struggle mapping sequences to the underlining taxonomic delineation for species-level clusters. The main reason is that taxonomic nomenclature, especially at the species level, is not necessarily reflected in adequate differences in the 16S rRNA gene. Instead, functional characteristics, phenotypes, or pathogenicity differences of the bacteria are used to designate species. A well-known example is the *Escherichia-Shigella* clade, with otherwise almost identical 16S rRNA genes, but even different genus names. Other such examples exist. Thus, since classifiers based on 16S rRNA cannot (yet) assign taxonomy up to the species level, TIC cannot overcome the absence of this information in the molecular species-level prediction. In any case, all classification tools finally rely on reference databases that affect their performance. That is why the usage of the latest and most comprehensive iteration is the recommended practice.

Naive similarity-based clustering tools' results are affected only by their underlying algorithm regardless of other available information. In contrast, TIC's performance is bound to the completeness of the classifier-provided taxonomic information. Already with the current level of knowledge extractable from commonly used classifiers, TIC outperforms naive clustering tools despite some novel sequences existing in most studies. Furthermore, as the classifiers improve in their capacity to translate sequence signatures to finer taxonomic classifications, TIC-produced clusters will also be affected and improved in terms of purity and quality.

4.3 Evaluation of Taxonomy Informed Clustering in Single Amplicon Studies

There has been a growing tendency to abandon "traditional" OTUs based pipelines due to their problems in clusters purity, reproducibility, and interoperability in favor of denoised sequences. Denoised sequences are called with different names depending on the tool used (e.g., ASVs, zOTUs). Although there are clear benefits in such pipelines using denoised sequences, limiting processing to the molecular strain level also is often problematic. To the extreme, a strain can be any single bacterium differing by a single mutation across its genome, effectively accounting for nearly as many strains as individual bacteria in a sample. However, commonly "strains" are viewed as relatives belonging to a given species and differing in few to several phenotypic characteristics. In any case, strains are not well defined, especially when derived by molecular sequences. Sequence fragments of the 16S rRNA gene of, e.g., about 300 bases may be identical and, therefore, different strains are assigned to one amplicon variant, although originating from several. Increasing the length of the fragment, e.g., by different sequencing technologies, may reveal an increased number of strains/variants for the same sample. Therefore, alpha-diversity measures based on denoised sequences of different lengths offer a non-comparable sample diversity measure only. Common OTU clustering of 16S rRNA genes to a fixed similarity cutoff for accommodating molecular species is more defined, stable across sequencing lengths and technologies and, reflects a more meaningful ecological entity. However, other problems with OTUs, as mentioned above, exist. Concerning, beta-diversity measures similar problems arise. For instance, methods like Jaccard and Bray-Curtis do not consider the similarity (i.e., taxonomy) among the different strains in a sample and, therefore, tend to inflate the distances across microbial profiles between samples. Finally, it defeats its purpose when studies perform strain-level processing at first, but use the binned family abundances or even higher taxonomies for their comparisons. In contrast, TIC offers an incremental, structured dissection of the sequencing outputs from zOTUs to sOTUs, and then proceeds to gOTUs and fOTUs. Since the taxonomic placement of the sequence is clear, exploring the different hierarchical levels is easy depending on the question. For instance, the test run using real amplicon data showed that multiple well differentiated fOTUs and gOTUs were revealed. These would otherwise be collectively treated as unknowns and not contribute to understanding a sample's or experiment's ecology. Clearly, the refined taxonomic classification of every sequence assists downstream comparisons among higher taxonomic levels and reveal differential patterns across yet undescribed groups. Since taxonomy-informed clustering always results in purer clusters and more informative outcomes, we strongly recommend integrating tools like TIC in future amplicon analysis pipelines.

4.4 Diversity Analysis of *Bifidobacteriaceae*

The *Bifidobacteriaceae* family has attracted much interest due to its mostly positive effects on humans and other mammals. Microbes of this family colonize the infant gut, aiding in nutrient absorption (Turroni et al., 2011) and can act as probiotics with beneficial effects in patients with irritable bowel syndrome (Yuan et al., 2017) and other intestinal diseases (Matsuoka et al., 2018). This family is currently composed of 10 genera containing 124 valid species in taxonomic nomenclature. Specifically, the genus *Bifidobacterium* covers most of the family's diversity with 105 species, representing the most diverse genus of the *Bifidobacteriaceae*. This genus is most frequently associated with the gastrointestinal tract of humans (Scardovi and Trovatelli, 1974). However, molecular evidence has shown the presence of *Bifidobacterium* in other niches beyond the mammalian gut (Watanabe et al., 2009), (Dong et al., 2000). Species within the *Bifidobacteriaceae* show varying degrees of ecological adaptation with few cosmopolitan taxa within an otherwise specialized majority. This is due to the intense selective pressure for acquiring and retaining genes responsible for utilizing various carbohydrates to compete in their respective ecological niches (Milani et al., 2014), (Milani et al., 2015).

Our findings indicate an even larger *Bifidobacterium* genus, followed by also prolific, but less known genera and numerous novel candidate genera. Interestingly, the distribution of the novel genera and species detected here by molecular data seems to follow the distribution of currently known and described species within the recognized genera (normalized chi-square *p*-value: 0.12). It is safe to assume that part of this discrepancy in species numbers (i.e., known vs unknown) is attributed to uneven sampling and isolation efforts devoted to human and mammalian gut environments in general, which the genus *Bifidobacterium* seems to dominate. Nevertheless, the observed pattern is so pronounced that it calls for further research to unravel the ecological constraints that dictate this massive differentiation of *Bifidobacterium* and the modes of persistence and dispersal of this vital family of bacteria in contrast to the other genera in this family.

5 CONCLUSION AND FUTURE WORK

The TIC pipeline is a modular set of tools that facilitate fast and easy analysis of microbial data to produce the data files most commonly used in microbial ecology. In the present manuscript, we demonstrate the advantages of reversing the current practice of *de novo* sequence clustering followed by taxonomic classification. In contrast, taxonomically placed sequences allow utilizing the classifier's information in guided clustering and this approach results in higher cluster quality and purity, and allows proper placing of yet unassigned sequences in the taxonomy.

Currently, the TIC pipeline will soon be integrated in online analytical services while further simplifying the technical requirements for users. New features and outputs, such as making the TIC pipeline available to distributed systems, enhanced graphical representations, and other features, which can be requested by the community, will be added.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/ **Supplementary Material**.

AUTHOR CONTRIBUTIONS

AK and IL conceived and designed the experiments, performed the experiments, analyzed the data, contributed materials, analysis tools, prepared figures and tables, and wrote the draft. MP and KN contributed in discussions, tool testing and helped in writing. The final manuscript was read and approved by all authors.

FUNDING

This project has received funding from the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT) under grant agreement No 710. IL also received funding by German Research Foundation (SFB 1371, Project No. 395357507). The Technical University of Munich supported this publication within the Open Access Publishing Program.

ACKNOWLEDGMENTS

We would like to thank Charlie Pauvert for reviewing and providing constructive feedback on the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.864597/full#supplementary-material

REFERENCES

Abellan-Schneyder, I., Matchado, M. S., Reitmeier, S., Sommer, A., Sewald, Z., Baumbach, J., et al. (2021). Primer, Pipelines, Parameters: Issues in 16s Rrna Gene Sequencing. *Msphere* 6, e01202–20. doi:10.1128/mSphere.01202-20

Asnicar, F., Weingart, G., Tickle, T. L., Huttenhower, C., and Segata, N. (2015). Compact Graphical Representation of Phylogenetic Data and Metadata with Graphlan. *PeerJ* 3, e1029. doi:10.7717/peerj.1029

Blaxter, M., Mann, J., Chapman, T., Thomas, F., Whitton, C., Floyd, R., et al. (2005). Defining Operational Taxonomic Units Using Dna Barcode Data. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 360, 1935–1943. doi:10.1098/rstb.2005.1725

Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J., and Holmes, S. P. (2016). Dada2: High-Resolution Sample Inference from Illumina Amplicon Data. *Nat. Methods* 13, 581–583. doi:10.1038/nmeth.3869

Dassi, E., Ballarini, A., Covello, G., Quattrone, A., Quattrone, A., Jousson, O., et al. (2014). Enhanced Microbial Diversity in the Saliva Microbiome Induced by Short-Term Probiotic Intake Revealed by 16s Rrna Sequencing on the Iontorrent Pgm Platform. *J. Biotechnol.* 190, 30–39. doi:10.1016/j.jbiotec.2014.03.024

DeSantis, T. Z., Hugenholtz, P., Larsen, N., Rojas, M., Brodie, E. L., Keller, K., et al. (2006). Greengenes, a Chimera-Checked 16s Rrna Gene Database and Workbench Compatible with Arb. *Appl. Environ. Microbiol.* 72, 5069–5072. doi:10.1128/AEM.03006-05

Dong, X., Xin, Y., Jian, W., Liu, X., and Ling, D. (2000). Bifidobacterium Thermacidophilum Sp. nov., Isolated from an Anaerobic Digester. *Int. J. Syst. Evol. Microbiol.* 50 Pt 1, 119–125. doi:10.1099/00207713-50-1-119

Edgar, R. C. (2010). Search and Clustering Orders of Magnitude Faster Than Blast. *Bioinformatics* 26, 2460–2461. doi:10.1093/bioinformatics/btq461

Edgar, R. C. (2013). Uparse: Highly Accurate Otu Sequences from Microbial Amplicon Reads. *Nat. Methods* 10, 996–998. doi:10.1038/nmeth.2604

Edgar, R. C. (2018). Updating the 97% Identity Threshold for 16s Ribosomal Rna otus. *Bioinformatics* 34, 2371–2375. doi:10.1093/bioinformatics/bty113

Edgar, R. C. (2016). Unoise2: Improved Error-Correction for Illumina 16s and its Amplicon Sequencing. *BioRxiv*, 081257.

Eren, A. M., Morrison, H. G., Lescault, P. J., Reveillaud, J., Vineis, J. H., and Sogin, M. L. (2015). Minimum Entropy Decomposition: Unsupervised Oligotyping for Sensitive Partitioning of High-Throughput Marker Gene Sequences. *ISME J.* 9, 968–979. doi:10.1038/ismej.2014.195

Huse, S. M., Welch, D. M., Morrison, H. G., and Sogin, M. L. (2010). Ironing Out the Wrinkles in the Rare Biosphere through Improved Otu Clustering. *Environ. Microbiol.* 12, 1889–1898. doi:10.1111/j.1462-2920.2010.02193.x

James, B. T., Luczak, B. B., and Girgis, H. Z. (2018). Meshclust: an Intelligent Tool for Clustering Dna Sequences. *Nucleic Acids Res.* 46, e83. doi:10.1093/nar/gky315

Johnson, J. S., Spakowicz, D. J., Hong, B. Y., Petersen, L. M., Demkowicz, P., Chen, L., et al. (2019). Evaluation of 16s Rrna Gene Sequencing for Species and Strain-Level Microbiome Analysis. *Nat. Commun.* 10, 5029–5111. doi:10.1038/s41467-019-13036-1

Koeppel, A. F., and Wu, M. (2013). Surprisingly Extensive Mixed Phylogenetic and Ecological Signals Among Bacterial Operational Taxonomic Units. *Nucleic Acids Res.* 41, 5175–5188. doi:10.1093/nar/gkt241

Kopylova, E., Noé, L., and Touzet, H. (2012). Sortmerna: Fast and Accurate Filtering of Ribosomal Rnas in Metatranscriptomic Data. *Bioinformatics* 28, 3211–3217. doi:10.1093/bioinformatics/bts611

Lagkouvardos, I., Joseph, D., Kapfhammer, M., Giritli, S., Horn, M., Haller, D., et al. (2016). Imngs: a Comprehensive Open Resource of Processed 16s Rrna Microbial Profiles for Ecology and Diversity Studies. *Sci. Rep.* 6, 33721–33729. doi:10.1038/srep33721

Lagkouvardos, I., Overmann, J., and Clavel, T. (2017). Cultured Microbes Represent a Substantial Fraction of the Human and Mouse Gut Microbiota. *Gut microbes* 8, 493–503. doi:10.1080/19490976.2017.1320468

Lagkouvardos, I., Weinmaier, T., Lauro, F. M., Cavicchioli, R., Rattei, T., and Horn, M. (2014). Integrating Metagenomic and Amplicon Databases to Resolve the Phylogenetic and Ecological Diversity of the Chlamydiae. *ISME J.* 8, 115–125. doi:10.1038/ismej.2013.142

Lan, Y., Wang, Q., Cole, J. R., and Rosen, G. L. (2012). Using the Rdp Classifier to Predict Taxonomic novelty and Reduce the Search Space for Finding Novel Organisms. *PLoS one* 7, e32491. doi:10.1371/journal.pone.0032491

Li, J., Quinque, D., Horz, H. P., Li, M., Rzhetskaya, M., Raff, J. A., et al. (2014). Comparative Analysis of the Human Saliva Microbiome from Different Climate Zones: Alaska, germany, and Africa. *BMC Microbiol.* 14, 316–413. doi:10.1186/s12866-014-0316-1

Li, J. F. (2015). A Fast Neighbor Joining Method. *Genet. Mol. Res.* 14, 8733–8743. doi:10.4238/2015.July.31.22

Liu, Y., Schmidt, B., and Maskell, D. L. (2009). "Parallel Reconstruction of Neighbor-Joining Trees for Large Multiple Sequence Alignments Using Cuda," in 23rd IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2009, Rome, Italy, May 23–29, 2009, 1. doi:10.1109/ipdps.2009.5160923

Liu, Z., DeSantis, T. Z., Andersen, G. L., and Knight, R. (2008). Accurate Taxonomy Assignments from 16s rRNA Sequences Produced by Highly Parallel Pyrosequencers. *Nucleic Acids Res.* 36, e120. doi:10.1093/nar/gkn491

Mahé, F., Rognes, T., Quince, C., de Vargas, C., and Dunthorn, M. (2014). Swarm: Robust and Fast Clustering Method for Amplicon-Based Studies. *PeerJ* 2, e593. doi:10.7717/peerj.593

Matsuoka, K., Uemura, Y., Kanai, T., Kunisaki, R., Suzuki, Y., Yokoyama, K., et al. (2018). Efficacy of Bifidobacterium Breve Fermented Milk in Maintaining Remission of Ulcerative Colitis. *Dig. Dis. Sci.* 63, 1910–1919. doi:10.1007/s10620-018-4946-2

Milani, C., Lugli, G. A., Duranti, S., Turroni, F., Bottacini, F., Mangifesta, M., et al. (2014). Genomic Encyclopedia of Type Strains of the Genus Bifidobacterium. *Appl. Environ. Microbiol.* 80, 6290–6302. doi:10.1128/AEM.02308-14

Milani, C., Lugli, G. A., Duranti, S., Turroni, F., Mancabelli, L., Ferrario, C., et al. (2015). Bifidobacteria Exhibit Social Behavior through Carbohydrate Resource Sharing in the Gut. *Sci. Rep.* 5, 15782–15814. doi:10.1038/srep15782

Müller, V. M., Zietek, T., Rohm, F., Fiamoncini, J., Lagkouvardos, I., Haller, D., et al. (2016). Gut Barrier Impairment by High-Fat Diet in Mice Depends on Housing Conditions. *Mol. Nutr. Food Res.* 60, 897–908. doi:10.1002/mnfr.201500775

Navlakha, S., White, J., Nagarajan, N., Pop, M., and Kingsford, C. (2010). Finding Biologically Accurate Clusterings in Hierarchical Tree Decompositions Using the Variation of Information. *J. Comput. Biol.* 17, 503–516. doi:10.1089/cmb.2009.0173

Nocker, A., Burr, M., and Camper, A. K. (2007). Genotypic Microbial Community Profiling: a Critical Technical Review. *Microb. Ecol.* 54, 276–289. doi:10.1007/s00248-006-9199-5

Ondov, B. D., Bergman, N. H., and Phillippy, A. M. (2011). Interactive Metagenomic Visualization in a Web Browser. *BMC bioinformatics* 12, 385–410. doi:10.1186/1471-2105-12-385

Pham, V. T., Lacroix, C., Braegger, C. P., and Chassard, C. (2016). Early Colonization of Functional Groups of Microbes in the Infant Gut. *Environ. Microbiol.* 18, 2246–2258. doi:10.1111/1462-2920.13316

Porter, T. M., and Hajibabaei, M. (2018). Scaling up: A Guide to High-Throughput Genomic Approaches for Biodiversity Analysis. *Mol. Ecol.* 27, 313–338. doi:10.1111/mec.14478

Preheim, S. P., Perrotta, A. R., Martin-Platero, A. M., Gupta, A., and Alm, E. J. (2013). Distribution-based Clustering: Using Ecology to Refine the Operational Taxonomic Unit. *Appl. Environ. Microbiol.* 79, 6593–6603. doi:10.1128/AEM.00342-13

Pruesse, E., Peplies, J., and Glöckner, F. O. (2012). Sina: Accurate High-Throughput Multiple Sequence Alignment of Ribosomal Rna Genes. *Bioinformatics* 28, 1823–1829. doi:10.1093/bioinformatics/bts252

Quast, C., Pruesse, E., Yilmaz, P., Gerken, J., Schweer, T., Yarza, P., et al. (2012). The Silva Ribosomal Rna Gene Database Project: Improved Data Processing and Web-Based Tools. *Nucleic Acids Res.* 41, D590–D596. doi:10.1093/nar/gks1219

Rognes, T., Flouri, T., Nichols, B., Quince, C., and Mahé, F. (2016). Vsearch: a Versatile Open Source Tool for Metagenomics. *PeerJ* 4, e2584. doi:10.7717/peerj.2584

Scardovi, V., and Trovatelli, L. D. (1974). Bifidobacterium Animalis (Mitsuoka) Comb. Nov. And the "Minimum" and "Subtile" Groups of New Bifidobacteria Found in Sewage. *Int. J. Syst. Bacteriol.* 24, 21–28. doi:10.1099/00207713-24-1-21

Schloss, P. D., Gevers, D., and Westcott, S. L. (2011). Reducing the Effects of Pcr Amplification and Sequencing Artifacts on 16s Rrna-Based Studies. *PloS one* 6, e27310. doi:10.1371/journal.pone.0027310

Schloss, P. D., and Westcott, S. L. (2011). Assessing and Improving Methods Used in Operational Taxonomic Unit-Based Approaches for 16s Rrna Gene Sequence Analysis. *Appl. Environ. Microbiol.* 77, 3219–3226. doi:10.1128/AEM.02810-10

Sierra, M. A., Li, Q., Pushalkar, S., Paul, B., Sandoval, T. A., Kamer, A. R., et al. (2020). The Influences of Bioinformatics Tools and Reference Databases in Analyzing the Human Oral Microbial Community. *Genes (Basel)* 11, 878. doi:10.3390/genes11080878

Simonsen, M., Mailund, T., and Pedersen, C. N. (2010). "Inference of Large Phylogenies Using Neighbour-Joining," in International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2010, Valencia, Spain, January 22–23 , 2010 (Springer), 334

Steinley, D. (2004). Properties of the Hubert-Arabie Adjusted Rand index. *Psychol. Methods* 9, 386–396. doi:10.1037/1082-989X.9.3.386

Turroni, F., Van Sinderen, D., and Ventura, M. (2011). Genomics and Ecological Overview of the Genus Bifidobacterium. *Int. J. Food Microbiol.* 149, 37–44. doi:10.1016/j.ijfoodmicro.2010.12.010

Vinh, N. X., Epps, J., and Bailey, J. (2010). Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. *J. Machine Learn. Res.* 11, 2837.

Wang, Q., Garrity, G. M., Tiedje, J. M., and Cole, J. R. (2007). Naive Bayesian Classifier for Rapid Assignment of Rrna Sequences into the New Bacterial Taxonomy. *Appl. Environ. Microbiol.* 73, 5261–5267. doi:10.1128/AEM.00062-07

Watanabe, K., Makino, H., Sasamoto, M., Kudo, Y., Fujimoto, J., and Demberel, S. (2009). Bifidobacterium Mongoliense Sp. nov., from Airag, a Traditional Fermented Mare's Milk Product from Mongolia. *Int. J. Syst. Evol. Microbiol.* 59, 1535–1540. doi:10.1099/ijs.0.006247-0

White, J. R., Navlakha, S., Nagarajan, N., Ghodsi, M. R., Kingsford, C., and Pop, M. (2010). Alignment and Clustering of Phylogenetic Markers-Iimplications for Microbial Diversity Studies. *BMC bioinformatics* 11, 152–210. doi:10.1186/1471-2105-11-152

Woese, C. R., Magrum, L. J., Gupta, R., Siegel, R. B., Stahl, D. A., Kop, J., et al. (1980). Secondary Structure Model for Bacterial 16s Ribosomal Rna: Phylogenetic, Enzymatic and Chemical Evidence. *Nucleic Acids Res.* 8, 2275–2293. doi:10.1093/nar/8.10.2275

Yarza, P., Richter, M., Peplies, J., Euzeby, J., Amann, R., Schleifer, K. H., et al. (2008). The All-Species Living Tree Project: a 16s Rrna-Based Phylogenetic Tree of All Sequenced Type Strains. *Syst. Appl. Microbiol.* 31, 241–250. doi:10.1016/j.syapm.2008.07.001

Yuan, F., Ni, H., Asche, C. V., Kim, M., Walayat, S., and Ren, J. (2017). Efficacy of Bifidobacterium Infantis 35624 in Patients with Irritable Bowel Syndrome: a Meta-Analysis. *Curr. Med. Res. Opin.* 33, 1191–1197. doi:10.1080/03007995.2017.1292230

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Kioukis, Pourjam, Neuhaus and Lagkouvardos. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

TECHNOLOGY AND CODE
published: 10 May 2022
doi: 10.3389/fbinf.2022.846922

MIntO: A Modular and Scalable Pipeline For Microbiome Metagenomic and Metatranscriptomic Data Integration

*Carmen Saenz, Eleonora Nigro, Vithiagaran Gunalan and Manimozhiyan Arumugam**

Novo Nordisk Foundation Center for Basic Metabolic Research, Faculty of Health and Medical Sciences, University of Copenhagen, Copenhagen, Denmark

Omics technologies have revolutionized microbiome research allowing the characterization of complex microbial communities in different biomes without requiring their cultivation. As a consequence, there has been a great increase in the generation of omics data from metagenomes and metatranscriptomes. However, pre-processing and analysis of these data have been limited by the availability of computational resources, bioinformatics expertise and standardized computational workflows to obtain consistent results that are comparable across different studies. Here, we introduce MIntO (Microbiome Integrated meta-Omics), a highly versatile pipeline that integrates metagenomic and metatranscriptomic data in a scalable way. The distinctive feature of this pipeline is the computation of gene expression profile through integrating metagenomic and metatranscriptomic data taking into account the community turnover and gene expression variations to disentangle the mechanisms that shape the metatranscriptome across time and between conditions. The modular design of MIntO enables users to run the pipeline using three available modes based on the input data and the experimental design, including *de novo* assembly leading to metagenome-assembled genomes. The integrated pipeline will be relevant to provide unique biochemical insights into microbial ecology by linking functions to retrieved genomes and to examine gene expression variation. Functional characterization of community members will be crucial to increase our knowledge of the microbiome's contribution to human health and environment. MIntO v1.0.1 is available at https://github.com/arumugamlab/MIntO.

Keywords: omics integration, metagenomic, metatranscriptomic, pipeline, gene expression, community turnover, microbial ecology, microbiome

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Bin Hu,
Los Alamos National Laboratory (DOE), United States
Martin Hölzer,
Robert Koch Institute (RKI), Germany
Fabio Sanchez,
University of São Paulo, Brazil

****Correspondence:***
Manimozhiyan Arumugam
arumugam@sund.ku.dk

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *31 December 2021*
Accepted: *11 April 2022*
Published: *10 May 2022*

Citation:
Saenz C, Nigro E, Gunalan V and Arumugam M (2022) MIntO: A Modular and Scalable Pipeline For Microbiome Metagenomic and Metatranscriptomic Data Integration. Front. Bioinform. 2:846922. doi: 10.3389/fbinf.2022.846922

INTRODUCTION

The human microbiome is a complex congregation of microbes comprising trillions of microbial cells present in our bodies (Bashan et al., 2016). Microbe-microbe and microbe-host interactions confer a variety of physiological benefits to the hosts and impact their susceptibility to disease. For instance, the microbial niche can provide metabolic functions different from the host genome, most of which are encoded by genes that have not yet been discovered (Nicholson et al., 2012; Donia and Fischbach, 2015).

TABLE 1 | Features of pipelines that handle metagenomic and metatranscriptomic data in comparison to MIntO: Steps, capacities and approaches.

	FMAP Kim et al. (2016); Salazar et al. (2019)	**IMP** Narayanasamy et al. (2016)	**MOSCA** Sequeira et al. (2019)	**SqueezeMeta** Tamames and Puente-Sánchez, (2018)	**MUFFIN** Van Damme et al. (2021)	**MIntO (2021)**
data source	short reads	paired-end short reads	paired-end short reads	paired-end short reads	paired-end Illumina reads (short reads) and Nanopore-based reads (long reads)	paired-end Illumina reads (short reads) and Nanopore-based reads (long reads)
quality and read length control	only quality control	Yes	Yes	Yes	Yes	Yes
host genome removal	only human genome removal	Yes	No	No	No	Yes
rRNA removal	No	Yes	Yes	No	No	Yes
taxonomy assignment	No	Yes	Yes	Yes	Yes	Yes
de novo assembly/co-assembly	No	Yes	Yes	Yes	combining short and long reads	optionally, include long reads
binning	No	Yes	Yes	Yes	Yes	Yes
gene prediction	Yes	Yes	Yes	Yes	Yes	Yes
function annotation	Yes	Yes	Yes	Yes	Yes	Yes
alignment to reference database/genomes	alignment to reference database	Yes	No	No	No	Yes
alignment to retrieved MAGs	No	Yes	Yes	Yes	Yes	Yes
normalization	RPKM	RPKM	TMM, RLE	RPKM	TPM	TPM, Marker genes
visualization	Yes	Yes	Yes	No	Yes	Yes
local installation	Yes	Yes	Yes	Yes	Yes	Yes
gene expression computation	No	No	No	No	No	Yes
differential analysis/ Downstream analysis	differentially-abundant genes analysis	No	differential gene expression analysis	No	No	No
Software dependencies installed by the user before using the pipeline	Perl, R, Statistics::R, DIAMOND or USEARCH, Bio::DB::Taxonomy, XML::LibXML	Python3, pip, impy, Conda, Docker/ Singularity	MOSGUITO, and Conda, or Docker/ Singularity	Conda	Nextflow and Conda or Docker/Singularity	FetchMGs, Conda

Studying these microbial communities is a challenging task, which has recently been made easier by high-throughput sequencing approaches which generate omics data such as metagenomes and metatranscriptomes. These omics methods have revolutionized microbiome research by allowing the characterization of complex microbial communities in different biomes without requiring their cultivation. Metagenomic data enables the genomic and taxonomic characterization of microbial community composition and, depending on the sequencing strategy employed, can allow the recovery of Metagenome-Assembled Genomes (MAGs) (Almeida et al., 2019; Stewart et al., 2019; Saheb Kashaf et al., 2022). However, it can only unravel the functional potential in a sample (Quince et al., 2017). In contrast, metatranscriptomic data identifies the pool of genes that are transcribed under a specific condition, which gives a more accurate picture of the processes and molecular activity occurring in the microbial community (Satinsky et al., 2014; Salazar et al., 2019). Hence, by analyzing both metagenomes and metatranscriptomes, we can have deeper insights into the functional potential as well as the actual activity of microbial communities (Wang et al., 2020; Tláskal et al., 2021).

In recent years, the application of high-throughput sequencing approaches in microbiome research has greatly increased together with the generation of large amounts of data (Qin et al., 2010; Human Microbiome Project Consortium, 2012; Pasolli et al., 2019). As a consequence, the pre-processing and analysis of such data have been limited by the availability of computational resources and bioinformatics expertise. In addition, there is a lack of standardized protocols to handle and analyze multi-omics data sets in a more consistent manner, making the comparisons between different studies and findings more challenging. Standardizing the way omics data are handled ensures a degree of consistency of the results across different studies. Furthermore, making the workflows semi-automatic will allow the analysis of complex microbial communities by users with limited bioinformatic skills.

Standard metagenomic and metatranscriptomic approaches entail 1) read curation, 2) *de novo* assembly and/or co-assembly, 3) binning, 4) gene prediction, 5) annotation of predicted genes at taxonomic and functional level and 6) quantification of gene abundances and transcripts. However, most of the computational pipelines developed so far can only analyze metagenomic or metatranscriptomic data individually and only few, reported in

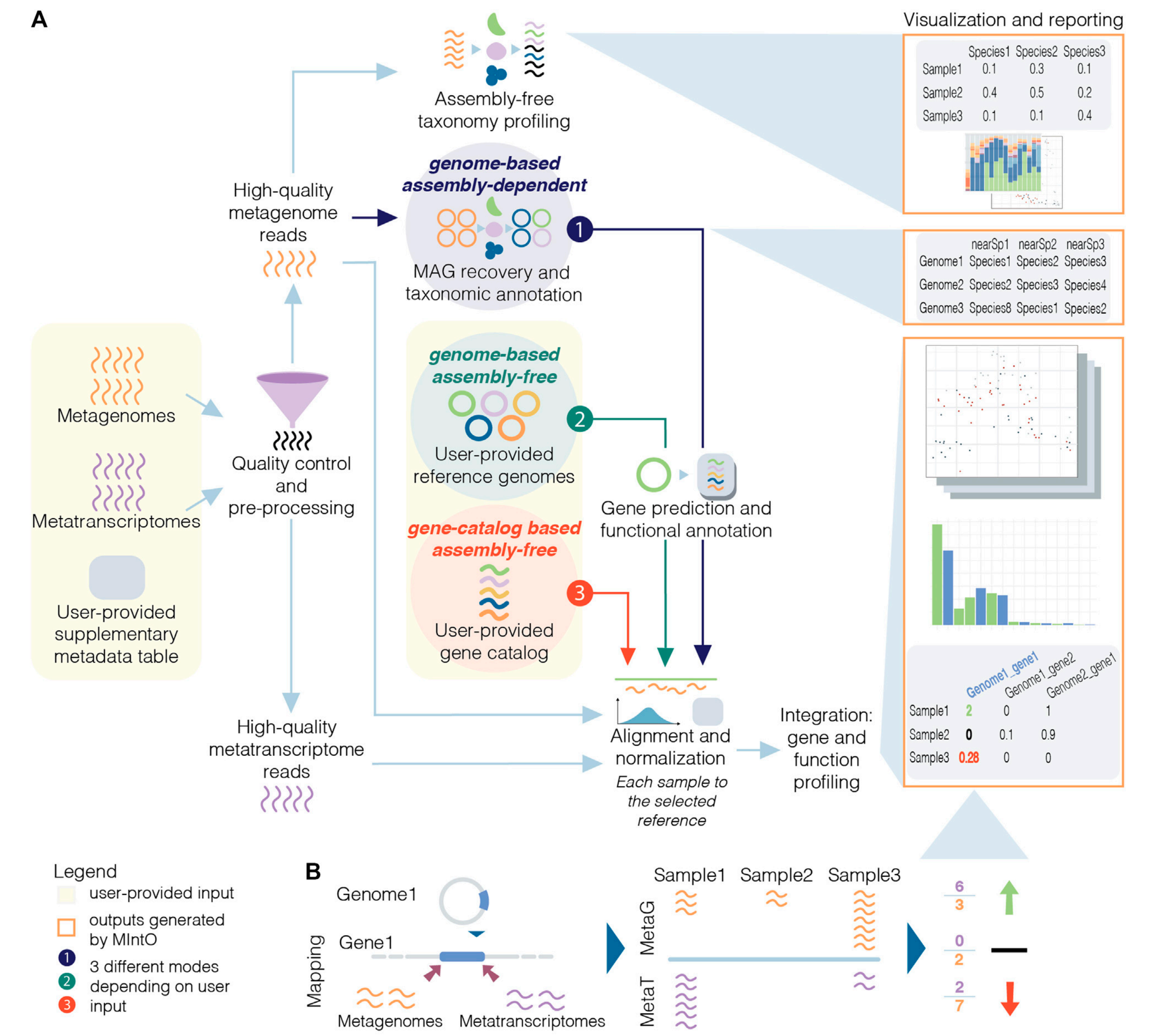

A
Visualization and reporting
Species1 Species2 Species3
Sample1 0.1 0.3 0.1
Sample2 0.4 0.5 0.2
Sample3 0.1 0.1 0.4
nearSp1 nearSp2 nearSp3
Genome1 Species1 Species2 Species3
Genome2 Species2 Species3 Species4
Genome3 Species8 Species1 Species2
Assembly-free taxonomy profiling
High-quality metagenome reads
genome-based assembly-dependent
MAG recovery and taxonomic annotation
1
Metagenomes
Metatranscriptomes
User-provided supplementary metadata table
Quality control and pre-processing
genome-based assembly-free
User-provided reference genomes
2
gene-catalog based assembly-free
User-provided gene catalog
3
Gene prediction and functional annotation
Alignment and normalization
Each sample to the selected reference
Integration: gene and function profiling
Genome1_gene1 Genome1_gene2 Genome2_gene1
Sample1 2 0 1
Sample2 0 0.1 0.9
Sample3 0.28 0 0
High-quality metatranscriptome reads
Legend
user-provided input
outputs generated by MIntO
1 2 3 3 different modes depending on user input
B
Genome1
Gene1
Mapping
Metagenomes
Metatranscriptomes
Sample1 Sample2 Sample3
MetaG
MetaT
6/3
0/2
2/7

FIGURE 1 | Schematic overview of metagenomic and metatranscriptomic integration to quantify gene expression levels. **(A)** Three modes are available based on the input data and the experiment design: the *genome-based assembly-dependent* mode (1, in dark purple) recovers MAGs from metagenomic samples, while the *genome-based assembly-free* (2, in dark green) and the *gene-catalog-based assembly-free* (3, in red) modes use publicly available genomes or a gene catalog, respectively, provided by the user. In the three modes, the pipeline workflow includes quality control and preprocessing; assembly-free taxonomy profiling of high-quality metagenomic reads (in orange) by identifying phylogenetic markers (coloured); alignment of the high-quality reads to the selected reference and normalization; integration: gene and functional profiling; and visualization and reporting. The gene prediction and functional annotation step is run using the recovered MAGs (mode 1) or publicly available genomes (mode 2). **(B)** The variation of gene expression depends on the abundance of transcripts from the organisms in the community and/or by changes in the abundance of these members and their related genes (community turnover).

Table 1, can handle both meta-omics data (Kim et al., 2016; Narayanasamy et al., 2016; Tamames and Puente-Sánchez, 2018; Salazar et al., 2019; Sequeira et al., 2019; Van Damme et al., 2021). Furthermore, only one of them (Van Damme et al., 2021) can combine two sequencing technologies (Nanopore or long-sequences and Illumina or short-sequences) to recover MAGs.

Overall, the pipelines shown in **Table 1** integrate metagenomic and metatranscriptomic data by comparing the abundances of genes and their respective transcripts. To the best of our knowledge, none of these (**Table 1**) considers the community composition and gene expression alterations as the underlying processes that shape the community transcript levels (Salazar et al., 2019) when integrating metagenomic and metatranscriptomic data. However, perturbations of the transcript levels can be a consequence of two factors: the variation in the expression of genes encoded by the organisms in the community, and/or by changes in the abundance of these members and their related genes in a process known as community turnover (Satinsky et al., 2014; Salazar et al., 2019). Hence, the integration of abundances of genes and the respective transcripts represents the gene expression profiles, which are the relative amount of transcripts per gene in a specific time (Salazar et al., 2019). Additionally, being able to recover genomes from metagenomic raw reads is crucial for an optimal computation of gene expression levels and provides a more accurate ecological description of the community's functioning (Tamames and Puente-Sánchez, 2018).

Here, we introduce MIntO (Microbiome Integrated meta-Omics), a pipeline that includes state of the art tools to integrate microbiome metagenomic and metatranscriptomic data in a scalable way for read pre-processing, species composition profiling, MAG generation, gene and function expression profiling, as well as the visualization of the results and comparison of multiple samples. Optionally, MIntO can combine long-read sequences for more contiguous assemblies and short-read sequences for higher accuracy, which helps recover more accurate as well as complete MAGs (Bertrand et al., 2019; Overholt et al., 2020; Brown et al., 2021). Depending on the data availability and research question, the pipeline can be run in three modes: (A) *genome-based assembly-dependent*, (B) *genome-based assembly-free* and (C) *gene-catalog-based assembly-free* (**Figure 1A**).

MIntO enables the study of microbial ecology by linking functions to genomes and environmental context, helping to understand the dynamics of the molecular activities captured by the whole community-level changes in composition and gene expression (**Figure 1B**).

METHODS

MIntO v1.0.1 has been developed using R software (v4.0.3) (The R Project for Statistical Computing, 2021), Python 3 (Van Rossum and Drake, 2009) and Perl (Wall, Christiansen and Orwant, 2000) programming languages, and has been tested on a 64-bit Linux server with 2 × AMD EPYC 7742 64-Core Processors and 2 terabytes of memory.

Conda Environment and Singularity Containers

MIntO has been designed to use publicly available software that are available as conda environments (Anaconda Inc, 2020) or singularity containers (Kurtzer, Sochat and Bauer, 2017) to minimize the installation of individual software packages by the user. All software dependencies are tied to specific versions in conda or singularity containers to ensure reproducibility and record-keeping of versions of the different libraries. It is encapsulated within a user-friendly framework using Snakemake (Mölder, 2021) to facilitate the scalability of the pipeline by optimizing the number of parallel processes from a single-core workstation to compute clusters. This pipeline enables consistency of the results and straightforward application by users with basic informatics skills to analyze complex omics data.

Pipeline Inputs

MIntO requires a configuration file as an input indicating the metagenomic (metaG) and/or metatranscriptomic (metaT) sample names and the corresponding raw FASTQ files location together with the path of the pipeline dependencies, currently only FetchMGs (Kultima et al., 2012). MIntO generates the necessary directories and outputs the required files for further analysis, including the configuration files needed in each step of the pipeline, but they should be filled out by the user. Optionally, the required databases can be downloaded and installed by MIntO.

In addition, if MIntO is run under *genome-based assembly-free* mode, the user should provide input genomes as FASTA files, genome features as GFF files, and amino acid sequences of protein-coding genes as FASTA files, while in the case of *gene-catalog based assembly-free* mode the user should provide a multi FASTA file with the nucleotide sequences of the genes, such as the one published with the Integrated Gene Catalog (IGC) (Li et al., 2014) (**Figure 1A**, user-provided input).

Pre-Processing of Metagenomic and Metatranscriptomic Short Reads

MIntO pre-processes metagenomic and metatranscriptomic short reads independently of each other. The pre-processing step can be subdivided into three different steps: quality and read length, host genome and ribosomal RNA (rRNA) filtering.

1. Quality and read length filtering.

We use Trimmomatic v0.39 (Bolger, Lohse and Usadel, 2014) to first remove sequencing adapters and low quality bases from raw reads and a second time to remove reads that are too short.

a. In the first step, the option *TRAILING:5 LEADING:5 SLIDINGWINDOW:4:20 ILLUMINACLIP:{adapters.fa}:2:30:10* is used if a sequence adapters file is provided by the user (*trimmomatic_adaptors = <PathTo>/adapters.fa*). Otherwise, a custom script retrieves the adapters by selecting the most abundant index in the first 10,000 headers of the raw FASTQ files (*trimmomatic_adaptors = False*). The user can decide to skip this step if adapter sequences have already been removed (*trimmomatic_adaptors = Skip*).
b. For the second filtering, the *MINLEN* parameter in Trimmomatic is used to remove reads that are too short. This cutoff is estimated as the maximum length above which a predefined percentage of the reads from the previous step are retained (default parameter is 95% of the reads, *perc_remaining_reads: 95*). If the estimated read length cutoff is below 50bp, trimmomatic will use 50bp as the minimum sequence length (**Supplementary Figure S3**).

2. Host genome filtering.

In the second step to remove putative host-derived sequences, the filtered read-pairs are aligned to a reference genome given by the user. The BWA aligner (Vasimuddin et al., 2019) version 2.2.1 is used to generate the index (*bwa-mem2 index*) and to map the read-pairs to the host genome (*bwa-mem2 mem -a*). Read-pairs aligned to this reference genome are identified by msamtools v1.1.0 (Arumugam, 2022) (*filter -S -l 30*) and excluded from the FASTQ files by mseqtools (https://github.com/arumugamlab/mseqtools) version 0.9.1, even if only one end is mapped (*subset --exclude --paired --list {listfile}*).

3. Ribosomal RNA filtering.

Prior to sequencing, it is recommended to deplete the rRNA in the metatranscriptomic samples. Nevertheless, it is common that metatranscriptomic sequence data still contains rRNA after such a depletion step. MIntO uses SortMeRNA v4.3.4 (Kopylova, Noé and Touzet, 2012) to map the metatranscriptomic reads to an rRNA sequence database consisting prokaryotic (16S and 23S) and eukaryotic (18S and 28S) rRNA sequences (*--paired_in --fastx --blast 1 --sam --other --ref*). Reads classified as rRNA by SortMeRNA are excluded from the FASTQ files using mseqtools (*subset --exclude --paired --list {listfile}*).

The remaining high-quality filtered (host-free for metagenomic and host- and rRNA-free for metatranscriptomic) reads are then passed to the sequence analysis and post processing steps.

Assembly-Free Taxonomic Profiling From High-Quality Filtered Reads

High-quality filtered reads can be profiled by the default program, MetaPhlAn3 v3.0.13 (Beghini et al., 2021) (*--input_type fastq --bowtie2out -t rel_ab_w_read_stats*). Alternatively, users can choose to run mOTUs2 v2.1.1 (Milanese et al., 2019) in two different modes to generate a taxonomic profile as relative abundance (taxa_profile: *motus_rel*, *profile -u -q*) or as counts (taxa_profile: *motus_raw*, *profile -c -u -q*). If the latter one is chosen, MIntO estimates the relative abundance of the taxonomic profile. To explore the similarities and dissimilarities of the data, the relative abundance of the species composition is used to generate two visual outputs: 1) the 15 most abundant genera across the samples, and 2) a principal coordinate analysis (PCoA) using Bray-Curtis distance. These visualizations provide users with a general idea of the microbial composition in the different samples. For a more detailed downstream analysis, MIntO outputs the combined table of the taxonomy profiles of all samples in CSV format and as a phyloseq object (McMurdie and Holmes, 2013), the latter including the abundance of the species, taxonomic classification and metadata tables.

Retrieving MAGs From Metagenomic High-Quality Host-Free Reads

MIntO's approach to reconstruct MAGs from high-quality host-free reads exploits metagenomic assembly of single samples as well as co-assembly of pre-defined sample groups followed by binning preparation and contig binning.

1. Assembly:

a. Long-read assembly: If available, Nanopore reads are assembled individually using metaFlye assembler (Kolmogorov et al., 2020) v2.9 (*--nano-raw <FASTQ> --meta --min-overlap 3000 --iterations 3*)
b. Short-read assembly: MetaSPAdes assembler v3.15.3 (Nurk et al., 2017) is used to correct paired-end short reads from individual samples (*--only-error-correction,* the default *--phred-offset* is auto) followed by their single-assembly (*--meta --only-assembler*, the default kmer option is k = *21,33,55,77,99,127*).
c. Hybrid assembly: Optionally, we can combine metagenomic Nanopore-based long reads and Illumina paired-end short reads to perform hybrid assembly by MetaSPAdes using the parameters as step (b) with an additional *--nanopore* option.
d. Co-assembly: MEGAHIT (Li et al., 2015) v1.2.9 is run with two different parameters (*--meta-sensitive* and *--meta-large*) per co-assembly, where by default all samples used in the single-assembly are assembled together. Users can also define their own subsets of samples that should be co-assembled in the configuration file.

2. Binning preparation:

Contigs longer than 2,500 bp from all the combinations of assemblies above are combined together in preparation for binning. Metagenomic reads from individual short-read metagenomes are first mapped to this set of contigs using BWA aligner (Vasimuddin et al., 2019) v2.2.1 (*bwa-mem2 mem -a*) in paired-end mode. Sequencing depth of the contigs in each sample is estimated by *jgi_summarize_bam_contig_depths* program included in MetaBAT2 (Kang et al., 2019).

3. Contig binning:

Contig binning is then performed by executing VAMB (Nissen et al., 2021), a binner using an unsupervised deep learning approach in the form of variational autoencoders that can be run with or without GPUs. GPU use is highly recommended if available in order to speed up the binning process, especially if working with a large number of samples. By default, MIntO runs VAMB four times, each time with a different set of parameters *-l 16 -n 256,256*; *-l 24 -n 384,384*; *-l 32 -n 512,512*; and *-l 40 -n 768,768*. However the user(s) can choose to perform just one run or a set of runs of their choice.

4. Non-redundant MAGs:

Bins generated by VAMB are split into MAGs derived from individual metagenomic samples. Only the MAGs that pass quality control using CheckM (Parks et al., 2015) (completeness > 90% and contamination < 5%) are kept. The MAGs are then subjected to cluster analysis performed with CoverM v0.6.0 (https://github.com/wwood/CoverM#usage, module cluster) in order to dereplicate them at 99% average nucleotide identity (ANI) (Jain et al., 2018). For each genome, a score is retrieved with the formula below.

$$assembly\ score = log10(longest\ contig\ length/\#contigs) + log10(N50/L50)$$
$$genome\ score = completeness - 2*contamination$$
$$final\ score = 0.1*genome\ score + assembly\ score$$

Then for each cluster the genome with the highest score is chosen, generating a unique set of non-redundant MAGs which will be used in the next step.

Taxonomic Assignment of MAGs

Once the unique set of MAGs is retrieved, taxonomy is assigned using the module *phylophlan_metagenomic* in PhyloPhlAn3 (Asnicar et al., 2020). MIntO uses SGB.Jul20 or SGB.Dec20 databases depending on user's choice (*--database*) which will be automatically downloaded in the program folder if no other location is specified. Additionally, if the users have previously downloaded one of the PhyloPhlAn3 databases of their interest, they can use that by giving their path.

Genome Annotation on the Retrieved MAGs

First, Prokka (Seemann, 2014) (version 1.14) (with options *--addgenes --centre X --compliant*) is used to identify and annotate the genes from the recovered MAGs, retrieving the corresponding nucleotide and amino acid sequences.

Next, predicted genes are annotated with several databases:

- eggNOG database (Huerta-Cepas et al., 2019) (COG ids) with eggNOG-mapper v2.1.6 (Huerta-Cepas et al., 2017; Cantalapiedra et al., 2021) (*--no_annot --no_file_comments --report_no_hits --override -m diamond* and *--annotate_hits_table -m no_search --no_file_comments --override*, emapperdb v5.0.2).
- KEGG functions (Kanehisa and Goto, 2000) (*-k -p prokaryote.hal --create-alignment -f mapper*, Kofam_scan (Aramaki et al., 2020) version 1.3.0 and ko_list from November 2021).
- Carbohydrate-active enzyme database [CAZyme, (Huang et al., 2018; Zhang et al., 2018)] with dbCAN annotation tool v2.0.11 (Zhang et al., 2018) (*run_dbcan.py protein*).
- Pfam database (Mistry et al., 2021) with eggNOG-mapper (Huerta-Cepas et al., 2017; Cantalapiedra et al., 2021).

These databases are installed locally by the user. The pipeline integrates the different gene annotations: Gene ID, eggNOG, KEGG_ko, KEGG_Pathway, KEGG_Module, dbCAN.mod, dbCAN.enzclass and Pfam.

Functional Profiling

The high-quality filtered (host-free for metagenomic and host- and rRNA-free for metatranscriptomic) reads are used to generate the functional profiles following four steps: metagenomic and metatranscriptomic read alignments, mappability ratio, read count normalization, and gene and function expression computation.

Metagenomic and Metatranscriptomic Reads Alignment

To estimate gene and transcript abundances, the high-quality filtered reads can be aligned to 1) genomes such as the recovered MAGs or publicly available genomes (*genome-based*) or 2) a gene catalog (*gene-based*), depending on the mode that the pipeline is run.

1. *Genome-based* alignment: The retrieved MAGs or the reference genomes are concatenated and indexed using the BWA aligner (Vasimuddin et al., 2019) v2.2.1 (*bwa-mem2 index*). Mapping reads to the reference (*bwa-mem2 mem -a*) is followed by highest-scoring alignment(s) filtering for each read with msamtools v1.1.0 (Arumugam, 2022) (*filter -S -b -l 50 -p 95 -z 80 --besthit*). The filtered BAM files are indexed by samtools v1.14 (Danecek et al., 2021) (*sort --output-fmt = BAM*; *index*) and the GFF file with the genome features is used to quantify the raw number of aligned reads to each gene by bedtools *multicov* v2.29.2 (Quinlan and Hall, 2010).
2. *Gene-based* alignment: As an alternative, the gene catalog given by the user is indexed using *bwa-mem2 index* [BWA aligner v2.2.1 (Vasimuddin et al., 2019)]. The aligned reads (*bwa-mem2 mem -a*) are filtered for highest-scoring alignment(s) per read with msamtools v1.1.0 (Arumugam, 2022) (*filter -S -b -l 50 -p 95 -z 80 --besthit*).

Optionally, the user can filter the aligned reads by establishing the minimum number of mapped reads to a gene, using the *MIN_mapped_reads* parameter. While the default value for this parameter is 0, for metagenomes with sequencing depth higher than 10 million paired-end reads, we recommend setting this threshold at 10 mapped reads to a gene (MIN_mapped_reads: 10), which is what we used for IBDMDB dataset.

Mappability Ratio

In addition, to estimate how representative the gene or genome databases are of the metagenomic and metatranscriptomic samples, the filtered BAM files are used to calculate the mappability ratio by msamtools v1.1.0 (Arumugam, 2022) (*profile --total {total_reads} --multi prop --unit all --nolen*). Here, we used the IGC (Li et al., 2014) and recovered MAGs as references.

Read Count Normalization

Normalization of read counts makes possible the comparison within or between different samples. Based on the users' selection, TPM (Transcripts Per Kilobase Million) or MGs (Marker Genes) normalized gene and transcript abundance profiles are generated from the metagenomic and metatranscriptomic read alignments, respectively.

1. *TPM normalization*. Sequencing depth and gene length are used to obtain the relative abundance of genes or transcripts (Wagner, Kin and Lynch, 2012). The TPM value of the gene *i*, TPM(*i*), is calculated by employing the equation:

$$TPM\,(i) = \frac{reads\,mapped\,to\,gene/gene\,length}{sum\,(reads\,mapped\,to\,gene/gene\,length)} \times 10^6$$
$$= \frac{n_i/l_i}{\Sigma_j\,(n_j/l_j)} \times 10^6$$

where n_i is the number of reads mapped to the gene i, l_i is the length of that gene and j iterates over all genes identified in the sample.

2. *MGs normalization.* In a similar approach to Salazar et al. study, but more customized to MAG-based analysis, the gene or transcript abundances of a MAG are divided by the median abundance of 10 universal single-copy phylogenetic MGs from the corresponding MAG (Salazar et al., 2019). These MGs are identified in each MAG by FetchMGs v1.2 (available at http://motu-tool.org/fetchMG.html) as OGs: COG0012, COG0016, COG0018, COG0172, COG0215, COG0495, COG0525, COG0533, COG0541, and COG0552. In addition, these MGs are constitutively expressed housekeeping genes across many different conditions (Sunagawa et al., 2013; Milanese et al., 2019; Salazar et al., 2019). Thus, the MGs-normalized metagenomic and metatranscriptomic profiles can be interpreted as the gene and transcript abundances in a MAG relative to housekeeping MGs abundance and transcript, respectively. The MGs value of the gene i, MGs(i), is calculated by employing the equation:

$$MG\,(i) = \frac{reads\,mapped\,to\,gene/gene\,length}{median\,10\,MGs\,from\,a\,genome} \times 10^6$$
$$= \frac{n_i/l_i}{M\,(MGs)} \times 10^6$$

where n_i is the number of reads mapped to the gene i in the gene's MAG, l_i is the length of that gene and $M(MGs)$ is the median abundance of the 10 MGs from the gene's genome. When the reads are mapped to a gene database, msamtools v1.1.0 (Arumugam, 2022) is used to normalize the number of aligned reads per gene to TPM (*profile --total {total_reads} --multi prop --unit tpm*). However, if the reads are mapped to a set of MAGs or publicly available genome(s), the user can choose to obtain TPM or MGs normalized abundances.

Computing Gene and Function Expression Profiles

The levels of gene expression are computed by the integration of gene and transcript abundance profiles, which is, the relative amount of RNA molecules per DNA copy of that gene (TPM normalization):

$$\text{gene expression} = \text{transcript abundance/gene copy number}$$

Or gene expression in that MAG relative to housekeeping MGs expression (MGs normalization):

$$\text{MGs-normalized gene expression}$$
$$= \text{gene expression /median MGs gene expression}$$

Finally, functional profiles are obtained by grouping the genes into functions.

Visualization

All the visualization outputs are generated in R software (v4.0.3) (The R Project for Statistical Computing, 2021), using the following packages: BiocManager (v1.30.16) (Morgan, 2021), data.table (v1.14.2) (Dowle and Srinivasan, 2021), reshape2 (v1.4.4) (Wickham, 2007), phyloseq (v1.34.0) (McMurdie and Holmes, 2013), tidyverse (v1.3.1) (Wickham et al., 2019), ggplot2 (v3.3.5) (Wickham, 2016), ggrepel (v0.9.1) (Wickham, 2007; Slowikowski, 2021), dplyr (v1.0.7) (Wickham et al., 2021), tidyr (v1.1.4) (Wickham and Girlich, 2021), stringr (v1.4.0), rlang (v0.4.11) (Henry and Wickham, 2021), haven (v2.4.3) (Wickham and Miller, 2021), vegan (v2.5-7) (Oksanen et al., 2020), keggrest (v1.30.1) (Tenenbaum, 2017), and pfam.db (v3.12.0). To have a better representation of the result, it is recommended to provide a metadata table by including the file path in the config file (*METADATA*) with sample ID, conditions and sample alias columns. If no metadata are provided, the sample IDs are used to generate the plots. However, the user can always use MIntO outputs for further downstream analysis.

Data

Inflammatory Bowel Disease Multi'Omics Database Samples

We used 91 human fecal metagenomes from the Inflammatory Bowel Disease Multi'omics Database [IBDMDB, (Lloyd-Price et al., 2019)]. The IBDMDB study provides matching Illumina metagenomic and metatranscriptomic data. We selected six participants diagnosed as non-IBD [P6018 (nIBD1), M2072 (nIBD2)]; Crohn's disease [H4006 (CD1) and H4020 (CD2)]; and ulcerative colitis [H4019 (UC1) and H4035 (UC2)] that were followed for 1 year each (**Supplementary Table S1**). Sample H4019_20 was not included due to a parsing error. Sequence data were retrieved from NCBI Short Read Archive under BioProject identifier PRJNA398089.

Paired-End Illumina and Nanopore-Based Metagenomic Data From Head and Neck Cancer Patients

We used human fecal metagenomes from head and neck cancer (HNC) patients (Wongsurawat et al., 2019), where samples were sequenced using Illumina and Nanopore technologies. We selected a subset of five patients: PatientHNC_03, PatientHNC_05, PatientHNC_06, PatientHNC_08 and PatientHNC_10. These were obtained from NCBI Short Read Archive under the accession numbers SRR7947170, SRR7947175, SRR7947177, SRR7947178, SRR7947179, SRR7947181, SRR7947184, SRR7947185, SRR7947186 and SRR7947187.

Human Genome

During MIntO pre-processing, the human genome (build hg38) was used to remove putative host-derived sequences (host genome filtering step).

Implementation of the Pipeline

MIntO implementation and automation are achieved by Snakemake (Mölder, 2021), a user-friendly framework that facilitates the scalability of the pipeline by optimizing the number of parallel processes from a single-core workstation to compute clusters. MIntO leverages singularity containers (Kurtzer, Sochat and Bauer, 2017) and Conda environments (Anaconda Inc, 2020) to ensure version control of the different libraries and implements a pipeline connecting several state of the art bioinformatic tools. In this way, MIntO enables consistency of the results and straightforward application by users with basic informatics skills to analyze complex omics data. The only dependencies are FetchMGs and Conda.

RESULTS

MIntO can be run in three different modes, thanks to its modular design, depending on the user's preference and available data: *genome-based assembly-free*, *gene-catalog-based assembly-free* and *genome-based assembly-dependent*. For all the three modes, users have to input FASTQ files from metagenomic and/or metatranscriptomic paired-end raw short reads and optionally, nanopore-based long reads, as well as a configuration file indicating the metagenomic and/or metatranscriptomic sample names and the corresponding location of raw FASTQ files. In the *genome-based assembly-dependent* mode, the given metagenomes are used to retrieve MAGs, while in the two *assembly-free* modes, *genome-based* or *gene-catalog-based*, the user also has to provide a set of reference genomes or a gene-catalog database, respectively, to generate the gene and functional profiles. These two options could be used when the user is working with a defined community or when there are not enough metagenomic samples to generate representative MAGs. These three modalities are illustrated in **Figure 1A**.

MIntO can be divided into seven major steps, which will be discussed in the next paragraphs using our analysis of example data (**Figure 1A**):

1. Quality control and pre-processing
2. Assembly-free taxonomy profiling
3. Recovery of MAGs and taxonomic annotation (only run in *genome-based assembly-dependent* mode)
4. Gene prediction and functional annotation (only run in *genome-based* modes)
5. Alignment and normalization
a. *genome-based* mode: recovered MAGs or publicly available genomes
b. *gene-based* mode: gene catalog
6. Integration: Gene and functional profiling
7. Visualization and reporting

The third step is skipped if an assembly-free mode is selected, and the fourth step is skipped when *gene catalog-based assembly-free* mode is chosen (**Figure 1A**). An overview of the directories generated can be seen in **Supplementary Figure S1**.

To illustrate the use of MIntO, a set of 91 human fecal metagenomes from the Inflammatory Bowel Disease Multi'omics Database (IBDMDB) was selected (Lloyd-Price et al., 2019). These samples correspond to six participants diagnosed as non-IBD (nIBD1 and nIBD2), Crohn's disease, (CD1 and CD2) and ulcerative colitis, (UC1 and UC2), which were followed for 1 year each (**Supplementary Figure S2**, **Supplementary Table S1**). The IBDMDB study provides matching Illumina metagenomic and metatranscriptomic data. The subset of samples used here correspond to 933.4 and 612 million read-pairs (2 × 101 bp) from metagenomic and metatranscriptomic sequencing, respectively (mean 10.85 million read-pairs, ranging from 0.26 to 21.04 million for metagenomic; mean 6.18 million read-pairs, ranging from 0.01 to 15.72 million for metatranscriptomic).

TABLE 2 | Median (minimum and maximum) of raw and high-quality million read-pairs in the 91 human fecal microbiome samples from the IBDMDB.

	metagenomic	metatranscriptomic
Raw read-pairs (millions)	10.85 (10.15–21.04)	6.18 (6.65–15.72)
High quality read-pairs (millions)	10.56 (9.9–20.58)	6.04 (6.52–15.45)

Here, we present the results from the *genome-based assembly-dependent* and *gene catalog-based assembly-free* modes, where we used recovered MAGs and the Integrated Gene Catalog (IGC) (Li et al., 2014), respectively, as reference to profile genes and functions.

Quality Control and Pre-Processing

The IBDMDB dataset was already filtered by quality and sequence adapters, therefore the first step in the pre-processing of the 91 samples was skipped (*trimmomatic_adaptors = Skip*, see Methods). We then used a minimum read length cutoff of 53 bp for metagenomic and 54 bp for metatranscriptomic to keep 95% of the longest sequences using Trimmomatic (Bolger, Lohse and Usadel, 2014) (**Supplementary Figure S3**).

Subsequently, putative host-derived sequences were removed using the human genome (build hg38). In silico rRNA sequences screening was exclusively applied to metatranscriptomic reads using SortMeRNA (Kopylova, Noé and Touzet, 2012). This resulted in a total number of 599.4 million high-quality read-pairs for metagenomic and 910.9 million high-quality read-pairs for metatranscriptomic data (**Table 2**, **Supplementary Figure S4**).

Assembly-Free Taxonomy Profiling

Once the reads were pre-processed, high-quality reads were profiled at species level using MetaPhlAn3 (Beghini et al., 2021) (**Figure 1A**, assembly-free taxonomy profiling step). In **Figure 2A**, we can see the temporal shifts and dynamics exhibited by microbes over the course of 1 year and the difference of microbial composition between the six participants focusing on the 15 most abundant genera across the samples. In general, the most predominant genera are *Bacteroides*, *Faecalibacterium* and *Roseburia*. The constitution of a separate cluster by samples from participant nIBD2 in **Figure 2B** cannot be explained by the 15 most abundant genera across the samples (**Figure 2A**), but it could be due to the difference in composition of lower-abundance bacteria.

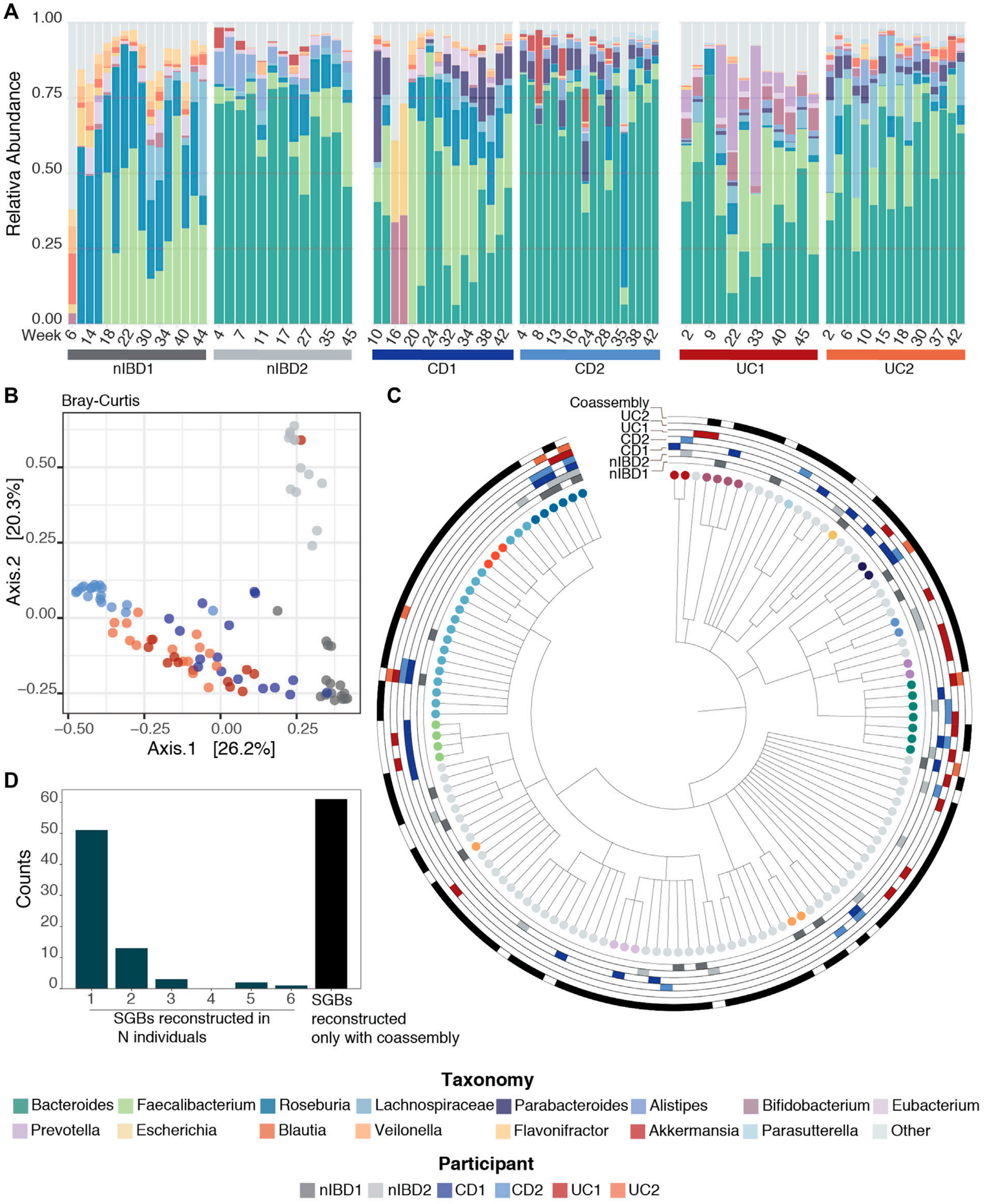


FIGURE 2 | Taxonomic profiles. **(A)** Relative abundance for the 91 samples for the 15 most abundant genera across the samples using MetaPhlAn3 (Beghini et al., 2021). **(B)** Projection of the first two principal coordinates based on Bray–Curtis dissimilarity from the microbiome composition using MetaPhlAn3 (Beghini et al., 2021). **(C)** Taxonomy tree representing the 131 SGBs taxonomies after running PhyloPhlAn3 (Asnicar et al., 2020) on the retrieved MAGs. The first six rings mark MAGs that were retrieved in the 6 patients with the different conditions used in this work (nIBD, CD and UC), while the last ring marks the MAGs obtained from co-assembly. **(D)** Distribution of the SGBs in the 6 patients: 51 SGBs taxonomies were retrieved from just one sample, 13 from two samples, 3 from three samples, 2 from five samples and 1 in all the samples. The last bar represents the 61 taxonomies that were found only by having performed co-assembly.

TABLE 3 | Number of SGB taxonomies retrieved per sample.

Sample/Method	Number of Taxa
nIBD1	18
nIBD2	21
CD1	24
CD2	15
UC1	21
UC2	24
Co-assembly	100

Recovery of MAGs and Taxonomic Annotation

In parallel, the pre-processed reads underwent the assembly step in the *genome-based assembly-dependent* mode (**Figure 1A**, recovery of MAGs and taxonomic annotation step). As this dataset consists of short-read metagenomes only, we used two assembly approaches to recover high-quality scaffolds: 1) assembly of each metagenome individually (single-assembly) using MetaSPAdes assembler (Nurk et al., 2017) and 2) assembly of all metagenomes together (co-assembly) using MEGAHIT (Li et al., 2015) assembler. Genome bins were generated from assembled scaffolds that were at least 2,500 bp long by mapping the 91 samples individually to the scaffolds, calculating the sequence depth of each scaffold in the 91 samples, and finally running VAMB (Nissen et al., 2021) four times with different parameters and GPU mode (see Methods).

After binning, 5,048 MAGs were retrieved from the 91 metagenomic samples. Using CheckM (Parks et al., 2015), we identified high-quality (HQ) MAGs (completeness > 95% and contamination < 5%) and kept 957 MAGs. We then obtained unique high-quality MAGs when clustering the HQ MAGs at 99% ANI distance (Jain et al., 2018) with CoverM (https://github.com/wwood/CoverM#usage) and choosing the best genome in a given cluster using a genome quality score (see Methods). This de-replication process resulted in 163 MAGs which constituted a set of non-redundant genomes (available at 10.5281/zenodo.6360083). These MAGs are useful to collectively explain the ecological description and biodiversity in the samples, and to capture sample-specific variation at functional and abundance level without relying on publicly available reference genomes. Additionally, working with a restricted number of genomes is helpful to speed up the next steps of the pipeline.

The taxonomic annotation of the 163 MAGs was performed by *phylophlan_metagenomic* module in PhyloPhlAn3 (Asnicar et al., 2020), which also provides taxonomic lineage information about the 10 nearest genomes in the PhyloPhlAn3 genome database. Each MAG was assigned to a species-level genome bin (SGB) if its closest genome in the database was within 5% average nucleotide identity. This resulted in the 163 MAGs falling into 131 SGBs (**Figure 2C**). In general, MAGs with a distance higher than 5% to the closest genome in the database can be considered as putative novel species (Manara et al., 2019; Pasolli et al., 2019). However, we did not recover any MAGs from putative novel species in this dataset.

By default, MIntO performs co-assembly, which although time consuming, is an extremely important step. In fact, we obtained the highest number of unique taxa from the co-assembled samples compared to any single-sample assembly (**Table 3**). Remarkably, 61 of the 131 taxonomies (~46%) could be retrieved only by performing co-assembly (**Figure 2D**). With single-sample assembly we still retrieved 31 (~23%) unique taxonomies not covered by the co-assembled samples, of which 13 (~10% of the total) are only found in one sample (**Figure 2C**). This is helpful to better distinguish sample-specific composition, as for example *Akkermansia muciniphila* SGB9228, which is the second *Akkermansiacae* species by presence in the human population (Karcher et al., 2021) can only be found in patient CD1. These results are achievable only by performing both single and co-assembly.

In addition, we performed our own benchmark to show that combining long and short reads improves the assembly contiguity. MIntO assembled paired-end metagenomes from the gut microbiota of five patients with head and neck cancer (Wongsurawat et al., 2019), which were generated by 1) Illumina-only, or 2) Illumina and Nanopore sequencing platforms. The number of generated scaffolds (127,315 and 172,888 for Illumina and Illumina + Nanopore, respectively), and their mean length (9.44 kb and 9.72 kb for Illumina and Illumina + Nanopore, respectively), were greater when long-reads were included in the assembly. Furthermore, Illumina + Nanopore assembly generated 13 scaffolds longer than 600 kb with a maximum of 1,119 kb, whereas the assembly of Illumina-only data generated 2 scaffolds longer than 600 kb with a maximum of 736 kb. Finally, the scaffold length distribution shows that scaffolds from Illumina + Nanopore assemblies are more contiguous than Illumina-only assemblies (**Supplementary Figure S6**).

Gene Prediction and Functional Annotation

The unique set of MAGs recovered in the previous step underwent gene prediction and functional annotation (**Figure 1A**, gene prediction and functional annotation). Prokka (Seemann, 2014) was used to identify and annotate the genes, retrieving the corresponding nucleotide and amino acid sequences. A total of 412,394 genes were predicted in the 163 recovered MAGs. These were annotated with seven different functional databases: eggNOG (Yin et al., 2012; Huerta-Cepas et al., 2019), KEGG Pathways, Modules and KOs (Kanehisa and Goto, 2000), dbCAN modules and enzyme classes (Yin et al., 2012), and Pfam (Mistry et al., 2021) (**Figure 3**). The same process could also be applied to user-provided genome sequences under *genome-based assembly-free* mode.

The gene and function annotation step was skipped in the *gene catalog-based assembly-free* mode as we used existing eggNOG, KEGG Pathways, KEGG Modules, KEGG KO, dbCAN modules, dbCAN enzymes class, Pfam function annotation for IGC (available at https://db.cngb.org/microbiome/genecatalog/genecatalog_human/). The number of expressed genes and functions for both modes are summarized in **Figure 3**. Even though we detected > 5 × genes by mapping the metagenomes to IGC compared to genes encoded in the 163 MAGs, genes from the MAGs covered the vast majority of the functions detected via

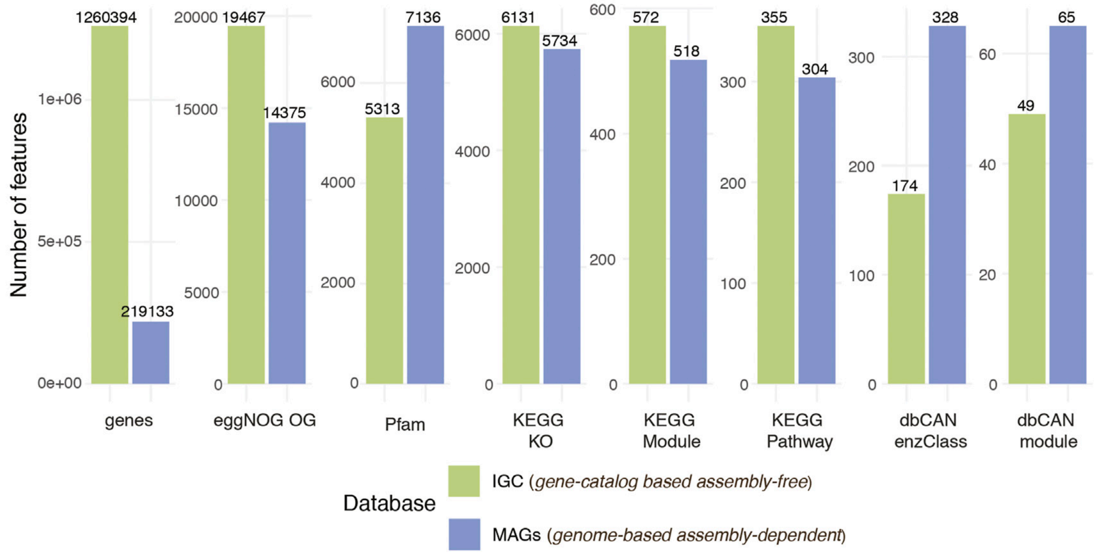


FIGURE 3 | Comparison of number of genes and features per function database between non-redundant high quality 163 MAGs and IGC.

IGC. In some cases such as Pfam and CAZy databases, MAGs recovered more functions suggesting that contiguous assemblies and more complete genes could improve the quality of functional annotations.

Alignment and Normalization

The metagenomic and metatranscriptomic high-quality reads were mapped to a reference database followed by TPM normalization to obtain the relative abundance of genes from metagenomic read alignments (i.e., gene abundance profile) and transcripts from metatranscriptomic read alignments (i.e., gene transcript profile) (**Figure 1A**, alignment, normalization and integration). We used as a reference database the 163 recovered MAGs for the *genome-based* mapping and the IGC (Li et al., 2014) for the *gene-based* alignment. Overall, the mappability rate at 95% of sequence identity for MAGs (median 72.26%) was lower than for IGC (median 92.47%) with the highest difference for participant CD2 (**Supplementary Figure S5**), which could be due also to the lower number of taxonomies retrieved for the samples (**Table 3**). However, this difference was not as remarkable when using metatranscriptomic reads (77.61 and 73.9% median, respectively).

Integration: Gene and Function Expression Profiling

The variation of microbial community transcript levels may be affected by the changes in gene expression and/or by the community turnover. To disentangle the individual contributions of these mechanisms across the different samples, we integrated gene abundance and transcript abundance profiles (Salazar et al., 2019) (see Methods). The obtained levels of gene expression represent the relative amount of expressed transcripts per gene (**Figure 1A**, integration: gene and functional profiling). From the 412,394 predicted genes in the 163 recovered MAGs, 219,133 genes were expressed in at least one sample, while we detected the expression of 1,260,394 genes from the 9.9 million genes in IGC.

Furthermore, the corresponding gene profiles were used to generate the function abundance, transcript and expression profiles by grouping the annotated genes into functions. The highest number of features detected in the samples corresponded to the eggNOG database on both modes, followed by Pfam or KEGG KO (**Figure 3**). We identified 5,734 and 6,131 KEGG KO expressed features when we used the recovered MAGs and IGC as a reference, respectively. Among the 7,217 KEGG KO functions identified between the two profiles, 64.4% (4,651 features) were found in both. The 15% of features (1,086) uniquely identified in the MAGs could correspond to genomes not included in the database and the 20.5% of the functions (1,481) detected in IGC could belong to low abundant bacteria whose genomes could not be retrieved or were missed due to MAGs filtered out based on our quality criteria.

We used MIntO's visualization features to perform principal coordinate analysis (PCoA) on the different gene and functional profiles to observe the longitudinal compositional changes and to compare the dissimilarities between participants. In **Figure 4A** we show the gene expression PCoA plot for the *assembly-free gene catalog* mode using IGC (Li et al., 2014). In general, the samples were clustered by Crohn's disease and Ulcerative colitis diagnosis suggesting a similar bacterial abundance and expressed genes due to the presence of the disease (Kostic, Xavier and Gevers, 2014; Lloyd-Price et al., 2019). Samples from participants used as control (nIBD1 and nIBD2) were clustered separately, probably due to the inter-individual variations in the microbiome composition. In fact, the most abundant genus in all participants was *Bacteroides*, with the exception of nIBD1 where *Roseburia* and *Faecalibacterium* were predominant. At transcript level (**Supplementary Figure S7**), the dissimilarity between the samples explained by the first two principal

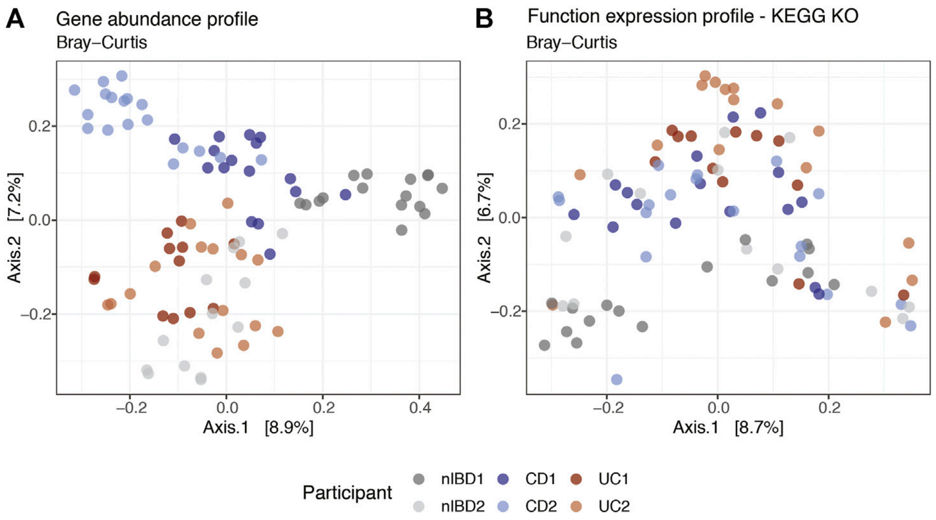


FIGURE 4 | Projection of the first two principal coordinates based on expression profiles Bray–Curtis dissimilarity at **(A)** gene and **(B)** function KEGG KO levels using a subset of 91 samples from IBDMDB. Labels correspond to the *sample alias* and are colored by *condition* (patients diagnosis).

coordinates (18.7% and 12.2%) was higher than at gene expression level (8.9% and 7.2%). The transcript abundance changes might be mainly attributed either to differences in the expression of genes encoded by the microbes in the community or changes in the abundance of these members and their related genes or a combination of these mechanisms. Hence, the computation of gene expression profiles by the integration of abundances of genes and the respective transcripts is of crucial importance to obtain a more accurate representation of ecologically relevant processes that are occurring.

Overall, the dissimilarities between the samples were visible at the gene expression, gene abundance and transcript abundance profiles (**Figure 4A** and **Supplementary Figure S7**). However, at function expression level (**Figure 4B**) the clusters were not as well defined, suggesting that genes from different species could harbor the same functions in different microbial communities. Although the taxonomic composition differed between the six participants and consequently the gene composition and expression, the functional profiles across individuals and time were more conserved (functional redundancy) (Tian et al., 2020). Differences in functional profiles between nonIBD and IBD diagnosed participants could provide insights into the functions involved in microbiome–host interactions at states of health or disease (Heintz-Buschart and Wilmes, 2018).

Visualization and Reporting

Further analyses can be done using the output files (**Figure 1A**, visualization and reporting; **Supplementary Figure S1**). MIntO generates three different types of table: 1) assembly-free and assembly-based taxonomic profiles; 2) gene profiles, including the gene IDs [generated by Prokka (Seemann, 2014; Beghini et al., 2021) when selecting *assembly-dependent* mode or sequence IDs when choosing *assembly-free* mode] and normalized gene abundance, transcript or expression; and 3) functional profiles per database, including the function IDs, function description and function abundance, transcript or expression normalized counts. For an easier downstream analysis of these data, phyloseq objects are generated for the taxonomic, gene and functional profiles.

MIntO also outputs the shown plots as preliminary results to help the user in the downstream analysis (**Figures 2A,B**, **Figures 3**, **4**, **Supplementary Figures S3, S7**).

The metadata provided in IBDMDB (**Supplementary Table S1**) was given as an input to the pipeline, which colored the samples by *sample_alias* (participant's ID) in the output plots.

DISCUSSION

MIntO is a versatile pipeline that integrates metagenomic and metatranscriptomic data, beyond a comparison of the gene and transcript abundances, in order to quantify gene and function expression in a very straightforward way. The modular design of MIntO enables the user to run the pipeline using three available modes based on the input data and the experimental design.

In order to illustrate the pipeline, a subset of 91 human fecal microbiome samples from the IBDMDB (Illumina metagenomic and metatranscriptomic paired-reads) was used to run the full version of the pipeline with default parameters. Here, we show the complementary results from two of the three available modes, *genome-based assembly-dependent* and *gene catalog-based assembly-free*. In the former, MIntO retrieved 163 high-quality non-redundant MAGs that encoded 412,394 genes, among which 219,133 genes were expressed in at least one sample, while 1,260,394 genes from IGC were expressed in the *gene catalog-based assembly-free* mode. Overall, the dissimilarities between the samples were visible at the taxonomic and gene levels, while the functional profiles across individuals and time were more conserved (functional redundancy), indicating that strain-specific genes from different microbiomes represented similar functions. Interestingly, among the 7,217 KEGG KO functions identified between the two profiles, 15% of the features were uniquely identified in the MAGs and 20.5% of the functions were detected in IGC.

The distinctive feature of this pipeline is the integration of the metagenomic and metatranscriptomic data, to obtain the expression profiles and furthermore the functional profiles by annotating the sequences with several databases. This enables us to study in detail the variation in expression of the genes and functions in the different samples across time and experiment conditions, thus the community behavior. Overall, the IBDMDB-samples clustered by the participant ID using the genes and transcript abundances and gene expression. However, using the KEGG KO annotations at function expression level, the clusters are not as well defined, due to the functional redundancy (Tian et al., 2020).

Another important feature of MIntO is performing *de novo* assembly and contig binning to recover high-quality MAGs from metagenomic reads, which compared to other methods utilizes an accurate unsupervised deep learning approach in the form of variational autoencoders (Nissen et al., 2021). The *assembly-dependent* mode could be helpful to retrieve novel genomes that are missed by reference-dependent profiling methods (Pasolli et al., 2019). The recovery of MAGs is indispensable to uncover the diversity of bacteria in an environment and it is crucial for an optimal calculation of the variation of gene expression, including unknown or functional genes from biosynthetic gene clusters (Youngblut et al., 2020). Additionally, new putative genomes can increase the number of known species in the available databases, especially when the analyses are performed on metagenomes coming from new environmental sources.

In conclusion, in this paper we show how MIntO can be a useful tool to analyze metagenomic and metatranscriptomic data in a standardized way, enabling the study of microbial ecology by linking functions to genomes and environmental context. We foresee that this pipeline will contribute to the understanding of the dynamics of the molecular activities captured by the community turnover and gene expression alterations as the cause that shapes community transcript levels. Elucidating the functions and characterizing the specific strains of a community will be crucial to increase our knowledge of the microbiome's contribution to human health and environment.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. Matching Illumina metagenomic and metatranscriptomic data from IBDMDB can be found here: https://ibdmdb.org/tunnel/public/summary.html, IBDMDB, BioProject identifier PRJNA398089. Matching shotgun metagenomic data generated from both Illumina and Nanopore technologies can be found in NCBI Short Read Archive under the accession numbers SRR7947170, SRR7947175, SRR7947177, SRR7947178, SRR7947179, SRR7947181, SRR7947184, SRR7947185, SRR7947186 and SRR7947187. Non-redundant MAGs constructed by MIntO from 91 metagenomes from IBDMDB are available at https://doi.org/10.5281/zenodo.6360083.

AUTHOR CONTRIBUTIONS

CS and MA conceived and designed the tool. CS, EN, VG, and MA created the software. CS, EN, and MA wrote the manuscript and performed all necessary testing. All authors read, revised, and approved the manuscript.

FUNDING

Novo Nordisk Foundation Center for Basic Metabolic Research is an independent Research Center, based at the University of Copenhagen, Denmark, and partially funded by an unconditional donation from the Novo Nordisk Foundation (www.cbmr.ku.dk) (Grant no. NNF18CC0034900). This work was supported by the Danish Council of Independent Research (Grant no. 6111-00471B). VG and EN were supported by the European Union's Horizon 2020 research and innovation program (GALAXY: Grant no. 668031).

ACKNOWLEDGMENTS

We are thankful to the members of the Arumugam group and Gabriel Fernandes for inspiring discussions and feedback on the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.846922/full#supplementary-material

REFERENCES

Almeida, A., Mitchell, A. L., Boland, M., Forster, S. C., Gloor, G. B., Tarkowska, A., et al. (2019). A New Genomic Blueprint of the Human Gut Microbiota. *Nature* 568 (7753), 499–504. doi:10.1038/s41586-019-0965-1

Anaconda Inc (2020). Anaconda Software Distribution, *Anaconda Documentation* [Preprint]. Available at: https://docs.anaconda.com/.

Aramaki, T., Blanc-Mathieu, R., Endo, H., Ohkubo, K., Kanehisa, M., Goto, S., et al. (2020). KofamKOALA: KEGG Ortholog Assignment Based on Profile HMM and Adaptive Score Threshold. *Bioinformatics* 36 (7), 2251–2252. doi:10.1093/bioinformatics/btz859

Arumugam, M. (2022). msamtools: Microbiome-Related Extension to Samtools. Available at: https://github.com/arumugamlab/msamtools (Accessed: March 31, 2022).

Asnicar, F., Thomas, A. M., Beghini, F., Mengoni, C., Manara, S., Manghi, P., et al. (2020). Precise Phylogenetic Analysis of Microbial Isolates and Genomes from Metagenomes Using PhyloPhlAn 3.0. *Nat. Commun.* 11 (1), 2500. doi:10.1038/s41467-020-16366-7

Bashan, A., Gibson, T. E., Friedman, J., Carey, V. J., Weiss, S. T., Hohmann, E. L., et al. (2016). Universality of Human Microbial Dynamics. *Nature* 534 (7606), 259–262. doi:10.1038/nature18301

Beghini, F., McIver, L. J., Blanco-Míguez, A., Dubois, L., Asnicar, F., Maharjan, S., et al. (2021). Integrating Taxonomic, Functional, and Strain-Level Profiling of

Diverse Microbial Communities with bioBakery 3. *eLife* 10, e65088. doi:10.1101/2020.11.19.388223

Bertrand, D., Shaw, J., Kalathiyappan, M., Ng, A. H. Q., Kumar, M. S., Li, C., et al. (2019). Hybrid Metagenomic Assembly Enables High-Resolution Analysis of Resistance Determinants and mobile Elements in Human Microbiomes. *Nat. Biotechnol.* 37 (8), 937–944. doi:10.1038/s41587-019-0191-2

Bolger, A. M., Lohse, M., and Usadel, B. (2014). Trimmomatic: a Flexible Trimmer for Illumina Sequence Data. *Bioinformatics* 30 (15), 2114–2120. doi:10.1093/bioinformatics/btu170

Brown, C. L., Keenum, I. M., Dai, D., Zhang, L., Vikesland, P. J., and Pruden, A. (2021). Critical Evaluation of Short, Long, and Hybrid Assembly for Contextual Analysis of Antibiotic Resistance Genes in Complex Environmental Metagenomes. *Sci. Rep.* 11 (1), 3753. doi:10.1038/s41598-021-83081-8

Cantalapiedra, C. P., Hernández-Plaza, A., Letunic, I., Bork, P., and Huerta-Cepas, J. (2021). EggNOG-Mapper V2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. *Mol. Biol. Evol.* 38, 5825–5829. [Preprint]. doi:10.1093/molbev/msab293

Danecek, P., Bonfield, J. K., Liddle, J., Marshall, J., Ohan, V., Pollard, M. O., et al. (2021). Twelve Years of SAMtools and BCFtools. *GigaScience* 10 (2), giab008. doi:10.1093/gigascience/giab008

Donia, M. S., and Fischbach, M. A. (2015). HUMAN MICROBIOTA. Small Molecules from the Human Microbiota. *Science* 349 (6246), 1254766. doi:10.1126/science.1254766

Dowle, M., and Srinivasan, A. (2021). data.table: Extension of "data.frame" [R Package data.table Version 1.14.2]. Available at: https://CRAN.R-project.org/package=data.table (Accessed: December 6, 2021).

Heintz-Buschart, A., and Wilmes, P. (2018). Human Gut Microbiome: Function Matters. *Trends Microbiol.* 26 (7), 563–574. doi:10.1016/j.tim.2017.11.002

Henry, L., and Wickham, H. (2021). rlang: Functions for Base Types and Core R and "Tidyverse" Features [R Package rlang Version 0.4.11]. Available at: https://CRAN.R-project.org/package=rlang (Accessed: December 6, 2021).

Huang, L., Zhang, H., Wu, P., Entwistle, S., Li, X., Yohe, T., et al. (2018). dbCAN-Seq: a Database of Carbohydrate-Active Enzyme (CAZyme) Sequence and Annotation. *Nucleic Acids Res.* 46 (D1), D516–D521. doi:10.1093/nar/gkx894

Huerta-Cepas, J., Forslund, K., Coelho, L. P., Szklarczyk, D., Jensen, L. J., von Mering, C., et al. (2017). Fast Genome-wide Functional Annotation through Orthology Assignment by eggNOG-Mapper. *Mol. Biol. Evol.* 34 (8), 2115–2122. doi:10.1093/molbev/msx148

Huerta-Cepas, J., Szklarczyk, D., Heller, D., Hernández-Plaza, A., Forslund, S. K., Cook, H., et al. (2019). eggNOG 5.0: a Hierarchical, Functionally and Phylogenetically Annotated Orthology Resource Based on 5090 Organisms and 2502 Viruses. *Nucleic Acids Res.* 47, D309–D314. doi:10.1093/nar/gky1085

Human Microbiome Project Consortium (2012). A Framework for Human Microbiome Research. *Nature* 486 (7402), 215–221. doi:10.1038/nature11209

Jain, C., Rodriguez-R, L. M., Phillippy, A. M., Konstantinidis, K. T., and Aluru, S. (2018). High Throughput ANI Analysis of 90K Prokaryotic Genomes Reveals clear Species Boundaries. *Nat. Commun.* 9 (1), 5114. doi:10.1038/s41467-018-07641-9

Kanehisa, M., and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* 28 (1), 27–30. doi:10.1093/nar/28.1.27

Kang, D. D., Li, F., Kirton, E., Thomas, A., Egan, R., An, H., et al. (2019). MetaBAT 2: an Adaptive Binning Algorithm for Robust and Efficient Genome Reconstruction from Metagenome Assemblies. *PeerJ* 7, e7359. doi:10.7717/peerj.7359

Karcher, N., Nigro, E., Punčochář, M., Blanco-Míguez, A., Ciciani, M., Manghi, P., et al. (2021). Genomic Diversity and Ecology of Human-Associated Akkermansia Species in the Gut Microbiome Revealed by Extensive Metagenomic Assembly. *Genome Biol.* 22 (1), 209. doi:10.1186/s13059-021-02427-7

Kim, J., Kim, M. S., Koh, A. Y., Xie, Y., and Zhan, X. (2016). FMAP: Functional Mapping and Analysis Pipeline for Metagenomics and Metatranscriptomics Studies. *BMC bioinformatics* 17 (1), 420. doi:10.1186/s12859-016-1278-0

Kolmogorov, M., Bickhart, D. M., Behsaz, B., Gurevich, A., Rayko, M., Shin, S. B., et al. (2020). metaFlye: Scalable Long-Read Metagenome Assembly Using Repeat Graphs. *Nat. Methods* 17 (11), 1103–1110. doi:10.1038/s41592-020-00971-x

Kopylova, E., Noé, L., and Touzet, H. (2012). SortMeRNA: Fast and Accurate Filtering of Ribosomal RNAs in Metatranscriptomic Data. *Bioinformatics* 28 (24), 3211–3217. doi:10.1093/bioinformatics/bts611

Kostic, A. D., Xavier, R. J., and Gevers, D. (2014). The Microbiome in Inflammatory Bowel Disease: Current Status and the Future Ahead. *Gastroenterology* 146 (6), 1489–1499. doi:10.1053/j.gastro.2014.02.009

Kultima, J. R., Sunagawa, S., Li, J., Chen, W., Chen, H., Mende, D. R., et al. (2012). MOCAT: a Metagenomics Assembly and Gene Prediction Toolkit. *PloS one* 7 (10), e47656. doi:10.1371/journal.pone.0047656

Kurtzer, G. M., Sochat, V., and Bauer, M. W. (2017). Singularity: Scientific Containers for Mobility of Compute. *PloS one* 12 (5), e0177459. doi:10.1371/journal.pone.0177459

Li, D., Liu, C.-M., Luo, R., Sadakane, K., and Lam, T.-W. (2015). MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* 31, 1674–1676. doi:10.1093/bioinformatics/btv033

Li, J., Jia, H., Cai, X., Zhong, H., Feng, Q., Sunagawa, S., et al. (2014). An Integrated Catalog of Reference Genes in the Human Gut Microbiome. *Nat. Biotechnol.* 32 (8), 834–841. doi:10.1038/nbt.2942

Lloyd-Price, J., Arze, C., Ananthakrishnan, A. N., Schirmer, M., Avila-Pacheco, J., Poon, T. W., et al. (2019). Multi-omics of the Gut Microbial Ecosystem in Inflammatory Bowel Diseases. *Nature* 569 (7758), 655–662. doi:10.1038/s41586-019-1237-9

Manara, S., Asnicar, F., Beghini, F., Bazzani, D., Cumbo, F., Zolfo, M., et al. (2019). Microbial Genomes from Non-human Primate Gut Metagenomes Expand the Primate-Associated Bacterial Tree of Life with over 1000 Novel Species. *Genome Biol.* 20 (1), 299. doi:10.1186/s13059-019-1923-9

McMurdie, P. J., and Holmes, S. (2013). Phyloseq: an R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. *PloS one* 8 (4), e61217. doi:10.1371/journal.pone.0061217

Milanese, A., Mende, D. R., Paoli, L., Salazar, G., Ruscheweyh, H. J., Cuenca, M., et al. (2019). Microbial Abundance, Activity and Population Genomic Profiling with mOTUs2. *Nat. Commun.* 10 (1), 1014. doi:10.1038/s41467-019-08844-4

Mistry, J., Chuguransky, S., Williams, L., Qureshi, M., Salazar, G. A., Sonnhammer, E. L. L., et al. (2021). Pfam: The Protein Families Database in 2021. *Nucleic Acids Res.* 49 (D1), D412–D419. doi:10.1093/nar/gkaa913

Mölder, F. (2021). Sustainable Data Analysis with Snakemake. *F1000Research* 10, 33. doi:10.12688/f1000research.29032.2

Morgan, M. (2021). Access the Bioconductor Project Package Repository [R Package BiocManager Version 1.30.16]. Available at: https://CRAN.R-project.org/package=BiocManager (Accessed: December 6, 2021).

Narayanasamy, S., Jarosz, Y., Muller, E. E., Heintz-Buschart, A., Herold, M., Kaysen, A., et al. (2016). IMP: a Pipeline for Reproducible Reference-independent Integrated Metagenomic and Metatranscriptomic Analyses. *Genome Biol.* 17 (1), 260. doi:10.1186/s13059-016-1116-8

Nicholson, J. K., Holmes, E., Kinross, J., Burcelin, R., Gibson, G., Jia, W., et al. (2012). Host-gut Microbiota Metabolic Interactions. *Science* 336 (6086), 1262–1267. doi:10.1126/science.1223813

Nissen, J. N., Johansen, J., Allesøe, R. L., Sønderby, C. K., Armenteros, J. J. A., Grønbech, C. H., et al. (2021). Improved Metagenome Binning and Assembly Using Deep Variational Autoencoders. *Nat. Biotechnol.* 39 (5), 555–560. doi:10.1038/s41587-020-00777-4

Nurk, S., Meleshko, D., Korobeynikov, A., and Pevzner, P. A. (2017). metaSPAdes: a New Versatile Metagenomic Assembler. *Genome Res.* 27 (5), 824–834. doi:10.1101/gr.213959.116

Oksanen, J., Blanchet, F. G., Friendly, M., Kindt, R., Legendre, P., McGlinn, D., et al. (2020). vegan: Community Ecology Package [R Package vegan Version 2.5-7]. Available at: https://CRAN.R-project.org/package=vegan (Accessed: December 6, 2021).

Overholt, W. A., Hölzer, M., Geesink, P., Diezel, C., Marz, M., and Küsel, K. (2020). Inclusion of Oxford Nanopore Long Reads Improves All Microbial and Viral Metagenome-Assembled Genomes from a Complex Aquifer System. *Environ. Microbiol.* 22 (9), 4000–4013. doi:10.1111/1462-2920.15186

Parks, D. H., Imelfort, M., Skennerton, C. T., Hugenholtz, P., and Tyson, G. W. (2015). CheckM: Assessing the Quality of Microbial Genomes Recovered from Isolates, Single Cells, and Metagenomes. *Genome Res.* 25 (7), 1043–1055. doi:10.1101/gr.186072.114

Pasolli, E., Asnicar, F., Manara, S., Zolfo, M., Karcher, N., Armanini, F., et al. (2019). Extensive Unexplored Human Microbiome Diversity Revealed by Over 150,000 Genomes from Metagenomes Spanning Age, Geography, and Lifestyle. *Cell* 176 (3), 649–e20. doi:10.1016/j.cell.2019.01.001

Qin, J., Li, R., Raes, J., Arumugam, M., Burgdorf, K. S., Manichanh, C., et al. (2010). A Human Gut Microbial Gene Catalogue Established by Metagenomic Sequencing. *Nature* 464 (7285), 59–65. doi:10.1038/nature08821

Quince, C., Walker, A. W., Simpson, J. T., Loman, N. J., and Segata, N. (2017). Shotgun Metagenomics, from Sampling to Analysis. *Nat. Biotechnol.* 35, 833–844. doi:10.1038/nbt.3935

Quinlan, A. R., and Hall, I. M. (2010). BEDTools: a Flexible Suite of Utilities for Comparing Genomic Features. *Bioinformatics* 26 (6), 841–842. doi:10.1093/bioinformatics/btq033

Saheb Kashaf, S., Proctor, D. M., Deming, C., Saary, P., and Hölzer, M. (2022). Integrating Cultivation and Metagenomics for a Multi-Kingdom View of Skin Microbiome Diversity and Functions. *Nat. Microbiol.* 7 (1), 169–179. doi:10.1038/s41564-021-01011-w

Salazar, G., Paoli, L., Alberti, A., Huerta-Cepas, J., Ruscheweyh, H. J., Cuenca, M., et al. (2019). Gene Expression Changes and Community Turnover Differentially Shape the Global Ocean Metatranscriptome. *Cell* 179 (5), 1068–e21. doi:10.1016/j.cell.2019.10.014

Satinsky, B. M., Crump, B. C., Smith, C. B., Sharma, S., Zielinski, B. L., Doherty, M., et al. (2014). Microspatial Gene Expression Patterns in the Amazon River Plume. *Proc. Natl. Acad. Sci. U S A.* 111 (30), 11085–11090. doi:10.1073/pnas.1402782111

Seemann, T. (2014). Prokka: Rapid Prokaryotic Genome Annotation. *Bioinformatics* 30 (14), 2068–2069. doi:10.1093/bioinformatics/btu153

Sequeira, J. C., Rocha, M., Madalena Alves, M., and Salvador, A. F. (2019). “MOSCA: An Automated Pipeline for Integrated Metagenomics and Metatranscriptomics Data Analysis,” in Practical Applications of Computational Biology and Bioinformatics, 12th International Conference, 183–191. doi:10.1007/978-3-319-98702-6_22

Slowikowski, K. (2021). Automatically Position Non-Overlapping Text Labels with “ggplot2” [R Package ggrepel Version 0.9.1]. Available at: https://CRAN.R-project.org/package=ggrepel (Accessed: December 6, 2021).

Stewart, R. D., Auffret, M. D., Warr, A., Walker, A. W., Roehe, R., and Watson, M. (2019). Compendium of 4,941 Rumen Metagenome-Assembled Genomes for Rumen Microbiome Biology and Enzyme Discovery. *Nat. Biotechnol.* 37 (8), 953–961. doi:10.1038/s41587-019-0202-3

Sunagawa, S., Mende, D. R., Zeller, G., Izquierdo-Carrasco, F., Berger, S. A., Kultima, J. R., et al. (2013). Metagenomic Species Profiling Using Universal Phylogenetic Marker Genes. *Nat. Methods* 10 (12), 1196–1199. doi:10.1038/nmeth.2693

Tamames, J., and Puente-Sánchez, F. (2018). SqueezeMeta, A Highly Portable, Fully Automatic Metagenomic Analysis Pipeline. *Front. Microbiol.* 9, 3349. doi:10.3389/fmicb.2018.03349

Tenenbaum, D.Bioconductor Package Maintainer (2017). KEGGREST: Client-Side REST Access to the Kyoto Encyclopedia of Genes and Genomes (KEGG). R package version 1.30.1. doi:10.18129/B9.bioc.KEGGREST

The R Project for Statistical Computing (2021). The R Project for Statistical Computing. Available at: https://www.R-project.org/(Accessed: December 6, 2021).

Tian, L., Wang, X. W., Wu, A. K., Fan, Y., Friedman, J., Dahlin, A., et al. (2020). Deciphering Functional Redundancy in the Human Microbiome. *Nat. Commun.* 11 (1), 6217. doi:10.1038/s41467-020-19940-1

Tláskal, V. (2021). Metagenomes, Metatranscriptomes and Microbiomes of Naturally Decomposing deadwood. *Scientific data* 8 (1), 198. doi:10.6084/m9.figshare.14821752

Van Damme, R., Hölzer, M., Viehweger, A., Müller, B., Bongcam-Rudloff, E., and Brandt, C. (2021). Metagenomics Workflow for Hybrid Assembly, Differential Coverage Binning, Metatranscriptomics and Pathway Analysis (MUFFIN). *Plos Comput. Biol.* 17 (2), e1008716. doi:10.1371/journal.pcbi.1008716

Van Rossum, G., and Drake, F. L. (2009). *Python 3 Reference Manual: (Python Documentation Manual Part 2)*. Scotts Valley, CA: CreateSpace.

Vasimuddin, M., Misra, S., Li, H., and Aluru, S. (2019). “Efficient Architecture-Aware Acceleration of BWA-MEM for Multicore Systems,” in 2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS). [Preprint]. doi:10.1109/ipdps.2019.00041

Wagner, G. P., Kin, K., and Lynch, V. J. (2012). Measurement of mRNA Abundance Using RNA-Seq Data: RPKM Measure Is Inconsistent Among Samples. *Theor. Biosci* 131 (4), 281–285. doi:10.1007/s12064-012-0162-3

Wall, L., Christiansen, T., and Orwant, J. (2000). *Programming Perl*. Sebastopol, CA: O’Reilly Media.

Wang, Y., Hu, Y., Liu, F., Cao, J., Lv, N., Zhu, B., et al. (2020). Integrated Metagenomic and Metatranscriptomic Profiling Reveals Differentially Expressed Resistomes in Human, Chicken, and Pig Gut Microbiomes. *Environ. Int.* 138, 105649. doi:10.1016/j.envint.2020.105649

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., et al. (2019). Welcome to the Tidyverse. *Joss* 4 (43), 1686. doi:10.21105/joss.01686

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York: Springer. Available at: https://ggplot2.tidyverse.org.

Wickham, H. (2007). Reshaping Data with thereshapePackage. *J. Stat. Soft.* 21 (12), 1. doi:10.18637/jss.v021.i12

Wickham, H., François, R., Henry, L., and Müller, K. (2021). dplyr: A Grammar of Data Manipulation [R Package dplyr Version 1.0.7]. Available at: https://CRAN.R-project.org/package=dplyr (Accessed: December 6, 2021).

Wickham, H., and Girlich, M. (2021). tidyr: Tidy Messy Data [R Package tidyr Version 1.1.4]. Available at: https://CRAN.R-project.org/package=tidyr (Accessed: December 6, 2021).

Wickham, H., and Miller, E. (2021). haven: Import and Export ‘SPSS’, ‘Stata’ and ‘SAS’ Files [R Package haven Version 2.4.3]. Available at: https://CRAN.R-project.org/package=haven (Accessed: December 6, 2021).

Yin, Y., Mao, X., Yang, J., Chen, X., Mao, F., and Xu, Y. (2012). dbCAN: a Web Resource for Automated Carbohydrate-Active Enzyme Annotation. *Nucleic Acids Res.* 40, W445–W451. doi:10.1093/nar/gks479

Youngblut, N. D., de la Cuesta-Zuluaga, J., Reischer, G. H., Dauser, S., Schuster, N., Walzer, C., et al. (2020). Large-Scale Metagenome Assembly Reveals Novel Animal-Associated Microbial Genomes, Biosynthetic Gene Clusters, and Other Genetic Diversity. *mSystems* 5 (6), e01045-20. doi:10.1128/mSystems.01045-20

Zhang, H., Yohe, T., Huang, L., Entwistle, S., Wu, P., Yang, Z., et al. (2018). dbCAN2: a Meta Server for Automated Carbohydrate-Active Enzyme Annotation. *Nucleic Acids Res.* 46 (W1), W95–W101. doi:10.1093/nar/gky418

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Saenz, Nigro, Gunalan and Arumugam. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 12 May 2022
doi: 10.3389/fbinf.2022.864382

DivCom: A Tool for Systematic Partition of Groups of Microbial Profiles Into Intrinsic Subclusters and Distance-Based Subgroup Comparisons

Evangelia Intze [1] *and Ilias Lagkouvardos* [2,3]*

[1]School of Science and Technology, Hellenic Open University, Patras, Greece, [2]Core Facility Microbiome, ZIEL – Institute for Food and Health, Technical University Munich, Freising, Germany, [3]Institute of Marine Biology, Biotechnology and Aquaculture, Hellenic Centre for Marine Research, Heraklion, Greece

When analyzing microbiome data, one of the main objectives is to effectively compare the microbial profiles of samples belonging to different groups. Beta diversity measures the level of similarity among samples, usually in the form of dissimilarity matrices. The use of suitable statistical tests in conjunction with those matrices typically provides us with all the necessary information to evaluate the overall similarity of groups of microbial communities. However, in some cases, this approach can lead us to deceptive conclusions, mainly due to the uneven dispersions of the groups and the existence of unique or unexpected substructures in the dataset. To address these issues, we developed divide and compare (DivCom), an automated tool for advanced beta diversity analysis. DivCom reveals the inner structure of groups by dividing their samples into the appropriate number of clusters and then compares the distances of every profile to the centers of these clusters. This information can be used for determining the existing interrelation of the groups. The proposed methodology and the developed tool were assessed by comparing the response of anemic patients with or without inflammatory bowel disease to different iron replacement therapies. DivCom generated results that revealed the inner structure of the dataset, evaluated the relationship among the clusters, and assessed the effect of the treatments. The DivCom tool is freely available at: https://github.com/Lagkouvardos/DivCom.

Keywords: microbial profiles, beta diversity, *de novo* clustering, reference distance, PAM

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Alexander Hübner,
Max Planck Institute for Evolutionary Anthropology, Germany
Andrew Maltez Thomas,
University of Trento, Italy

****Correspondence:***
Ilias Lagkouvardos
ilias.lagkouvardos@tum.de

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *28 January 2022*
Accepted: *04 April 2022*
Published: *12 May 2022*

Citation:
Intze E and Lagkouvardos I (2022) DivCom: A Tool for Systematic Partition of Groups of Microbial Profiles Into Intrinsic Subclusters and Distance-Based Subgroup Comparisons. Front. Bioinform. 2:864382. doi: 10.3389/fbinf.2022.864382

1 INTRODUCTION

Over the last 20 years, the field of microbiome research has been experiencing exponential growth, mainly powered by advances in sequencing technology. A significant amount of this body of research has been focused on how dysbiotic microbial communities are linked with pathological conditions (Coker et al., 2018; Harbison et al., 2019; Harbison et al., 2019; Hufnagl et al., 2020). In addition, the importance of microbes has been recognized in other fields spanning from agricultural and biotechnological applications to ecological and environmental interventions (Lian et al., 2018; Qiu et al., 2019).

Nowadays, several pipelines, tools, and platforms are dedicated to analyzing microbiome datasets. Specialized tools include R-based packages such as vegan (Oksanen et al., 2015), phyloseq (McMurdie and Holmes, 2013), and SIAMCAT (Wirbel et al., 2021). Pipelines, like QIIME2 (Bolyen et al., 2019), mothur (Schloss et al., 2009), and Rhea (Lagkouvardos et al., 2017), usually offer streamlined analytical functionalities with minimal programming requirements. However, more tools and methodologies are under development to reflect the growth in our understanding of the topic and accommodate our needs for more specialized analytics.

Beta diversity, the measure of diversity between two samples, is one of the most widely used concepts in microbiome data analysis (Lin et al., 2015; Wagner et al., 2018). Beta diversity does not focus on the abundance of specific bacterial taxa but takes into account the overall microbial community structure. The usage of an appropriate metric function results in a single measurement (distance) of similarity or dissimilarity that can be used to examine the relations among the samples in a study. Metrics like Bray-Curtis (Bray and Curtis, 1957), weighted or unweighted Unifrac (Lozupone et al., 2011), and Jaccard distance (Jaccard, 1912) are commonly used for exploratory and ordination analyses. In a limited number of studies, the quantification of beta diversity measures has been utilized to gain better insights into the community dynamics (Halfvarson et al., 2017; Suzuki et al., 2020).

Clustering a group without using labels or prior knowledge of the data is defined as unsupervised clustering. Unsupervised clustering does not use any external information and relies only on the pairwise distances of the samples. Since this type of clustering shares similar principles with the *de novo* OTU picking (Navas-Molina et al., 2013), here in this study, we will borrow this term, and we will call the process of the unsupervised clustering as "*de novo* clustering" of the microbial profiles. This procedure can be extremely helpful for revealing substructures of a dataset that are unknown or have not been predicted during the study design (Ramette, 2007). The proposed concept of the enterotypes (Arumugam et al., 2011) is one of the most known cases where *de novo* clustering revealed intrinsic substructures in the human gut microbiota. Also, *de novo* clustering contributed significantly to drawing conclusions in the studies of Paetzold et al. (2019) and later García-Mantrana et al. (2020), which investigated skin and maternal microbiomes, respectively. Although both beta diversity and *de novo* clustering techniques are commonly used by individual researchers, no standardized procedure, pipeline, or tool integrates and automates their combined use for group comparisons.

Comparing the microbial profile of two or more groups against each other or exploring the relationship between control and intervention groups is part of a typical workflow for many studies (Morris et al., 2013; Prast-Nielsen et al., 2019; Ventura et al. 2019). Through this process, the dissimilarity between the members of each group can be used to determine the level of differentiation among the examined groups.

The problem that arises is that the approaches used to analyze the microbial datasets can lead us, in some cases, to wrong assumptions or incomplete conclusions. Among others, there are three main obstacles in the currently applied methodologies: the first is derived by the dimensionality reduction process (Calle, 2019), the second by the statistical tests (Xia and Sun, 2017), and the third by not taking into consideration the unique substructure of the data (Gupta et al., 2017; He et al., 2018). Because of the dimensionality reduction process and the selected distance metric, there is a high chance of producing a distorted image of the data (Calle, 2019; Hawinkel et al., 2019). Relying only on the visual representation of the ordination plots can lead us to misleading conclusions about the existing relationship among the profiles of the different groups. The suggested practice for evaluating groups' dissimilarities is through the application of a multivariable statistical test (Knight et al., 2018) like PERMANOVA (permutational multivariate analysis of variance) (Anderson, 2001), PERMDISP (permutational analysis of multivariate dispersions) (Anderson, 2006), or ANOSIM (analysis of similarities) (Clarke, 1993). However, even this practice can also produce inaccurate or deceptive outcomes mainly due to the lack of homogeneity between the groups, the different levels of their dispersion (Anderson et al., 2008; Warton et al., 2012), and the wrong use and interpretation of the results of the statistical tests.

To illustrate these issues, we present two hypothetical cases in which wrong conclusions can easily be drawn if we follow the widely applied microbiome data analysis practices. In the first case, we simulated two groups that have the same center and a similar number of samples but significantly different dispersions (**Figure 1A**). The visual inspection of the plot suggests structural differences among these two groups; however, the PERMANOVA ($p = 0.838$) and ANOSIM ($p = 0.556$) tests affected by the same centers and the different dispersions (PERMDISP: $p < 0.001$) returned a high probability that both groups originate from the same distribution. Although PERMANOVA and ANOSIM tests are fairly robust methods, they have their own limitations and are sensitive to different dispersions. Relying only on these results can obscure the information about the substantially different structures of the dataset.

In the next case, according to PERMDISP, the two groups have similar dispersions ($p = 0.35$) but appear to have different centers (**Figure 1C**); this observation is also supported by the results of the PERMANOVA ($p < 0.001$) and ANOSIM tests ($p < 0.001$). These facts could lead the researcher to conclude that samples originating from group "Test" have significantly different profiles than those of group "Reference." The PERMANOVA test assumes that there is only one distribution from which every group is sampling. However, *de novo* clustering of both groups reveals that each of them consists of two well-defined subgroups (**Figure 1D**). The use of statistical tests like PERMANOVA on groups composed of multiple subgroups can lead to misleading conclusions. In our instance, the two pairs of subclusters have similar centers and dispersions but different sampling sizes and also uneven number of samples belonging to each subgroup. This unequal representation of the two otherwise related compositions, from which both groups were composed, was the reason for the distorted and misinformative initial image. Thus, revealing the internal structure of the groups could provide

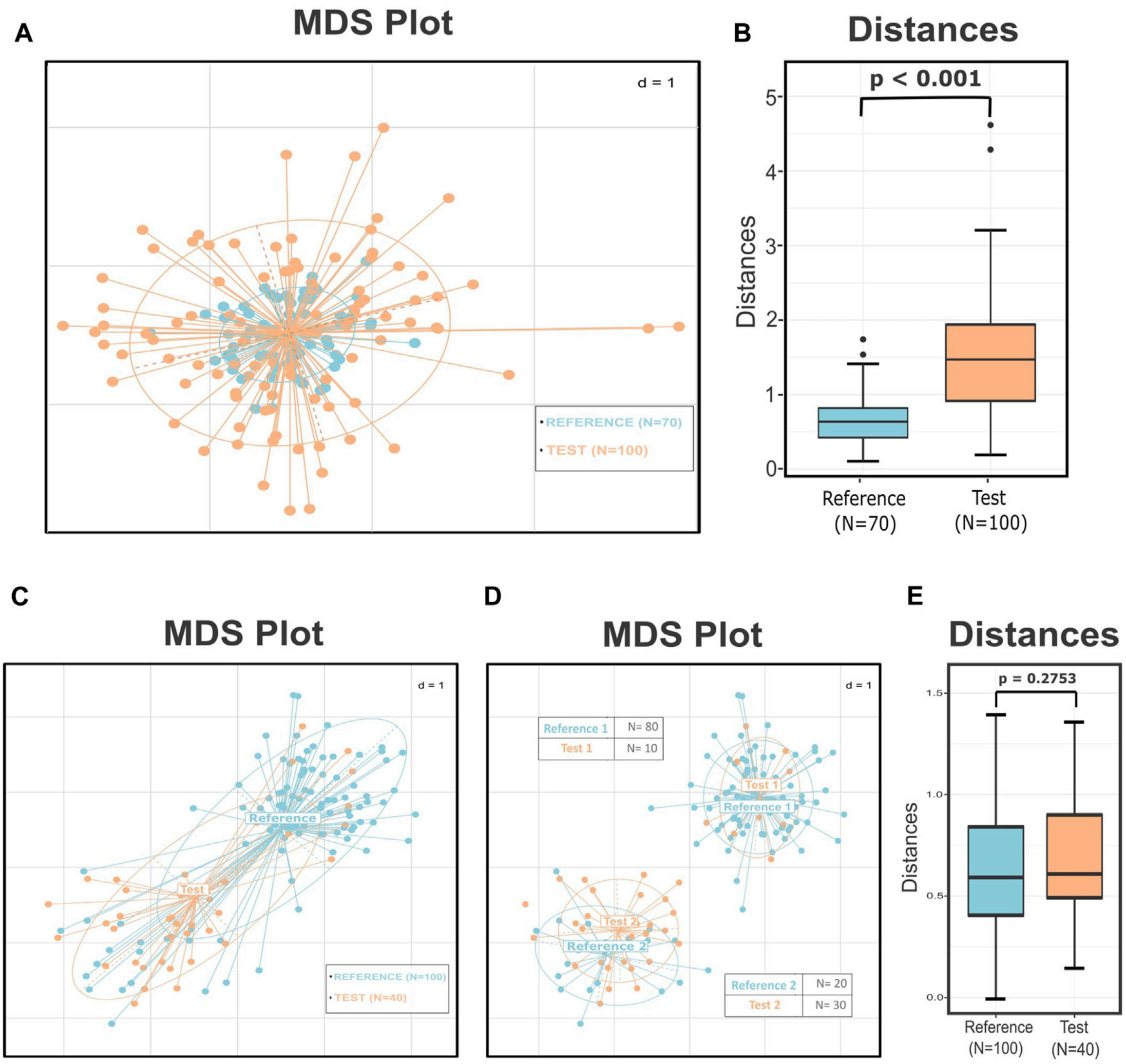


FIGURE 1 | Simulated data demonstrating how different dispersion among groups influences the results of the statistical tests and how the substructures within the groups and the uneven sampling of these subclusters can lead to a misleading interpretation of the data. **(A)** MDS plot presenting two groups with the same center but according to the PERMDISP test significantly different variances ($p < 0.001$). The p-value of the PERMANOVA and ANOSIM tests for these groups is 0.838 and 0.556, respectively, leading to not rejecting the null hypothesis that the two groups are drawn from the same distribution. **(B)** Boxplots presenting the distances of the test points from the center of the reference group. The p-value of the Wilcoxon rank sum test is less than 0.001, implying correctly that these two groups are significantly different. **(C)** MDS plot that presents two groups with similar dispersions ($p = 0.35$) but seemingly different centers. The results of the PERMANOVA ($p < 0.001$) and ANOSIM ($p < 0.001$) tests indicate that these two groups are well-separated. **(D)** MDS plot illustrating the subgroups derived by the *de novo* clustering of the two groups. The dataset now consists of two pairs of highly related clusters with different representations in the two subgroups. **(E)** Boxplots presenting the distribution of the distances of every test sample from their closest reference center. According to the p-value of the Wilcoxon rank sum test ($p = 0.2753$), the points of the two groups do not differ significantly in their values. The perceived difference comes from the unequal representation of the two subgroups in the final dataset.

us with additional information about the dataset and assist us in preventing errors like those mentioned earlier.

In these two cases, we summarized some of the existing problems in the microbiome beta diversity analysis that sometimes are difficult to be detected and overpassed. The problem presented in the first case is the improper use of the statistical tests or the wrong interpretation of their results. In the first instance, the PERMDISP test provided a clear view that the groups have significantly different dispersions; this should have been a hindrance to applying multivariate tests like PERANOVA or ANOSIM as their results could have been inaccurate. However, in many cases, the power of these tests is overestimated by the researchers, leading them to wrong conclusions. In the second instance, even though we did not have any misuse of the statistical tests, the inner structure of the groups was a key factor in having incorrect results. Unfortunately, in these cases, there is not an easy and reliable alternative for the user to follow: either the researcher will have to rely on the results of the statistical tests with the fear of drawing the wrong conclusions or should alter the exploratory approach of the study.

Herein, we introduce DivCom, a new approach that can be used as an answer to the challenges mentioned earlier. This approach aims to compare different groups in a more efficient and detailed way, and reveal their relations. The central notion behind DivCom is that groups of microbial profiles should not be treated as entireties because valuable information about their unique structural characteristics could be lost. DivCom employs the idea of dividing the groups using *de novo* clustering and then comparing these clusters using beta diversity measures as metrics. According to the methodology of DivCom, the samples of the control group are clustered, and then, the most representative point (centroid/medoid) for each of these clusters is selected. Consequently, all the distances of the remaining test samples from these preselected points are calculated and then assessed.

Applying the DivCom methodology to the previously mentioned simulated cases of **Figure 1**, we can infer that in the first instance, the distinct structure of the dataset was revealed by comparing the distances of the samples of the two groups from the center ($p < 0.001$) (**Figure 1B**). Also, in the second case, the distance of the samples from their closest reference center showed that there is no significant difference between the two pairs of groups (**Figure 1E**). Therefore, even though the commonly used techniques failed to uncover the true relationship of the data, DivCom achieved this by using a distance- and structure-based approach. Also, the use of the centroids reduced the required calculations and comparisons, and produced results that are analogous with those we would have obtained if we had compared all test samples with all the samples of the closest reference cluster (**Supplementary Figure S1**).

The effectiveness of the method was also evaluated using publicly available gut microbiome data from the study of Lee et al. (2017). The selected research aimed to compare the effect of iron replacement in anemic patients who suffered from inflammatory bowel disease against a group of non-inflamed anemic individuals. All the subjects were randomly separated into two groups, and they followed two different treatments of iron replacement for 3 months. Simply by applying the DivCom approach, we were able to reproduce some of the main findings of the original analysis and also reveal some additional aspects of the data that were unnoticed in the original work. DivCom provides us with a better insight into the data and can be used complementary to the currently applied data analysis pipelines. The proposed methodology is implemented as an automated, open-source, user-friendly, and easily-editable R-based program. The DivCom tool has minimal input requirements, produces several detailed outputs, and is available at: https://github.com/Lagkouvardos/DivCom.

2 MATERIALS AND METHODS

2.1 Overview

DivCom has been implemented in R programming language under version 4.1.2. The tool relies on the functions provided by R packages: ade4, ape, caTools, cluster, cowplot, data.table, dplyr, factoextra, fpc, ggplot2, ggpubr, ggtree, graphics, grid, gridExtra, gtable, GUniFrac, mclust, phangorn, RColorBrewer, stats, tidyr, tools, and vegan. Many of these packages have their own dependencies. In the detailed description of the scripts, some of the key functions are provided, along with the package to which they belong. Also, selected sections of the Rhea pipeline (Lagkouvardos et al., 2017) were modified accordingly and incorporated in the DivCom scripts. The tool consists of two scripts, named "Beta-Diversity.R" and "DivCom.R." The former is an ancillary script, while the latter is the main script of the tool (**Figure 2A**).

DivCom is a purely distance-based tool that compares different groups by taking into consideration the phylogenetic distances between observed organisms, and using statistical measures to evaluate the results. Therefore, the Partitioning Around Medoids (PAM) algorithm (Kaufman and Rousseeuw, 2009) is applied to cluster the samples (cluster::pam), and Generalized Unifrac (Chen et al., 2012) is the default distance metric used by the program (GUniFrac::GUniFrac). The statistical hypothesis testing relies on the Wilcoxon rank sum test (Mann and Whitney, 1947; Wilcoxon, 1992) for the continuous variables (stats::wilcox.test), the chi-square test for the categorical variables (stats::chisq.test), permutational analysis of multivariate dispersions (PERMDISP) (Anderson, 2006) for the dispersion similarity comparison of the groups (vegan:: betadisper and permutest), and permutational multivariate analysis of variance (PERMANOVA) (Anderson, 2001) for the similarity comparison of the groups (vegan::adonis). All the p-values are adjusted using the Benjamini–Hochberg method (Benjamini and Hochberg, 1995) (stats::p.adjust). The multidimensional scaling (MDS) algorithm (Gower, 1966) is applied for the ordination analysis (stats::cmdscale), and finally, scatterplots, boxplots, barplots, and phylograms are used to visualize the findings (ade4::s.class, ggtree, ggtree, ggplot2).

2.2 Inputs

The input requirements are minimal as the user has to provide only three mandatory files.

- The first file is an OTU or ASV abundance table which can be either normalized or not. In this table, the rows should represent the OTUs or ASVs, and the columns should represent the samples. In case the table is not normalized, then the first step will be the normalization of the table so the sum of the counts will be equal across all the samples.
- Considering that the generalized Unifrac distance is used as the default distance metric, the second necessary input file is a phylogenetic tree that corresponds to the OTUs or ASVs of the abundance table. If a tree is not available, the user can instead provide a dissimilarity matrix of the samples.
- The final requirement is a mapping file that contains the labels of the samples. The information of the mapping file is necessary for the labeling and assigning of the reference and test groups.

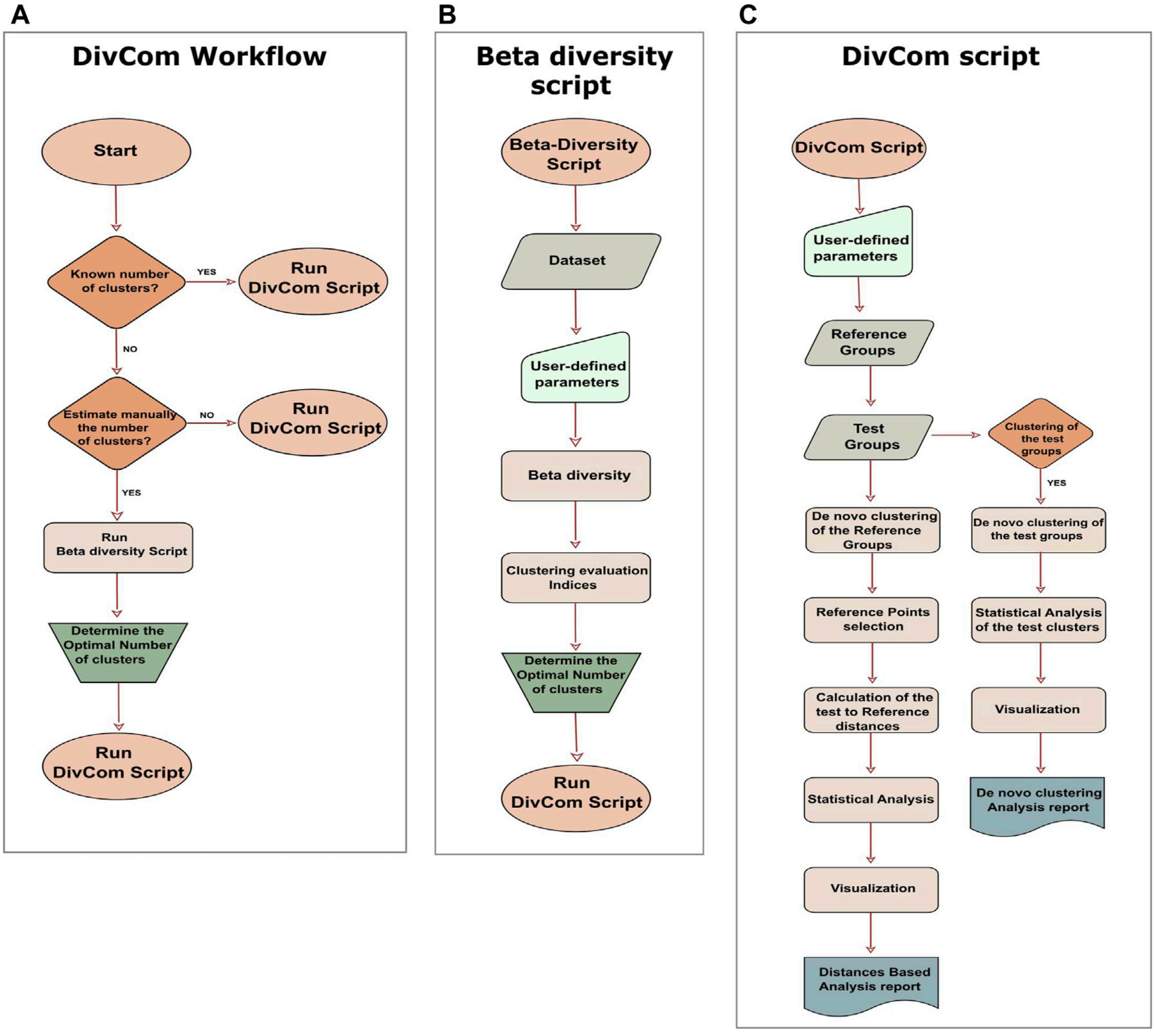

A
DivCom Workflow
Start
Known number of clusters?
YES
Run DivCom Script
NO
Estimate manually the number of clusters?
NO
Run DivCom Script
YES
Run Beta diversity Script
Determine the Optimal Number of clusters
Run DivCom Script
B
Beta diversity script
Beta-Diversity Script
Dataset
User-defined parameters
Beta diversity
Clustering evaluation Indices
Determine the Optimal Number of clusters
Run DivCom Script
C
DivCom script
DivCom Script
User-defined parameters
Reference Groups
Test Groups
Clustering of the test groups
YES
De novo clustering of the Reference Groups
De novo clustering of the test groups
Reference Points selection
Statistical Analysis of the test clusters
Calculation of the test to Reference distances
Visualization
Statistical Analysis
De novo clustering Analysis report
Visualization
Distances Based Analysis report

FIGURE 2 | Workflow of the DivCom tool, and the two scripts of the program. **(A)** According to the workflow of the DivCom, the user can execute the beta-diversity to calculate the optimal number of clusters or to directly run the DivCom script. **(B)** The script "Beta-Diversity.R" calculates and visualizes beta diversity between the samples and produces the plots of four different clustering evaluation indices (Calinski-Harabasz, silhouette, prediction strength, and Within Sum of Squares). These outputs provide the user with the necessary information in order to determine the optimal number of clusters for each group. **(C)** The main script is called "DivCom.R" and performs *de novo* clustering to both the reference and test groups, calculates the pairwise distances of the reference and test samples, and finally conducts an automated statistical analysis and produces the final reports. This information contributes to a better understanding of the interrelation between the different groups under study.

In addition to the files mentioned earlier, the user has to fill out some additional parameters. The desired number of clusters for each group, the name of the reference and test groups, and the type of the produced plots are among these additional requirements. A detailed description for each of these parameters is provided in the scripts and the accompanying documents of the DivCom tool on its GitHub page.

Also, in the initiation phase, the user has to define the names of the input files and then determine which group or groups will serve as the reference dataset. The rest of the samples will be compared with this reference group.

2.3 Beta-Diversity Script

Moving on to the actual scripts of the program, the first is named "Beta-Diversity.R" (**Figure 2B**), and it is a slightly revised version of the "Beta-Diversity" script of the Rhea pipeline. Its purpose is to calculate Beta-Diversity for microbial communities but mainly to provide us with all the necessary information about each group's optimal number of clusters. The script produces the plots of the Calinski-Harabasz (Caliński and Harabasz, 1974) and the silhouette (Rousseeuw, 1987) index. (fpc::cluster.stats), the Within Sum of Squares (WSS) (factoextra::fviz_nbclust), and the prediction strength (Tibshirani and Walther, 2005) (fpc:: prediction. strength) plots and also the plot of the BIC values for six models as they are produced by the model-based clustering based on finite Gaussian mixtures (mclust::Mclust). The purpose of the last plot is to inform us if the dataset consists of a homogenous and uniform distribution so that no substructures exist. If this is true, then the program will propose just one cluster. These measures were selected as they are among the most widely used techniques for clustering validation (Baarsch and Celebi, 2012; Bouveyron and Brunet-Saumard, 2014).

2.3.1 Optimal Number of Clusters

The appraisal of these graphs in conjunction with the prior knowledge of the dataset can help the user decide about the optimal number of clusters for each group. Although we recommend that users make this decision based on their preferences and understandings, among the default outputs of the script a report is included with a recommendation about the optimal number of clusters for each group. To make this suggestion, the script first calculates the optimal number of clusters for each index and then selects the number with the highest frequency. In case of a tie, this suggestion is based on the results of the Calinski-Harabasz index. Even though all indices have their own strengths and weaknesses, we chose to highlight the role of the Calinski-Harabasz index because it is a variance-dependent index that produces higher values when the clusters are compact and well-separated; these characteristics are necessary and highly desirable in our approach. Alternatively, if the user does not wish to evaluate the optimal number of clusters manually, they can omit the Beta-Diversity script and use the integrated option in the main script for automatic calculation of the optimal number of clusters for each group based on the values of the Calinski-Harabasz index. Depending on their preferences, the users can manually evaluate the optimal number of clusters, follow our recommendation, or choose to be automatically calculated by the program (**Figure 2A**).

2.4 DivCom Script

After determining the optimal number of clusters for each group, the user has to run the main script of the tool, which is named “DivCom.R.” DivCom script consists of two main sections (**Figure 2C**): the first is called “Distances-Based analysis” and the second “*De novo* clustering analysis.” The main difference between them is that in the first part of the analysis, *de novo* clustering is applied only to the reference dataset, while in the second and optional stage, all the groups are clustered, and then compared and analyzed against each other.

2.4.1 Distances-Based Analysis

Proceeding to the actual procedures of the tool, in order to achieve a better insight into the data and take into consideration the unique substructure of the groups, the script performs *de novo* clustering to the samples of the reference group. The PAM algorithm performs this task using the desired number of clusters and the produced distances matrix as inputs. Through this process, the most representative points of the reference group are determined and stored for further use. The medoids of the clusters can be used as the representative points; this is the default and recommended option. Also, the mean or median counts of the OTUs or ASVs can be used as an alternative option to the medoids.

Following the clustering process, the program calculates the distances of the remaining samples to these representative points. Then, each sample is assigned to the closest and probably more relevant to it, in terms of their microbial composition, reference cluster. This procedure results in an indirect clustering of the test samples based only on the distances from the most representative points of the reference group.

Next, a fully automated statistical analysis is conducted. MDS plots visualize the relationship of the reference clusters with their closest test samples. Boxplots present the distances of the samples under study from the nearest reference cluster. Also, tables containing the *p*-values and various statistical measures are printed. Finally, a part of the process is dedicated to analyzing the distribution of the test samples across the clusters of the reference group. This part can assist the user in discovering similar patterns between the reference and test groups.

2.4.2 *De Novo* Clustering Analysis

The second part of the analysis is complementary to the previous section. The main difference is that *de novo* clustering is applied not only to the reference but also to each of the test groups. The user has to specify the desired number of clusters for each test group in the initiation phase. If this information is not provided correctly, then this part of the analysis is omitted.

Assuming that the aforementioned information has been provided, every test group is clustered using the PAM algorithm. Subsequently, every subcluster is compared with the representative points of the reference group. This process results in outputs that compare the structures of the reference and test groups. Therefore, it is easier for the user to reveal the substructural similarities and existing relations between the groups.

Once again, an entirely automated statistical analysis is performed following this procedure. Various descriptive statistics measures for the clusters of the reference and test groups are produced. MDS and boxplots which visualize the relation between the subclusters, and the tables containing *p*-values, distances, and statistical measures are printed. Similar to the previous stage, the distributions of the test samples across the clusters are analyzed and assessed.

Considering that these two sections of the program produce analogous results, the user can compare their outputs and uncover aspects of the dataset that would be difficult to be discovered in any other way.

2.5 Outputs

The program produces two detailed reports in the PDF format, one for each of the two steps described earlier. The first file is named “Distances-based report,” and its goal is to present the information about the discrepancy between the reference and the test groups. This report visualizes and statistically investigates the relation of the reference clusters to their closest test samples. The second output is a PDF file called “*De novo* clustering report,” and it aims to present the relationship between the reference and test subclusters. Since *de novo* clustering has been applied to both the control and test groups, this file focuses more on the relationship and the distance-based similarities of the reference clusters with their closest test subclusters.

TABLE 1 | Number of samples of each category that were used in the DivCom analysis. In total, 19 samples of the NI group, 26 of the CD group, and 17 of the UC group were selected to participate in the study analysis.

Iron intake	NI (non-inflamed reference group)	CD (Crohn's disease test group)	UC (ulcerative colitis test group)
PO	9	12	10
IV	10	14	7
Total	19	26	17

Each of these reports includes MDS plots and phylograms that illustrate the relationship between the samples. Boxplots present the distances from the selected representative points and tables containing various statistical measures derived from the analysis. In order for the results to be more interpretable by the user, a detailed description is included for each of these elements. Additionally, all the outputs are printed in the PNG or tab format in a separate folder.

2.6 Test Dataset

To demonstrate the performance of DivCom and allow users to test the functionality of the tool, a previously unpublished raw sequencing dataset from the iron replacement study of Lee et al. (2017) was used and is publicly available from now on.

This particular dataset was selected as the objective of the study was in line with the requirements of DivCom. This research aimed to assess the effects of Per Oral (PO) and intravenous (IV) iron replacement therapy (IRT) in patients with two types of inflammatory bowel disease (IBD) and a group of non-inflamed (NI) individuals with iron deficiency. The cohort consisted of Crohn's disease patients (CD, N = 31), ulcerative colitis patients (UC, N = 22), and non-inflamed individuals (NI, N = 19); in total, 62 subjects were involved in this study. The NI individuals were used as the control/reference group, while the CD and UC patients were used as the test groups. All the subjects were randomly separated into PO or IV groups, and they followed the corresponding therapy for 3 months. Therefore, the dataset consisted of two timepoints based on the sampling time; the first timepoint referred to the samples at the baseline (B) and the second to the samples after the 3-month treatment (3M).

The raw sequences were processed through the IMNGS platform (Lagkouvardos et al., 2016), implementing the UNOISE3 (Edgar, 2016) and UPARSE (Edgar, 2013) pipelines, using the default parameters. The number of samples of each category that fulfilled the quality assessment and eventually took part in the final analysis is summarized in **Table 1**.

3 RESULTS

As presented in the introduction, the DivCom approach surpassed the limitations and pitfalls of the currently applied methodology and revealed the true relationship between the groups (**Figures 1B and E**). Here, using the test dataset of the iron replacement study, we evaluated the performance of our tool in real and complex data, its ability to reproduce parts of the initial analysis, and its contribution to a better understanding of the dataset.

In the first step of the analysis, we evaluated the effect of the treatment on the non-inflamed (NI) control samples. As indicated by the Calinski-Harabasz index and verified by the suggestion of the Beta-Diversity script, the pretreatment samples of the NI group (NI.B) were partitioned into four clusters (**Supplementary Figure S2A**). The distances of all the after-treatment individuals (NI.3M) from these clusters were calculated and then evaluated. These distances indicated that there was no significant difference for the profiles of the non-inflamed (NI) anemic patients before (B) and after (3M) iron treatment ($p = 0.3908$) (**Figure 3A**). Therefore, since the IRT did not result in consistent changes in the overall microbial profile of the samples, all the NI individuals were merged and used as a unified reference group consisting of 38 profiles. The Calinski-Harabasz index and the recommendation of the Beta-Diversity script supported the existence of two clusters for the entirety of the reference group (**Supplementary Figure S2B**). Therefore, for the rest of the analysis, the control group of the NI was subdivided into two clusters. The type of treatment (IV, PO) and the sampling time (B, 3M) did not contribute to the formation of these two groups as the chi-square p-values were 0.217 for the first case and 0.602 for the second.

Continuing the analysis, we investigated whether the intervention shifted the IBD samples (UC and CD) closer to the NI reference points. The distances of the IBD groups (B and 3M) from the NI reference points were significantly higher than those of the reference group ($p < 0.001$), highlighting in this way the disturbed nature of the IBD profiles. Nevertheless, those distances were not significantly different among time points (IBD.B-IBD.3M) ($p = 0.96$), indicating that the treatment did not affect the median distances of the IBD sample from the NI reference samples (**Figure 3B**).

Next, we repeated the analysis using the sampling time (B or 3M) and the type of disease (CD or UC) as the independent variables. The boxplots of the distances from the closest reference medoid and the statistical testing indicated that the UC and CD groups at the baseline and after the iron replacement were once again significantly farther from the control group of the NI compared to the reference samples ($p < 0.05$) (**Figure 4A**). Regarding the distances, the CD patients seemed to have a more substantial level of dissimilarity with the NI group than the UC patients. Also, DivCom automatically assigned each IBD profile to its closest NI reference medoid. The integrated chi-square analysis

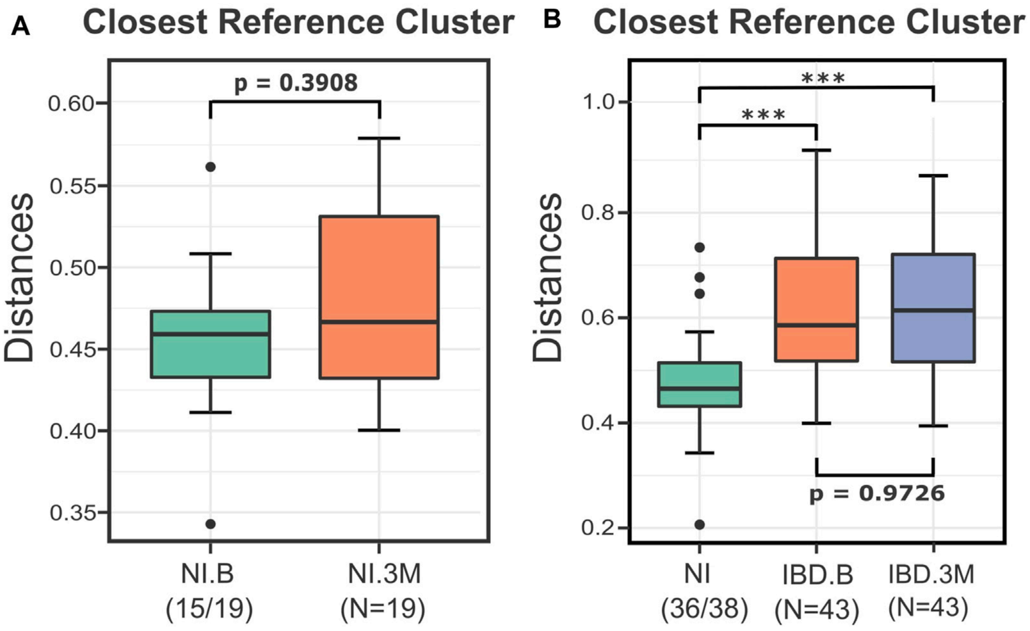


FIGURE 3 | Boxplots presenting the distances of the NI and IBD samples from their closest reference medoids before (B) and after (3M) the treatments. **(A)** The distances of the NI.3M samples from their closest medoid of the NI.B group implied that the iron replacement therapy (IRT) did not affect the microbial composition of NI samples considerably. The two groups were not significantly different (*p* = 0.3908), so for the rest of the DivCom analysis, the samples were merged and used as a unified reference group. **(B)** Boxplots displaying the impact of the iron replacement therapy on the IBD samples. The IRT did not shift the IBD samples closer to the NI group. The distances of the IBD groups (B and 3M) from the reference group of the NI samples are significantly different compared to those of the NI group (*p*<0.001). However, the two groups are highly related to each other (*p* = 0.97). *p*-values: *<0.05; ***<0.01.

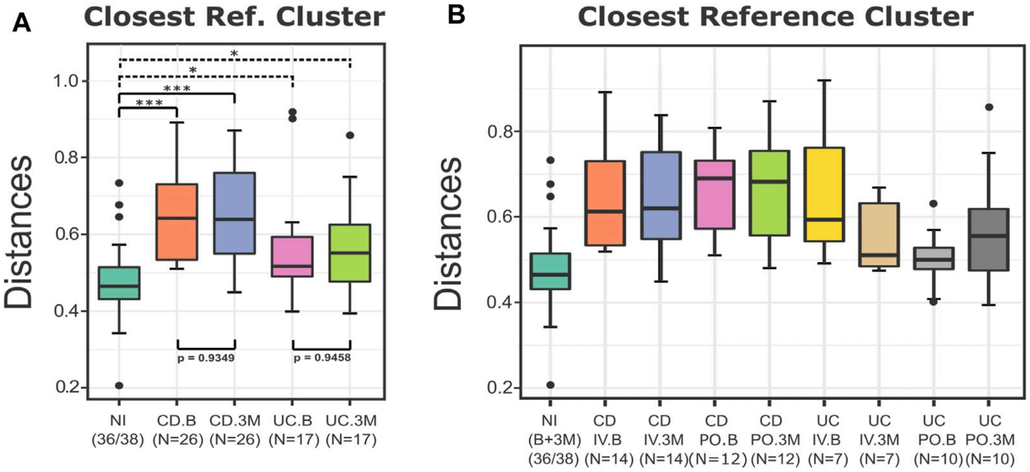


FIGURE 4 | Boxplots of the distances of the test groups from the closest reference medoid of the NI group. **(A)** For the UC and CD groups, the type of the disease and the sampling time did not affect their distances from the NI samples. All the groups were significantly farther from the control group compared to the reference samples (*p* < 0.05). The distances of the CD patients from the control group seem to present overall higher values than the distances of the UC patients from the NI group. **(B)** All the IBD samples were grouped based on the type of the disease, the treatment, and the sampling time. The boxplots of the distances from the closest medoid indicate that the CD groups have a higher level of homogeneity but are farther from the control group compared to the UC groups. On the contrary, most of the UC samples are closer to the NI group, but their distances from the reference group present a higher variance. In particular, the distances of the UC.PO.B group are related to the NI group (*p* = 0.34). *p*-values: *<0.05; ***<0.01.

revealed that the samples of the UC group before and after the intervention had similar distribution around the reference medoids to the samples of the NI group (*p* = 0.5786 at the baseline, *p* = 0.2602 at 3 months). On the other hand, this trend was not present for the CD patients (*p* = 0.0070 at the baseline, *p* = 0.0003 at 3 months).

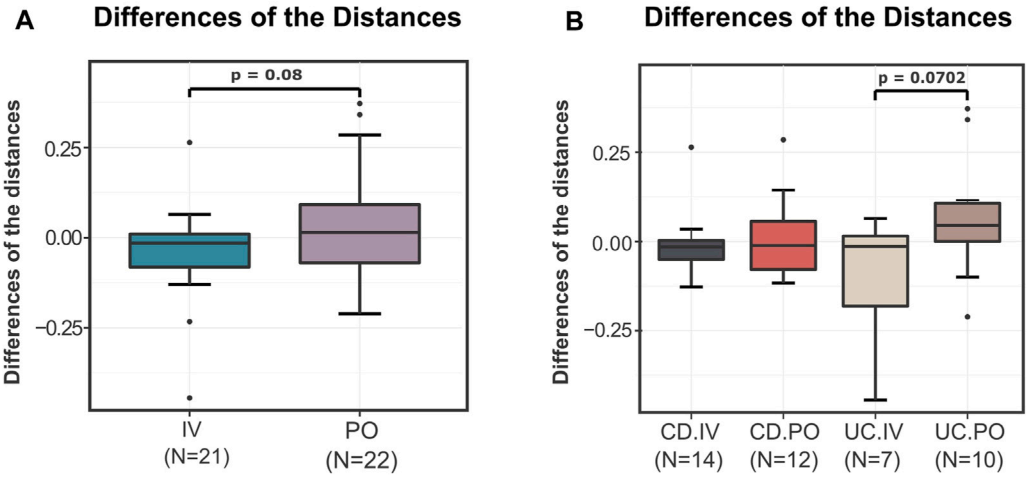


FIGURE 5 | Differences of the "before–after" distances of the test samples from the closest reference medoid of the NI group. **(A)** For the samples of the IV and PO treatments, the differences of the distances from the closest reference medoid before and after the IRT were calculated. The boxplots illustrate the distribution of these differences. The IV treatment seems to have slightly but not significantly better results than the PO treatment (p = 0.08). The average differences of the IV samples are negative (-0.0609), while the average differences of the PO treatment are positive (0.09715). **(B)** The IBD samples were compared based on the type of the disease and the treatment. In terms of distance to the reference samples, a differential treatment response is observed on UC patients (p = 0.0702). CD patients do not seem to be affected by the mode of treatment, with both resulting in a slight convergence to the reference profiles.

When the IBD samples were labeled based on the disease (CD and UC), the type of the treatment (IV and PO), and the sampling time (B and 3M), it was more evident that the CD groups appeared to have greater distances from the control group than the UC samples (**Figure 4B**). The IRT seemed to have a more pronounced effect on the UC groups as their samples exhibited a higher variance in terms of their distances from the reference group before and after the treatments. In the UC patients, the type of iron replacement showed trends of differential effect, with the IV group demonstrating a slight decrease and the PO group exhibiting a small increase in the overall distances from the NI. However, in both cases (UC and CD), the distances from the reference group did not change considerably, independent of the type of the iron replacement. On a side note, we revealed that at the baseline, the UC samples chosen to follow the PO treatment seemed to be considerably closer to the NI group than the remaining samples of the UC or CD patients.

Subsequently, taking advantage of the outputs of the DivCom, a secondary analysis was conducted. The intention was to determine whether the PO or IV treatment had a more profound impact on the distances of the test samples to the reference dataset. Thus, the differences between the distances from the closest medoid after the intervention and those at the baseline were calculated. The average differences of the distances for the IV treatment in both the CD and UC groups were negative (CD = −0.0053, UC = −0.1165). On the other hand, the corresponding differences for the PO treatment in the CD and UC patients were positive (CD = 0.0267, UC = 0.1676). Overall, the statistical comparison of those differences for the two types of treatment showed a trend (p = 0.08), indicating that PO treatment led to an increase of distance from the reference samples, while the IV treatment resulted in a decrease (**Figure 5A**). This difference was mainly due to the differential effect of the treatment type on the UC patients, with the CD patients remaining mostly unaffected (**Figure 5B**).

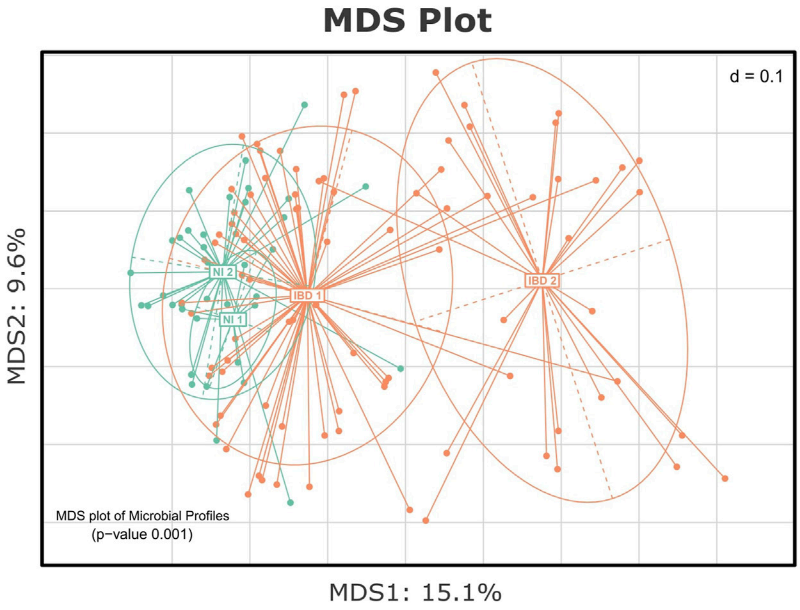
In order to reveal the test group's unique substructure, *de novo* clustering was applied to the IBD profiles. As suggested by the Calinski-Harabasz index (**Supplementary Figure S2C**) and the majority of the other indices as they were produced by the Beta-Diversity script, the IBD group was partitioned into two clusters (**Figure 6**). One cluster was closer to the NI group, and the other was considerably more distant from the reference samples. This finding was further evaluated through the statistical testing of the corresponding distances of each subcluster from the nearest reference medoid (**Supplementary Figure S3**). Both the IBD clusters were significantly farther from the NI groups ($p < 0.05$), confirming that the profiles of the patients diverged from those of the control group. Through the automated statistical testing of the DivCom, we verified that the iron therapy did not affect the CD patients. The distribution of the CD samples across the IBD clusters did not change significantly before and after the intervention (p = 0.2379). On the contrary, the distribution of UC samples was significantly different after the IRT ($p < 0.001$).

In total, we executed the program five times using the appropriate variables and number of clusters each time (**Supplementary Material S1**). All the plots and statistical results except **Figure 5** were produced directly and automatically by DivCom. The generated outputs underwent only minor editing for complying with the formatting requirements.

4 DISCUSSION

4.1 DivCom, Iron Dataset, and Beyond

Comparing microbial profiles of different groups can be a challenging process mainly due to the multivariate and



Cluster	Closest reference cluster
IBD1	0.558
IBD2	0.765

FIGURE 6 | MDS plot presenting the *de novo* clustering of the NI and IBD groups. Both the NI and IBD samples were clustered into two clusters. The IBD 1 subcluster is closer to the reference groups of the NI, and the IBD 2 is considerably more distant. The table presents the median level of dissimilarity of each IBD cluster from its nearest reference cluster.

multifactorial nature of the data (Ramette, 2007; Paliy and Shankar, 2016). DivCom proposes a new approach for microbial communities' comparisons that is easily applied using the developed tool. In order to evaluate whether DivCom can produce meaningful results, we employed this methodology to previously produced data that were made public with this work. Next, the basic conclusions of the DivCom analysis are presented and compared with the outcomes of the original publication.

Similar to the results of Lee et al. (2017), we found that the NI group was more homogenous, and the treatment did not considerably affect their overall community composition. Relying on this fact, we treated all the NI samples as a unity during our analysis. In the original article, it was not emphasized that the samples of the UC group were more related to the group of the NI than the group of the CD. In particular, the samples of the UC group chosen to follow the PO treatment were consistently closer to the NI samples. This observation was not mentioned or taken into account in the initial study but was among the default outputs of the DivCom. Sampling imbalances like the above can lead to misleading results when they are not taken into consideration.

In studies exploring the possible differential effect of a treatment on the microbial profiles of two or more groups, the labeling of the subjects is commonly based on their demographics or status characteristics. This process results in dividing the dataset into test and control groups. Traits like age, gender, or disease severity should always be considered and be part of a typical study design in order to avoid biases caused by these factors (Kim et al., 2017; Bharti and Grimm, 2021). However, in addition to the demographic and status characteristics, we argue that the subjects' baseline microbiome is a significant confounder we should always bear in mind in such analyses. We recommend that an initial screening be performed to map the microbial structure of the cohort, and then subjects be assigned to groups so that the underlying microbial groups are equally represented among the test and control groups. DivCom could assist in this process by revealing the different communities present in the cohorts and creating more balanced experiments.

Also, we verified that although the two treatments overall did not shift the IBD samples significantly closer to the reference group, the IV treatment had slightly better results concerning the distances from the reference medoids. The CD samples seemed to have the same response to the IRT independently of the followed method. In contrast, the UC patients' microbiomes seemed to be more sensitive to the type of iron replacement, with IV treatment resulting in overall decreased distance from the reference groups and PO negatively affecting the structure of the community reflected in increased distance from the controls. This observation was not accentuated in the original article as the relationship between the treatments and the type of the disease had not been investigated. DivCom can easily highlight such observations through its integral utilization of the distances as the primary method of group comparison.

The *de novo* clustering of the IBD samples showed that one of the two subclusters was close to the reference group, and the other one was farther away. The IRT method did not seem to affect the structure of the IBD clusters. According to chi-square analysis, the after-treatment PO and IV samples were similarly dispersed across the clusters. However, the type of the disease appeared to influence the way the samples are distributed across the IBD clusters. Almost all the UC samples were in the cluster closer to the NI group, while the CD samples were equally dispersed to the two subclusters. Many of these details went unnoticed in the original publication as the structure of the groups had not been taken into account. However, additional variables like the disease severity, age, or diet could also contribute to the observed clustering pattern. In general, *de novo* clustering can provide us with a sense of how well our recorded metadata reflect and explain the grouping of the microbial profiles. The existence of unexpected structures in the dataset could be an indication that factors that had not been predicted or taken into consideration could have a severe impact on the results (Goodrich et al., 2014; Alashwal et al., 2019). Therefore, it is important to always perform this type of analysis in any experiment dealing with microbiome data.

Considering the differences between the original and DivCom analysis, the former was based mainly on the study of dominant bacterial taxa, while the DivCom analysis used beta diversity metrics to summarize the overall community composition. However, both the initial and the current analyses were conducted with respect to the reference group of the NI. Although the samples of the NI group did not have any type of inflammatory bowel disease, the sampling occurred during their hospitalization. Therefore, it would not be appropriate to generalize the results to the wider healthy population. Considering this fact, a universal baseline reference dataset of healthy individuals would be useful for quickly and easily assessing the level of dysbiosis in individual samples (Lloyd-Price et al., 2016; King et al., 2019). If this becomes a reality, then the way will open for more personalized-focused treatments (Zmora et al., 2016; Behrouzi et al., 2019).

4.2 Strengths and Limitations of DivCom

Beta diversity is one of the most important parts of the microbiome data analysis; it allows us to explore the relationship between the samples and, by extension, the relation between the different groups under study. As presented and described in the introduction, statistical and structural limitations can produce deceptive outcomes that will consequently affect the rest of the analysis. Most of the time, it is not easy to overcome these obstacles, mainly due to the lack of alternative options. DivCom tries to solve some of these problems by using a distance-based approach that considers the inner structure of the data and reducing the dependency on the results of the statistical tests.

The primary purpose of DivCom is to compare different groups and reveal their interrelation. Therefore, it should be used in studies where two or more groups are compared against each other. The ideal scenario would be when the test groups are compared with control/reference samples. Since the proposed approach uses the pairwise distances of every sample from the reference points, the wrong selection of this dataset may lead to misleading and uninterpretable results. Thus, selecting the reference dataset is an essential part of the process.

An advantage of DivCom is that the sampling size of the dataset and the distribution of the samples across the groups did not considerably affect the overall results. DivCom can produce accurate results independently of the dataset. Although the sampling size does not directly affect the procedure and the outcomes, it is recommended not to use extremely small groups (e.g., 2–3 samples), as in this case, it would be difficult to obtain strong statistical results and draw safe conclusions about the overall trend of the groups.

The *de novo* clustering is a fundamental part of the DivCom methodology. Numerous techniques and algorithms perform unsupervised clustering; among these approaches, model-based clustering methods like Dirichlet multinomial mixtures (DMM) (Holmes et al., 2012) and Dirichlet-tree multinomial mixtures (DTMM) (Bai et al., 2020), density-based clustering algorithms like density-based spatial clustering of applications with noise (DBSCAN) (Ester et al., 1996), or even neural network clustering algorithms like self-organizing tree algorithm (SOTA) (Dopazo and Carazo, 1997) are included. However, here in the DivCom tool, we chose a more conventional approach like the PAM algorithm. PAM does not make any assumptions about the distribution of the samples, can work with any dissimilarity matrix, and forms sphere-like clusters. In particular, the last characteristic is extremely useful as it allowed us to successfully use the concept of the central points as representative points of the clusters. As presented in the introduction, only the use of the medoids/centroids produced results analogous with those we would have obtained if we had performed all the possible calculations and comparisons. All the previously mentioned qualities lead DivCom to be fast and produce accurate and detailed outcomes.

Another benefit of using the DivCom approach is that each sample is studied separately, and the program produces various statistical measures for each of these points. In this way, the user can detect outliers and samples with abnormal behavior more easily, and then further assess them. Identifying and evaluating outliers is not always a straightforward task, so this is an important and maybe overlooked feature of the tool.

DivCom does not require any advanced programming knowledge as the users do not have to edit or modify the code in the scripts; they just need to fill out the required parameters and then execute the program. Each of these parameters is described in the scripts, and clear guidelines are provided so even inexperienced users to be able to use the tool. The workflow of DivCom is flexible and can be personalized depending on the requirements and needs of each user. Also, the results are printed in the form of reports in which each plot and table is explained so it will be easier for the user to interpret the results.

The computational and memory requirements of DivCom can be considered limited. The development and testing of the tool were performed mainly in spec-wise average personal computers (processor: Intel core i5, Ram: 8GB, operating system: Windows 10). The requisite time to complete the process ranged from a few seconds to several minutes, depending on the size of the abundance table. For the dataset used in this study (62 × 254 abundance table), the execution time for the whole analysis was approximately 1 min, while for much larger datasets (e.g., 700 × 5,985 abundance table) the execution time for the whole analysis was no more than 7 min. The only part of the process that has increased computational requirements and is time-consuming is the calculation of the beta diversity (Generalized Unifrac). Therefore, for even larger datasets, it is recommended that the user pre-calculates and provides the matrix of the pairwise distances between the samples in order to speed up the process.

5 CONCLUSION AND FUTURE WORK

In conclusion, we proposed a novel approach for the analysis of the microbiome datasets. This approach incorporates beta diversity measures used as distance metrics and the technique of *de novo* clustering. This new methodology offers more detailed and well-defined comparisons between different groups under study. An automated tool that applies the suggested method was developed and introduced.

Also, we assessed the performance of DivCom using existing data and comparing the findings with those of the original study. The outcomes showed its effectiveness as we were able to verify some of the key points of the original publication simply by running our tool while discovering and highlighting unnoticed details.

In some cases, the proposed approach outperforms the current methods and techniques that are applied to the beta diversity analysis. Of course, future improvements and optimizations to the tool will render it easier for the user, will simplify the process, and will expand its capability to handle a wider range of possible cases. The use of DivCom combined with the existing tools for downstream microbiome data analysis offers clear advantages and additional information, and therefore should be considered in every microbiome analysis.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. These data can be found here: European Nucleotide Archive (https://www.ebi.ac.uk/ena/browser/) with accession number: PRJEB48168.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by the local Ethics Committee of the University of Alberta (Canada) and registered with clinicaltrial.gov (NCT010675). The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

EI and IL conceived and designed the experiments, performed the experiments, analyzed the data, contributed materials/analysis tools, wrote the article, prepared figures and/or tables, and reviewed drafts of the article.

FUNDING

This project has received funding from the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under grant agreement no. 710. IL also received funding from the German Research Foundation (SFB 1371, Project No. 395357507). This publication was supported by the Technical University of Munich within the Open Access Publishing Program.

ACKNOWLEDGMENTS

We want to thank Dirk Haller for sharing the original data from the Lee et al. study that were used as the demonstration dataset of the DivCom. Also, we would like to thank Charlie Pauvert and Paschalia Patramani for their feedback on the manuscript text, as well as Antonios Kioukis for his help with editing the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.864382/full#supplementary-material

REFERENCES

Alashwal, H., El Halaby, M., Crouse, J. J., Abdalla, A., and Moustafa, A. A. (2019). The Application of Unsupervised Clustering Methods to Alzheimer's Disease. *Front. Comput. Neurosci.* 13, 31. doi:10.3389/fncom.2019.00031

Anderson, M., Gorley, R., and Clarke, K. (2008). *Permanova+ for Primer: Guide to Software and Statistical Methods*. plymouthuk. *manual.*

Anderson, M. J. (2006). Distance-based Tests for Homogeneity of Multivariate Dispersions. *Biometrics* 62, 245–253. doi:10.1111/j.1541-0420.2005.00440.x

Anderson, M. J. (2001). A New Method for Non-parametric Multivariate Analysis of Variance. *Austral Ecol.* 26, 32–46. doi:10.1111/j.1442-9993.2001.01070.pp.x

Arumugam, M., Raes, J., Pelletier, E., Le Paslier, D., Yamada, T., Mende, D. R., et al. (2011). Enterotypes of the Human Gut Microbiome. *nature* 473, 174–180. doi:10.1038/nature09944

Baarsch, J., and Celebi, M. E. (2012). "Investigation of Internal Validity Measures for K-Means Clustering," in Proceedings of the international multiconference of engineers and computer scientists (sn), 14–16.1.

Bai, B., Cao, Y., and Li, X. (2020). Dtmm: Evacuation Oriented Optimized Scheduling Model for Disaster Management. *Computer Commun.* 150, 661–671. doi:10.1016/j.comcom.2019.11.049

Behrouzi, A., Nafari, A. H., and Siadat, S. D. (2019). The Significance of Microbiome in Personalized Medicine. *Clin. Transl Med.* 8, 16–19. doi:10.1186/s40169-019-0232-y

Benjamini, Y., and Hochberg, Y. (1995). Controlling the False Discovery Rate: a Practical and Powerful Approach to Multiple Testing. *J. R. Stat. Soc. Ser. B (Methodological)* 57, 289–300. doi:10.1111/j.2517-6161.1995.tb02031.x

Bharti, R., and Grimm, D. G. (2021). Current Challenges and Best-Practice Protocols for Microbiome Analysis. *Brief Bioinform* 22, 178–193. doi:10.1093/bib/bbz155

Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, C. C., Al-Ghalith, G. A., et al. (2019). Reproducible, Interactive, Scalable and Extensible Microbiome

Data Science Using Qiime 2. *Nat. Biotechnol.* 37, 852–857. doi:10.1038/s41587-019-0209-9

Bouveyron, C., and Brunet-Saumard, C. (2014). Model-based Clustering of High-Dimensional Data: A Review. *Comput. Stat. Data Anal.* 71, 52–78. doi:10.1016/j.csda.2012.12.008

Bray, J. R., and Curtis, J. T. (1957). An Ordination of the upland forest Communities of Southern Wisconsin. *Ecol. Monogr.* 27, 325–349. doi:10.2307/1942268

Calinski, T., and Harabasz, J. (1974). A Dendrite Method for Cluster Analysis. *Comm. Stats. - Theor. Methods* 3, 1–27. doi:10.1080/03610927408827101

Calle, M. L. (2019). Statistical Analysis of Metagenomics Data. *Genomics Inform.* 17. doi:10.5808/gi.2019.17.1.e6

Chen, J., Bittinger, K., Charlson, E. S., Hoffmann, C., Lewis, J., Wu, G. D., et al. (2012). Associating Microbiome Composition with Environmental Covariates Using Generalized Unifrac Distances. *Bioinformatics* 28, 2106–2113. doi:10.1093/bioinformatics/bts342

Clarke, K. R. (1993). Non-parametric Multivariate Analyses of Changes in Community Structure. *Austral Ecol.* 18, 117–143. doi:10.1111/j.1442-9993.1993.tb00438.x

Coker, O. O., Dai, Z., Nie, Y., Zhao, G., Cao, L., Nakatsu, G., et al. (2018). Mucosal Microbiome Dysbiosis in Gastric Carcinogenesis. *Gut* 67, 1024–1032. doi:10.1136/gutjnl-2017-314281

Dopazo, J., and Carazo, J. M. (1997). Phylogenetic Reconstruction Using an Unsupervised Growing Neural Network that Adopts the Topology of a Phylogenetic Tree. *J. Mol. Evol.* 44, 226–233. doi:10.1007/pl00006139

Edgar, R. C. (2013). Uparse: Highly Accurate Otu Sequences from Microbial Amplicon Reads. *Nat. Methods* 10, 996–998. doi:10.1038/nmeth.2604

Edgar, R. C. (2016). Unoise2: Improved Error-Correction for Illumina 16s and its Amplicon Sequencing. *BioRxiv*, 081257. doi:10.1101/081257

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *kdd* 34, 226–231.

García-Mantrana, I., Selma-Royo, M., González, S., Parra-Llorca, A., Martínez-Costa, C., and Collado, M. C. (2020). Distinct Maternal Microbiota Clusters Are Associated with Diet during Pregnancy: Impact on Neonatal Microbiota and Infant Growth during the First 18 Months of Life. *Gut Microbes* 11, 962–978. doi:10.1080/19490976.2020.1730294

Goodrich, J. K., Di Rienzi, S. C., Poole, A. C., Koren, O., Walters, W. A., Caporaso, J. G., et al. (2014). Conducting a Microbiome Study. *Cell* 158, 250–262. doi:10.1016/j.cell.2014.06.037

Gower, J. C. (1966). Some Distance Properties of Latent Root and Vector Methods Used in Multivariate Analysis. *Biometrika* 53, 325–338. doi:10.1093/biomet/53.3-4.325

Gupta, V. K., Paul, S., and Dutta, C. (2017). Geography, Ethnicity or Subsistence-specific Variations in Human Microbiome Composition and Diversity. *Front. Microbiol.* 8, 1162. doi:10.3389/fmicb.2017.01162

Halfvarson, J., Brislawn, C. J., Lamendella, R., Vázquez-Baeza, Y., Walters, W. A., Bramer, L. M., et al. (2017). Dynamics of the Human Gut Microbiome in Inflammatory Bowel Disease. *Nat. Microbiol.* 2, 17004–17007. doi:10.1038/nmicrobiol.2017.4

Harbison, J. E., Roth-Schulze, A. J., Giles, L. C., Tran, C. D., Ngui, K. M., Penno, M. A., et al. (2019). Gut Microbiome Dysbiosis and Increased Intestinal Permeability in Children with Islet Autoimmunity and Type 1 Diabetes: A Prospective Cohort Study. *Pediatr. Diabetes* 20, 574–583. doi:10.1111/pedi.12865

Hawinkel, S., Kerckhof, F. M., Bijnens, L., and Thas, O. (2019). A Unified Framework for Unconstrained and Constrained Ordination of Microbiome Read Count Data. *PLoS One* 14, e0205474. doi:10.1371/journal.pone.0205474

He, Y., Wu, W., Zheng, H. M., Li, P., McDonald, D., Sheng, H. F., et al. (2018). Regional Variation Limits Applications of Healthy Gut Microbiome Reference Ranges and Disease Models. *Nat. Med.* 24, 1532–1535. doi:10.1038/s41591-018-0164-x

Holmes, I., Harris, K., and Quince, C. (2012). Dirichlet multinomial mixtures: generative models for microbial metagenomics. *PLoS One* 7 (2), e30126.

Hufnagl, K., Pali-Schöll, I., Roth-Walter, F., and Jensen-Jarolim, E. (2020). Dysbiosis of the Gut and Lung Microbiome Has a Role in Asthma. *Semin. Immunopathol* 42, 75–93. doi:10.1007/s00281-019-00775-y

Jaccard, P. (1912). The Distribution of the Flora in the Alpine Zone.1. *New Phytol.* 11, 37–50. doi:10.1111/j.1469-8137.1912.tb05611.x

Kaufman, L., and Rousseeuw, P. J. (2009). *Finding Groups in Data: An Introduction to Cluster Analysis*, 344. John Wiley & Sons, 346. doi:10.1002/9780470316801

Kim, D., Hofstaedter, C. E., Zhao, C., Mattei, L., Tanes, C., Clarke, E., et al. (2017). Optimizing Methods and Dodging Pitfalls in Microbiome Research. *Microbiome* 5, 52–14. doi:10.1186/s40168-017-0267-5

King, C. H., Desai, H., Sylvetsky, A. C., LoTempio, J., Ayanyan, S., Carrie, J., et al. (2019). Baseline Human Gut Microbiota Profile in Healthy People and Standard Reporting Template. *PloS one* 14, e0206484. doi:10.1371/journal.pone.0206484

Knight, R., Vrbanac, A., Taylor, B. C., Aksenov, A., Callewaert, C., Debelius, J., et al. (2018). Best Practices for Analysing Microbiomes. *Nat. Rev. Microbiol.* 16, 410–422. doi:10.1038/s41579-018-0029-9

Lagkouvardos, I., Fischer, S., Kumar, N., and Clavel, T. (2017). Rhea: a Transparent and Modular R Pipeline for Microbial Profiling Based on 16s Rrna Gene Amplicons. *PeerJ* 5, e2836. doi:10.7717/peerj.2836

Lagkouvardos, I., Joseph, D., Kapfhammer, M., Giritli, S., Horn, M., Haller, D., et al. (2016). Imngs: a Comprehensive Open Resource of Processed 16s Rrna Microbial Profiles for Ecology and Diversity Studies. *Sci. Rep.* 6, 33721–33729. doi:10.1038/srep33721

Lee, T., Clavel, T., Smirnov, K., Schmidt, A., Lagkouvardos, I., Walker, A., et al. (2017). Oral versus Intravenous Iron Replacement Therapy Distinctly Alters the Gut Microbiota and Metabolome in Patients with Ibd. *Gut* 66, 863–871. doi:10.1136/gutjnl-2015-309940

Lian, J., Wijffels, R. H., Smidt, H., and Sipkema, D. (2018). The Effect of the Algal Microbiome on Industrial Production of Microalgae. *Microb. Biotechnol.* 11, 806–818. doi:10.1111/1751-7915.13296

Lin, S. W., Freedman, N. D., Shi, J., Gail, M. H., Vogtmann, E., Yu, G., et al. (2015). Beta-diversity Metrics of the Upper Digestive Tract Microbiome Are Associated with Body Mass index. *Obesity (Silver Spring)* 23, 862–869. doi:10.1002/oby.21020

Lloyd-Price, J., Abu-Ali, G., and Huttenhower, C. (2016). The Healthy Human Microbiome. *Genome Med.* 8, 51–11. doi:10.1186/s13073-016-0307-y

Lozupone, C., Lladser, M. E., Knights, D., Stombaugh, J., and Knight, R. (2011). Unifrac: an Effective Distance Metric for Microbial Community Comparison. *ISME J.* 5, 169–172. doi:10.1038/ismej.2010.133

Mann, H. B., and Whitney, D. R. (1947). On a Test of whether One of Two Random Variables Is Stochastically Larger Than the Other. *Ann. Math. Statist.* 18, 50–60. doi:10.1214/aoms/1177730491

McMurdie, P. J., and Holmes, S. (2013). Phyloseq: an R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. *PloS one* 8, e61217. doi:10.1371/journal.pone.0061217

Morris, A., Beck, J. M., Schloss, P. D., Campbell, T. B., Crothers, K., Curtis, J. L., et al. (2013). Comparison of the Respiratory Microbiome in Healthy Nonsmokers and Smokers. *Am. J. Respir. Crit. Care Med.* 187, 1067–1075. doi:10.1164/rccm.201210-1913OC

Navas-Molina, J. A., Peralta-Sánchez, J. M., González, A., McMurdie, P. J., Vázquez-Baeza, Y., Xu, Z., et al. (2013). Advancing Our Understanding of the Human Microbiome Using Qiime. *Methods Enzymol.* 531, 371–444. doi:10.1016/B978-0-12-407863-5.00019-8

Oksanen, J., Blanchet, F. G., Kindt, R., Legendre, P., Minchin, P., O'Hara, R., et al. (2015). Vegan Community Ecology Package: Ordination Methods, Diversity Analysis and Other Functions for Community and Vegetation Ecologists. *R. Package Ver*, 2–3.

Paetzold, B., Willis, J. R., Pereira de Lima, J., Knödlseder, N., Brüggemann, H., Quist, S. R., et al. (2019). Skin Microbiome Modulation Induced by Probiotic Solutions. *Microbiome* 7, 95–99. doi:10.1186/s40168-019-0709-3

Paliy, O., and Shankar, V. (2016). Application of Multivariate Statistical Techniques in Microbial Ecology. *Mol. Ecol.* 25, 1032–1057. doi:10.1111/mec.13536

Prast-Nielsen, S., Tobin, A. M., Adamzik, K., Powles, A., Hugerth, L. W., Sweeney, C., et al. (2019). Investigation of the Skin Microbiome: Swabs vs. Biopsies. *Br. J. Dermatol.* 181, 572–579. doi:10.1111/bjd.17691

Qiu, Z., Egidi, E., Liu, H., Kaur, S., and Singh, B. K. (2019). New Frontiers in Agriculture Productivity: Optimised Microbial Inoculants and *In Situ* Microbiome Engineering. *Biotechnol. Adv.* 37, 107371. doi:10.1016/j.biotechadv.2019.03.010

Ramette, A. (2007). Multivariate Analyses in Microbial Ecology. *FEMS Microbiol. Ecol.* 62, 142–160. doi:10.1111/j.1574-6941.2007.00375.x

Rousseeuw, P. J. (1987). Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis. *J. Comput. Appl. Math.* 20, 53–65. doi:10.1016/0377-0427(87)90125-7

Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, E. B., et al. (2009). Introducing Mothur: Open-Source, Platform-independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Appl. Environ. Microbiol.* 75, 7537–7541. doi:10.1128/AEM.01541-09

Suzuki, T., Sutani, T., Nakai, H., Shirahige, K., and Kinoshita, S. (2020). The Microbiome of the Meibum and Ocular Surface in Healthy Subjects. *Invest. Ophthalmol. Vis. Sci.* 61, 18. doi:10.1167/iovs.61.2.18

Tibshirani, R., and Walther, G. (2005). Cluster Validation by Prediction Strength. *J. Comput. Graphical Stat.* 14, 511–528. doi:10.1198/106186005x59243

Ventura, R. E., Iizumi, T., Battaglia, T., Liu, M., Perez-Perez, G. I., Herbert, J., et al. (2019). Gut Microbiome of Treatment-Naïve MS Patients of Different Ethnicities Early in Disease Course. *Sci. Rep.* 9 (1), 1–10. doi:10.1038/s41598-019-52894-z

Wagner, B. D., Grunwald, G. K., Zerbe, G. O., Mikulich-Gilbertson, S. K., Robertson, C. E., Zemanick, E. T., et al. (2018). On the Use of Diversity Measures in Longitudinal Sequencing Studies of Microbial Communities. *Front. Microbiol.* 9, 1037. doi:10.3389/fmicb.2018.01037

Warton, D. I., Wright, S. T., and Wang, Y. (2012). Distance-based Multivariate Analyses Confound Location and Dispersion Effects. *Methods Ecol. Evol.* 3, 89–101. doi:10.1111/j.2041-210x.2011.00127.x

Wilcoxon, F. (1992). "Individual Comparisons by Ranking Methods," in *Breakthroughs in Statistics* (Springer), 196–202. doi:10.1007/978-1-4612-4380-9_16

Wirbel, J., Zych, K., Essex, M., Karcher, N., Kartal, E., Salazar, G., et al. (2021). Microbiome Meta-Analysis and Cross-Disease Comparison Enabled by the Siamcat Machine Learning Toolbox. *Genome Biol.* 22, 93–27. doi:10.1186/s13059-021-02306-1

Xia, Y., and Sun, J. (2017). Hypothesis Testing and Statistical Analysis of Microbiome. *Genes Dis.* 4, 138–148. doi:10.1016/j.gendis.2017.06.001

Zmora, N., Zeevi, D., Korem, T., Segal, E., and Elinav, E. (2016). Taking it Personally: Personalized Utilization of the Human Microbiome in Health and Disease. *Cell Host Microbe* 19, 12–20. doi:10.1016/j.chom.2015.12.016

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Intze and Lagkouvardos . This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 13 May 2022
doi: 10.3389/fbinf.2022.867111

VirHunter: A Deep Learning-Based Method for Detection of Novel RNA Viruses in Plant Sequencing Data

Grigorii Sukhorukov[1,2]*, *Maryam Khalili*[3], *Olivier Gascuel*[4], *Thierry Candresse*[3], *Armelle Marais-Colombel*[3] and *Macha Nikolski*[1,2]*

[1]CNRS, IBGC, UMR 5095, Université de Bordeaux, Bordeaux, France, [2]Bordeaux Bioinformatics Center, Université de Bordeaux, Bordeaux, France, [3]Université de Bordeaux, INRAE, UMR BFP, CS20032, CEDEX, Villenave d'Ornon, France, [4]Institut de Systématique, Biodiversité, Evolution (ISYEB - UMR7205, Muséum National d'Histoire Naturelle, CNRS, SU, EPHE, UA), Paris, France

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Deyvid Amgarten,
Albert Einstein Israelite Hospital, Brazil
Bruno Koshin Vázquez Iha,
Universidade Anhanguera de São Paulo, Brazil

Correspondence:
Grigorii Sukhorukov
grsukhorukov@gmail.com
Macha Nikolski
macha.nikolski@u-bordeaux.fr

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *31 January 2022*
Accepted: *24 March 2022*
Published: *13 May 2022*

Citation:
Sukhorukov G, Khalili M, Gascuel O, Candresse T, Marais-Colombel A and Nikolski M (2022) VirHunter: A Deep Learning-Based Method for Detection of Novel RNA Viruses in Plant Sequencing Data.
Front. Bioinform. 2:867111.
doi: 10.3389/fbinf.2022.867111

High-throughput sequencing has provided the capacity of broad virus detection for both known and unknown viruses in a variety of hosts and habitats. It has been successfully applied for novel virus discovery in many agricultural crops, leading to the current drive to apply this technology routinely for plant health diagnostics. For this, efficient and precise methods for sequencing-based virus detection and discovery are essential. However, both existing alignment-based methods relying on reference databases and even more recent machine learning approaches are not efficient enough in detecting unknown viruses in RNAseq datasets of plant viromes. We present VirHunter, a deep learning convolutional neural network approach, to detect novel and known viruses in assemblies of sequencing datasets. While our method is generally applicable to a variety of viruses, here, we trained and evaluated it specifically for RNA viruses by reinforcing the coding sequences' content in the training dataset. Trained on the NCBI plant viruses data for three different host species (peach, grapevine, and sugar beet), VirHunter outperformed the state-of-the-art method, DeepVirFinder, for the detection of novel viruses, both in the synthetic leave-out setting and on the 12 newly acquired RNAseq datasets. Compared with the traditional tBLASTx approach, VirHunter has consistently exhibited better results in the majority of leave-out experiments. In conclusion, we have shown that VirHunter can be used to streamline the analyses of plant HTS-acquired viromes and is particularly well suited for the detection of novel viral contigs, in RNAseq datasets.

Keywords: novel virus detection, RNA viruses, plant virome, alignment-free method, deep learning, artificial neural network

INTRODUCTION

Study of viromes at an unprecedented scale has been enabled by the adoption of high-throughput sequencing (HTS) technologies and is now frequently undertaken across an increasing range of host species. In particular, sequencing of plant viromes has become quite common, partly due to its relevance to the agricultural sector. The acquired datasets help to elucidate important questions such as virus spread among host reservoirs and effects of agriculture on the ecosystems and their biodiversity as well as the identification of novel viruses in crops and natural environments (Lefeuvre et al., 2019). These developments are fast advancing our knowledge of viral diversity through the discovery of previously unknown viral species or variants and the identification of new hosts of known viruses (Roossinck et al., 2015; Massart et al., 2017). Following the classification proposed by Stobbe and Roossinck (2014), viruses identified in HTS datasets can be classified into three different groups as follows: 1) viruses that are already known to infect a given host; 2) novel viruses from a known family or known viruses that have not been found previously described to infect a given host; and finally 3) completely novel viruses that share little to no sequence similarity with known viruses already present in the databases.

Using an efficient virus detection method, including for the identification of novel viruses, is essential for efficient disease management. Standard diagnostic tests (ELISA assays and PCR-based assays) depend on specific antibodies or primers and thus require prior knowledge of the virus and of its phylogenetic neighbors. Precise identification of viruses is further complexified by the large diversity encountered in the majority of viral species which is linked to the high mutation rate of these agents. This is particularly true for plant viruses, the majority of which are RNA viruses whose mutation rate is very high (Jenkins et al., 2002). Moreover, the new variants emerging from genomic rearrangements or recombination events can also significantly differ from the parental viruses (Domingo 2010). Also, many of the plant viruses are multihost pathogens, and a single plant can be infected by multiple unrelated viral species (Roossinck, 1997). Such infections by multiple viruses represent an additional challenge for detection since the viral load of different pathogens can be very unequal (Martín and Elena, 2009). Moreover, in most cases, background contamination is currently unavoidable (Kleiner et al., 2015; Maree et al., 2018; Kutnjak et al., 2021). In this context, HTS combined with bioinformatics tools has been shown to be a valuable approach, both for detection of known viruses and for the discovery of novel ones (Maree et al., 2018; Villamor et al., 2019; Mehetre et al., 2021).

Viruses do not have a universal gene marker that could be used for their identification, contrary to the conserved regions of the 16S rRNA and ITS genes, commonly used to classify bacteria and fungi at the genus or species level (Mokili et al., 2012). Moreover, the abundance of viral genomic material in plant sequencing samples can be very low (Massart et al., 2019), due to the dominance of the host material. Hence, specific sample preparation to enrich plant RNA viral-specific sequences is an important step that makes the downstream detection of viruses by bioinformatics methods more reliable. They include approaches providing a high and targeted enrichment of viral sequences, such as the purification of viral double-stranded RNAs (dsRNAs) or that of virion-associated nucleic acids (VANAs) as well as less specific approaches generally affording lower enrichment, such as the sequencing of small interfering RNAs (siRNAs) or inclusion of a ribodepletion step prior to the sequencing of total cellular RNAs (Maree et al., 2018; Kutnjak et al., 2021). As already discussed in a range of reviews, each of these approaches have advantages and weaknesses. In particular, strategies providing high enrichment factors may improve detection sensitivity but often at the cost of introducing biases with the risk of compromising the detection of some particular viruses (Maree et al., 2018; Kutnjak et al., 2021). For example, dsRNA-based approaches are usually poor at detecting DNA viruses, while VANA-based ones may perform poorly for viruses with labile particles.

When interested in known viruses or potentially novel viruses but from a known family, bioinformatics methods that compare the sequenced reads to genomes in public databases are very efficient for virus detection and identification (Stobbe and Roossinck, 2014; Massart et al., 2019). Read-based analysis is thus particularly suited to study viral diversity of sequencing samples in terms of known viral species. Generalistic metagenome analysis tools such as, for example, Kaju (Menzel et al., 2016), Kraken 2 (Wood et al., 2019), and Centrifuge (Kim et al., 2016) show good performance in terms of sensitivity and precision in detection of present known viral species (De Vries et al., 2021).

For the discovery of novel viruses, use of *de novo* assembly to recover novel viral contigs from sequencing data is an essential step in order to overcome the incompleteness of virus reference databases, annotation errors and, importantly, the limited homology between novel viral sequences and reference genomes (Sutton et al., 2019). The assembly step is a staple of short-read sequencing studies, which are still the vast majority today (Maree et al., 2018; Kutnjak et al., 2021). It represents its own challenges, in particular, for very short reads such as those of siRNAs and for viral populations with multiple and microdiverse variants (Warwick-Dugdale et al., 2019), often leading to microdiversity-associated fragmentation and, sometimes, to chimeras in the resulting contigs (Martinez-Hernandez et al., 2017; Roux et al., 2017), which in turn affects the downstream analysis, including estimation of viral diversity and identification of novel viruses (Nayfach et al., 2021). Popular assembler choices are the generalistic de Bruijn graph assembly metaSPAdes (Nurk et al., 2017) and Trinity, for RNAseq (Grabherr et al., 2011).

Following the recent review (Kutnjak et al., 2021), the methods used to analyze assembled contigs can be grouped into three main categories: 1) alignment and mapping-based methods, 2) protein domain searches, and 3) k-mer-based approaches that can either rely on signatures or leverage machine learning. Among this large plethora of tools, alignment-based methods are widely adopted when working with assembled contigs since they provide a longer sequence for homology search against reference genomes using either BLAST (Altschul et al., 1990) and its derivatives or the amino acid alignment of protein-coding genes predicted from the assembled contigs using DIAMOND (Buchfink et al., 2015). Also, focusing the analysis on coding regions is particularly relevant for RNAseq data since the non-coding sequences of viruses are not highly represented in such samples, even if they can be well conserved in certain viral taxa. However, the main drawback of alignment- or mapping-based approaches lies on the fact that they are both computationally intensive and require expertise for filtering and interpreting the results. As for the generalistic k-mer signature approaches, they remain demanding in terms of memory and are best suited for diversity analysis tasks (Kutnjak et al., 2021).

The emergence of machine learning tools for contig-based analysis of virome sequencing data holds much promise to streamline the discovery of novel viruses in sequencing datasets by both avoiding the time-consuming sequence similarity analyses and modeling even highly divergent sequences. These methods build models based on sequences with known class labels such as "virus" and "host" and learn features that allow them to differentiate between the classes. VirFinder (Ren et al., 2017) and VirSorter2 (Guo et al., 2021) rely on classical machine learning, the former being based on a regularized logistic regression applied to the k-mer frequency matrix extracted from the sequence and the latter on a random forest model built from genomic features. Methods based on deep learning networks have also been proposed for virus detection, such as DeepVirFinder (Ren et al., 2020) and ViraMiner (Tampuu et al., 2019) that both rely on a combination of convolutional neural networks (CNNs) and dense neural networks, and VirNet (Abdelkareem et al., 2018) that relies on a long short-term memory (LSTM) architecture. These three deep learning methods were developed for identification of viral contigs in metagenomic samples and evaluated on bacterial and human metagenomes. However, DeepVirFinder has been recently successfully used in plant-related virome studies (Santos-Medellin et al., 2021).

In this work, we present VirHunter, a deep learning method that uses convolutional neural networks (CNNs), classifies previously assembled contigs to identify potential viral, host, and bacterial (contamination) sequences in RNAseq samples. The hybrid architecture of VirHunter combines a multi-network CNN-based module covering different k-mer sizes with a downstream random forest classifier module. We have trained VirHunter models for three different plant host species: peach, grapevine, and sugar beet. Importantly, we have shown that VirHunter is especially performant for the task of completely novel virus discovery by building 31 leave-out datasets, in which each viral family is excluded from the training dataset, and comparing the results with a standard BLAST-based solution on one side and a state-of-the-art deep learning method, DeepVirFinder, the other side. VirHunter not only systematically outperformed DeepVirFinder in terms of virus detection but also has considerably reduced the False Positive rate. Cross-evaluation has shown that host detection accuracy remained high and decreased slightly when test sequences originated from the plant species were further phylogenetically removed from that used to train the model. We have further evaluated the detection capacity of VirHunter on *in silico* mutated contigs sampled from the NCBI virus database and have shown that it decreased only slightly with a progressively increased mutation rate (e.g., True Positive rate of 0.898 for 20% mutation rate). Moreover, we generated 12 RNAseq datasets for a range of host species and have shown that VirHunter was not only able to uncover the viruses that were previously identified but also to streamline the analyses by considerably reducing the need for manual curation.

MATERIALS AND METHODS

Datasets

We downloaded all complete and incomplete viral sequences from the NCBI virus database for which the host's taxonomic id belongs to *Viridiplantae* on 26/10/2021, which yielded 122,832 sequences. Plant sequences have been downloaded for *Prunus persica* (peach), *Vitis vinifera* (grapevine), *Beta vulgaris* (sugar beet), and *Oryza sativa* (rice) from the NCBI RefSeq genomes database. On one hand, they consist of the latest available assemblies, GCF_000346465.2, GCF_000003745.3, GCF_000511025.2, and GCF_001433935.1 for peach, grapevine, sugar beet, and rice, respectively, and of the coding region sequences (CDSs), on the other hand. In the absence of the plastid sequence in the reference assembly of the sugar beet, we used the separately available sugar beet plastid sequence (NC_059012.1). All complete representative bacterial genomes have been downloaded from the NCBI RefSeq database on 28/10/2021 using the genome_updater.sh script.

To simulate the discovery of completely unknown viruses that do not have expected similarities with the available data, we constructed virus family leave-out datasets by excluding in turns all the sequences of a given plant viral family from the downloaded virus dataset. The NCBI taxonomy contains 45 viral families. We excluded the *Pospiviroidae* and the *Avsunviroidae* families of viroids as they have very small genomes (average length < 1,000). All families with the number of available sequences <100 were merged in one dataset called *small families*. Finally, all sequences without family labels constituted the *unclassified* dataset. This resulted in 31 leave-out datasets.

Moreover, we generated 12 novel virome-sequencing RNAseq datasets, sampled from peach, grapevine, and sugar beet (see *Sample Preparation and Sequencing*). Description of these datasets and presence of viruses identified by aligning assembled contigs against the NCBI GenBank database (see *Assembly of RNAseq Datasets and Annotation of Viral Contigs*) are listed in the **Supplementary Table S1**.

Sample Preparation and Sequencing

Total RNAs were extracted from three peach leaf samples, three grapevine phloem scrapping samples, and three sugar beet leaf samples using the CTAB method (Chang et al., 1993), the Spectrum™ Plant Total RNA Kit (Sigma-Aldrich, Saint Quentin-Fallavier, France), and the NucleoSpin RNA plant kit (Macherey-Nagel SAS, Hoerdt, France), respectively. RNAseq libraries were prepared either from total RNAs (peach and grapevine samples), messenger RNAs (grapevine samples), or ribodepleted RNAs (sugar beet samples). High-throughput sequencing was performed on an Illumina platform (Hiseq3000 or NovaSeq600) using a paired-end read length of 2 × 150 bp. Accession numbers for each of the three studies (peach, grapevine, and sugar beet) containing raw FASTQ sequencing files are provided in the **Supplementary Table S1**.

Assembly of RNAseq Datasets and Annotation of Viral Contigs

All of the 12 selected plant virome datasets (see *Datasets*) were processed with the QIAGEN CLC Genomics Workbench (v.21.0.5). Briefly, reads were first quality-controlled and trimmed using default parameters and then assembled using

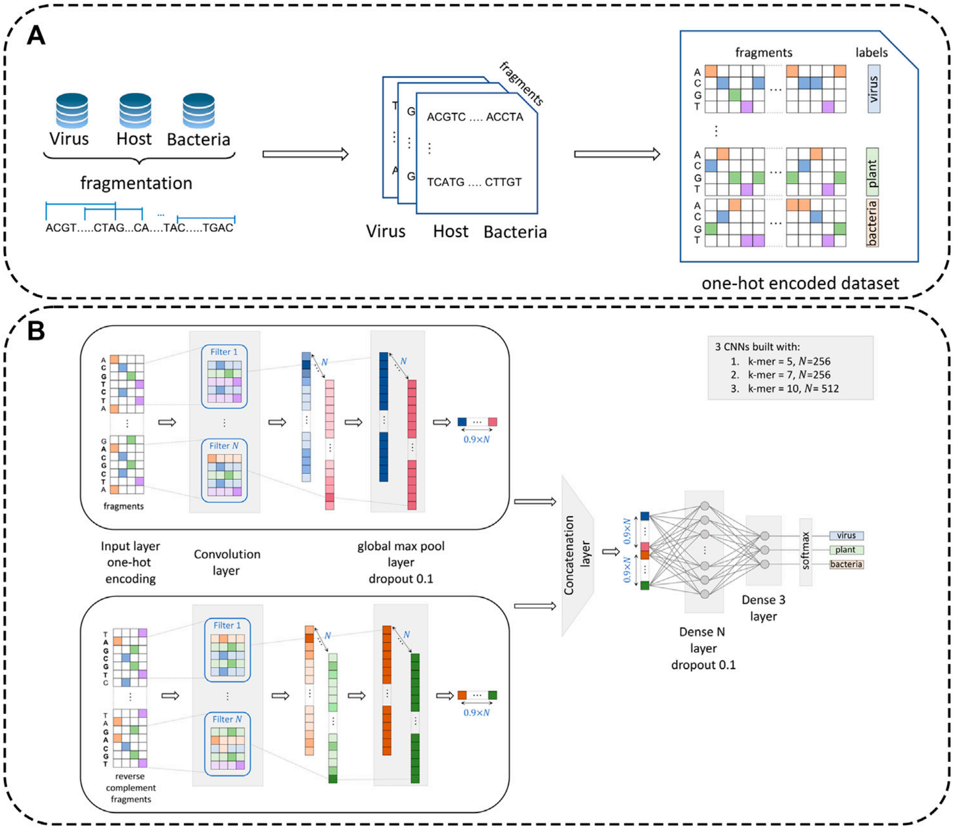


FIGURE 1 | Dataset preprocessing procedure and architecture of the multi-CNN module. Panel **(A)**: Reference datasets (virus, plant, and bacteria) are first fragmented with a pre-defined fragment size (500 and 1000 bp). Each fragment is further one-hot encoded and carries the class label. Panel **(B)**: Three CNN modules are built for k-mers of size $k = 5, 7$, and 10. One-hot encoded genomic fragments of a fixed size are processed by convolutional and global max-pooling layers before being concatenated. A total of two dense layers are followed by the softmax activation function to produce a 3-class classification.

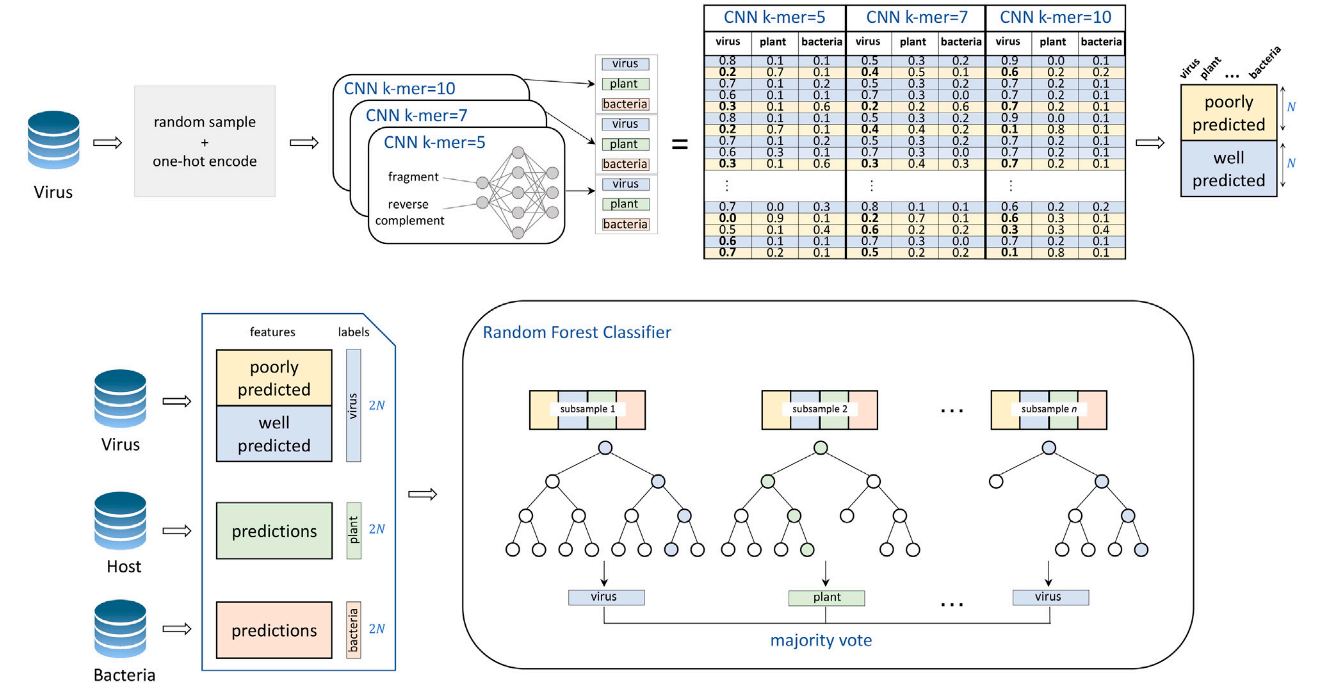
de novo assembly (word size 50, bubble size 300, and minimum contig length 250). To identify viral contigs present in these datasets, we followed a standard three step BLAST-based approach, see e.g., (Candresse et al., 2018). 1) All contigs were aligned using the CLC built in tBLASTx tool against the NCBI nucleotide non-redundant database limited to taxonomic identifiers of viruses. Contigs having significant hits (e-value below the 10^{-20} cut off) were selected. 2) Contigs were further filtered by aligning them using BLASTn and BLASTx with default parameters against the whole GenBank non-redundant nucleotide and protein databases, respectively, and keeping contigs for which the best hits correspond to plant viruses for both BLASTn and BLASTx. Additional manual expert curation allowed to discard contigs with incoherencies between the two alignment results. 3) Finally, all reads passing quality control were mapped against the plant viral contigs, resulting from step 2 using the CLC built-in mapping utility with default parameters with high stringency (90% identity of 90% of read's length). Only contigs with length >750 nucleotides and having sufficient read coverage (expert curation) were retained.

Annotation results together with the counts of thus identified viral contigs are listed in the **Supplementary Table S1**.

Data Preprocessing

To prepare the data for processing by the neural network module, datasets were preprocessed by creating representative one-hot encoded fragments (see **Figure 1**). Specifically, let us denote the virus dataset by V, the plant dataset by H (for "host")—composed of the full assembly G, the coding sequences C, the chloroplast sequence L, and the bacterial dataset by B. Given a fragment size n of 500 and 1,000 nucleotides, V was split into fragments of size n with a sliding window with an increment of $n/2$. Sequences shorter than n nucleotides and longer than $0.95 \times n$ were padded to n bp length with gaps (those shorter than $0.95 \times n$ are discarded), together yielding N viral fragments. Same number N of fragments of size n was randomly sampled from B. As for the plant, G was split into $0.6 \times N$ fragments using a sliding window with an increment of size n, C was split into $0.3 \times N$ fragments with a sliding window with increment of $n/2$, and finally $0.1 \times N$ fragments were sampled randomly from L.

Including plastids in relatively high proportion into the plant dataset H was important to avoid the potential incorrect assignment of contigs originating from plastids to B, given the phylogenetic proximity of plastids and bacteria (McFadden, 2001). Moreover, there are RNA viruses that are known to be replicated in tight association with plastids (mostly chloroplasts) -



CNN k-mer=5			CNN k-mer=7			CNN k-mer=10		
virus	plant	bacteria	virus	plant	bacteria	virus	plant	bacteria
0.8	0.1	0.1	0.5	0.3	0.2	0.9	0.0	0.1
0.2	0.7	0.1	**0.4**	0.5	0.1	**0.6**	0.2	0.2
0.7	0.1	0.2	0.5	0.3	0.2	0.7	0.2	0.1
0.6	0.1	0.1	0.7	0.3	0.0	0.7	0.2	0.1
0.3	0.1	0.6	**0.2**	0.2	0.6	**0.7**	0.2	0.1
0.8	0.1	0.1	0.5	0.3	0.2	0.9	0.0	0.1
0.2	0.7	0.1	**0.4**	0.4	0.2	**0.1**	0.8	0.1
0.7	0.1	0.2	0.5	0.3	0.2	0.7	0.2	0.1
0.6	0.3	0.1	0.7	0.3	0.0	0.7	0.2	0.1
0.3	0.1	0.6	**0.3**	0.4	0.3	**0.7**	0.2	0.1
⋮			⋮			⋮		
0.7	0.0	0.3	0.8	0.1	0.1	0.6	0.2	0.2
0.0	0.9	0.1	**0.2**	0.7	0.1	**0.6**	0.3	0.1
0.5	0.1	0.4	**0.6**	0.2	0.2	**0.3**	0.3	0.4
0.6	0.1	0.1	0.7	0.3	0.0	0.7	0.2	0.1
0.7	0.2	0.1	**0.5**	0.2	0.2	**0.1**	0.8	0.1

FIGURE 2 | Training of the VirHunter's machine learning module. The individual network predictions are subsetted to contain an equal number of both poorly predicted (prediction value for viral class < 0.8) and well-predicted (prediction value ≥ 0.8) viral fragments (with the goal to overselect poor predictions relative to their overall frequency in order to drive the model to recognize even completely novel viruses). The random forest classifier uses these subsetted predictions for its training.

see e.g., (Budziszewska and Obrępalska-Stęplowska, 2018; Delgado et al., 2019). Enriching for CDS sequences was necessary since the envisioned application of VirHunter is for RNAseq virome datasets. Five compositions of *G*/*C*/*L* proportions of *H* were tested (100/0/0, 90/0/10, 60/30/10, 50/40/10, and 45/45/10, data not shown) and the best was retained.

Fragments were further transformed from length *n* ACGT-character strings to $n \times 4$ one-hot encoded arrays, in which an A is encoded by [1, 0, 0, 0], a C is encoded by [0, 1, 0, 0] *etc.*, while gaps are encoded by [0, 0, 0, 0]. Moreover, the encoded dataset is augmented by adding the reverse complement of each original fragment. Indeed, it has been shown by Shrikumar et al. (2017) that CNN models in genomics require the reverse-complement data augmentation combined with parameter sharing between the forward- and reverse-complement representations of the model. Class labels *V*, *H*, or *B* are assigned to each fragment according to its provenance.

VirHunter Architecture

VirHunter architecture was defined with two main components the first component is a multi-path neural network shown in **Figure 1**, and the second component is a machine learning classification module shown in **Figure 2**.

1. Neural network component. The neural network module follows a k-mer-based approach. To alleviate a potential difficulty related to the choice of *k*, VirHunter implements a multi-model solution for $k = 5$, 7, and 10 (see **Figure 1**), with three independent CNN models having the same architecture. These values of *k* were chosen based on the accuracy of the individual CNN networks in the family leave-out experiment (see **Supplemental Figure S1**). The genomic DNA sequence and its reverse complement for each n-size fragment are transformed from nucleotides (in ACGTN alphabet) to an $n \times 4$ one-hot encoded array as presented in *Data Preprocessing*. A convolution layer with leaky rectified linear unit activation function ($a = 0.1$) and global max-pool and dropout layers are then applied independently to the forward fragments and their paired reverse-complement versions. The use of dropout layers was introduced to alleviate the issue of overfitting. Models with $k = 5, 7$ have the convolution layer with 256 filters, while the model for $k = 10$ has 512 filters. The two resulting vectors for the forward- and reverse-complement fragments are then concatenated. Finally, two dense layers are applied. The first dense layer has the number of units equal to 256 for the paths with $k = 5, 7$ and 512 for the path with $k = 10$. It employs a rectified linear unit activation function. The second dense layer has three units and uses the softmax activation function to enable three-class classification.

2. Random Forest component. The second module of the VirHunter implements a random forest classifier (see **Figure 2**) with the goal to aggregate the predictions from three neural networks. The classifier receives nine real-valued predictions from the multi-network module (three per network) and outputs one of the three classes using the majority vote implementation of random forest. The random forest classifier was chosen over other approaches such as linear regression and simple voting, based on performance (data not shown).

Training

The neural network and machine learning modules were trained separately for each of the three plant host species (peach, grapevine, and sugar beet) and for fragment sizes n of 500 and 1,000.

The training dataset for the CNN module was built as presented in *Data Preprocessing*. Training batches with size 512 were prepared in a balanced manner across the three classes (virus, plant, and bacteria) from the training dataset and are split into training and validation with the ratio of 9:1. Each of the three individual networks was trained for 10 epochs, followed by 1 epoch of training on the validation set to take into account all the data.

For training and testing the machine learning components, predictions for the three trained networks were obtained on $100,000$ randomly selected fragments of size n from each V and B. Likewise, $100,000$ fragments of size n were randomly sampled from H, following the ratio described in *Data Preprocessing*. Predictions for random viral fragments were further subsetted in the following manner. We split the test dataset viral fragments into those having good quality predictions (prediction value for viral class ≥ 0.8) and low-quality predictions (prediction value < 0.8) and maintained $10,000$ randomly selected fragments from each category, yielding $20,000$ predictions. These $20,000$ predictions were further selected for plant host H and bacterial B fragments. The resulting dataset with three predictions for each of $60,000$ fragments was further split in train and test datasets with 2:1 ratio, and the machine learning module was trained with parameters max_depth = 5, n_estimators = 10, max_features = 1, and max_samples = 0.2.

We verified that overfitting was successfully circumvented by the individual CNN networks that compose the neural network component of our model by comparing the accuracy on validation and test datasets obtained by these individual networks trained on families in the leave-out experiment for peach (see **Supplementary Table S9**). No significant difference was observed.

Contig Classification

VirHunter trained on fragments with $n = 500$ was used to classify contigs with length $750 < l < 1500$, while VirHunter trained on fragments with $n = 1000$ was used to classify contigs with $1500 < l$. Indeed, an ORF of 500 nucleotides corresponds to an 18 kDa protein, this size covering the vast majority of viral polymerases, movement proteins, and capsid proteins for plant viruses. Contigs with $l < 750$ were considered as very small for prediction by the smaller of the two models and were discarded.

Each fragment of an input contig was preprocessed following the procedure presented in *Data Preprocessing*. Predictions were produced by the neural network module for each of these one-hot encoded fragments, yielding three probabilities of belonging to a specific class (V, H, B). These class probabilities were further processed by the random forest component, resulting in a unique class label for each of the fragments. Finally, given class labels for each of the fragments of the input contig, a vote was applied to decide to which class belongs the whole contig, viral if the number of viral (V) fragments is greater than those from H and from B, host if the number of host (H) fragments is greater than those from V and from B, and bacterial otherwise.

RESULTS

VirHunter Outperforms State-of-the-Art Tools on Family Leave-Out Datasets

VirHunter was trained on GPU (Nvidia Tesla T4) with $n = 1000$ for 31 family leave-out datasets and three different plant datasets (peach, grapevine, and sugar beet), resulting in 63 leave-out models. The test datasets were prepared by random sampling of 30,000 fragments with $n = 1000$ from the corresponding left-aside families of viral sequences, bacteria, and plant.

Classification results for the viral fragments by VirHunter in this family leave-out experiment are shown in **Figure 3** and in **Supplementary Tables S2, S3**. We compared the capacity of VirHunter to detect novel viruses in the family leave-out setting with the BLAST-based approach on one hand and two state-of-the-art machine learning methods, DeepVirFinder and VirSorter2, on the other hand as also shown in **Figure 3**. Briefly, each test dataset was aligned using tBLASTx (v2.12.0), preserving one best hit with parameters -max_target_seqs 1 -max_hsps 1, against the respective virus database with the leave-out family removed, and filtered by e-value $< 10^{-10}$, percent identity > 0.5, and alignment length > 50 amino acids (see results in **Supplementary Table S4**) in order to emulate the annotation workflow without manual inspection; DeepVirFinder was trained on the same 31 leave-out datasets but excluding bacterial fragments from the training dataset since this method provides the possibility to have only two class labels and using the recommended parameters (Ren et al., 2020) on 10 CPUs Intel Xeon CPU E5-2630 v4 (see results in **Supplementary Table S5**); VirSorter2 was evaluated on each test dataset using pretrained models provided by authors (see results in **Supplementary Table S6**).

Variability of correct classification was observed for viral fragments of different left-out families for all three methods as shown in **Figure 3** (see for detailed results in **Supplementary Tables S2–S4**). We have split the families into three groups according to the lowest True Positive (TP) rate of VirHunter across the three plant host species: 21 "easy to classify" (TP rate > 0.7), 7 "moderately difficult to classify" (TP rate between 0.5 and 0.7), and 3 "difficult to classify" (TP rate < 0.5). VirHunter almost systematically outperformed DeepVirFinder in terms of TPs (virus fragments from the leave-out family classified as being viral). In total, there are four exceptions, namely, *Reoviridae*, *Mayoviridae*, *Phycodnaviridae*, and *small families*, out of which *Reoviridae* presented a considerable performance gap. After inspection, it appeared that VirHunter's false negatives for these four families mostly corresponded to viral fragments being classified as bacteria. This is possibly due to the fact that *Mayoviridae* are bacteriophages, *Reoviridae* concern a very wide range of

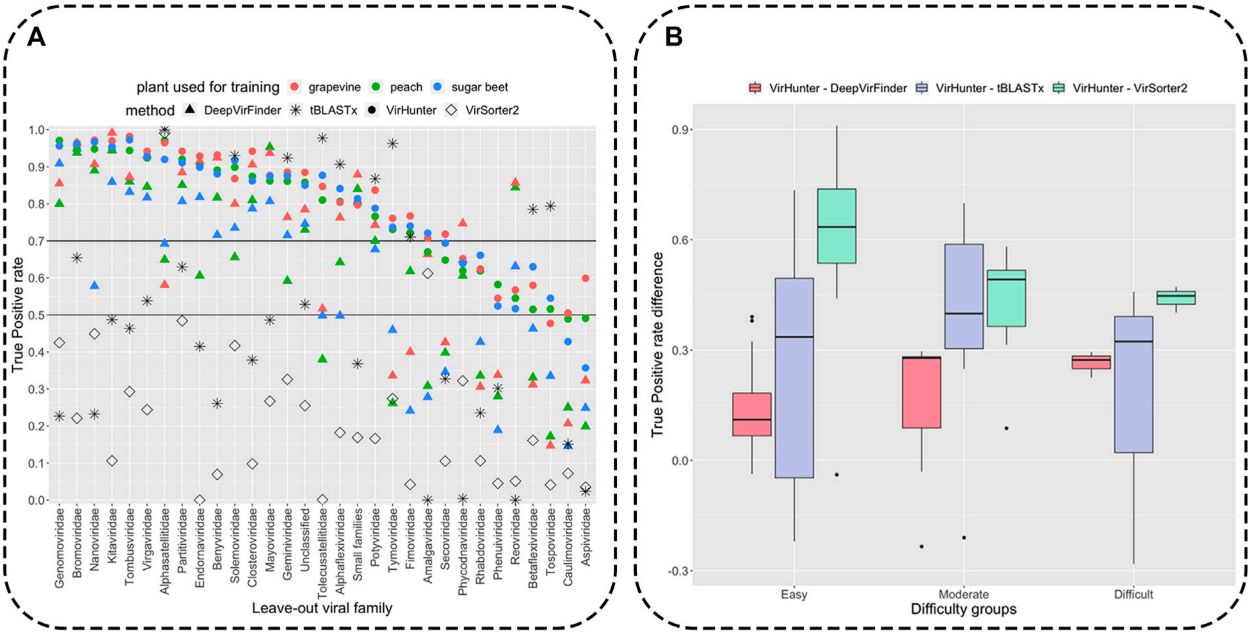


FIGURE 3 | Detection of novel viral fragments in the family leave-out setup. Panel **(A)**: Results for the percent of correctly classified fragments (out of 10, 000) with length $n = 1000$ from the corresponding left-aside families. VirHunter results are depicted by circles, tBLASTx by stars, DeepVirFinder by triangles, and VirSorter2 by diamonds. Black lines represent thresholds separating families into three difficulty groups for VirHunter as follows: easy to classify (minimum TP rate across the three plants > 0.7), difficult to classify (minimum TP rate < 0.5), and moderately difficult to classify (minimum TP rate between 0.5 and 0.7). Panel **(B)**: Differences in the True Positive rate between VirHunter, DeepVirFinder (red), tBLASTx (blue), and VirSorter2 (green).

hosts and present characteristics of bacteriophages [likely evolutionary relationship to the *Cystovirus* family of bacteriophage (Guglielmi et al., 2006)], while the *small families* contain a wide variety of viruses, and bacteriophages are one among them (*Mitoviridae*). This is to be counterbalanced by the fact that being trained only on plant and virus sequences due to the 2-class approach, DeepVirFinder systematically erroneously considers the majority of bacterial fragments as being viral (see **Supplementary Table S4**). As for the *Phycodnaviridae* family, it contains dsDNA viruses, which could potentially have contributed to the poorer performance of VirHunter relatively to DeepVirFinder for two of the host species. Altogether, VirHunter has shown consistently better capacity to detect novel viruses than DeepVirFinder.

Of note is also the difference in time requirement for training the VirHunter and DeepVirFinder models. On average, training a full model for one leave-out family for one plant host required 11 h for VirHunter (three CNNs, each for both fragment sizes 500 and 1000 – 6 CNNs in total—and the random forest) and 72 h for DeepVirFinder (four CNNs for fragment sizes 150, 300, 500, and 1000).

Compared to both VirHunter and DeepVirFinder, VirSorter2 has shown poorer performance in the family leave-out setup on all the families except two. Indeed, the TP rate was below 0.5 threshold for all families except for the *Amalgaviridae* and the *Alphasatellitidae*. For the former, VirSorter outperformed DeepVirFinder, while showing poorer results than VirHunter, while for latter it was the best performing method together with tBLASTx (see Panel A of **Figure 3**).

As shown in **Figure 3**, despite the reasonably permissive filtering criteria, tBLASTx shows best results comparable with VirHunter and for certain families exhibits particularly poor performance relative to the two machine learning methods. For the "easy to classify" families, the difference was mostly in favor of VirHunter, sometimes drastically (see for example, *Nanoviridae* and *Genomoviridae* in Panel A and the boxplot in Panel B). In seven cases, tBLASTx outperformed VirHunter, but this difference was mostly marginal (5.8% difference in TP rate on average), the outlier being *Tolecusatellitidae* and *Tymoviridae*, where the gain in favor of tBLASTx was the strongest. For the "moderately difficult to classify" families, VirHunter had a higher TP rate than tBLASTx in all cases. For the three "difficult to classify" families, even if VirHunter's performance was globally low, it still outperformed tBLASTx, with the notable exception of *Tospoviridae*. Altogether, VirHunter has shown consistently better results than that of tBLASTx, for which the TP rate was frequently below the threshold 0.5 (16 families out of 31).

As for the capacity to correctly classify bacterial fragments, VirHunter has shown a systematically high TP rate, ranging from 0.958 to 0.983, across all the leave-out experiments. As for plant fragments, the TP rate was also satisfactory, sugar beet TP from 0.950 to 0.961, grapevine TP from 0.983 to 0.991, and peach TP

TABLE 1 | VirHunter results for prediction of fragments from different plants. Classification results for three plant-specific models of $10,000$ fragments for length 1000 randomly drawn from three plants' reference genomes, from all viral sequences and bacteria are shown. In bold are predictions for the expected class.

Plant used for training	Plant used for testing	Predicted label		
		Plant	Virus	Bacteria
Peach	Peach	**0.988**	0.007	0.006
	Grapevine	**0.892**	0.064	0.044
	Sugar beet	**0.804**	0.113	0.083
	Virus	0.002	**0.996**	0.002
	Bacteria	0.005	0.017	**0.978**
Grapevine	Peach	**0.845**	0.106	0.005
	Grapevine	**0.986**	0.011	0.004
	Sugar beet	**0.78**	0.148	0.072
	Virus	0.002	**0.997**	0.002
	Bacteria	0.007	0.021	**0.973**
Sugar beet	Peach	**0.824**	0.132	0.045
	Grapevine	**0.878**	0.087	0.035
	Sugar beet	**0.956**	0.018	0.026
	Virus	0.002	**0.996**	0.002
	Bacteria	0.012	0.019	**0.969**

from 0.983 to 0.989 (see columns "Bacteria" and "Plant" in **Supplementary Table S2**).

Plant Fragments Are Accurately Classified When VirHunter Is Trained on Phylogenetically Close Plant Species

VirHunter was trained independently with $n = 1000$ for three selected plants (peach, grapevine, and sugar beet) and all the downloaded viruses and bacteria, generated as described in *Data Preprocessing*, yielding three models.

We cross-evaluated VirHunter's ability to correctly predict fragments from the plant absent in the training by sampling from the three studied plants, and $10,000$ random fragments with $n = 1000$ were selected. Those three plant test datasets were supplemented with two datasets with $n = 1000$, sampled randomly from all viral sequences and from bacteria, respectively.

As previously described (see *VirHunter Outperforms State-of-the art Tools on Family Leave-out Datasets*), plant fragments, coming from the same plant that the models were trained on, are consistently well classified for all the three models with the TP rate ranging between 0.95 ("sugar beet" model tested on random fragments from the sugar beet genome) and 0.99 (the "peach" model tested on random fragments from the peach genome) as shown in **Table 1**. When the plant host species used for training the model is reasonably phylogenetically close to the one of the test datasets, the impact on the TP rate is not very important. For example, the "peach" model tested on random fragments from the grapevine genome still produces the TP rate of 0.9, and the "grapevine" model tested on peach fragments gives the TP rate of 0.836. However, both these models generate a lower TP rate when tested on random fragments from the more phylogenetically distant sugar beet fragments, 0.827 and 0.781, for the "peach" and "grapevine" models, respectively. Similarly, the "sugar beet" model performs less well for both peach and grapevine random fragments, with TP rates of 0.854 and 0.887, respectively.

The three plants used for training models are phylogenetically distant from one another as they belong to different families, sugar beet belongs to the *Amaranthaceae* family, grapevine belongs to the *Vitaceae* family, and peach to the *Rosaceae* family; all the three are *eudicots*. Out of these three plants, sugar beet is the outlier. Peach and grapevine belong to the *Rosids* higher clade, while sugar beet belongs to the *Caryophyllids* higher clade. Given the phylogenetic distance, the lower bound of 0.78 for the true positive rate between these three plants is reasonable.

To evaluate how strongly the performance would be affected if the host of RNAseq dataset was to be from an even further phylogenetically removed plant (belonging to the *monocots*), we trained a model on the rice (*Oryza sativa*) dataset that belongs to *monocots* higher clade. As shown in the **Supplementary Table S7**, the performance drop was coherent with the increase of the phylogenetic distance (TP rate was 0.766, 0.759, and 0.702 for fragments from peach, grapevine, and sugar beet, respectively); however, the recall remained high for both viral and bacterial fragments. These results highlight that when the host of the RNAseq dataset is phylogenetically highly divergent from any of the plants used to train the available models, a new model for a phylogenetically closer plant has to be trained.

VirHunter Enables Classification of Long Mutated Viral Fragments

To evaluate the potential quality of VirHunter's predictions on contigs' classification, we randomly sampled 10,000 long fragments with $n \in [1500, 2000, 2500, 3000, 4500, 6000]$ from the whole virus dataset V. Furthermore, to better emulate contigs resulting from assembly of sequencing reads, we applied a point mutation rate $m \in [0, 0.05, 0.1, 0.15, 0.2]$ to these long fragments. Classification of the resulting mutated long fragments was performed using models trained for the three plants as described in *VirHunter Enables Classification of Long-Mutated Viral Fragments* and following the procedure for contig classification described in *Contig Classification*.

We observed that VirHunter generated highly accurate predictions for long viral fragments with 0 mutations and that across different fragment sizes (column "Mutation rate" 0 in **Supplementary Table S5**). The TP rate slowly decreased with the increase of the mutation rate: for example, the average TP rate across different fragment sizes with the mutation rate 0.2 was 0.885 for the "peach" model, 0.924 for the "grapevine" model, and 0.885 for the "sugar beet" model. Moreover, these results were consistent between the three plant host species used to build the models: the "peach" model's TP rate was 0.944 in average across different fragment lengths and mutation rates, the "grapevine" models' average TP rate was 0.960, and the "sugar beet" model's average TP rate was 0.936.

TABLE 2 | Performance of VirHunter, DeepVirFinder, and VirSorter2 on 12 RNAseq virome datasets. For each of the 12 datasets shown are the number of contigs that were annotated as being viral by experts and the number of contigs in the initial assembly with length >750. Columns "VirHunter," "DeepVirFinder," and "VirSorter2" show predictions run on these contigs by each method. Columns "# detected" show the total number of contigs detected as being viral by each of the two methods, and columns "detected ∩ annotated" indicates how many of these were previously identified by the curators. Finally, the "tBLASTx e-value < 10^{-10}" column indicates how many of "# detected" contigs align against viruses for VirHunter.

Dataset ID and plant origin		# Contig >750	# Contig annotated as viral	VirHunter			DeepVirFinder		VirSorter2	
				# detected	Detected ∩ annotated	tBLASTx hits (e-val < 10^{-10})	# detected	Detected ∩ annotated	# detected	Detected ∩ annotated
P1	Peach	1,009	2	35	2	14	45	2	10	1
P2	Peach	415	2	19	2	10	32	2	8	1
P3	Peach	685	2	23	2	10	49	2	7	1
G1	Grapevine	9,154	10	153	10	47	133	6	52	4
G2	Grapevine	17,024	10	178	10	40	131	9	117	6
G3	Grapevine	18,750	20	208	18	59	137	17	142	11
G4	Grapevine	4,332	15	95	14	32	81	11	24	4
G5	Grapevine	19,395	25	262	23	73	302	23	144	8
G6	Grapevine	2,932	15	70	14	30	86	13	26	12
S1	Sugar beet	6,082	11	236	10	48	335	11	28	6
S2	Sugar beet	8,902	16	277	16	49	419	16	37	7
S3	Sugar beet	6,912	11	203	11	51	307	11	21	4

VirHunter Uncovers Expected Novel and Known Viral Contigs in Virome

The capacity of VirHunter to detect novel viral contigs from real RNAseq-sequencing data was evaluated and compared to that of DeepVirFinder and VirSorter2. The 12 virome RNAseq datasets, sampled from peach, grapevine, and sugar beet (see **Supplementary Table S1**) were assembled as described in *Assembly of RNAseq Datasets and Annotation of Viral Contigs*. To imitate the novel virus discovery setting, we excluded from the virus dataset those viral species that were annotated as present in the studied plant viromes, and models for each plant species were trained accordingly for VirHunter and DeepVirFinder. For example, to train the "grapevine" model, all viral species present in samples from grapevine (**Supplementary Table S1** column "Present viruses") were deleted from the virus dataset. The same procedure was carried out for training the "peach" and "sugar beet" models. VirSorter2 pretrained models were used following the recommendations in Guo et al. (2021).

The assembled contigs >750 nt were analyzed by VirHunter, DeepVirFinder, and VirSorter2 (see **Table 2** and **Supplementary Table S8**). Importantly, VirHunter assigned a viral label to a lower number of contigs than DeepVirFinder in eight out of 12 datasets ("Viral contigs #" under VirHunter and DeepVirFinder columns). These are the contigs that have to undergo additional manual expert analysis. To better understand their nature, we aligned the contigs identified by VirHunter to the BLAST NCBI nucleotide database limited to "Viruses" taxonomic id as was performed for *Assembly of RNAseq Datasets and Annotation of Viral Contigs* analysis. Contigs getting at least one alignment with percent identity >0.5, length >50 amino acids, and e-value < $10^{-10}$are reported in the column "tBLASTx hits."

Moreover, for six datasets (P1, P2, P3, G4, S2, and S3) VirHunter and DeepVirFinder have correctly identified contigs that were previously annotated as viral. For four datasets (G1, G2, G3, and G6), VirHunter was able to discover additional 4, 3, 5, and 1 contigs, respectively. However, for two cases (G5 and S1), DeepVirFinder identified one more annotated contig relative to VirHunter. While VirSorter2 exhibited lower overprediction comparted to VirHunter and DeepVirFinder, its ability to correctly identify viral contigs was low, as it detected at best 60% of the expected viral contigs.

Remember that contigs annotated by experts were all removed from the virus dataset used for the training of VirHunter and DeepVirFinder, V. Consequently, strictly from the computational point of view, detection of these contigs as being viral can thus be considered as detection of novel viruses for those tools. Simple tBLASTx alignment of these expertly annotated contigs against V produced variable percent identity, which was as low as 32.4% for a contig from the G1 grapevine dataset and as high as 99% for a contig from the S1 sugar beet dataset (see **Supplementary Table S1**). According to the classification of Stobbe and Roossinck, (2014), discovery of these viruses could thus be assimilated in our setup with the discovery of "novel viruses from a known family" and potentially of "completely novel viruses."

Moreover, it is possible that at least some potentially novel viruses were missed during expert annotation and that the overprediction in columns "# detected" and "tBLASTx hits" (e-val < 10^{-10}) is lower in reality. Indeed, a large number of unidentified novel viruses have been recently shown to be present in public RNAseq datasets by Edgar et al. (2021), where the authors have identified 10^5 novel RNA viruses. Finally, of note is the considerable gain of time left for expert curation of contigs by approaches similar to that presented in *Assembly of RNAseq Datasets and Annotation of Viral Contigs*, given the numbers in the "# detected" column, where VirHunter has shown improvement over DeepVirFinder in eight out of 12 datasets.

DISCUSSION

High-throughput sequencing (HTS) is capable of broad virus detection for both known and unknown viruses in a variety of hosts and habitats. It has been successfully applied for novel virus discovery in many agricultural crops, leading to the current drive to apply this technology routinely for plant health diagnostics. For this, efficient and precise methods for HTS-based virus detection and discovery are essential.

RNA viruses are the most abundant pathogens infecting plants. However, RNA plant virus detection using HTS presents a number of challenges due to their genetic diversity, lack of conserved regions across viral species, short genome lengths, high mutation rate, and incomplete knowledge present in reference databases. To address this challenge, we developed a novel deep learning method, VirHunter, to detect novel and known plant viruses in RNAseq datasets.

VirHunter is particularly well-suited for the discovery of novel viruses as it was exemplified on 31 synthetic leave-out family datasets, where VirHunter systematically outperformed DeepVirFinder and VirSorter2, reference machine learning tools for virus detection. When compared with the standard tBLASTx approach, we have shown that for most (21 out of 31) leave-out families, VirHunter obtained a higher TP rate. In six cases, tBLASTx was slightly better (5.8% on average). However, there remained four cases where we have seen a much worse performance in VirHunter results. For these specific families, it can be noted that they are particularly well-suited to the alignment-based virus identification, for example, *Alphasatellitidae* viruses carry high sequence similarity to *Geminiviridae* (which was confirmed by the majority of tBLASTx hits).

We have shown that the 3-class classification design of VirHunter, accounting for possible bacterial contamination, was justified by evaluating how such contaminating contigs would be classified. Not surprisingly, VirHunter efficiently dealt with bacterial contamination, while DeepVirFinder classified bacteria mostly (65%) as viruses, which should have been "plants" if the goal is to identify viruses. We have also demonstrated that VirHunter is also perfectly suited for the detection of known divergent viruses, by evaluating classification accuracy on contigs with progressively increasing the mutation rate.

Note the fact that VirHunter is designed to be trained separately for a specific plant host species. However, classification of plant contigs still remained reasonable (minimum 0.78 TP rate) when we performed a cross-evaluation by classifying sequences coming from three phylogenetically distant plants (peach, grapevine, and sugar beet) by each of the three models. As expected, VirHunter performed better, when the plants it was trained and tested on were phylogenetically closer: grapevine and peach belong to the same *rosids* higher clade resulted in better mutual predictions, while sugar beet as an outgroup belonging to the *caryophyllids* higher clade has shown a relative drop in performance. All these three plants are *eudicots* (Pin 2012). When the model was trained on an even further phylogenetically distant plant, rice that belongs to *monocots* and tested on fragments from peach, grapevine, and sugar beet, the classification accuracy of VirHunter was expectedly lower. Together this implies that to classify contigs from an RNAseq experiment, using a pretrained model trained on the exact same plant species as the host of the experimental dataset is not mandatory, but it is preferable to use one trained on a phylogenetically close plant, ideally from the same family and at least belonging to the same *eudicots/monocots* higher clade. A possible avenue to explore in the future work is the feasibility of transfer learning (Eraslan et al., 2019), to enable fast on-demand retraining for a new plant or building a generalistic plant model.

Finally, we validated VirHunter's capacity to detect novel viruses on 12 newly acquired RNAseq datasets for peach, grapevine, and sugar beet. In these datasets, VirHunter detected at least 90% (73% for DeepVirFinder and 26% for VirSorter2) of all expert-annotated viral contigs, and in seven datasets it was 100%. Another contribution is the low rate of false positives generated by VirHunter, leaving from 19 to 277 contigs depending on the dataset to be inspected by an expert. These results indicate that VirHunter efficiently reduces the number of contigs requiring manual expert curation.

In conclusion, we have shown that VirHunter can be used to streamline the analyses of plant HTS-acquired viromes and is particularly well suited for the detection of novel viral contigs, in RNAseq datasets.

DATA AVAILABILITY STATEMENT

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/**Supplementary Material**.

AUTHOR CONTRIBUTIONS

MN and OG conceptualized the approach. MN and TC designed the study. MN, TC, and AM-C supervised the research. MN and GS contributed to the computational experimental design. GS implemented VirHunter. GS and MK performed genome assembly. TC, AM-C, and MK collected the samples and generated sequencing data. MK and TC performed data annotation. All the authors contributed to writing the manuscript.

FUNDING

This work was supported by the funding from Horizon 2020 Marie Skłodowska-Curie Actions Innovative Training Network (H2020 MSCA-ITN) project "INEXTVIR" (GA 813542).

REFERENCES

Abdelkareem, A. O., Khalil, M. I., Elaraby, M., Abbas, H., and Elbehery, A. H. A. (2018). "VirNet: Deep Attention Model for Viral Reads Identification," in Procs of the 2018 13th Intl Conf. on Computer Engineering and Systems (ICCES), Cairo, Egypt, 18-19 Dec. 2018, 623–626. doi:10.1109/icces.2018.8639400

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic Local Alignment Search Tool. *J. Mol. Biol.* 215 (3), 403–410. doi:10.1016/S0022-2836(05)80360-2

Buchfink, B., Xie, C., and Huson, D. H. (2015). Fast and Sensitive Protein Alignment Using DIAMOND. *Nat. Methods* 12, 59–60. doi:10.1038/nmeth.3176

Budziszewska, M., and Obrępalska-Stęplowska, A. (2018). The Role of the Chloroplast in the Replication of Positive-Sense Single-Stranded Plant RNA Viruses. *Front. Plant Sci.* 9, 1776. doi:10.3389/fpls.2018.01776

Candresse, T., Theil, S., Faure, C., and Marais, A. (2018). Determination of the Complete Genomic Sequence of Grapevine Virus H, a Novel Vitivirus Infecting grapevine. *Arch. Virol.* 163, 277–280. doi:10.1007/s00705-017-3587-7

Chang, S., Puryear, J., and Cairney, J. (1993). A Simple and Efficient Method for Isolating RNA from pine Trees. *Plant Mol. Biol. Rep.* 11, 113–116. doi:10.1007/BF02670468

de Vries, J. J. C., Brown, J. R., Fischer, N., Sidorov, I. A., Morfopoulou, S., Huang, J., et al. (2021). Benchmark of Thirteen Bioinformatic Pipelines for Metagenomic Virus Diagnostics Using Datasets from Clinical Samples. *J. Clin. Virol.* 141, 104908. doi:10.1016/j.jcv.2021.104908

Delgado, S., Navarro, B., Serra, P., Gentit, P., Cambra, M. Á., Chiumenti, M., et al. (2019). How Sequence Variants of a Plastid-Replicating Viroid with One Single Nucleotide Change Initiate Disease in its Natural Host. *RNA Biol.* 16 (7), 906–917. doi:10.1080/15476286.2019.1600396

Domingo, E. (2010). Mechanisms of Viral Emergence. *Vet. Res.* 41, 38. doi:10.1051/vetres/2010010

Edgar, R. C., Taylor, J., Lin, V., Altman, T., Barbera, P., Meleshko, D., et al. (2021). *Petabase-scale Sequence Alignment Catalyses Viral Discovery*. BioRxiv. 2020-08.

Eraslan, G., Avsec, Ž., Gagneur, J., and Theis, F. J. (2019). Deep Learning: New Computational Modelling Techniques for Genomics. *Nat. Rev. Genet.* 20, 389–403. doi:10.1038/s41576-019-0122-6

Grabherr, M. G., Haas, B. J., Yassour, M., Levin, J. Z., Thompson, D. A., Amit, I., et al. (2011). Full-length Transcriptome Assembly from RNA-Seq Data without a Reference Genome. *Nat. Biotechnol.* 29 (7), 644–652. doi:10.1038/nbt.1883

Guglielmi, K. M., Johnson, E. M., Stehle, T., and Dermody, T. S. (2006). Attachment and Cell Entry of Mammalian Orthoreovirus. *Curr. Top. Microbiol. Immunol.* 309, 1–38. doi:10.1007/3-540-30773-7_1

Guo, J., Bolduc, B., Zayed, A. A., Varsani, A., Dominguez-Huerta, G., and Delmont, T. O. (2021). VirSorter2: A Multi-Classifier, Expert-Guided Approach to Detect Diverse DNA and RNA Viruses. *Microb.* 9 (1), 1–13. doi:10.1186/s40168-020-00990-y

Jenkins, G. M., Rambaut, A., Pybus, O. G., and Holmes, E. C. (2002). Rates of Molecular Evolution in RNA Viruses: A Quantitative Phylogenetic Analysis. *J. Mol. Evol.* 54, 156–165. doi:10.1007/s00239-001-0064-3

Kim, D., Song, L., Breitwieser, F. P., and Salzberg, S. L. (2016). Centrifuge: Rapid and Sensitive Classification of Metagenomic Sequences. *Gen. Res.* 26 (12), 1721–1729. doi:10.1101/gr.210641.116

Kleiner, M., Hooper, L. V., and Duerkop, B. A. (2015). Evaluation of Methods to Purify Virus-like Particles for Metagenomic Sequencing of Intestinal Viromes. *BMC Genomics* 16 (1), 7–15. doi:10.1186/s12864-014-1207-4

Kutnjak, D., Tamisier, L., Adams, I., Boonham, N., Candresse, T., Chiumenti, M., et al. (2021). A Primer on the Analysis of High-Throughput Sequencing Data for Detection of Plant Viruses. *Microorganisms* 9, 841. doi:10.3390/microorganisms9040841

Lefeuvre, P., Martin, D. P., Elena, S. F., Shepherd, D. N., Roumagnac, P., and Varsani, A. (2019). Evolution and Ecology of Plant Viruses. *Nat. Rev. Microbiol.* 17, 632–644. doi:10.1038/s41579-019-0232-3

Maree, H. J., Fox, A., Al Rwahnih, M., Boonham, N., and Candresse, T. (2018). Application of HTS for Routine Plant Virus Diagnostics: State of the Art and Challenges. *Front. Plant Sci.* 9, 1082. doi:10.3389/fpls.2018.01082

Martín, S., and Elena, S. F. (2009). Application of Game Theory to the Interaction between Plant Viruses during Mixed Infections. *J. Gen. Virol.* 90, 2815–2820. doi:10.1099/vir.0.012351-0

Martinez-Hernandez, F., Fornas, O., Lluesma Gomez, M., Bolduc, B., de la Cruz Peña, M. J., Martínez, J. M., et al. (2017). Single-virus Genomics Reveals Hidden Cosmopolitan and Abundant Viruses. *Nat. Commun.* 8, 15892. doi:10.1038/ncomms15892

Massart, S., Candresse, T., Gil, J., Lacomme, C., Predajna, L., Ravnikar, M., et al. (2017). A Framework for the Evaluation of Biosecurity, Commercial, Regulatory, and Scientific Impacts of Plant Viruses and Viroids Identified by NGS Technologies. *Front. Microbiol.* 8, 45. doi:10.3389/fmicb.2017.00045

Massart, S., Chiumenti, M., De Jonghe, K., Glover, R., Haegeman, A., Koloniuk, I., et al. (2019). Virus Detection by High-Throughput Sequencing of Small RNAs: Large-Scale Performance Testing of Sequence Analysis Strategies. *Phytopathology* 109 (3), 488–497. doi:10.1094/PHYTO-02-18-0067-R

McFadden, G. I. (2001). Primary and Secondary Endosymbiosis and the Origin of Plastids. *J. Phycology* 37 (6), 951–959. doi:10.1046/j.1529-8817.2001.01126.x

Mehetre, G. T., Leo, V. V., Singh, G., Sorokan, A., Maksimov, I., Yadav, M. K., et al. (2021). Current Developments and Challenges in Plant Viral Diagnostics: A Systematic Review. *Viruses* 13, 412. doi:10.3390/v13030412

Menzel, P., Ng, K. L., and Krogh, A. (2016). Fast and Sensitive Taxonomic Classification for Metagenomics With Kaiju. *Nature Communications* 7 (1), 1–9. doi:10.1038/ncomms11257

Mokili, J. L., Rohwer, F., and Dutilh, B. E. (2012). Metagenomics and Future Perspectives in Virus Discovery. *Curr. Opin. Virol.* 2, 63–77. doi:10.1016/j.coviro.2011.12.004

Nayfach, S., Camargo, A. P., Schulz, F., Eloe-Fadrosh, E., Roux, S., and Kyrpides, N. C. (2021). CheckV Assesses the Quality and Completeness of Metagenome-Assembled Viral Genomes. *Nat. Biotechnol.* 39, 578–585. doi:10.1038/s41587-020-00774-7

Nurk, S., Meleshko, D., Korobeynikov, A., and Pevzner, P. A. (2017). metaSPAdes: a New Versatile Metagenomic Assembler. *Genome Res.* 27 (5), 824–834. doi:10.1101/gr.213959.116

Pin, P. A. (2012). *Life Cycle and Flowering Time Control in Beet.* PhD Thesis (Sweden, Umeå: Swedish University of Agricultural Sciences).

Ren, J., Ahlgren, N. A., Lu, Y. Y., Fuhrman, J. A., and Sun, F. (2017). VirFinder: A Novel k-mer Based Tool for Identifying Viral Sequences From Assembled Metagenomic Data. *Microb.* 5 (1), 1–20. doi:10.1186/s40168-017-0283-5

Ren, J., Song, K., DengAhlgren, C. N. A., Ahlgren, N. A., Fuhrman, J. A., Li, Y., et al. (2020). Identifying Viruses from Metagenomic Data Using Deep Learning. *Quant. Biol.* 8, 64–77. doi:10.1007/s40484-019-0187-4

Roossinck, M. J., Martin, D. P., and Roumagnac, P. (2015). Plant Virus Metagenomics: Advances in Virus Discovery. *Phytopathology* 105, 716–727. doi:10.1094/PHYTO-12-14-0356-RVW

Roossinck, M. J. (1997). Mechanisms of Plant Virus Evolution. *Annu. Rev. Phytopathol* 35, 191–209. doi:10.1146/annurev.phyto.35.1.191

Rott, M., Xiang, Y., Boyes, I., Belton, M., Saeed, H., Kesanakurti, P., et al. (2017). Application of Next Generation Sequencing for Diagnostic Testing of Tree Fruit Viruses and Viroids. *Plant Dis.* 101, 1489–1499. doi:10.1094/PDIS-03-17-0306-RE

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.867111/full#supplementary-material

Roux, S., Emerson, J. B., Eloe-Fadrosh, E. A., and Sullivan, M. B. (2017). Benchmarking Viromics: an In Silico Evaluation of Metagenome-Enabled Estimates of Viral Community Composition and Diversity. *PeerJ* 5, e3817. doi:10.7717/peerj.3817

Santos-Medellin, C., Zinke, L. A., Ter Horst, A. M., Gelardi, D. L., Parikh, S. J., and Emerson, J. B. (2021). Viromes Outperform Total Metagenomes in Revealing the Spatiotemporal Patterns of Agricultural Soil Viral Communities. *The ISME Journ* 15, 1–15. doi:10.1038/s41396-021-00897-y

Shrikumar, A., Greenside, P., and Kundaje, A. (2017). *Reverse-complement Parameter Sharing Improves Deep Learning Models for Genomics*. bioRxiv. 103663.

Stobbe, A. H., and Roossinck, M. J. (2014). Plant Virus Metagenomics: What We Know and Why We Need to Know More. *Front. Plant Sci.* 5, 150. doi:10.3389/fpls.2014.00150

Sutton, T. D. S., Clooney, A. G., Ryan, F. J., Ross, R. P., and Hill, C. (2019). Choice of Assembly Software Has a Critical Impact on Virome Characterisation. *Microbiome* 7 (1), 12–15. doi:10.1186/s40168-019-0626-5

Tampuu, A., Bzhalava, Z., Dillner, J., and Vicente, R. (2019). ViraMiner: Deep Learning on Raw DNA Sequences for Identifying Viral Genomes in Human Samples. *PLoS ONE* 14, e0222271. doi:10.1371/journal.pone.0222271

Villamor, D. E. V., Ho, T., Al Rwahnih, M., Martin, R. R., and Tzanetakis, I. E. (2019). High Throughput Sequencing for Plant Virus Detection and Discovery. *Phytopathology* 109 (5), 716–725. doi:10.1094/PHYTO-07-18-0257-RVW

Warwick-Dugdale, J., Solonenko, N., Moore, K., Chittick, L., Gregory, A. C., Allen, M. J., et al. (2019). Long-read Viral Metagenomics Captures Abundant and Microdiverse Viral Populations and Their Niche-Defining Genomic Islands. *PeerJ* 7, e6800. doi:10.7717/peerj.6800

Wood, D. E., Lu, J., and Langmead, B. (2019). Improved Metagenomic Analysis With Kraken 2. *Gen. Biol.* 20 (1), 1–13. doi:10.1186/s13059-019-1891-0

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Sukhorukov, Khalili, Gascuel, Candresse, Marais-Colombel and Nikolski. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 16 May 2022
doi: 10.3389/fbinf.2022.867386

Scalable Microbial Strain Inference in Metagenomic Data Using StrainFacts

Byron J. Smith[1,2†], *Xiangpeng Li*[3], *Zhou Jason Shi*[1,4], *Adam Abate*[3,4†] and *Katherine S. Pollard*[1,2,4*†]

[1]The Gladstone Institute of Data Science and Biotechnology, San Francisco, CA, United States, [2]Department of Epidemiology and Biostatistics, University of California, San Francisco, San Francisco, CA, United States, [3]Department of Bioengineering and Therapeutic Sciences, University of California, San Francisco, San Francisco, CA, United States, [4]Chan-Zuckerberg Biohub, San Francisco, CA, United States

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Guido Zampieri,
Università degli Studi di Padova, Italy
Matt Olm,
Stanford University, United States

***Correspondence:**
Katherine S. Pollard
katherine.pollard@gladstone.ucsf.edu

†ORCID:
Byron J. Smith
orcid.org/0000-0002-0182-404X
Adam Abate
orcid.org/0000-0001-9614-4831
Katherine S. Pollard
orcid.org/0000-0002-9870-6196

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *01 February 2022*
Accepted: *14 April 2022*
Published: *16 May 2022*

Citation:
Smith BJ, Li X, Shi ZJ, Abate A and Pollard KS (2022) Scalable Microbial Strain Inference in Metagenomic Data Using StrainFacts. Front. Bioinform. 2:867386. doi: 10.3389/fbinf.2022.867386

While genome databases are nearing a complete catalog of species commonly inhabiting the human gut, their representation of intraspecific diversity is lacking for all but the most abundant and frequently studied taxa. Statistical deconvolution of allele frequencies from shotgun metagenomic data into strain genotypes and relative abundances is a promising approach, but existing methods are limited by computational scalability. Here we introduce StrainFacts, a method for strain deconvolution that enables inference across tens of thousands of metagenomes. We harness a "fuzzy" genotype approximation that makes the underlying graphical model fully differentiable, unlike existing methods. This allows parameter estimates to be optimized with gradient-based methods, speeding up model fitting by two orders of magnitude. A GPU implementation provides additional scalability. Extensive simulations show that StrainFacts can perform strain inference on thousands of metagenomes and has comparable accuracy to more computationally intensive tools. We further validate our strain inferences using single-cell genomic sequencing from a human stool sample. Applying StrainFacts to a collection of more than 10,000 publicly available human stool metagenomes, we quantify patterns of strain diversity, biogeography, and linkage-disequilibrium that agree with and expand on what is known based on existing reference genomes. StrainFacts paves the way for large-scale biogeography and population genetic studies of microbiomes using metagenomic data.

Keywords: metagenomics, strains, microbiome, biogeography, population genetics, model-based inference

INTRODUCTION

Intra-specific variation in microbial traits are widespread and are biologically important in human associated microbiomes. Strains of a species may differ in their pathogenicity (Loman et al., 2013), antibiotic resistance (Shoemaker et al., 2001), impacts on drug metabolism (Haiser et al., 2014), and ability to utilize dietary components (Patrick et al., 2010; Ostrowski et al., 2022). Standard methods for analysis of complex microbial communities are limited to coarser taxonomic resolution due to their reliance on slowly evolving marker genes (Case et al., 2007-January) or on genome reference databases lacking diverse strain representation (Nayfach et al., 2020). Approaches that quantify microbiomes at the level of strains may better capture variation in microbial function (Albanese and Donati, 2017), provide insight into ecological and evolutionary processes (Garud and Pollard, 2019), and discover previously unknown microbial etiologies for disease (Yan et al., 2020).

Shotgun metagenomic data can in principle be used to track strains by looking for distinct patterns of alleles observed across single nucleotide polymorphisms (SNPs) within the species. Several tools have recently been developed that count the number of metagenomic reads containing alleles across SNP sites (Nayfach et al., 2016; Costea P. I. et al., 2017; Truong et al., 2017; Beghini et al., 2021; Olm et al., 2021; Shi et al., 2021). Comparisons of the resulting "metagenotypes" across samples has been used to track shared strains (Li et al., 2016; Olm et al., 2021), or to interrogate the biogeography (Costea PI. et al., 2017; Truong et al., 2017) and population genetics of species (Garud et al., 2019). The application of this approach is limited, however, by low sequencing coverage, which results in missing values at some SNP sites, and co-existing mixtures of strains, which introduce ambiguity about the taxonomic source of each metagenomic read.

One promising solution to these challenges is statistical strain deconvolution, which harnesses multiple metagenotypes (e.g., a collection of related samples) to simultaneously estimate the genotypes and relative abundances of strains across samples. Several tools have been developed that take this approach, including Lineage (O'Brien et al., 2014), Strain Finder (Smillie et al., 2018), DESMAN (Quince et al., 2017), and ConStrains (Luo et al., 2015). These methods have been used to track the transmission of inferred strains from donors' to recipients' microbiomes after fecal microbiota transplantation (FMT) (Smillie et al., 2018; Chu et al., 2021; Watson et al., 2021; Smith et al., 2022). The application of strain deconvolution has been limited, however, by the computational demands of existing methods, where runtimes scale poorly with increasing numbers of samples, latent strains, and SNPs considered. One reason for this poor scaling is the discreteness of alleles at each SNP, which has led existing methods to use expectation maximization algorithms to optimize model parameters (Smillie et al., 2018), or Markov chain Monte Carlo to sample from a posterior distribution (O'Brien et al., 2014; Luo et al., 2015; Quince et al., 2017).

Here we take a different approach, extending the strain deconvolution framework by relaxing the discreteness constraint and allowing genotypes to vary continuously between alleles. The use of this "fuzzy" genotype approximation makes our underlying model fully differentiable, and allows us to apply modern, gradient-based optimization algorithms to estimate strain genotypes and abundances. Here we show that the resulting tool, StrainFacts, can scale to tens of thousands of samples, hundreds of strains, and thousands of SNPs, opening the door to strain inference in large metagenome collections.

TABLE 1 | Symbols used to describe the StrainFacts model.

Symbols	Description
$i = 1, \ldots, N$	Index and number of samples
$s = 1, \ldots, S$	Index and number of strains
$g = 1, \ldots, G$	Index and number of SNP sites
y_{ig}, m_{ig}	Counts of reads with the alternative allele; the total count of both reference and alternative alleles at SNP g in sample i
p_{ig}	Alternative allele frequency at SNP g in sample i
γ_{sg}, $\vec{\gamma}_g$	Allele at SNP g in strain s; vector of alleles for all strains
π_{is}, $\vec{\pi}_i$	Relative abundance of strain s in sample i; vector of relative abundances for all strains
ε_i	Sequencing error rate in sample i
α	Concentration parameter of the BetaBinomial distribution
$\vec{\rho}$	Metacommunity strain composition
Y, M, P, Γ, Π	Matrices composed of the above elements

MATERIALS AND METHODS

A Fully Differentiable Probabilistic Model of Metagenotype Data

Metagenotypes

A metagenotype is represented as a count matrix of the number of reads with each allele at a set of SNP sites for a single species in each sample. This can be gathered directly from metagenomic data, for instance by aligning reads to a reference genome and counting the number of reads with each allele at SNP sites. In this study we use GT-Pro (Shi et al., 2021), which instead counts exact k-mers associated with known single nucleotide variants. Although the set of variants at a SNP may include any of the four bases, here we constrain metagenotypes to be biallelic: reference or alternative. For a large majority of SNPs, only two alleles are observed across reference genomes (Shi et al., 2021). Metagenotypes from multiple samples are subsequently combined into a 3-dimensional array.

Deconvolution of Metagenotype Data

StrainFacts is based on a generative, graphical model of biallelic metagenotype data (summarized in **Supplementary Figure S1**) which describes the allele frequencies at each SNP site in each sample (p_{ig} for sample i and SNP g) as the product of the relative abundance of strains ($\vec{\pi}_i$) and their genotypes, γ_{sg}, where 0 indicates the reference and one indicates the alternative allele for strain s. This functional relationship is therefore $p_{ig} = \sum_s \gamma_{sg} \times \pi_{is}$, In matrix form, equivalently, we notate this as $P = \Gamma\Pi$ (**Table 1**).

The crux of strain deconvolution is taking noisy observations of P—based on the observed alternative allele counts Y and total counts M obtained from metagenotypes across multiple samples—and determining suitable matrices Γ and Π. This notation highlights parallels to non-negative matrix factorization (NMF). Like NMF, given a choice of loss function, L, this inference task can be transformed into a constrained optimization problem, where $\underset{\Pi,\Gamma}{\arg\min}\, L(\Pi, \Gamma | Y)$ is a scientifically useful estimate of these two unobserved matrices. We take the approach of explicitly modeling the stochasticity of observed metagenotypes, placing priors on Π and Γ, and taking the resulting posterior probability as the loss function. This "maximum a posteriori" (MAP) approach has also been applied to NMF (Schmidt et al., 2009). However, unlike NMF, where the key constraint is that all matrices are non-negative, the metagenotype deconvolution model also constrains the elements of P and Γ to lie in the closed interval $[0, 1]$, and the rows of Π are are "on the $(s - 1)$-simplex", i.e. they sum to one.

Fuzzy Genotypes and the Shifted-Scaled Dirichlet Distribution

StrainFacts does *not* constrain the elements of Γ to be discrete—i.e. in the set $\{0, 1\}$ for biallelic sites—in contrast to prior tools: DESMAN (Quince et al., 2017), Lineage (O'Brien et al., 2014), and Strain Finder's (Smillie et al., 2018) exhaustive search. Instead, we allow genotypes to vary continuously in the open interval between fully reference (0) and fully alternative (1). The use of fuzzy-genotypes serves a key purpose: by replacing the only discrete parameter with a continuous approximation, our posterior function becomes fully differentiable, and therefore amenable to efficient, gradient-based optimization. When not using the exhaustive search strategy, Strain Finder also treats genotypes as continuous to accelerate inference, but these are discretized after each iteration. We show below that inference with StrainFacts is faster than with Strain Finder.

Since true genotypes are in fact discrete, we place a prior on the elements of Γ that pushes estimates towards zero or one and away from intermediate—ambiguous—values. Similarly, we put a hierarchical prior on Π that regularizes estimates towards lower strain heterogeneity within samples, as well as less strain diversity across samples. This makes strain inferences more parsimonious and interpretable. We harness the same family of probability distributions, the shifted-scaled Dirichlet distribution (SSD) (Monti et al., 2011), for all three goals. We briefly describe our rationale and parameterization of the SSD distribution in the **Supplementary Methods**.

For each element of Γ we set the prior as $(\gamma, 1 - \gamma) \sim SSD(1, 1, \frac{1}{\gamma^*})$. (Note that we trivially transform the 1-simplex valued $(\gamma, 1 - \gamma)$ to the unit interval by dropping the second element.) Smaller values of the hyperparameter γ^* correspond to more sparsity in Γ. We put a hierarchical prior on Π, with the rows subject to the prior $\vec{\pi}_i \sim SSD(1, \vec{\rho}, \frac{1}{\pi^*})$ given a "metacommunity" hyperprior $\vec{\rho} \sim SSD(1, 1, \frac{1}{\rho^*})$, reflecting the abundance of strains across all samples. Decreasing the values of γ^*, ρ^*, and π^* increases the strength of regularization imposed by the respective priors.

Model Specification

The underlying allele frequencies P are not directly observed due to sequencing error, and we include a measurement process in our model. We assume that the true allele is replaced with a random allele at a rate ε_i for all SNP sites g in sample i: $\tilde{p}_{ig} = p_{ig}(1 - \varepsilon_i/2) + (1 - p_{ig})(\varepsilon_i/2)$. Given the total counts, M, we then model the observed alternative allele counts, Y, with the Beta-Binomial likelihood, parameterized with $\tilde{P}$ and one additional parameter—α^*—controlling count overdispersion relative to the Binomial model.

To summarize, our model is as follows (in random variable notation; see **Supplementary Figure S1** for a plate diagram):

$$
\begin{aligned}
y_{ig} &\sim \mathrm{BetaBinom}\left(\tilde{p}_{ig}, \alpha^* \,\middle|\, m_{ig}\right)\\
\tilde{p}_{ig} &= p_{ig}(1 - \varepsilon_i/2) + \left(1 - p_{ig}\right)(\varepsilon_i/2)\\
p_{ig} &= \sum_s \pi_{is}\gamma_{sg}\\
\left(\gamma_{sg}, 1 - \gamma_{sg}\right) &\sim \mathrm{SSD}\left(\mathbf{1}, \mathbf{1}, \frac{1}{\gamma^*}\right)\\
\vec{\pi}_i &\sim \mathrm{SSD}\left(\mathbf{1}, \vec{\rho}, \frac{1}{\pi^*}\right)\\
\vec{\rho} &\sim \mathrm{SSD}\left(\mathbf{1}, \mathbf{1}, \frac{1}{\rho^*}\right)\\
\varepsilon &\sim \mathrm{Beta}\left(\varepsilon_a^*, \frac{\varepsilon_a^*}{\varepsilon_b^*}\right)
\end{aligned}
$$

Model Fitting

StrainFacts takes a MAP-based approach to inference on this model, using gradient-based methods to find parameter values that maximize the posterior probability of our model conditioned on the observed counts. We rely heavily on the probabilistic programming framework Pyro (Bingham et al., 2019), which is built on the PyTorch library (Paszke et al., 2019) for numerical methods.

Initial values for Γ and Π are selected using NMF, and all other parameters are initialized randomly (**Supplementary Methods**). In order to promote global convergence, we take a prior annealing approach (**Supplementary Methods**). While it is impossible to know in practice if we converge to a global optimum, we find that this procedure often leads to accurate estimates without the need for replicate fits from independent initializations.

Simulation and Evaluation

Metagenotype data was simulated in order to enable direct performance benchmarking against ground-truth genotypes and strain compositions. For each independent simulation, discrete genotypes of length G for S strains were sampled as $S \times G$ independent draws from a symmetric Bernoulli distribution. The composition of strains in each of N samples were generated as independent draws from a Dirichlet distribution over S components having a symmetric concentration parameter of 0.4. Per-sample allele frequencies were generated as the product of the genotypes and the strain-composition matrices. Sequence error was set to $\varepsilon = 0.01$ for all samples. Finally metagenotypes at each SNP site were drawn from a $\mathrm{Binomial}(m, \tilde{p}_{ig})$ distribution, with a sequencing depth of $m = 10$ across all sites.

Estimates were evaluated against the simulated ground truth using five different measures of error (see Results).

Metagenotypes and Reference Genomes

We applied StrainFacts to data from two previously compiled human microbiome metagenomic datasets: stool samples from a fecal microbiota transplantation (FMT) study described in (Smith et al., 2022, BioProject PRJNA737472) and 20,550 metagenomes from a meta-analysis of publicly available data in (Shi et al., 2021, various accessions). As described in that publication, metagenotypes for gut prokaryotic species were tallied using GT-Pro version 1.0.1 with the default database, which includes up to 1,000 of the highest quality genomes for each species from the Unified Human Gastrointestinal Genome (UHGG) V1.0 (Almeida et al., 2021). This includes both cultured isolates and high-quality metagenomic assemblies. This same database was used as a reference set to which we compared our inferred genotypes. Estimated genomic distances between SNPs were based on the UHGG representative genome.

We describe detailed results for *Escherichia coli* (id: 102506, MGYG-HGUT-02506), *Agathobacter rectalis* (id: 102492, MGYG-HGUT-02492), *Methanobrevibacter smithii* (id: 102163, MGYG-HGUT-02163), and CAG-279 sp1 (id: 102556, MGYG-HGUT-02556). These were selected to demonstrate application of StrainFacts to prevalent gram-positive and gram-negative bacteria in the human gut, the most prevalent archaeon, as well as an unnamed, uncultured, and largely unstudied species. We also describe detailed results for *Streptococcus thermophilus* (GT-Pro species id: 104345, representative UHGG genome: MGYG-HGUT-04345), selected for its high diversity in one sample of our single-cell sequencing validation.

Single-Cell Genome Sequencing

Of the 159 samples with metagenomes described in the FMT study, we selected two samples for single-cell genomics (which we refer to as the "focal samples"). These samples were obtained from two different study subjects; one is a baseline sample and the other was collected after several weeks of FMT doses as described in (Smith et al., 2022). A full description of the single-cell genomics pipeline is included in the **Supplementary Methods**, and will be briefly summarized here. For each of the focal samples, microbial cells were isolated from whole feces by homogenization in phosphate buffered saline, 50 μM filter-based removal of large fecal particles, and density gradient separation. After isolating and thoroughly washing the density layer corresponding to the microbiota, this cell suspension was mixed with polyacrylamide precursor solution, and emulsified with a hydrofluoric oil. Aqueous droplets in oil were allowed to gellate before separating the resulting beads from the oil phase and washing. Beads were size selected to between 5 and 25 μM, with the goal of enriching for those encapsulated a single microbial cell. Cell lysis was carried out inside the hydrogel beads by incubating with zymolyase, lysostaphin, mutanolysin, and lysozyme. After lysis, proteins were digested with proteinase K, before thoroughly washing the beads. Tn5 tagmentation and barcode PCR were carried out using the MissionBio Tapestri microfluidics DNA workflow with minor modifications. After amplification, the emulsion was broken and the aqueous phase containing the barcoded amplicons was used for sequencing library preparation with Nextera primers including P5 and P7 sequences followed by Ampure XP bead purification. Libraries were sequenced by Novogene on an Illumina NovaSeq 6000, BioProject PRJNA737472.

Demultiplexed sequence data for each droplet barcode were independently processed with GT-Pro identically to metagenomic sequences. For each barcode, GT-Pro allele counts for a given species were assumed to be representative of a single strain of that species. Horizontal coverage was calculated as the fraction of GT-Pro positions with ≥2 reads, unlike metagenotypes where ≥1 read was used to calculate horizontal coverage. These single-cell genotypes (SCGs) were filtered to those with > 1% horizontal coverage over SNP sites, leaving 87 species with at least one SCG from either of the two focal samples. During analysis, a number of SCGs were found to have nearly identical patterns of horizontal coverage. These may have been formed by merging of droplets during barcoding PCR, which could have resulted in multiple barcodes in the same amplification. To reduce the impact of this artifact, allele counts from multiple SCGs were summed by complete-linkage, agglomerative clustering based on their depth profiles across SNP sites, at a 0.3 cosine dissimilarity threshold.

Computational Analysis

Metagenotype Filtering

From GT-Pro metagenotypes, we extracted allele counts for select species and removed SNPs that had < 5% occurance of the minor allele across samples. Species with more than 5,000 SNPs after filtering, were randomly down-sampled without replacement to this number of sites. Samples with less than 5% horizontal coverage were also filtered out.

Strain Inference

For all analyses, StrainFacts was run with the following hyperparameters $\rho^* = 0.5$, $\pi^* = 0.3$, $\gamma^* = 10^{-10}$, $\alpha^* = 10$, $\varepsilon_a^* = 1.5$, $\varepsilon_b^* = 0.01$. The learning rate was initially set to 0.05. Prior annealing was applied to both Γ and $\vec{\rho}$ by setting γ^* and ρ^* to 1.0 and 5, respectively, for the first 2,000 steps of gradient descent, before exponentially relaxing these hyperparameters to their final values over the next 8,000 steps. After this annealing period, when parameters had not improved for 100 steps, the learning rate was halved until it had fallen below 10^{-6}, at which point we considered parameters to have converged. These hyperparameters were selected through manual optimization and we found that they gave reasonable performance across the diverse datasets in this study.

The number of strains parameterized by our model was chosen as follows. For comparisons to SCGs, the number of strains was set at 30% of the number of samples—e.g. 33 strains were parameterized for *S. thermophilus* because metagenotypes from 109 samples remained after coverage filtering. For the analysis of thousands of samples described in (Shi et al., 2021), we parameterized our model with 200 strains and increased the numerical precision from 32 to 64 bits. After strain inference using the 5,000 subsampled SNPs, full-length genotypes were estimated post-hoc by conditioning on our estimate of Π and iteratively refitting subsets of all SNPs (**Supplementary Methods**).

For computational reproducibility we set fixed seeds for random number generators: 0 for all analyses where we only report one estimate, and 0, 1, 2, 3, and 4 for the five replicate estimates described for simulated datasets. Strain Finder was run with flags *--dtol 1 --ntol 2 --max_reps 1*. We did not use *--exhaustive*, Strain Finder's exhaustive genotype search strategy, as it is much more computationally intensive.

Genotype Comparisons

Inferred fuzzy genotypes were discretized to zero or one for downstream analyses. SNP sites without coverage were treated as unobserved. Distances between genotypes were calculated as the masked, normalized Hamming distance, the fraction of alleles that do not agree, ignoring unobserved SNPs. Similarly, SCG genotypes and the metagenotype consensus were discretized to the majority allele. In comparing the distance between SCGs and these inferred genotypes sites missing from either the SCG or the metagenotype were treated as unobserved. Metagenotype entropy, a proxy for strain heterogeneity, was calculated for each sample as the depth weighted mean allele frequency entropy:

$$\frac{1}{\sum_g m_{ig}} \sum_g -m_{ig}\left[\hat{p}_{ig}\log_2\left(\hat{p}_{ig}\right) + \left(1-\hat{p}_{ig}\right)\log_2\left(1-\hat{p}_{ig}\right)\right]$$

where $\hat{p}_{ig}$ is the observed alternative allele frequency.

Where indicated, we dereplicated highly similar strains by applying average-neighbor agglomerative clustering at a 0.05 genotype distance threshold. Groups of these highly similar strains were replaced with a single composite strain with a genotype derived from the majority allele at each SNP site and assigned the sum of strain relative abundances in each sample. Subsequent co-clustering of these dereplicated inferred and reference strains was done in the same way, but at a 0.15 genotype distance threshold. After co-clustering, to test for enrichment of strains in "shared" clusters, we permuted cluster labels and re-tallied the total number of strains found in clusters with both inferred and reference strains. Likewise, to test for enrichment of "inferred-only" clusters we tallied the total number of strains found in clusters without reference strains after this shuffling. By repeating the permutation 9,999 times, we arrived at an empirical null distribution to which we compared our true, observed values to calculate a p-value.

Pairwise linkage disequilibrium (LD) was calculated as the squared Pearson correlation coefficient across genotypes of dereplicated strains. Genome-wide 90th percentile LD, was calculated from a random sample of 20,000 or, if fewer, all available SNP positions. To calculate the 90th percentile LD profile, SNP pairs were binned at either an exact genomic distance or within a window of distances, as indicated. In order to encourage a smooth distance-LD relationship, windows at larger pairwise-distance spanned a larger range. Specifically the ith window covers the span $\lfloor 10^{(i-1)/c} \rfloor, \lfloor 10^{i/c} \rfloor$ where $c = 30$ so that 120 windows span the full range $[1, 10^4)$.

Software and Code Availability

StrainFacts is implemented in Python3 and is available at https://github.com/bsmith89/StrainFacts and v0.1 was used for all results reported here. Strain Finder was not originally designed to take a random seed argument, necessitating minor modifications to the code. Similarly, we made several modifications to the MixtureS (Li et al., 2021) code allowing us to run it directly on simulated metagenotypes and compare the results to StrainFacts and Strain Finder outputs. Patch files describing each set of changes, as well as all other code and metadata needed to re-run our analyses are available at https://doi.org/10.5281/zenodo.5942586. For reproducibility, analyses were performed using Snakemake (Mölder et al., 2021) and with a Singularity container (Kurtzer et al., 2017) that can be obtained at https://hub.docker.com/repository/docker/bsmith89/compbio.

Runtime and Memory Benchmarking

Runtimes were determined using the Snakemake *benchmark:* directive, and memory requirements using the GNU time utility, version 1.8 with all benchmarks run on the Wynton compute cluster at the University of California, San Francisco. Across strain numbers and replicates, maximum memory usage for models with 10,000 samples and 1,000 SNPs was, counterintuitively, less than for smaller models, likely because portions of runtime data were "swapped" to disk instead of staying in RAM. We therefore excluded data for these largest models from our statistical analysis of memory requirements.

RESULTS

Scaling Strain Inference to Hundreds of Genotypes in Thousands of Samples

Inferring the genotypes and relative abundance of strains in large metagenome databases requires a deconvolution tool that can scale to metagenotypes with thousands of SNPs in tens-of-thousands of samples, while simultaneously tracking hundreds of microbial strains. To accomplish this we developed StrainFacts, harnessing fuzzy genotypes to accelerate inference on large datasets. We evaluated the practical scalability of the StrainFacts algorithm by applying it to simulated datasets of increasing size, and comparing its time and memory requirements to Strain Finder, a previously described method for strain inference. While several tools have been developed to perform strain deconvolution (e.g. Lineage O'Brien et al., 2014; and DESMAN Quince et al., 2017), Strain Finder's model and approach to inference are the most similar to StrainFacts. We therefore selected it for comparison in order to directly assess the value of fuzzy genotypes.

We simulated five replicate metagenotypes for 120 underlying strains in 400 samples, and 250 SNPs, and then applied both StrainFacts and Strain Finder to these data parameterizing them with 120 strains. Both tools use random initializations, which can result in convergence to different optima. We therefore benchmarked runtimes for five independent initializations on each dataset, resulting in 25 total runs for each tool. In this setting, the median runtime for StrainFacts was just 17.2 min, while Strain Finder required a median of 6.4 h. When run on a GPU instead of CPU, StrainFacts was able to fit these data in a median of just 5.1 min.

Since the correct strain number is not known a priori in real-world applications, existing strain inference tools need to be parameterized across a range of plausible strain counts, a step that significantly impacts runtime. To assess performance in this

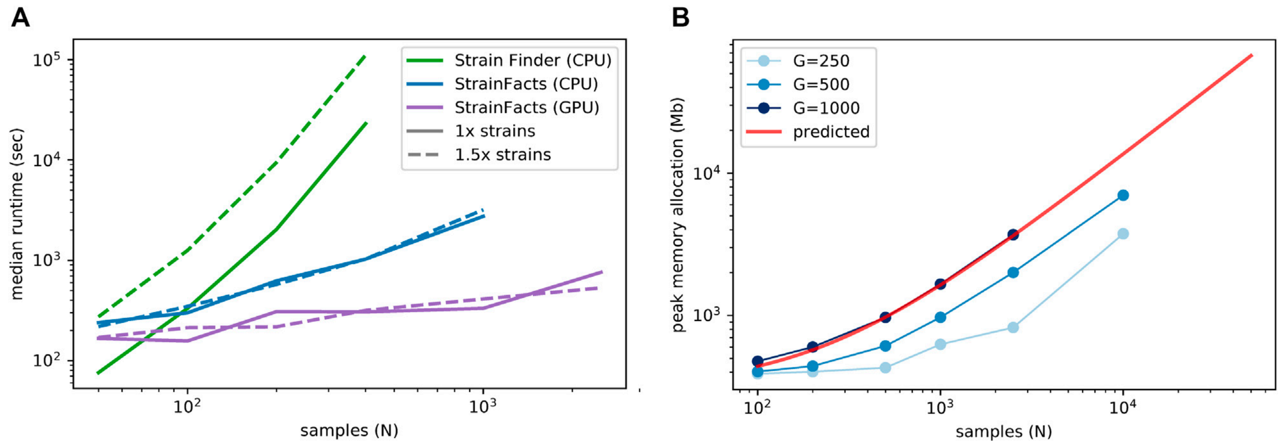


FIGURE 1 | Computational scalability of strain inference on simulated data. **(A)** Runtime (in seconds, log scale) is plotted at a range of sample counts for both Strain Finder and StrainFacts, as well for the latter with GPU acceleration. Throughout, 250 SNPs are considered, and simulated strains are fixed at a 1:5 ratio with samples. Models are specified with this same number of strains ("1x strains", solid lines) or 50% more ("1.5x strains", dashed lines). Median of 25 simulation runs is shown. **(B)** Maximum memory allocation in a model with 100 strains is plotted for StrainFacts models across a range of sample counts (N) and SNP counts (G, line shade). Median of nine replicate runs is shown. Maximum memory requirements are extrapolated to higher numbers of samples for a model with 1,000 SNPs (red line). A version of this panel that includes a range of strain counts is included as **Supplementary Figure S2**.

setting, we also fit versions of each model with 50% more strains than the ground-truth, here referred to as the "1.5x parameterization" in contrast to the 1x parameterization already described. In this setting, StrainFacts' performance advantage was even more pronounced, running in a median of 17.1 min and just 5.3 min on GPU, while Strain Finder required 30.8 h. Given the speed of StrainFacts, we were able to fit an even larger simulation with 2,500 samples and 500 strains. On a GPU, this took a median of 12.6 min with the 1x parameterization and, surprisingly, just 8.9 min with the 1.5x parameterization. We did not attempt to run Strain Finder on this dataset.

We next examined runtime scaling across a range of sample counts between 50 and 2,500. We applied Strain Finder and StrainFacts (both CPU and GPU) to simulated metagenotypes with 250 SNPs, and a fixed 1:5 ratio of strains to samples. Median runtimes for each tool at both the 1x and 1.5x parameterization demonstrate a substantially slower increase for StrainFacts as model size increases (**Figure 1A**). Strain Finder was faster than StrainFacts on the 1x parameterization of a small simulation with 50 samples and 10 strains: 1.3 min median runtime versus 4 min for StrainFacts on a CPU and 2.8 min on a GPU. However, StrainFacts had faster median runtimes on all other datasets.

Given the good runtime scaling properties of StrainFacts, we next asked if computer memory constraints would limit its applicability to the largest datasets (**Figure 1A**). A model fitting 10,000 samples, 400 strains, and 500 SNPs had a maximum memory allocation of 7.7 GB, indicating that StrainFacts' memory requirements are satisfied on most contemporary CPU or GPU hardware and opening the door to even larger models. Using ordinary least squares, we fit the observed memory requirements to the theoretical, asymptomatic expectations, $\mathcal{O}(NS + NG + SG)$, resulting in a regression R^2 of 0.997. We then used this empirical relationship to extrapolate for even larger models (**Figure 1B**), estimating that for a model of 400 strains and 1,000 SNPs, 32 GB of memory would be able to simultaneously perform strain inference for more than 22,000 samples. This means StrainFacts can realistically analyse tens of thousands of samples on commercial GPUs.

StrainFacts Accurately Reconstructs Genotypes and Population Structure

We next set out to evaluate the accuracy of StrainFacts and to compare it to Strain Finder. We simulated 250 SNPs for 40 strains, generating metagenotypes across 200 samples. For both tools, we specified a model with the true number of strains, fit the model to this data, and compared inferences to the simulated ground-truth. For each of five replicate simulations we performed inference with five independent initializations, thereby gathering 25 inferences for each tool. As in (Smillie et al., 2018), we use the weighted UniFrac distance (Lozupone et al., 2007) as an integrated summary of both genotype and relative abundance error. By this index, StrainFacts and Strain Finder performed similarly well when applied to the simulated data (**Figure 2A**). We repeated this analysis with the 1.5x parameterization to assess the robustness of inferences to model misspecification, finding that both tools maintained similar performance to the 1x parameterization. By comparison, considering too few strains (the 0.8x parameterization, fitting 32 strains) degraded performance dramatically for both tools, with StrainFacts performing slightly better. Thus, we conclude based on UniFrac distance that StrainFacts is as accurate as Strain Finder and that both models are robust to specifying too many strains.

To further probe accuracy, we quantified the performance of StrainFacts and Strain Finder with several other measures. First, we evaluated pairwise comparisons of strain composition by calculating the mean absolute error of pairwise Bray-Curtis dissimilarities (**Figure 2B**). While, with the 1x parameterization, Strain Finder slightly outperformed

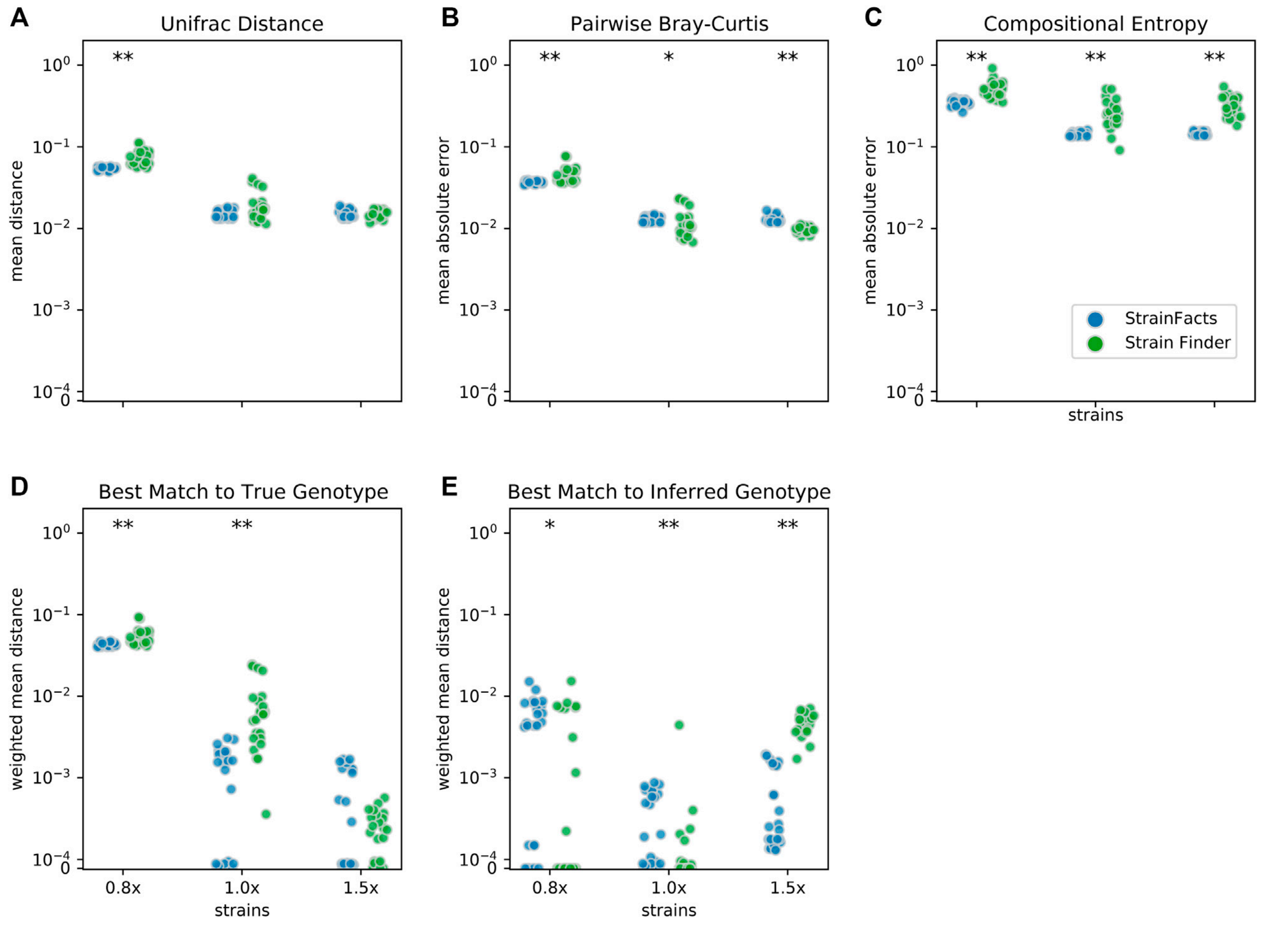


FIGURE 2 | Accuracy of strain inference on simulated data. Performance of StrainFacts and Strain Finder are compared across five distinct accuracy indices, with lower scores reflecting better performance on each index. Simulated data had 200 samples, 40 underlying strains, and 250 SNPs. For each tool, 32, 40, and 60 strain models were parameterized ("0.8x", "1x", and "1.5x" respectively), and every model was fit with five independent initializations to each simulation. All 25 estimates for each tool-parameterization combination are shown. Scores reflect **(A)** mean Unifrac distance between simulated and inferred strain compositions, **(B)** mean absolute difference between all-by-all pairwise Bray-Curtis dissimilarities calculated on simulated versus inferred strain compositions, **(C)** mean absolute difference in Shannon entropy calculated on simulated versus inferred strain compositions, **(D)** abundance weighted mean Hamming distance from each ground-truth strain to its best-match inferred genotype, and **(E)** the reverse: abundance weighted mean Hamming distance from each inferred strain to its best-match true genotype. Markers at the top of each panel indicate a statistical difference between tools at a $p < 0.05$ (*) or $p < 0.001$ (**) significance threshold by Wilcoxon signed-rank test. A version of this figure that includes accuracy comparisons to MixtureS is included as **Supplementary Figure S3**.

StrainFacts on this index, the magnitude of the difference was small. This suggests that StrainFacts can be used for applications in microbial ecology that rely on measurements of beta-diversity.

Ideally, inferences should conform to Occam's razor, estimating "as few strains as possible, but no fewer". Unfortunately, Bray-Curtis error is not sensitive to the splitting or merging of co-abundant strains and UniFrac error is not sensitive to the splitting or merging of strains with very similar genotypes. To overcome this limitation, we calculated the mean absolute error of the Shannon entropy of the inferred strain composition for each sample (**Figure 2C**). This score quantifies how accurately inferences reflect within-sample strain heterogeneity. StrainFacts performed substantially better on this score than Strain Finder for all three parameterizations, indicating more accurate estimation of strain heterogeneity.

Finally, we assessed the quality of genotypes reconstructed by StrainFacts compared to Strain Finder using the abundance weighted mean Hamming distance. For each ground-truth genotype, normalized Hamming distance is computed based on the best matching inferred genotype (**Figure 2D**), then summarized as the mean weighted by the true strain abundance across all samples. We assessed the reverse as well: the abundance weighted mean, best-match Hamming distance for each inferred genotype among the ground-truth genotypes (**Figure 2E**). These two scores can be interpreted as answers to the distinct questions "how well were the true genotypes recovered?" and "how well do the inferred genotypes reflect the truth?", respectively. While StrainFacts and Strain Finder performed similarly on these indexes—which tool had higher accuracy varied by score and parameterization—StrainFacts' accuracy

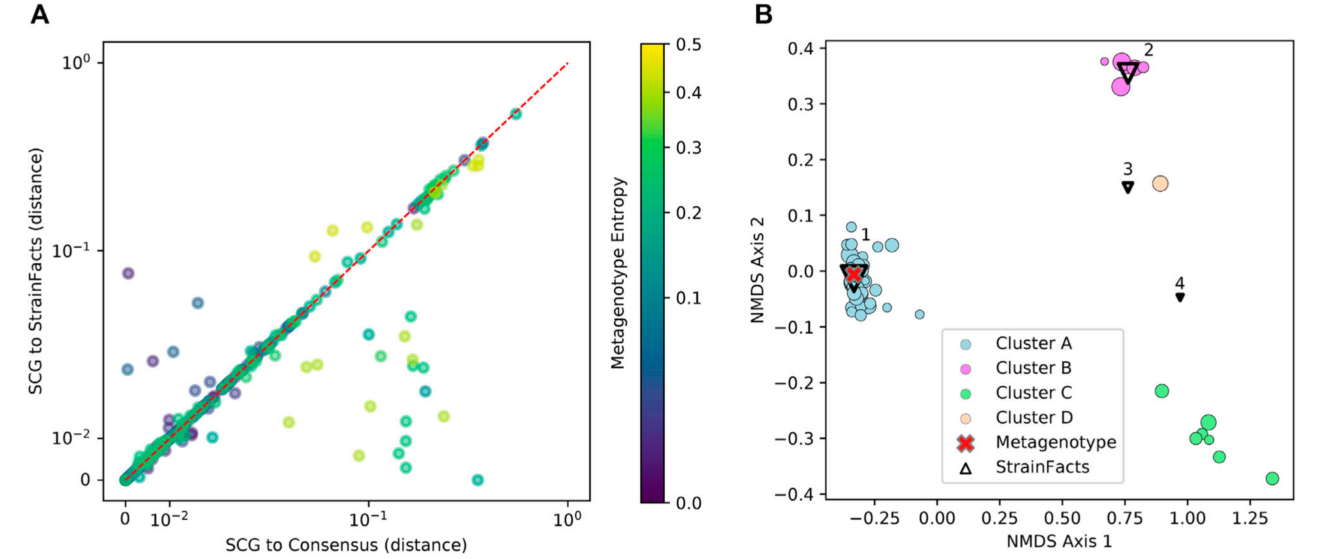


FIGURE 3 | Inferred strains reflect genotypes from a single-cell sequencing experiment. **(A)** Distance between observed SCGs and StrainFacts inferences (X-axis) versus consensus genotypes (Y-axis). Points below and to the right of the red dotted line reflecting an improvement of our method over the consensus, based on the normalized, best-match Hamming distance. Each dot represents an individual SCG reflecting a putative genotype found in the analysed samples. SCGs from all species found in either of the focal samples are represented, and marker colors reflect the metagenotype entropy of that species in the relevant focal sample, a proxy for the potential strain diversity represented. Axes are on a "symmetric" log scale, with linear placement of values below 10^{-2}. **(B)** A non-metric multidimensional scaling ordination of 68 SCGs and inferred genotypes for one species, S. thermophilus, with notably high strain diversity in one of the two focal samples. Circles represent SCGs, are colored by their assignment to one of four identified clusters, and larger markers indicate greater horizontal coverage. Triangles represent StrainFacts genotypes inferred to be at greater than 1% relative abundance, and larger markers reflect a higher inferred relative abundance. The red cross represents the consensus metagenotype of the focal sample.

was more stable across the 1x and 1.5x parameterizations. It should be noted that since strain genotypes are only inferred for SNP sites, the genome-wide genotype reconstruction error (which includes invariant sites) will likely be much lower than this Hamming distance. We examine the relationship between genotype distances and average nucleotide identity (ANI) in **Supplementary Figure S4** in order to contextualize our simulation results for those more familiar with ANI comparisons.

To expand our performance comparison to a second tool designed for strain inference, we also ran MixtureS on a subset of the simulations. MixtureS estimates strain genotype and relative abundance on each metagenotype individually and therefore does not leverage variation in strain abundance across samples. We found that it performed worse than Strain Finder and Strain Facts on the benchmarks (see **Supplementary Figure S3**).

Overall, these results suggest that StrainFacts is capable of state-of-the-art performance with respect to several different scientific objectives in a realistic set of simulations. Performance was surprisingly robust to model misspecification with more strains than the simulation. Eliminating the computational demands of a separate model selection step further improves the scaling properties of StrainFacts.

Single-Cell Sequencing Validates Inferred Strain Genotypes

Beyond simulations, we sought to confirm the accuracy of strain inferences in a real biological dataset subject to forms of noise and bias not reflected in the generative model we used for simulations. To accomplish this, we applied a recently developed, single-cell, genomic sequencing workflow to obtain ground-truth, strain genotypes from two fecal samples collected in a previously described, clinical FMT experiment (Smith et al., 2022) from two independent subjects. We ran StrainFacts on metagenotypes derived from these two focal samples as well as the other 157 samples in the same study.

Genotypes that StrainFacts inferred to be present in each of these metagenomes matched the observed SCGs, with a mean, best-match normalized Hamming distance of 0.039. Furthermore, the median distance was just 0.013, reflecting the outsized influence of a small number of SCGs with more extreme deviations. For many species, SCGs also match a consensus genotype—the majority allele at each SNP site in each metagenotype (see **Figure 3A**). We found a mean distance to the consensus of 0.037 and a median of 0.009. Because this distance excludes sites without observed counts in the metagenotype, we masked these same sites in our inferred genotypes to uniformly contrast the consensus approach to StrainFacts genotypes. Overall, inferred genotypes were similar to the consensus, with a mean, masked distance of 0.031 (median of 0.009). However, the consensus approach fails for species with a mixture of multiple, co-existing strains. When we select only species with a metagenotype entropy of greater than 0.05, an indication of strain heterogeneity, we see that StrainFacts inferences have a distinct advantage, with a mean distance of 0.055 versus 0.069 for the consensus approach (median of 0.018

TABLE 2 | Concordance among SCGs of cluster assignments and the closest-match StrainFacts inferred genotype, among the four strains inferred to be at greater than 1% relative abundance in the analysed sample. The total number of SCGs in each cluster and the relative abundance of each inferred strain are indicated in parentheses in the column and row labels, respectively. Numbers in each cell indicate the number of SCGs at that intersection and values in parentheses indicate the median normalized Hamming distance of those SCGs to the inferred strain genotype.

	Cluster A (48)	Cluster B (7)	Cluster C (6)	Cluster D (1)
Strain 1 (57%)	48 (0.006)	1 (0.18)		
Strain 2 (32%)		3 (0.19)	6 (0.008)	
Strain 3 (7%)				1 (0.02)
Strain 4 (3%)		3 (0.19)		

versus 0.022, $p < 0.001$). These results validate inferred genotypes in a stool microbiome using single-cell genomics and demonstrate that StrainFacts accounts for strain-mixtures better than consensus genotypes do.

Of the 75 species represented in our SCG dataset, one stood out for having numerous SCGs while reflecting a remarkably high degree of strain heterogeneity. Among 68 high-quality SCGs for *S. thermophilus*, cluster analysis identified four distinct types (here referred to as Clusters A—D), accounting for 48, 7, 6, and one SCGs, respectively (**Figure 3B**). Independently, StrainFacts inferred four strains in the metagenomic data from the same stool sample, (Strain 1—4) with 57, 32, and 7, and 3% relative abundance, respectively. We explored the concordance between clusters and StrainFacts inferences by assigning a best-match Hamming distance genotype among the inferred strains to each SCG (**Table 2**). For SCGs in three of the four clusters there was a low median distance to StrainFacts genotypes as well as a perfect 1-to-1 correspondence between strains and clusters. While this genotype concordance was broken for SCGs in cluster B, strain 4 was also inferred to be at the lowest relative abundance, suggesting that there may have been too little information encoded in the metagenotype data to accurately reconstruct that strain's genotype. While SCG counts and inferred strain fractions do not match perfectly in this sample, this may be due to large differences between SCG and metagenomic sequencing technologies that could result in differentially biased sampling of strains. The SCG cluster with the largest membership was, however, matched with the strain inferred to be at the highest relative abundance. Our findings for *S. thermophilus* show that StrainFacts' estimates of genotypes and relative abundances are remarkably accurate for samples with high strain heterogeneity, despite the challenges presented by real biological samples and low abundance strains.

Analysis of Genomic Diversity Using *de novo* Strain Inferences on Thousands of Samples

Having established the accuracy and scalability of StrainFacts, we applied it to a corpus of metagenotype data derived from 20,550 metagenomes across 44 studies, covering a large fraction of all publicly available human-associated microbial metagenomes (Shi et al., 2021). We performed strain inference on GT-Pro metagenotypes for four species: *Escherichia coli*, *Agathobacter rectalis*, *Methanobrevibacter smithii*, and CAG-279 sp1. *E. coli* and *A. rectalis* are two highly prevalent and well studied bacterial inhabitants of the human gut microbiome, and *M. smithii* is the most prevalent and abundant archaeon detected in the human gut (Scanlan et al., 2008). CAG-279, on the other hand, is an unnamed and little-studied genus and a member of the family *Muribaculaceae*. This family is common in mice (Lagkouvardos et al., 2019), but to our knowledge does not have representatives cultured from human samples.

For each species, we compared strains inferred by StrainFacts to those represented in the GT-Pro reference database, which is derived from the UHGG (Almeida et al., 2021). In order to standardize comparisons, we dereplicated inferred and reference strains at a 0.05 genotype distance threshold. Interestingly, dereplication had a negligible effect on StrainFacts results, reducing the number of *E. coli* strains by just 4 (to 119) with no reduction for the three other species. This suggests that the diversity regularization built into the StrainFacts model is sufficient to collapse closely related strains as part of inference.

As GT-Pro only tallies alleles at a fixed subset of SNPs, the relationship between genotype distances and ANI is not fixed. In order to anchor our results to this widely-used measure of genome similarity, we compared the genotype distance to genome-wide ANI for all pairs of genomes in the GT-Pro reference database for all four species. We find that the fraction of positions differing genome wide (calculated as 1—ANI) was nearly proportional to the fraction of genotyped positions differing, but with a different constant of proportionality for each species: *E. coli* (0.0776, uncentered $R^2 = 0.994$), *A. rectalis* (0.1069, $R^2 = 0.990$), *M. smithii* (0.0393, $R^2 = 0.967$), and CAG-279 (0.0595, $R^2 = 0.991$). Additional details of this analysis can be found in **Supplementary Figure S4.**

StrainFacts Recapitulates Known Diversity in Well Studied Species

E. coli, *A. rectalis*, and *M. smithii* all have many genome sequences in GT-Pro reference database, presenting an opportunity to contrast inferred against reference strains. In order to evaluate the concordance between the two (**Table 3** and **Figure 4**), we co-clustered all dereplicated strains (both reference and inferred) at a 0.15 normalized Hamming distance threshold—note, crucially, that this distance reflects a much smaller full-genome dissimilarity, as it is based only on genome positions with polymorphism across metagenomes, ignoring conserved positions.

For *E. coli*, we identified 40 strain clusters with 93% of inferred strains and 94% of references falling into clusters containing strains

TABLE 3 | Dereplication and co-clustering of strains inferred from metagenomes or from a reference database.

Species	Metagenome samples fit	Reference strains[a]	Inferred strains[a]	Total clusters[b]	Novel clusters[b] (%)	Shared clusters[b] (%)
E. coli	9,232	176	119	40	20	60
A. rectalis	11,860	752	198	456	13	25
M. smithii	3,528	384	178	205	7	38
CAG-279	3,579	135	200	228	50	25

[a] *Dereplicated at 0.05 distance threshold.*

[b] *Co-clustered at a 0.15 distance threshold.*

from both sources ("shared" clusters), which is significantly more overlap than expected after random shuffling of cluster labels (p = 0.002 by permutation test). While most metagenome-inferred genotypes are similar to those found in genome reference databases, we observed some clusters composed only of StrainFacts strains, representing novel lineages. However, these strains are no more common than after random permutation (p = 0.81), matching our expectations for this well-studied species.

We next asked if these trends hold for the other species. While *A. rectalis* had a much greater number of clusters (456), 69% of inferred strains and 45% of reference strains are nonetheless found to be in shared clusters, significantly more than would be expected with random shuffling of cluster labels (p = 0.002 by permutation test). Correspondingly, we do not find evidence for enrichment of inferred strains in novel clusters (p = 0.71). We find similar results for *M. smithii* and CAG-279—the fraction of strains in shared clusters is significantly greater than after random reassignment ($p < 0.001$ for both), and there is no evidence for enrichment of inferred strains in novel clusters (p = 1.0 for both). Overall, the concordance between reference and inferred strains supports not only the credibility of StrainFacts' estimates, but also suggests that our *de novo* inferences capture a substantial fraction of previously documented strain diversity, even in well studied species.

Going beyond the extensive overlap of strains with reference genomes and StrainFacts inferences, we examined clusters in which references are absent or relatively rare. Visualizing a dendrogram of consensus genotypes from co-clustered strains (**Figure 4**) we observe some sections of the *A. rectalis* dendrogram with many novel strains. Similarly, for CAG-279, the sheer number of inferred strains relative to genomes in reference databases means that fully half of all genotype clusters are entirely novel, emphasizing the power of StrainFacts inferences in understudied species. Future work will be needed to determine if these represent new subspecies currently missing from reference databases.

Species Inhabiting the Human Gut Exhibit Distinct Biogeography Observed Across Independent Metagenomic Studies

Large metagenomic collections allow us to examine intraspecific microbial diversity at a global scale and among dozens of studies. Towards this end, we identified the most abundant StrainFacts strain of *E. coli*, *A. rectalis*, *M. smithii*, and CAG-279 in stool samples across 34 independent studies. Across all four species, we observe some strains that are distributed globally as well as others that appear specific to one country of origin (**Figure 5**, **Supplementary Figure S5**). For example, among the 198 dereplicated, inferred strains of *A. rectalis*, 75 were found as the dominant strain (i.e. highest relative abundance) in samples collected on three or more continents. While this makes it challenging to consistently discern where a sample was collected based on its dominant strain of a given species, we nonetheless find that studies with samples collected in the United States of America form a distinct cluster, as do those from China, and the two are easily distinguished from one another and from most other studies conducted across Europe and North America (**Figure 5**). Our observation of a distinct group of *A. rectalis* strains enriched in samples from China is consistent with previous results (Scholz et al., 2016; Costea PI. et al., 2017; Truong et al., 2017).

These general trends hold across the other three species. In *M. smithii*, independent studies in the same country often share very similar strain dominance patterns (e.g. see clustering of studies performed in each of China, Mongolia, Denmark, and Spain in **Figure 5**). In *E. coli*, while many strains appear to be distributed globally, independent studies from China still cluster together based on patterns in strain dominance (see **Supplementary Figure S5**). Notably, in CAG-279, studies with individuals in westernized societies do not cluster separately from the five other studies, suggesting that host lifestyle is not highly correlated with specific strains of this species. The variety of dominance patterns across the four species described here suggest that different mechanisms may drive intraspecific biogeography depending on the biology and natural history of the species. As the coverage of diverse microbiomes grows, StrainFacts will enable future studies disentangling the contributions of lifestyle, dispersal limitation and other factors in the global distribution of strains.

Linkage Disequilibrium Decay Suggests Variation in Recombination Rates Across Microbial Species

Studying the population genetics of host-associated microbes has the potential to elucidate processes of transmission, diversification, and selection with implications for human health and perhaps even our understanding of human origins (Linz et al., 2007; Garud and Pollard, 2019). To demonstrate the application of StrainFacts to the study of microbial evolution, we examined patterns in pairwise LD, here calculated as the squared Pearson correlation coefficient (r^2). This statistic can inform understanding of recombination rates in microbial populations

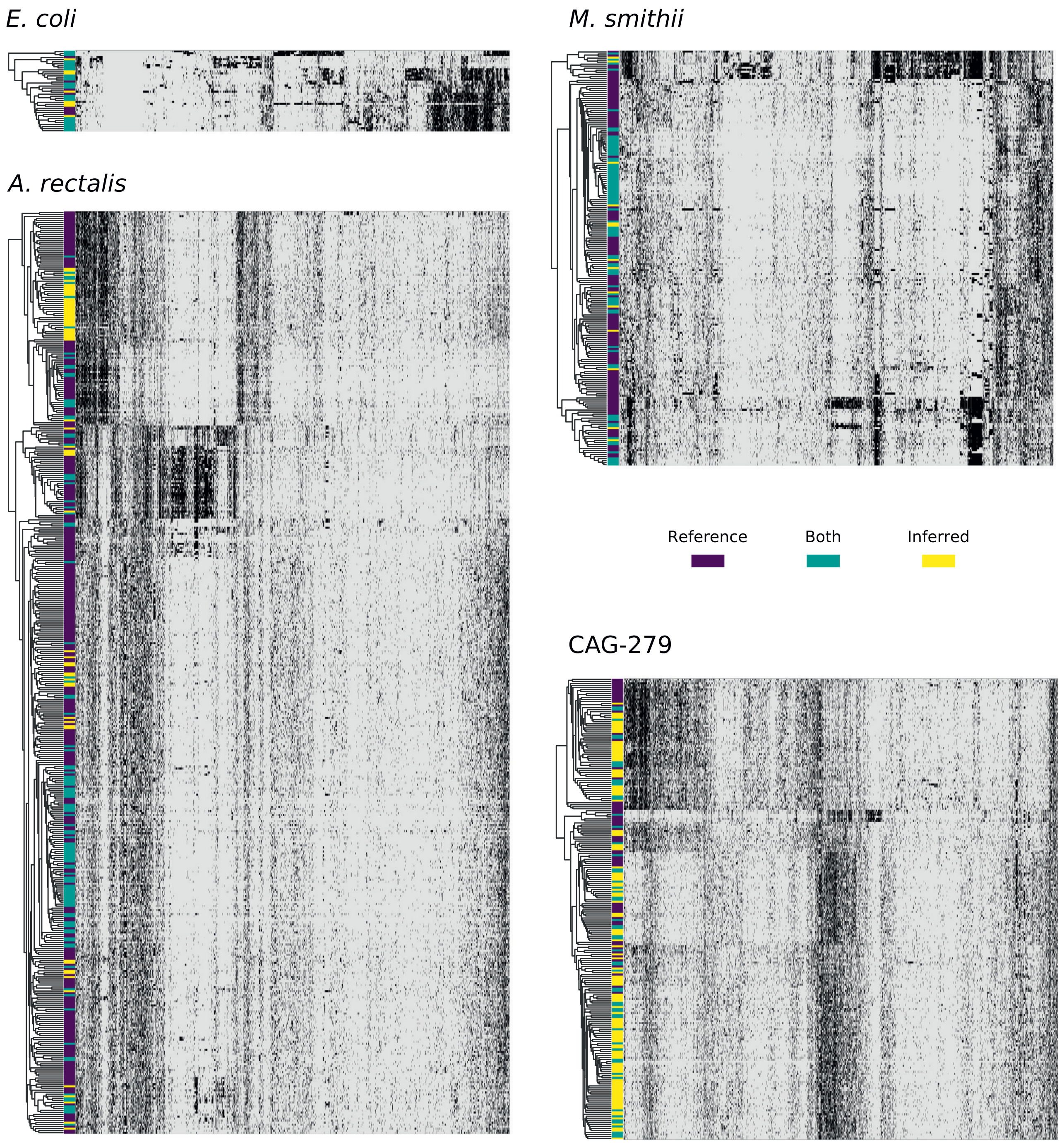


FIGURE 4 | Concordance between reference and StrainFacts inferred strain genotypes for four prevalent species in the human gut microbiome. Heatmap rows represent consensus genotypes from co-clustering of reference and inferred strains and columns are 3,500 randomly sampled SNP sites (grey: reference and black: alternative allele). Colors to the left of the heatmap indicate clusters with only reference strains (dark purple), only inferred strains (yellow), or both (teal). Rows are ordered by hierarchical clustering built on distances between consensus genotypes and columns are ordered arbitrarily to highlight correlations between SNPs.

(Vos, 2009; Garud et al., 2019). Genome-wide, LD, summarized as the 90th percentile r^2 (LD_{90}, Vos et al., 2017), was substantially higher for *E. coli* (0.24) than *A. rectalis* (0.04), *M. smithii* (0.11), or CAG-279 (0.04), perhaps suggesting greater population structure in the species and less panmictic recombination.

We estimated LD distance-decay curves for SNPs, using a single, high-quality reference genome for each species to obtain estimates of pairwise distance between SNP sites. For all four species, adjacent SNPs were much more tightly linked, with an LD_{90} of > 0.999. LD was still dramatically above background at 50

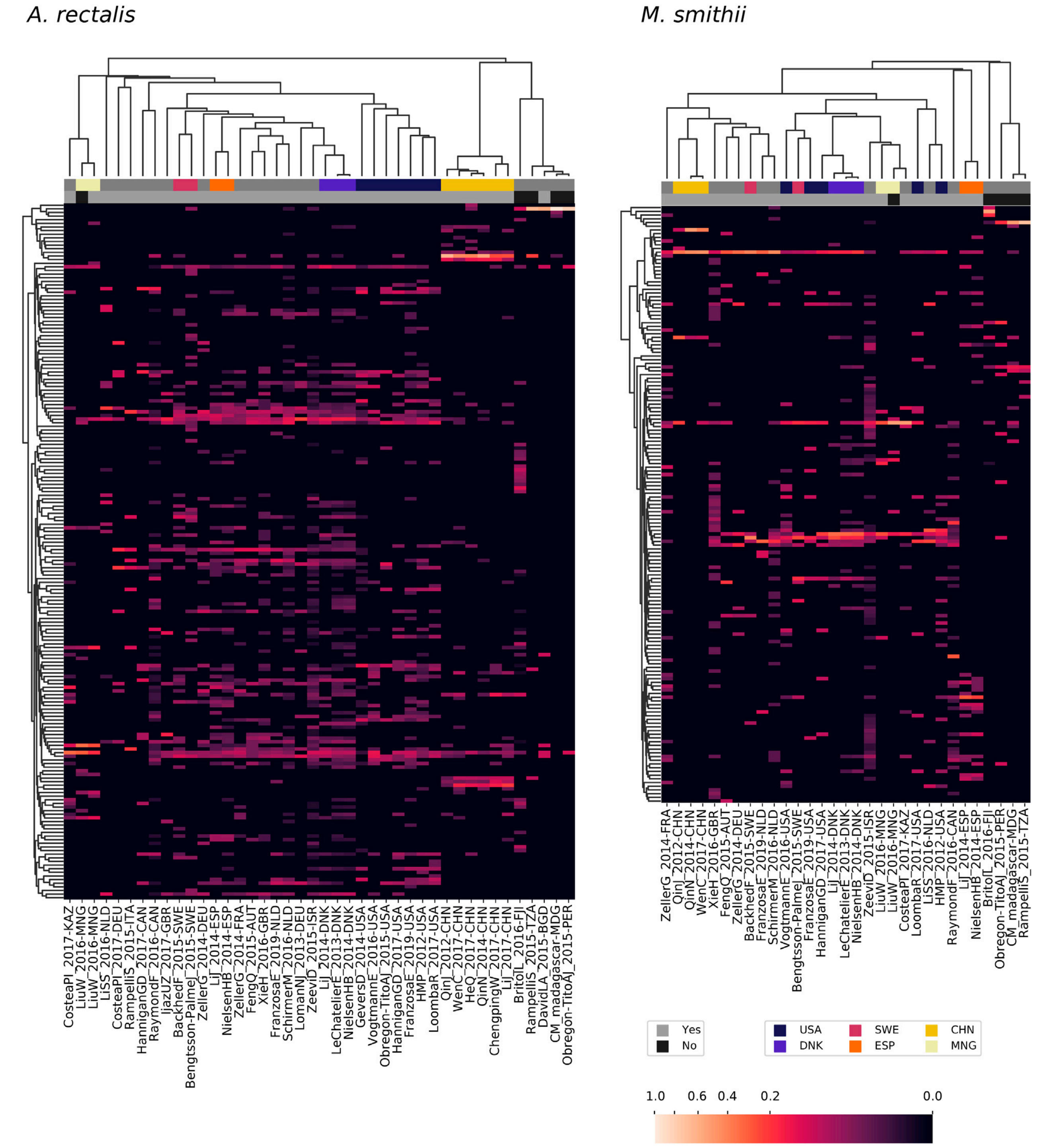


FIGURE 5 | Patterns in strain dominance according to geography and lifestyle across thousands of publicly available metagenomes in dozens of independent studies for two common members of the human gut microbiome. Columns represent collections of samples from individual studies and are further segmented by country and lifestyle (westernized or not). Rows represent strains inferred by StrainFacts. Cell colors reflect the fraction of samples in that study segment with that strain as the most abundant member. Study segments are omitted if they include fewer than 10 samples. Row ordering and the associated dendrogram reflect strain genotype distances, while the dendrogram for columns is based on their cosine similarity. Studies with samples collected in several countries with notable clustering for one or more species are highlighted with colors above the heatmap. Additionally, studies from westernized populations are indicated. Both a study identifier and the ISO 3166-ISO country-code are included in the column labels.

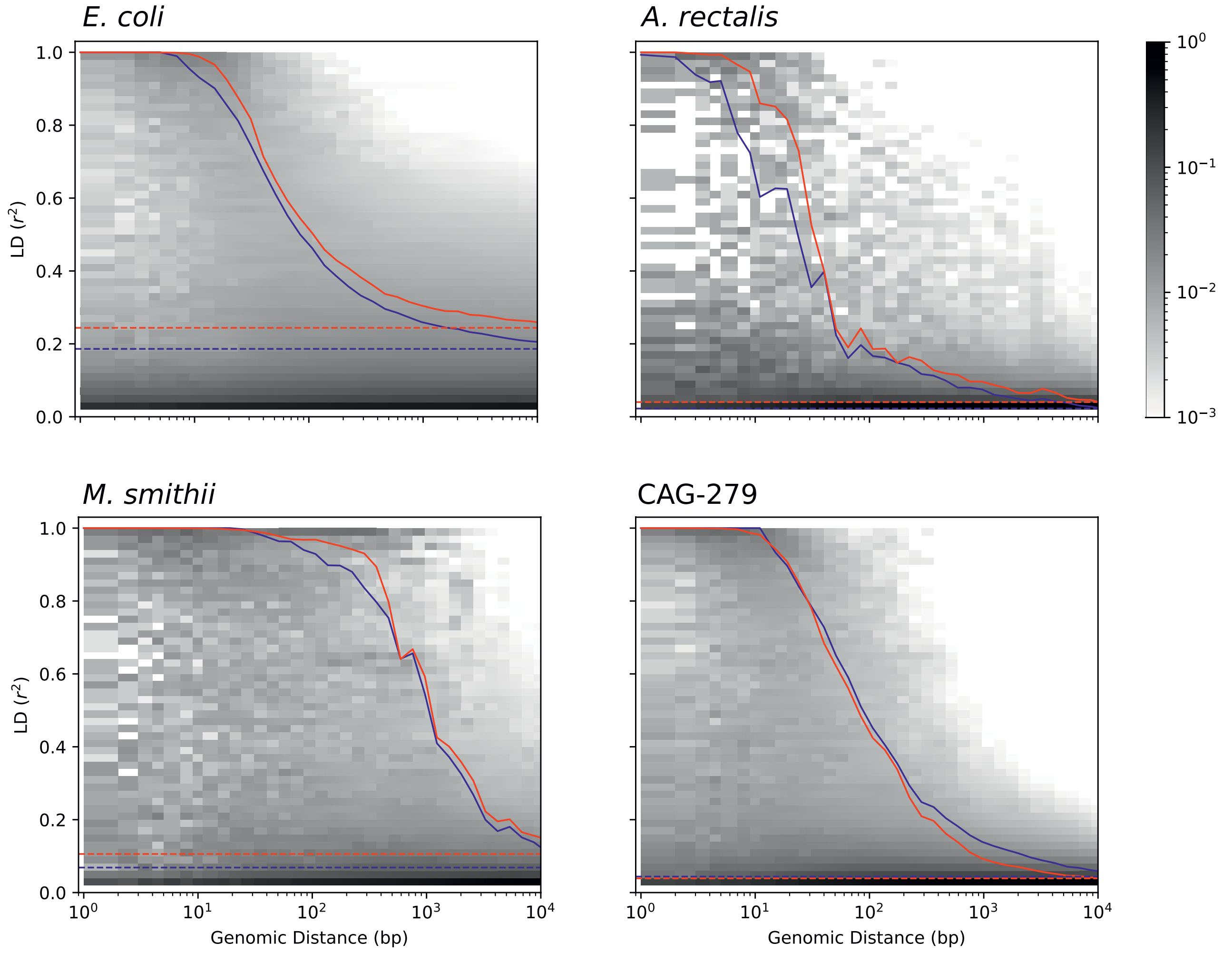


FIGURE 6 | Pairwise LD across genomic distance estimated from inferred genotypes for four species. LD was calculated as r^2 and genomic distance between polymorphic loci is based on distances in a single, representative genome. The distribution of SNP pairs in each distance window is shown as a histogram with darker colors reflecting a larger fraction of the pairs in that LD bin, and the LD_{90} for pairs at each distance is shown for inferred strains (red), along with an identical analysis on strains in the reference database (blue). Genome-wide LD_{90} (dashed lines) is also indicated.

bases of separation, and fell rapidly with increasing distance (**Figure 6**). The speed of this decay was different between species, which we characterized with the $LD_{½,90}$: the distance at which the LD_{90} was less than 50% of the value for adjacent SNPs (Vos et al., 2017). *M. smithii* exhibited by far the slowest decay, with a $LD_{½,90}$ of 520 bases, followed by *E. coli* at 93 bases, CAG-279 at 66, and *A. rectalis* at just 28 bases. This variation in decay profiles may reflect major differences in recombination rates across populations.

To validate our findings, we ran identical analyses with dereplicated genotypes from genomes in the GT-Pro reference database. Estimates of both LD and the distance-decay relationship are very similar between inferred and reference strains, reinforcing the value of genotypes inferred from metagenomes for microbial population genetics. Interestingly, for three of the four species (*E. coli*, *A. rectalis*, and *M. smithii*), LD estimates from StrainFacts strains were significantly higher than from references ($p <$ 1e-10 for all three by Wilcoxon test), while CAG-279 exhibited a trend towards the reverse ($p = 0.85$). It is not clear what might cause these quantitative discrepancies, but they could reflect differences in the set of strains in each dataset. Future studies expanding this analysis to additional species will identify patterns in recombination rates across broader microbial diversity.

DISCUSSION

Here we have described StrainFacts, a novel tool for strain inference in metagenomic data. StrainFacts models metagenotype data using a fuzzy-genotype approximation, allowing us to estimate both the relative abundance of strains across samples as well as their genotypes. To accelerate analysis compared to the current state-of-the-art, we harness the differentiability of our model to apply modern, gradient-based optimization and GPU-parallelization. Consequently, StrainFacts can scale to tens-of-thousands of samples while inferring genotypes for hundreds of strains. On simulated benchmarks, we show that StrainFacts has comparable accuracy to Strain Finder, and we validate strain inferences *in vivo* against genotypes observed by single-cell genomics. Finally, we apply StrainFacts to a database of tens of thousands of metagenomes from the human microbiome to estimate strains *de novo*, allowing us to characterize strain diversity, biogeography, and population genetics, without the need for cultured isolates.

Beyond Strain Finder, other alternatives exist for strain inference in metagenomic data. While we do not directly compare to DESMAN, runtimes of several hours have been reported for that tool on substantially smaller simulated datasets (Quince et al., 2017), and hence we believe that StrainFacts is likely the most scalable implementation of the metagenotype deconvolution approach. Still other methods apply regularized regression (e.g. Lasso Albanese and Donati, 2017) to decompose metagenotypes—essentially solving the abundance half of the deconvolution problem but not the genotypes half—or look for previously determined strain signatures (e.g. k-mers Panyukov et al., 2020) based on known strain genotypes. However, both of these require an *a priori* database of the genotypes that might be present in a sample. An expanding database of strain references can make these approaches increasingly useful, and StrainFacts can help to build this reference.

Several studies avoid deconvolution by directly examining allele frequencies inferred from metagenotypes. While consensus (Truong et al., 2017; Zolfo et al., 2017) or phasing (Garud et al., 2019) approaches can accurately recover genotypes in some cases, their use is limited to low complexity datasets, with sufficient sequencing depth and low strain heterogeneity. Likewise, measuring the dissimilarity of metagenotypes among pairwise samples indicates shared strains (Podlesny and Fricke, 2020), but this approach risks confounding strain mixing with genotype similarity. Finally, assembly (Li et al., 2019) and read-based methods (Cleary et al., 2015) for disentangling strains are most applicable when multiple SNPs can be found in each sequencing read. With ongoing advancements in long-read (Vicedomini et al., 2021) and read-cloud sequencing (Kuleshov et al., 2016; Kang et al., 2018), these approaches will become increasingly feasible. Thus, StrainFacts occupies the same analysis niche as Strain Finder and DESMAN, and it expands upon these reliable approaches by providing a scalable model fitting procedure.

Fuzzy genotypes enable more computationally efficient inference by eliminating the need for discrete optimization. Specifically, we used well-tested and optimized gradient descent algorithms implemented in PyTorch for parameter estimation. In addition, fuzzy genotypes may be more robust to deviations from model assumptions. For instance, an intermediate genotype could be a satisfactory approximation when gene duplications or deletions are present in some strains. While we do not explore the possibility here, fuzzy genotypes may provide a heuristic for capturing uncertainty in strain genotypes. Future work could consider propagating intermediate genotype values instead of discretizing them.

StrainFacts builds on recent advances in metagenotyping, in particular our analyses harnessed GT-Pro (Shi et al., 2021) to greatly accelerate SNP counting in metagenomic reads. While we leave a comparison of StrainFacts performance on the outputs of other metagenotypers to future work, StrainFacts itself is agnostic to the source of input data. It would be straightforward to extend StrainFacts to operate on loci with more than two alleles or to use metagenotypes from a tool other than GT-Pro. It would also be interesting to extend StrainFacts to use SNPs outside the core genome that vary in their presence across strains.

Combined with the explosive growth in publicly available metagenomic data and the development of rapid metagenotyping tools, StrainFacts enables the *de novo* exploration of intraspecific microbial diversity at a global scale and on well-powered cohorts with thousands of samples. Future applications could examine intraspecific associations with disease, track the history of recombination between microbial subpopulations, and measure the global transmission of host-associated microbes across human populations.

DATA AVAILABILITY STATEMENT

Metagenomic sequencing data from the FMT study are available through the SRA under BioProject PRJNA737472, The two single-cell genomics experiments are also under that project with accessions SRR18748374 and SRR18748375. Publicly available metagenomes are available under various other accessions described in (Shi et al., 2021). Strain genotypes from the GT-Pro reference database are publicly available at https://fileshare.czbiohub.org/s/waXQzQ9PRZPwTdk. All other code and metadata needed to reproduce these results are available at https://doi.org/10.5281/zenodo.5942586.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by the UCSF Committee on Human Research. The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

BS: conceptualization, data curation, formal analysis, methodology, software, visualization, writing—original draft, writing—review and editing. XL: investigation, data curation, methodology, writing—original draft, writing—review and editing. ZS: data curation, methodology, writing—review and editing. AA: funding acquisition, supervision. KP: conceptualization, writing—review and editing, funding acquisition, supervision.

FUNDING

BS was supported by the National Institutes of Health grant number 5T32DK007007 and from the UCSF Initiative for Digital Transformation in Computational Biology & Health. BS, ZS, and KP were supported by the National Science Foundation grant number 1563159 and by the Gladstone Institutes. BS, XL, ZS, AA, and KP were supported by the funding from Chan Zuckerberg Biohub.

ACKNOWLEDGMENTS

Barbara Engelhardt provided valuable feedback on this project.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.867386/full#supplementary-material

REFERENCES

Albanese, D., and Donati, C. (2017). Strain Profiling and Epidemiology of Bacterial Species from Metagenomic Sequencing. *Nat. Commun.* 8, 2260. doi:10.1038/s41467-017-02209-5

Almeida, A., Nayfach, S., Boland, M., Strozzi, F., Beracochea, M., Shi, Z. J., et al. (2021). A Unified Catalog of 204,938 Reference Genomes from the Human Gut Microbiome. *Nat. Biotechnol.* 39, 105–114. doi:10.1038/s41587-020-0603-3

Beghini, F., McIver, L. J., Blanco-Míguez, A., Dubois, L., Asnicar, F., Maharjan, S., et al. (2021). Integrating Taxonomic, Functional, and Strain-Level Profiling of Diverse Microbial Communities with bioBakery 3. *eLife* 10, e65088. doi:10.7554/eLife.65088

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., et al. (2019). Pyro: Deep Universal Probabilistic Programming. *J. Mach. Learn. Res.* 20, 1–6. Available at: http://jmlr.org/papers/v20/18-403.html. (Accessed April 8, 2021). doi:10.48550/arXiv.1810.09538

Case, R. J., Boucher, Y., Dahllöf, I., Holmström, C., Doolittle, W. F., and Kjelleberg, S. (2007). Use of 16S rRNA and rpoB Genes as Molecular Markers for Microbial Ecology Studies. *Appl. Environ. Microbiol.* 73, 278. doi:10.1128/AEM.01177-06

Chu, N. D., Crothers, J. W., Nguyen, L. T. T., Kearney, S. M., Smith, M. B., Kassam, Z., et al. (2021). Dynamic Colonization of Microbes and Their Functions after Fecal Microbiota Transplantation for Inflammatory Bowel Disease. *mBio* 12, e0097521. doi:10.1128/mBio.00975-21

Cleary, B., Brito, I. L., Huang, K., Gevers, D., Shea, T., Young, S., et al. (2015). Detection of Low-Abundance Bacterial Strains in Metagenomic Datasets by Eigengenome Partitioning. *Nat. Biotechnol.* 33, 1053–1060. doi:10.1038/nbt.3329

Costea, P. I., Coelho, L. P., Sunagawa, S., Munch, R., Huerta-Cepas, J., Forslund, K., et al. (2017a). Subspecies in the Global Human Gut Microbiome. *Mol. Syst. Biol.* 13, 960. doi:10.15252/msb.20177589

Costea, P. I., Hildebrand, F., Arumugam, M., Bäckhed, F., Blaser, M. J., Bushman, F. D., et al. (2017b). Enterotypes in the Landscape of Gut Microbial Community Composition. *Nat. Microbiol.* 3, 8. doi:10.1038/s41564-017-0072-8

Garud, N. R., Good, B. H., Hallatschek, O., and Pollard, K. S. (2019). Evolutionary Dynamics of Bacteria in the Gut Microbiome within and across Hosts. *Plos Biol.* 17, e3000102. doi:10.1371/journal.pbio.3000102

Garud, N. R., and Pollard, K. S. (2019). Population Genetics in the Human Microbiome. *Trends Genet.* 36, 53. doi:10.1016/j.tig.2019.10.010

Haiser, H. J., Seim, K. L., Balskus, E. P., and Turnbaugh, P. J. (2014). Mechanistic Insight into Digoxin Inactivation by Eggerthella Lenta Augments Our Understanding of its Pharmacokinetics. *Gut Microbes* 5, 233–238. doi:10.4161/gmic.27915

Kang, J. B., Siranosian, B., Moss, E., Andermann, T., and Bhatt, A. (2018). Read Cloud Sequencing Elucidates Microbiome Dynamics in a Hematopoietic Cell Transplant Patient. *IEEE Int. Conf. Bioinforma. Biomed. BIBM.* 2018, 234. doi:10.1109/bibm.2018.8621297

Kuleshov, V., Snyder, M. P., and Batzoglou, S. (2016). Genome Assembly from Synthetic Long Read Clouds. *Bioinformatics* 32, i216–i224. doi:10.1093/bioinformatics/btw267

Kurtzer, G. M., Sochat, V., and Bauer, M. W. (2017). Singularity: Scientific Containers for Mobility of Compute. *PLOS ONE* 12, e0177459. doi:10.1371/journal.pone.0177459

Lagkouvardos, I., Lesker, T. R., Hitch, T. C. A., Gálvez, E. J. C., Smit, N., Neuhaus, K., et al. (2019). Sequence and Cultivation Study of *Muribaculaceae* Reveals Novel Species, Host Preference, and Functional Potential of This yet Undescribed Family. *Microbiome* 7, 28. doi:10.1186/s40168-019-0637-2

Li, S. S., Zhu, A., Benes, V., Costea, P. I., Hercog, R., Hildebrand, F., et al. (2016). Durable Coexistence of Donor and Recipient Strains after Fecal Microbiota Transplantation. *Science* 352, 586–589. doi:10.1126/science.aad8852

Li, X., Saadat, S., Hu, H., and Li, X. (2019). BHap: A Novel Approach for Bacterial Haplotype Reconstruction. *Bioinformatics* 35, 4624–4631. doi:10.1093/bioinformatics/btz280

Li, X., Hu, H., and Li, X. (2021). MixtureS: A Novel Tool for Bacterial Strain Reconstruction from Reads. *Bioinformatics* 37, 575. doi:10.1093/bioinformatics/btaa728

Linz, B., Balloux, F., Moodley, Y., Manica, A., Liu, H., Roumagnac, P., et al. (2007). An African Origin for the Intimate Association between Humans and Helicobacter pylori. *Nature* 445, 915–918. doi:10.1038/nature05562

Loman, N. J., Constantinidou, C., Christner, M., Rohde, H., Chan, J. Z., Quick, J., et al. (2013). A Culture-independent Sequence-Based Metagenomics Approach to the Investigation of an Outbreak of Shiga-Toxigenic Escherichia coli O104:H4. *JAMA* 309, 1502–1510. doi:10.1001/jama.2013.3231

Lozupone, C. A., Hamady, M., Kelley, S. T., and Knight, R. (2007). Quantitative and Qualitative β Diversity Measures Lead to Different Insights into Factors that Structure Microbial Communities. *Appl. Environ. Microbiol.* 73, 1576–1585. doi:10.1128/aem.01996-06

Luo, C., Knight, R., Siljander, H., Knip, M., Xavier, R. J., and Gevers, D. (2015). ConStrains Identifies Microbial Strains in Metagenomic Datasets. *Nat. Biotechnol.* 33, 1045–1052. doi:10.1038/nbt.3319

Mölder, F., Jablonski, K. P., Letcher, B., Hall, M. B., Tomkins-Tinch, C. H., Sochat, V., et al. (2021). Sustainable Data Analysis with Snakemake. *F1000Research* 10, 33. doi:10.12688/f1000research.29032.2

Monti, G. S., Mateu-Figueras, G., Pawlowsky-Glahn, V., and Egozcue, J. J. (2011). "The Shifted-Scaled Dirichlet Distribution in the Simplex," in 4th International Workshop on Compositional Data Analysis, Sant Feliu de Guíxols, Girona Spain.

Nayfach, S., Rodriguez-Mueller, B., Garud, N., and Pollard, K. S. (2016). An Integrated Metagenomics Pipeline for Strain Profiling Reveals Novel Patterns of Bacterial Transmission and Biogeography. *Genome Res.* 26, 1612–1625. doi:10.1101/gr.201863.115

Nayfach, S., Roux, S., Seshadri, R., Udwary, D., Varghese, N., Schulz, F., et al. (2020). A Genomic Catalog of Earth's Microbiomes. *Nat. Biotechnol.* 39, 499. doi:10.1038/s41587-020-0718-6

O'Brien, J. D., Didelot, X., Iqbal, Z., Amenga-Etego, L., Ahiska, B., and Falush, D. (2014). A Bayesian Approach to Inferring the Phylogenetic Structure of Communities from Metagenomic Data. *Genetics* 197, 925–937. doi:10.1534/genetics.114.161299

Olm, M. R., Crits-Christoph, A., Bouma-Gregson, K., Firek, B. A., Morowitz, M. J., and Banfield, J. F. (2021). inStrain Profiles Population Microdiversity from Metagenomic Data and Sensitively Detects Shared Microbial Strains. *Nat. Biotechnol.* 39, 727–736. doi:10.1038/s41587-020-00797-0

Ostrowski, M. P., La Rosa, S. L., Kunath, B. J., Robertson, A., Pereira, G., Hagen, L. H., et al. (2022). Mechanistic Insights Into Consumption of the Food Additive Xanthan Gum by the Human Gut Microbiota. *Nat. Microbiol.* 7 (4), 556–569.

Panyukov, V. V., Kiselev, S. S., and Ozoline, O. N. (2020). Unique K-Mers as Strain-specific Barcodes for Phylogenetic Analysis and Natural Microbiome Profiling. *Int. J. Mol. Sci.* 21, 944. doi:10.3390/ijms21030944

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in *Advances In Neural Information Processing Systems*(ancouver, Canada: Curran Associates, Inc.). Available at: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html (Accessed January 30, 2022).

Patrick, S., Blakely, G. W., Houston, S., Moore, J., Abratt, V. R., Bertalan, M., et al. (2010). Twenty-eight Divergent Polysaccharide Loci Specifying within- and Amongst-Strain Capsule Diversity in Three Strains of Bacteroides Fragilis. *Microbiology (Reading)* 156, 3255–3269. doi:10.1099/mic.0.042978-0

Podlesny, D., and Fricke, W. F. (2020). Microbial Strain Engraftment, Persistence and Replacement after Fecal Microbiota Transplantation. *medRxiv* 2020, 20203638. doi:10.1101/2020.09.29.20203638

Quince, C., Delmont, T. O., Raguideau, S., Alneberg, J., Darling, A. E., Collins, G., et al. (2017). DESMAN: A New Tool for De Novo Extraction of Strains from Metagenomes. *Genome Biol.* 18, 181–222. doi:10.1186/s13059-017-1309-9

Scanlan, P. D., Shanahan, F., and Marchesi, J. R. (2008). Human Methanogen Diversity and Incidence in Healthy and Diseased Colonic Groups Using mcrA Gene Analysis. *BMC Microbiol.* 8, 79. doi:10.1186/1471-2180-8-79

Schmidt, M. N., Winther, O., and Hansen, L. K. (2009). “Bayesian Non-negative Matrix Factorization,” in Independent Component Analysis And Signal Separation *Lecture Notes in Computer Science*. Editors T. Adali, C. Jutten, J. M. T. Romano, and A. K. Barros (Berlin, Heidelberg: Springer), 540–547. doi:10.1007/978-3-642-00599-2_68

Scholz, M., Ward, D. V., Pasolli, E., Tolio, T., Zolfo, M., Asnicar, F., et al. (2016). Strain-level Microbial Epidemiology and Population Genomics from Shotgun Metagenomics. *Nat. Methods* 13, 435–438. doi:10.1038/nmeth.3802

Shi, Z. J., Dimitrov, B., Zhao, C., Nayfach, S., and Pollard, K. S. (2021). Fast and Accurate Metagenotyping of the Human Gut Microbiome with GT-Pro. *Nat. Biotechnol.* 1–10. doi:10.1038/s41587-021-01102-3

Shoemaker, N. B., Vlamakis, H., Hayes, K., and Salyers, A. A. (2001). Evidence for Extensive Resistance Gene Transfer Among Bacteroides Spp. And Among Bacteroides and Other Genera in the Human Colon. *Appl. Environ. Microbiol.* 67, 561–568. doi:10.1128/AEM.67.2.561-568.2001

Smillie, C. S., Sauk, J., Gevers, D., Friedman, J., Sung, J., Youngster, I., et al. (2018). Strain Tracking Reveals the Determinants of Bacterial Engraftment in the Human Gut Following Fecal Microbiota Transplantation. *Cell Host Microbe* 23, 229–e5. doi:10.1016/j.chom.2018.01.003

Smith, B. J., Piceno, Y., Zydek, M., Zhang, B., Syriani, L. A., Terdiman, J. P., et al. (2022). Strain-Resolved Analysis in a Randomized Trial of Antibiotic Pretreatment and Maintenance Dose Delivery Mode with Fecal Microbiota Transplant for Ulcerative Colitis. *Sci. Rep.* 12 (1), 5517. doi:10.1038/s41598-022-09307-5

Truong, D. T., Tett, A., Pasolli, E., Huttenhower, C., and Segata, N. (2017). Microbial Strain-Level Population Structure and Genetic Diversity from Metagenomes. *Genome Res.* 27, 626–638. doi:10.1101/gr.216242.116

Vicedomini, R., Quince, C., Darling, A. E., and Chikhi, R. (2021). Strainberry: Automated Strain Separation in Low-Complexity Metagenomes Using Long Reads. *Nat. Commun.* 12, 4485. doi:10.1038/s41467-021-24515-9

Vos, M. (2009). Why Do Bacteria Engage in Homologous Recombination? *Trends Microbiol.* 17, 226–232. doi:10.1016/j.tim.2009.03.001

Vos, P. G., Paulo, M. J., Voorrips, R. E., Visser, R. G., van Eck, H. J., and van Eeuwijk, F. A. (2017). Evaluation of LD Decay and Various LD-Decay Estimators in Simulated and SNP-Array Data of Tetraploid Potato. *Theor. Appl. Genet.* 130, 123–135. doi:10.1007/s00122-016-2798-8

Watson, A. R., Füssel, J., Veseli, I., DeLongchamp, J. Z., Silva, M., Trigodet, F., et al. (2021). Adaptive Ecological Processes and Metabolic independence Drive Microbial Colonization and Resilience in the Human Gut. *bioRxiv*. doi:10.1101/2021.03.02.433653

Yan, Y., Nguyen, L. H., Franzosa, E. A., and Huttenhower, C. (2020). Strain-level Epidemiology of Microbial Communities and the Human Microbiome. *Genome Med.* 12, 71. doi:10.1186/s13073-020-00765-y

Zolfo, M., Tett, A., Jousson, O., Donati, C., and Segata, N. (2017). MetaMLST: Multi-Locus Strain-Level Bacterial Typing from Metagenomic Samples. *Nucleic Acids Res.* 45, e7. doi:10.1093/nar/gkw837

Conflict of Interest: KP is on the scientific advisory board of Phylagen.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Smith, Li, Shi, Abate and Pollard. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

frontiers | Frontiers in Bioinformatics

REVIEW
published: 26 May 2022
doi: 10.3389/fbinf.2022.871393

Metagenomic Analysis Using Phylogenetic Placement—A Review of the First Decade

Lucas Czech[1]*, *Alexandros Stamatakis*[2,3], *Micah Dunthorn*[4] *and Pierre Barbera*[5]*

[1]Department of Plant Biology, Carnegie Institution for Science, Stanford, CA, United States, [2]Computational Molecular Evolution Group, Heidelberg Institute for Theoretical Studies, Heidelberg, Germany, [3]Institute for Theoretical Informatics, Karlsruhe Institute of Technology, Karlsruhe, Germany, [4]Natural History Museum, University of Oslo, Oslo, Norway, [5]Independent Researcher, Bisingen, Germany

Phylogenetic placement refers to a family of tools and methods to analyze, visualize, and interpret the tsunami of metagenomic sequencing data generated by high-throughput sequencing. Compared to alternative (e. g., similarity-based) methods, it puts metabarcoding sequences into a phylogenetic context using a set of known reference sequences and taking evolutionary history into account. Thereby, one can increase the accuracy of metagenomic surveys and eliminate the requirement for having exact or close matches with existing sequence databases. Phylogenetic placement constitutes a valuable analysis tool *per se*, but also entails a plethora of downstream tools to interpret its results. A common use case is to analyze species communities obtained from metagenomic sequencing, for example via taxonomic assignment, diversity quantification, sample comparison, and identification of correlations with environmental variables. In this review, we provide an overview over the methods developed during the first 10 years. In particular, the goals of this review are 1) to motivate the usage of phylogenetic placement and illustrate some of its use cases, 2) to outline the full workflow, from raw sequences to publishable figures, including best practices, 3) to introduce the most common tools and methods and their capabilities, 4) to point out common placement pitfalls and misconceptions, 5) to showcase typical placement-based analyses, and how they can help to analyze, visualize, and interpret phylogenetic placement data.

Keywords: phylogenetic placement, evolutionary placement, phylogenetics, metagenomics, metabarcoding, species diversity, taxonomic assignment, sequence identification

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Burkhard Morgenstern,
University of Göttingen, Germany
José Patané,
Butantan Institute, Brazil
Benjamin Linard,
Spygen, France

Correspondence:
Lucas Czech
lczech@carnegiescience.edu
Pierre Barbera
pierre@barbera-bio.info

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *08 February 2022*
Accepted: *11 April 2022*
Published: *26 May 2022*

Citation:
Czech L, Stamatakis A, Dunthorn M and Barbera P (2022) Metagenomic Analysis Using Phylogenetic Placement—A Review of the First Decade. Front. Bioinform. 2:871393. doi: 10.3389/fbinf.2022.871393

1 INTRODUCTION

Advances in sequencing technologies enable the broad sequencing of genetic material in environmental samples (Edwards and Holt, 2013; Sunagawa et al., 2013), for instance, from water (Karsenti et al., 2011; Giner et al., 2016; Lacoursière-Roussel et al., 2016), soil (Dupont et al., 2016; Mahé et al., 2017), and air (Clare et al., 2022), which is known as environmental DNA (eDNA, Deiner et al., 2017; Ruppert et al., 2019), or from the human body (Curtis et al., 2012; Methé et al., 2012; Matsen, 2015; Wang et al., 2015) and other sources (Hanson et al., 2016; ElRakaiby et al., 2019; Gohli et al., 2019; Lorimer et al., 2019). Crucially, this enables the ecological survey of a community of organisms in their immediate environment (i. e., *in situ*), and allows to directly study the genetic composition of species communities (from viruses to megafauna); a field known as metagenomics (Thomas et al., 2012; Escobar-Zepeda et al., 2015; Oulas et al., 2015; Lindgreen et al., 2016).

Metagenomic data typically stem from so-called *High-Throughput Sequencing* (HTS, Pettersson et al., 2009; Reuter et al., 2015; Goodwin et al., 2016) technologies, such as *Next Generation Sequencing* (NGS, Logares et al., 2012; Mardis, 2013), as well as later generations (Niedringhaus et al., 2011; Pareek et al., 2011; Mignardi and Nilsson, 2014; Heather and Chain, 2016; Mardis, 2016). For a sample of biological material, these technologies typically produce thousands to millions or even billions of short genetic sequences (also called "reads") with a length of some hundred base pairs length each. Over the past decades, decreasing costs and increasing throughput of sequencing technologies have caused an exponential growth in sequencing data (Muir et al., 2016), which has now passed the peta-scale barrier (Katz et al., 2022).

A major analysis step in metagenomic studies is to characterize the reads obtained from an environment by means of comparison to *reference sequences* of known species (Desai et al., 2012). A straight-forward way to accomplish this is to quantify the similarity between the reads and reference sequences. We obtain an indication of possible novelty if the sequence similarity to known species is low (Temperton et al., 2012; Peabody et al., 2015). However, such approaches do not provide the user with the evolutionary context of the read, and have been found to incorrectly identify sequences (Koski and Golding, 2001; Clemente et al., 2011; Mahé et al., 2017).

Instead, general phylogenetic methods can be used directly to classify and characterize the reads, providing highly accurate and information-rich results (Brady and Salzberg, 2009; Segata et al., 2012; Truong et al., 2015; Jamy et al., 2019; Beghini et al., 2021). However, trying to resolve the phylogenetic relationships between millions of short reads and the given reference sequences represents a significant computational challenge. Furthermore, as most phylogenetic methods require an *alignment* of sequences, metagenomic data can often not be used directly, as whole-genome reference data might not be available or computationally intractable. Instead, specific *marker genes* can be targeted (or filtered from the metagenomic data), which are genetic regions that are well-suited for differentiating between species (Ren et al., 2016). The use of marker genes to identify species is called *DNA (meta-)barcoding* (Deiner et al., 2017; Hebert et al., 2003; Savolainen et al., 2005; Kress and Erickson, 2008); see **Section 2.2** for details.

A powerful and increasingly popular class of methods to identify and analyze diverse (meta-)genomic (barcode) data is the so-called *phylogenetic placement* (or *evolutionary placement*) of genetic sequences onto a given fixed phylogenetic *reference tree*. By placing unknown, anonymous sequences (in this context called *query sequences*) into the evolutionary context of a tree, these methods allow for the taxonomic assignment of the sequences (i. e., the association of genomic reads to existing species, for example Auladell et al., 2019; Jamy et al., 2019; Hleap et al., 2021). Moreover, they can also provide information on the evolutionary relationships between these query sequences and the reference species/sequences, and thus go beyond simple species identification. Phylogenetic placement has found applications in a variety of situations, such as data cleaning and retention (Mahé et al., 2017), inference of new clades (Dunthorn et al., 2014; Bass et al., 2018), estimation of ecological profiles (Keck et al., 2018), identification of low-coverage genomes of viral strains (Mühlemann et al., 2020), phylogenetic analysis of viruses such as SARS-CoV-2 (Morel et al., 2020; Turakhia et al., 2021), and in clinical studies of microbial diseases (Srinivasan et al., 2012).

When analyzing the resulting data, there are two complementary interpretations of phylogenetic placement: 1) as a set of individual sequences, placed with respect to the reference phylogeny, e. g., for taxonomic assignment, phylo-geographic tracing, or even possible clinical relevance; 2) as a combined distribution of sequences on the tree, characterizing the sampled environment at a given point in time or space to examine the composition of a species community as a whole, for instance as a means of sample ordination and visualization, and association with environmental variables.

In this review, we provide an overview of existing methods to conduct phylogenetic placement, as well as post-analysis methods for visualization and knowledge inference from placement data. We also discuss some practical aspects, such as common pitfalls and misconceptions, as well as caveats and limitations of these methods. We mainly refer to metagenomic input data (or more accurately, metabarcoding data, see below for details) as it represents the most common use case, but also highlight some alternative use cases where phylogenetic placement is employed for other types of sequence data.

2 PHYLOGENETIC PLACEMENT

2.1 Overview and Terminology

The modern approach to phylogenetic tree inference is based on molecular sequence data, and uses stochastic models of sequence evolution (Arenas, 2015) to infer the tree topology and its branch lengths (Felsenstein, 2004; Yang, 2006). Note that the computational cost to infer the optimal tree under the given optimality criterion grows super-exponentially in the number of sequences (Felsenstein, 2004). In addition, large trees comprising more than a couple of hundred sequences are often cumbersome to visualize, rendering the approach challenging for current (e. g., metagenomic) large datasets. Furthermore, the lack of phylogenetic signal contained in the short reads of most HTS technology usually does not suffice for a robust tree inference (Dunthorn et al., 2014; Bininda-Emonds et al., 2001; Moret et al., 2002; von Mering et al., 2007). Hence, *phylogenetic placement* emerged from the demand to obtain phylogenetic information about sequence sets that are too large in number and too short in length to infer comprehensive phylogenetic trees (Matsen et al., 2010; Berger et al., 2011). In a metagenomic context, a set of sequences obtained from an environment such as water, soil, or the human body, is here called a *sample*. This is often the data that we intend to place, and might have further metadata associated with it, e. g., environmental factors/variables such as temperature or geo-locations where the sample was taken.

Generally, the input of a phylogenetic placement analysis is a phylogenetic *Reference Tree* (RT) consisting of sequences spanning the genetic diversity that is expected in the sequences to be placed into the tree. The tree can be rooted or unrooted; in the latter case however, a "virtual" root (or top-level trifurcation) is used in the computation as a fixed point of reference (Czech et al., 2019). Then, for a single sequence (e. g., a short read), in this context called a *Query Sequence* (QS), the goal of phylogenetic placement is to determine the branches of the RT to which the QS is most closely evolutionarily related. Note that the RT is kept fixed, that is, the QSs are not inserted as new branches into the tree, but rather "mapped" onto its branches. Hence, the phylogenetic relationships *between* individual QSs are not resolved.

This is the key insight that makes it possible to efficiently compute the placement of large numbers of QSs. By only determining the evolutionary relationship between the sequences of the RT and each individual QS, the process can be efficiently parallelized, and the required processing time scales linearly in the number of QS. Furthermore, this allows us to consider multiple branches as potential *Placement Locations* for a given QS, representing uncertainty in the placement, often expressed as a probability (or confidence) of the QS being placed on that branch. This uncertainty might result from weak phylogenetic signal, or might indicate some other issue with the data, as explained later. In Maximum-Likelihood (ML) based placement (see Section "Maximum Likelihood Placement" for details), these probabilities are computed as the *Likelihood Weight Ratio* (LWR) resulting from the evaluation of placing the QS attached to an additional (hypothetical) branch into the tree. Hence, for historic reasons, the probability of a placement location (one QS placed on a specific branch) is often called its LWR, and for a given QS, the sum of LWRs over all branches is 1 (equivalent to the total probability). See **Figure 1** for a glossary of the terminology, and see **Table 1** for an overview of different placement tools, and which of the aforementioned quantities they can compute.

In other words, phylogenetic placement can be thought of as an all-to-all mapping from QSs to branches of the RT, with a probability for each placement location, as shown in **Figures 2D,E**. We can however also interpret each such placement location *as if* it was an extra branch inserted into the RT, as shown in **Figures 2B,C**. In particular, maximum likelihood placement makes use of its underlying evolutionary model to also estimate the involved branch lengths that are altered through the insertion of a QS, see **Figure 2B** for details. This interpretation highlights the aspect of each individual QS being part of the underlying phylogeny. For example, this allows its taxonomic assignment to that clade of the reference tree where the QS shows the highest accumulated placement probability, as explained later.

2.1.1 Misconceptions

In the existing literature, and from our experience in teaching the topic as well as supporting the users of our software, some concepts of phylogenetic placement are not always well explained or understood. Although we have introduced these concepts above already, we briefly address two common misconceptions here, for clarity.

Likelihood Weight Ratio (LWR). The probability (confidence) that a QS is placed onto a particular branch (i. e., a single Placement Location).

Maximum Likelihood (ML). A statistical framework to estimate the parameters of a probability distribution.

Phylogenetic Placement. A family of methods to place a set of QSs onto the branches of an RT, by mapping each QS to one or several most likely Placement Locations on the tree.

Placement Location. An individual location (branch and position along the branch) onto which a specific QS has been placed; often annotated with a probability score (LWR) whose sum over all branches is 1 for that QS.

Query Sequence (QS). A single sequence to be placed into the RT. Typically, this is a short read or amplicon obtained via metabarcoding or metagenomics.

Reference Alignment (RA). The underlying multiple sequence alignment (MSA), based on a set of RSs, that is used in ML-based phylogenetic placement and was used to infer the RT.

Reference Sequence (RS). A typically high-quality sequence of a species or strain that is used as reference to compare the QSs against. Used to compute the RA and infer the RT.

Reference Tree (RT). The (bifurcating) phylogenetic tree used as a scaffold to place the QSs into, mostly inferred via ML methods.

FIGURE 1 | Glossary and abbreviations.

TABLE 1 | General purpose placement tools. This table compares the features of the general purpose (i. e., not use-case specific) phylogenetic placement tools. Columns are as follows. Alignment: Does the tool need the QSs to be aligned against the reference alignment? Multiple: Does the tool produce multiple placement locations per QS, or just a single (best) one? Uncertainty: Is there some measure of uncertainty (such as LWR) assigned to each placement location? Branch Length: Does the tool compute the involved branch lengths at each placement location for each QS.

Placement Tool	Alignment	Multiple	Uncertainty	Branch Lengths
PPLACER	yes	yes	yes	yes
RAXML-EPA	yes	yes	yes	yes
EPA-NG	yes	yes	yes	yes
RAPPAS	no	yes	yes	no
APPLES	no	no	no	yes
APP-SPAM	no	no	no	yes

Firstly, a common misconception is that the tree is amended by the QSs, that is, that new branches are added to the RT, and that the phylogenetic relationships of the QSs with each other are hence resolved. This is not the case; instead, the RT is kept fixed, the QSs are only aligned against the reference alignment, but not against each other (in ML placement), and the QSs are mapped only to the existing branches in the RT. This mapping *can* however be interpreted "as if" the QS was a new terminal node (leaf or tip) of the tree, usually inserted (or "grafted") into the branch with the most probable placement location, which can be useful in some applications.

Secondly, a further common misconception is that a QS is only placed onto a single branch, or that only the best (most likely) placement location is taken as the result for each placed QS. Instead, each branch is seen as a potential placement

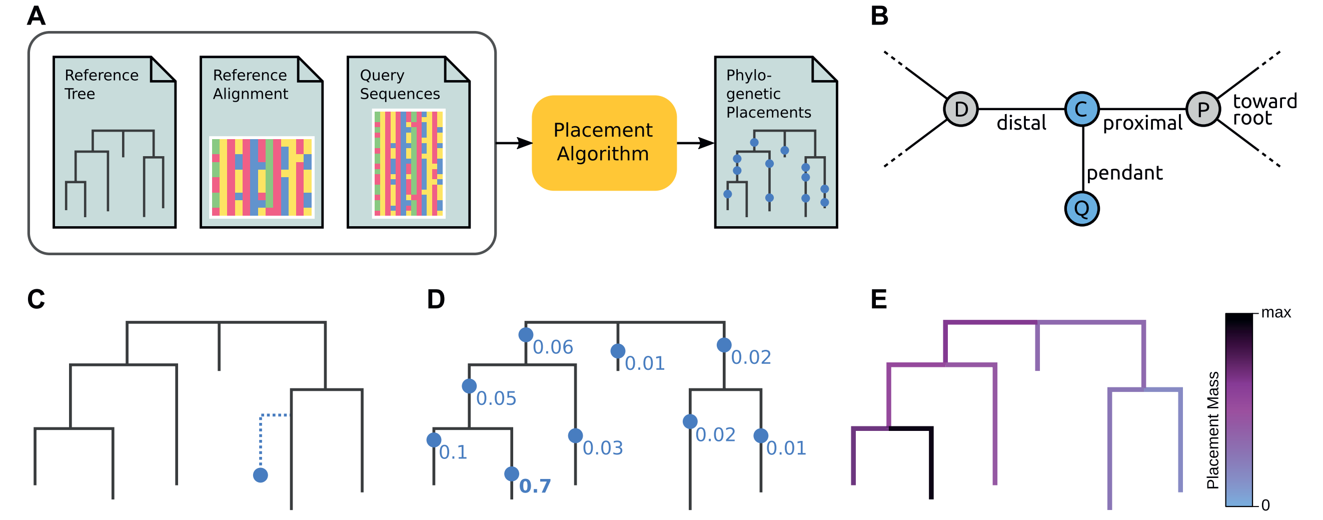


FIGURE 2 | Overview of phylogenetic placement. Here, we show the typical process, focused on ML-based placement. For the sake of simplicity, we here omit heuristics and other algorithmic improvements. Alignment-free placement works conceptually in an analogous way, but does not compute tree likelihoods. **(A)** Pipeline and data flow. The input to phylogenetic placement are the Reference Tree (RT) and its corresponding Reference Alignment (RA), as well as the set of Query Sequences (QSs) that we are interested in. The placement algorithm computes potential placement locations of a QS on the branches of the RT, for each QS in the input. **(B)** Terminology. The nodes D and P belong to the Reference Tree (RT). When placing a Query Sequence (QS), the branch between these nodes is split into two parts by a temporary new node C, which serves as the attachment point for another temporary new node Q that represents the QS. Note that these two new nodes are only conceptually inserted into the RT–they represent the mapping of the QS onto that branch. The *pendant* branch leads to Q. The original branch is split into the *proximal* branch, which leads towards the (possibly virtual) root of the RT, and the *distal* branch, which leads away from the root. **(C)** A single QS is placed onto a single branch (that is, one placement location). Vertical distances symbolize branch lengths. Note that the QS is located at a certain position along its Reference Tree branch (splitting that branch into distal and proximal parts), and has a (pendant) branch length of its own. At this step, ML-based placement computes the likelihood of the RT with the QS as a (temporary) extra branch. For one single QS, this step is then repeated at every branch of the tree. **(D)** Once the likelihoods of placing the QS onto every branch have been computed, the Likelihood Weight Ratios (LWRs) for this QS are computed. They express the confidence of placing the QS onto each branch, and can be interpreted as a probability distribution of the QS across the tree (and hence sum to one across all branches). In the image, we omit pendant branch lengths for the sake of simplicity. **(E)** The process is repeated for every QS, yielding an LWR-weighted "mapping" of each QS to each branch. We can visualize this as a cumulative distribution across all QSs on the tree, coloring branches according to the total sum of the LWRs at that branch over all QS. See **Figure 4A** for a real-world example of this.

location with a certain probability, which sum to one over the tree. It can however be useful to reduce the placement distribution of a QS to only its most probable placement location. Also, for practical reasons, typically not all locations are stored in the resulting file (or even considered in the computation by application of heuristics), as low probability locations can often be discarded to save storage space and downstream processing time; see Section "File Format" for details. Lastly, some placement methods do only output a single best placement, see **Table 1**.

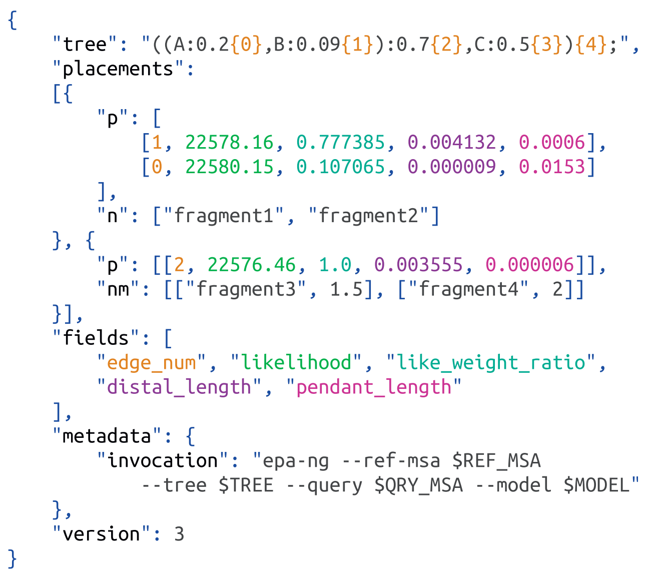
In summary, phylogenetic placement yields a distribution of potential locations of where a QS could be attached in the RT–but it does not extend the RT by the QS with an actual branch.

2.1.2 File Format

Placement data is usually stored in the so-called `jplace` format (Matsen et al., 2012), which is based on the `json` format (Bray, 2018; Douglas, 2018). See **Figure 3** for an example. It uses a custom augmentation of the `Newick` format (Archie et al., 1986) to store the reference tree, where each branch is additionally annotated by a unique edge number, so that placement locations can easily refer to the branches. For each QS (named via the list `"n"`), the format then stores a set of possible placement locations (in the list `"p"`), where each location is described by the values: 1) `"edge_num"`, which identifies the branch of this placement location, 2) `"likelihood"`, which is used by maximum likelihood based placement methods, 3) `"like_weight_ratio"` (LWR), which denotes the probability (or confidence) of this placement location for the given QS, 4) `"distal_length"` and 5) `"pendant_length"`, which are the branch lengths involved in the placement of the QS for the given placement location; see **Figure 2B** for an explanation of these lengths.

These five data fields are the standard fields of the `jplace` format; further fields can be added as needed. As noted above, typically not all placement locations for a given QS are stored in the file, as low probability placements unnecessarily increase the file size without providing substantial information; in that case, the sum of the stored LWR values might actually be smaller than 1.

The format furthermore allows for multiple names in the `"n"` list, as well as assigning a "multiplicity" to each such name (by using a list called `"nm"` instead of `"n"`). For instance, this allows to only store the placement locations for identical reads once, while keeping track of the original raw abundances of these reads or OTUs. A pair of a `"n"`/`"nm"` list and a `"p"` list is called a "pquery", and describes a set of placement locations for one or



```
{
    "tree": "((A:0.2{0},B:0.09{1}):0.7{2},C:0.5{3}){4};",
    "placements":
    [{
        "p": [
            [1, 22578.16, 0.777385, 0.004132, 0.0006],
            [0, 22580.15, 0.107065, 0.000009, 0.0153]
        ],
        "n": ["fragment1", "fragment2"]
    }, {
        "p": [[2, 22576.46, 1.0, 0.003555, 0.000006]],
        "nm": [["fragment3", 1.5], ["fragment4", 2]]
    }],
    "fields": [
        "edge_num", "likelihood", "like_weight_ratio",
        "distal_length", "pendant_length"
    ],
    "metadata": {
        "invocation": "epa-ng --ref-msa $REF_MSA
            --tree $TREE --query $QRY_MSA --model $MODEL"
    },
    "version": 3
}
```

FIGURE 3 | `Jplace` format for phylogenetic placement. The exemplary file consists of a reference `"tree"` in a custom `Newick` format that annotates edge numbers in curly brackets, followed by two pqueries, which is the term for combined lists of sequence names and their placement locations. The first pquery contains two placement locations (`"p"`) for two query sequences (`"n"`), and the second contains a single location (`"p"`) for two other sequences including their multiplicities/abundances (`"nm"`). The order to interpret the values per location is given via the `"fields"` list, and highlighted by colors here; additional `"metadata"` and a `"version"` of the file format can be given. Example adapted from (Matsen et al., 2012).

more (identical) QSs. This structure is then repeated for each QS that has been placed.

To our knowledge, the GENESIS library (Czech et al., 2020) is the only general purpose toolkit for working with, and manipulating, placement data in `jplace` format. It also incorporates many of the downstream visualization and analysis techniques we describe later on. Some other tools that offer basic capability to work with `jplace` files are BoSSA (Lefeuvre, 2018), GGTREE (Yu et al., 2017), and TREEIO (Wang et al., 2020), all of which can read `jplace` files for processing in R.

With the release of several placement tools that do not use the ML framework, see Section "Distance-Based Placement", the `jplace` file format (Matsen et al., 2012) may require an update. The standard is written currently (as of version 3) with placement properties such as branch lengths and likelihood scores in mind, which do not translate well to other types of placement algorithms (pers. comm. with S. Mirarab, July 2020). Furthermore, it might be helpful to support sample names, multiple samples per file, and additional per-sample or even per-query annotations and other metadata in the file format. Being based on `json`, this can already be achieved now by adding these entries ad-hoc, but would lack support by parsers if not properly standardized.

2.2 Types of Query Sequences

In principle, any type of genetic sequence data can be subjected to placement, as long as the reference sequences span the genomic regions where the query sequences originate from. Apart from the availability of suitable reference sequences used to construct a reference tree (see Section "Sequence Selection"), the primary limiting factor is the extent to which a given placement tool supports the data. Currently, the majority of placement tools supports nucleotide (DNA/RNA) and amino acid (protein) data. Many placement methods require query reads to be aligned to the reference, i. e. they need to be homologs.

2.2.1 Metabarcoding and Amplicons

For the above reasons, a common approach to obtain sequences is *metabarcoding* (Deiner et al., 2017; Hebert et al., 2003; Savolainen et al., 2005; Kress and Erickson, 2008). In metabarcoding, one or several *marker* or *barcoding* genes, such as 16S (Weisburg et al., 1991), 18S (Meyer et al., 2010), ITS, COI, etc. (Woese and Fox, 1977; Woese et al., 1990; Ji et al., 2013; Sunagawa et al., 2013) are typically chosen to compute the reference alignment, and appropriate primers are selected to enable metabarcode sequencing of the sample (Deiner et al., 2017). A marker gene should be universally present in the studied organisms, and ideally should only occur once in the genome of each organism (Dunthorn et al., 2014; Nguyen et al., 2014), i. e., be single-copy. In practice, marker genes often occur multiple times per genome, possibly requiring the need for copy number correction. A marker gene should exhibit sufficient between-species variation to distinguish them from each other, but show low within-species variation (Kress and Erickson, 2008). Using a metabarcoding approach has several advantages: it targets loci of interest and focuses the sequencing effort there (incidentally also limiting the size of the reference MSA), barcoding genes are typically well suited for phylogenetics (stable regions to aid alignment paired with variable regions to discriminate organisms), and the approach is generally cost-effective. Such approaches use amplicon sequencing (Peabody et al., 2015; Hugerth and Andersson, 2017), wherein only DNA originating from the targeted region is amplified using the Polymerase Chain Reaction (PCR, Bartlett and Stirling, 2003), thus yielding the subsequent sequencing of any remaining DNA fragments from other regions highly improbable. The resulting amplicon sequences have been shown to be well-suited for phylogenetic placement (Mahé et al., 2017; Janssen et al., 2018).

However, PCR-based amplifications are known to introduce biases in the abundance of the sequencing reads, as some fragments may be copied with a higher likelihood than others (Morgan et al., 2010; Logares et al., 2014). Similarly, a further bias that skews abundance results exists as different organisms may have a different number of copies of the targeted gene, ranging from single copies to 15 copies, depending on the organism (Lee et al., 2009). Some methods exist that attempt to account for copy number bias (Kembel et al., 2012; Angly et al., 2014; Pereira-Flores et al., 2019) as well as for PCR amplification bias (Love et al., 2016; Silverman et al., 2021).

When an untargeted sequencing approach is chosen instead (such as shotgun metagenomic sequencing), using a broader scope for the reference sequences may be advisable, such as using whole genome data. This might only be feasible for small genomes such as some viruses or mitochondrial DNA. Alternatively, a sensible approach is to filter out any reads that did likely not originate from the genetic regions that constitute the reference alignment. This can be achieved, for example, using HMMSEARCH from the HMMER-package (Eddy, 1995; Eddy, 1998), which allows the user to obtain a list of reads that have an alignment score above a given threshold. Similarly, so-called $_{\mathrm{mi}}$tags (Logares et al., 2014) represent a shotgun-based alternative to amplicon sequencing.

Recently, placement methods have emerged that do not require the alignment of query sequences to a reference, and some do not even require the references to be aligned against each other (see Section "Distance-Based Placement"). However, establishing that query reads and reference sequences are homologous is still necessary.

2.2.2 Sequencing Technologies

A further consideration is the choice of sequencing technology, with the primary property being the length of the resulting sequencing reads. So far, the vast majority of studies utilizing phylogenetic placement have relied on short-read sequencing technologies such as NGS, using by now well established protocols to perform broad low-cost sequencing (van Dijk et al., 2014). However, this approach produces very short (150-400 nucleotide) reads, that typically only cover fragments of a reference gene. For universal single-copy markers, this can limit their applicability to phylogenetics due to the lower information content. However, the approach has been applied successfully to other types of data (Piredda et al., 2021; Cardoni et al., 2022).

More recent sequencing technologies, called third generation sequencing, or long-read sequencing (LRS), yield individual reads that cover entire genes, or even entire genomes (Amarasinghe et al., 2020). While placement was originally developed for short read sequencing, longer read lengths typically increase the phylogenetic signal contained in reads, thus increasing the reliability of phylogenetic methods. Indeed, such sequence data have been shown to overcome this fundamental hurdle to phylogenetically resolving the relationships between query sequences that originally gave rise to phylogenetic placement (Jamy et al., 2019).

An emerging third way to obtain longer reads is to combine short reads into longer so-called Synthetic Long-Reads (SLRs), which have been used successfully to characterize metagenomes (Sharon et al., 2015; Kuleshov et al., 2016) and which improve upon short-read metabarcoding approaches for taxonomic classification (Jamy et al., 2019; Ritter et al., 2020; Jeong et al., 2021).

Related to this is the assembly of genomes from metagenomic sequences (MAGs, Tyson et al., 2020), a technique which has recently been shown to reliably obtain multi-loci data from highly diverse data sources and environments (Parks et al., 2017). MAGs may be a beneficial input for phylogenetic placement, especially for methods that are able to directly handle such assemblies in their entirety (Metin et al., 2021). Other placement methods may also benefit from sequence assemblies when combined with marker gene extraction, as it potentially increases the number of viable query sequences.

2.2.3 Clustering

Once the wet-lab sequencing strategy has been determined, a user eventually obtains a (typically large) set of sequences. After quality control, a potential next step is to consider if, and how, to cluster these raw sequences in order to reduce the amount of data that has to be processed, often at the cost of losing information. Common choices include clustering by similarity threshold ($\geq 97\%$) resulting in Operational Taxonomic Units (OTUs, Blaxter et al., 2005; Edgar, 2010; Fu et al., 2012; Westcott and Schloss, 2015; Rognes et al., 2016), more strictly based on single nucleotide differences resulting in Amplicon Sequencing Variants (ASVs, Callahan et al., 2016), or more recent alternatives such as SWARM clustering (Mahé et al., 2021). These methods are most commonly used for clustering reads from marker regions, and hence applicable in the placement context; for a comprehensive review of clustering methods, see (Zou et al., 2020).

If possible, it is recommended to avoid clustering, in order to retain potential phylogenetic signal; this choice however also depends on study design and goals. However, even if sequences are not clustered, we strongly recommend dereplication, that is, removal of exact (strict) duplicates of sequences, to avoid unnecessary redundant computations. For the same reason, sequence dereplication is also useful when pooling the sequences from multiple samples together and placing the resulting set via a single placement run. Tools that offer this capability include USEARCH (Edgar, 2010), and VSEARCH (Rognes et al., 2016), as well as the placement-specific CHUNKIFY command in GAPPA (Czech et al., 2020).

2.2.4 Outgroup Rooting

Finally, an often overlooked source of query sequences are high-quality reference sequence databases. Here, the use-case of placement shifts away from taxonomic assignment: instead such data can be used to attempt an outgroup rooting of an existing tree, using already classified sequences (Hubert et al., 2014; Liede-Schumann et al., 2020; Morel et al., 2020). The result of placement, in this case, is a set of suggested branches on which to root the tree, including a probability estimate for each root placement onto each branch (Liede-Schumann et al., 2020).

2.3 Reference Sequences, Alignment, and Tree

The phylogenetic reference tree (RT), inferred from a set of reference sequences (RSs) using their alignment (*Reference Alignment*, RA), is the foundation and scaffold for conducting phylogenetic placement. Ideally, to avoid duplicating work, to ensure high quality, and to provide stable points of reference for comparison between studies, suitable reference trees should be provided by the respective research/organismal communities. First efforts for microbial eukaryotes are on their way (Berney et al., 2017; Del Campo et al., 2018; Rajter and Dunthorn, 2021; Rajter et al., 2021), although some of these are not designed explicitly for phylogenetic placements, but more taxonomic groups will follow. Recently, efforts have also been made to produce reference trees for higher order animals, such as fish (Collins et al., 2021). As references are however not yet available for all taxonomic groups, we here provide an overview of the process (see also Mahé et al., 2017, Rajter et al., 2021, for practical examples).

2.3.1 Sequence Selection

As phylogenetic placement cannot infer evolutionary relationships below the taxonomic level of the reference tree, the first step is the selection of suitable RSs, which should 1) cover the diversity that is expected in the query sequences (QSs), and 2) be well-established and representative for their respective clades to facilitate meaningful interpretation. In order to capture unexpected diversity and potential outliers, it can be advantageous to include a wider range of sequences as well (Mahé et al., 2017), or to run preliminary tests and filtering (placement- or similarity-based) with a broad reference to ensure that all diversity in the QSs is accounted for.

In many cases, the selection process is (unfortunately) labor-intense, as it requires hand-selecting known sequences from reference databases such as SILVA (Pruesse et al., 2007; Quast et al., 2013; Yilmaz et al., 2014), NCBI (Benson et al., 2009; Sayers et al., 2009), GREENGENES (DeSantis et al., 2006; McDonald et al., 2012), or RDP (Wang et al., 2007; Cole et al., 2014). This manual process however also often provides the highest quality, and allows to optimally assemble the RSs for a given project. See also (Balvočiūtė and Huson, 2017) for a comparison of these databases.

Important selection criteria are the number of sequences to be selected, as well as their diversity; both of which depend on the study design and goals. Generally, a number of RSs in the order of hundreds to a few thousands has shown to provide enough coverage for most QS datasets, while still being small enough to properly visualize their phylogeny and to conduct all necessary computations in reasonable time. Often, it is sufficient to include a single species to represent a whole clade (Rajter and Dunthorn, 2021). Depending on the types of downstream analyses, it can be a disadvantage to select sequences that are too similar to each other (i. e., closely related species, or different strains of the same species), as this can spread the placement distribution across nearby branches. In other words, placements with similar probability in many branches are mostly a consequence of reference alignment regions for which large subtrees contain (almost) identical sequences. This is however expected when conducting taxonomic assignment at species or below-species level, and the reference should be built with the targeted taxonomic resolution in mind.

On the other hand, if the QSs contain enough phylogenetic signal (e. g., when using long reads, whole genome data, or when the target gene has sufficient variability), including multiple representatives of a taxonomic group might allow to obtain more finely resolved placements. For example, in short genomes such as HIV or arthropod mitochondria, where mutations are not concentrated in specific regions but spread all over the genome, reads matching a reference alignment region likely show a decent amount of variation, making placements exploitable (Linard et al., 2020).

Lastly, the RSs need to at least span the genomic region that the QSs come from. For a more robust inference of the RT however, it can be advantageous to include a larger region with more phylogenetic signal. Theoretically, if one wanted to place shotgun sequences from entire genomes, whole-genome RSs would be needed.

As an alternative to manual selection, the Phylogenetic Automatic Reference Tree (PhAT, Czech et al., 2018) is a method that uses reference taxonomic databases to select suitable RSs which represent the diversity of (subsets of) the database. In cases where taxonomic resolution at the species-level does not require expert curation, the PhAT method can provide a basis for rapid data exploration, and help to obtain an overview of the data and its intrinsic diversity.

2.3.2 Reference Alignment Computation

Next, for ML-based tree inference and placement, the RSs need to be aligned against each other to obtain the reference alignment (RA). Typically, this is conducted with *de novo* multiple sequence alignment tools such as T-COFFEE (Notredame et al., 2000), MUSCLE (Edgar, 2004), MAFFT (Katoh et al., 2002), and others; see (Kemena and Notredame, 2009; Pervez et al., 2014; Chatzou et al., 2016) for reviews. Recently, MUSCLE v5 introduced an interesting new approach that generates alignment ensembles to capture alignment uncertainty (Edgar, 2021, preprint). In the ML framework, the QSs also need to be aligned against the RA, see next section.

2.3.3 Tree Inference

Finally, given the RA, a phylogenetic tree of the RSs is inferred, which is henceforth used as the reference tree (RT); see (Kapli et al., 2020) for a general review on this topic. In theory, any method that yields a fully resolved (bifurcating) tree is applicable, e. g., neighbor joining (Saitou and Nei, 1987), maximum parsimony (Sankoff, 1975), or Bayesian inference (Holder and Lewis, 2003; Yang, 2006). In practice however, maximum likelihood (ML) tree inference (Yang, 2006; Dhar and Minin, 2016) is preferred, in particular when using ML-based placement, as otherwise inconsistencies in the assumed models of sequence evolution can affect placement accuracy. To this end, common software tools include IQ-TREE (Nguyen et al., 2015), FASTTREE2 (Price et al., 2010), and RAxML (Stamatakis, 2014; Kozlov et al., 2019); see (Zhou et al., 2018) for a review and evaluation of ML-based tree inference tools. An open research question in this context is how to incorporate uncertainty in the tree inference (and in the alignment computation) with phylogenetic placement (Huelsenbeck et al., 2001; Ronquist, 2004; Edgar, 2021).

2.3.4 Alignment of Query Sequences

For many placement methods, the query sequences need to be aligned against the reference alignment. In principle, *de novo* alignment methods can be deployed to obtain a comprehensive alignment of both the reference and query sequences. These tools are however not intended for HTS data, and are not well suited for handling the heterogeneity of phylogenetic placement data, with (typically) longer, curated, high-quality reference sequences, and short lower-quality reads (query sequences).

Hence, with the rise of high-throughput sequencing, specialized tools have been developed that extend a given (reference) alignment without fully recomputing the entire alignment. In the context of phylogenetic placement, there are two additional advantages that can be exploited to improve efficiency: 1) query sequences only need to be aligned against the reference, but not against each other (as their phylogenetic relationship is not resolved during placement), and 2) insertions into the reference that result from aligning a QS against the reference can be omitted as they do not contain any phylogenetic signal for the placement of the QS.

In the simplest case, only the reference alignment and query sequences are required as input. For instance, the hmmalign command of HMMER (Eddy, 1995; Eddy, 1998) can align query sequences to the reference alignment using a profile Hidden Markov Model (HMM) built from the reference alignment. Note that the option `-m` has to be set in order to not insert columns of gaps into the reference. Alternatively, the MAFFT command `--addfragments` (Katoh and Frith, 2012) uses an internally constructed guide tree built from a pairwise distance matrix of the reference alignment to aid the alignment process; here, the option `--keeplength` has to be set to not add columns of gaps to the reference.

Furthermore, the PAPARA tool (Berger and Stamatakis, 2011; Berger and Stamatakis, 2012) can be used that was specifically developed to target phylogenetic placement. It takes the RT as additional input, and uses inferred ancestral sequences at the inner nodes of the tree to improve the alignment process. Here, the option `-r` has to be set to not insert columns of gaps into the reference. Similarly, PAGAN (Löytynoja et al., 2012) also utilizes the information in the reference tree, but it *does* extend the reference alignment with gaps as needed for the query sequence, causing higher computational effort during placement.

Note that typically, read mapping tools such as BOWTIE2 (Langmead and Salzberg, 2012) or BWA (Li and Durbin, 2009; Li and Durbin, 2010) are not recommended for phylogenetic placement, as they expect low-divergent sequences as input, e. g., from a single species.

2.4 General Purpose Placement Methods

Once initial tasks such as reference tree creation and sequence alignment are completed, the actual placement can commence. There exist several distinct algorithmic approaches for conducting the core part of phylogenetic placement, which we introduce here; see **Table 1** for an overview.

2.4.1 Maximum Likelihood Placement

Maximum Likelihood (ML) is a statistically interpretable and robust general inference framework, and one of the most common approaches for phylogenetic tree inference (Felsenstein, 2004; Yang, 2006; Dhar and Minin, 2016). It works by searching through the super-exponentially large space of potential tree topologies for a given set of sequences (taxa), and computing the phylogenetic likelihood of the sequence data of these taxa being the result of the evolutionary relationships between the taxa as described by each potential tree, while also computing branch lengths of the tree. The result of this inference is the tree topology one is able to find using some heuristic search strategy that best (most likely) "explains" the underlying sequence data. Due to the NP-hardness of the tree search problem, the best tree one can find might not be the globally best one.

To calculate this likelihood, ML methods use statistical models of sequence evolution that describe substitutions between sequences (insertions and deletions are mostly ignored; it is hence also called a substitution model), see (Arenas, 2015) for a review. Consequently, the estimated parameters of these models are an inherent property of the resulting phylogenetic tree. The choice of model parameters also directly informs the specific branch lengths of a tree, interpreting a tree under a different set of model parameters thus may lead to inconsistencies. Therefore, under the ML framework, we strongly recommend to use the same substitution model and parameters for tree inference and for phylogenetic placement.

Based on the general ML tree inference framework, ML-based phylogenetic placement works in two steps: First, the QSs are aligned against the RA as described above, and second, using the resulting comprehensive alignment with both reference and query sequences, the QSs are placed on the RT using the maximum likelihood method to evaluate possible placement locations (Matsen et al., 2010; Stark et al., 2010; Berger et al., 2011).

Standard methods used in ML tree inference use search heuristics to explore some possible tree topologies for a given set of sequences. Instead, for a given QS, ML-based placement only searches through the branches of the reference tree (RT) as potential placement locations for the QS. That is, each branch of the RT is evaluated as a placement location, and branch lengths of the involved branches are optimized, following the same approaches as for *de novo* tree inference. However, the distal and proximal branch lengths of the placement (see **Figure 2B** for details) are typically re-scaled, so that their sum is equal to the original branch length in the RT. Finally, the phylogenetic likelihood of the tree with the QS amended as a temporary extra taxon is calculated.

For each QS and each branch of the RT, this process yields a likelihood score (which is stored in the `jplace` format, see Section "File Format"). The Likelihood Weight Ratio (LWR) of a placement location is then computed as the ratio between this likelihood score and the sum over all likelihood scores for the QS across the entire tree (von Mering et al., 2007; Strimmer and Rambaut, 2002). These likelihood scores sum to one across all branches, and hence express the confidence (or probability) of the QS being placed on a given branch.

The first two tools to conduct phylogenetic placement in an ML framework were the simultaneously published (as preprints) PPLACER (Matsen et al., 2010) and RAxML-EPA (Berger et al., 2011). Both build on the same general ML concepts, but use different strategies for improving computational efficiency, e. g., by heuristically limiting the number of evaluated branches (potential placement locations). Additionally, PPLACER offers a Bayesian placement mode. The more recent EPA-NG (Barbera et al., 2018) tool combines features from both PPLACER and RAxML-EPA, is substantially faster and more scalable on large numbers of cores, and hence is the recommended tool for ML-based placement.

2.4.2 Ancestral-Reconstruction-Based Placement

Recently, multiple methods were introduced that do not rely on aligning query sequences to a reference MSA. The first such group of methods is based on reconstructing ancestral states at interior nodes of the reference tree, again using an ML framework. From these ancestral sequences, *k*-mers are generated and associated with the branches of the reference tree. Subsequently, phylogenetic placement is performed by comparing the constituent *k*-mers of a QS with the set of *k*-mers indexing the reference tree branches, thereby obviating the need for QS alignment. This is the general approach used in both RAPPAS (Linard et al., 2019) and LSHPLACE (Brown and Truszkowski, 2012).

It should be noted that using this procedure, distal and pendant branch lengths of a given RT branch are determined during the association of *k*-mers with RT branches, meaning that all placements on a given branch have the same fixed location. This means that an additional step to conduct branch length optimization that is not directly offered by RAPPAS or LSHPLACE may be required to obtain more realistic placement branch lengths. RAPPAS however does produce multiple placements per QS and calculates a confidence measure akin to the LWR, yielding a distribution for placing a single QS onto different branches of the tree.

2.4.3 Distance-Based Placement

Finally, the most recent placement approaches utilize methods from distance-based phylogenetic inference.

For example, APPLES (Metin et al., 2019) is based on the least-squares criterion for tree reconstruction (Felsenstein, 2004). For a given tree, the least-squares method calculates the difference between the pairwise sequence distances and the pairwise patristic distances (i. e., the path lengths between two leaves). A least-squares optimal tree is the tree for which this difference is minimized. In APPLES, this criterion is used to score possible placement locations of a QS on an existing tree, returning the branch which minimizes the between-distances difference. A key advantage of the least-squares approach is its ability to efficiently handle reference trees with hundreds of thousands of leaves, which is currently not computationally feasible using ML methods. Further, the method does not require an alignment of the sequences involved, requiring only a measure of pairwise distance between them. Note however that as these methods still require a reference tree, computing a reference MSA may still be needed, unless the tree is inferred via distance-based methods as well. Consequently, even unassembled sequences, such as genome skims (Dodsworth, 2015), may be used both as reference and query sequences. Recently, an updated APPLES-2 was published that further improves upon the scalability and accuracy of the tool (Metin et al., 2021). Note also that APPLES can take as input, but does not require, aligned sequences.

The most recent alignment-free method is APP-SPAM (Blanke and Morgenstern, 2021). It utilizes the concept of a spaced-word, which can be understood as a type of *k*-mer for which only some characters have to be identical for two subsequences to be considered as having the same *k*-mer. This relaxed equality definition is informed by a binary pattern, indicating for each site of a spaced word whether it should be taken into account (1) or disregarded (0). Building on this, the tool calculates pairwise distances between a QS and the RSs based on the number of shared spaced-words. Subsequently, the tool identifies the placement branch of a QS as either the terminal branch of the closest RS, or the branch leading to the parental node of the LCA of the two closest RSs, depending on the strength of the signal of the closest RS. Notably, APP-SPAM is able to provide both distal and pendant branch lengths for the placements it produces, and does so using an estimated phylogenetic distance (the Jukes-Cantor distance, Jukes and Cantor, 1969). Note that both APPLES and APP-SPAM only produce a single placement per QS and can therefore not offer statistical measures of placement uncertainty such as the LWR.

Generally, distance-based placement methods produce results with lower accuracy compared to ML-based placement, though this gap appears to be narrowing. These newer approaches do however expand the scope of placement to sizes of reference trees, and lengths of reference sequences, that are orders of magnitude larger than what is currently possible with ML methods.

2.5 Application-Specific Placement Methods

Several additional placement methods exist. We provide a survey of these in this section. The placement methods covered in this section set themselves apart through their more specific use-cases, however this does not imply that their scope of use is necessarily limited.

2.5.1 Viral Data

A particularly challenging use case for phylogenetic methods is the investigation of viral data, with a highly relevant example coming from the SARS-CoV-2 pandemic. Due to the dense sampling involved in studying such viral outbreaks, differences between individual taxa in a prospective tree may only be due to a very low number of, or even single, mutations. Consequently the amount of phylogenetic signal is generally very low, complicating tree reconstruction (Morel et al., 2020). Yet, distinguishing between major viral variants and identifying them precisely from a given clinical sample is crucial for epidemiological studies. In this context the USHER software was introduced that specifically focuses on phylogenetic placement of SARS-CoV-2 sequences (Turakhia et al., 2021). In contrast to ML methods, USHER uses a Maximum Parsimony (MP) approach, and does not operate on the full sequence alignment. This allows the method to focus directly on individual mutations, and consequently only use a fraction of the runtime and memory footprint of conventional ML placement methods. Note that the accuracy of MP-based phylogenetic methods can suffer when one or more lineages in the tree have experienced rapid evolution that results in long branch lengths. In such cases MP may incorrectly determine such lineages to be closely related, an effect termed *long branch attraction* (Felsenstein, 1978; Bergsten, 2005). While this is less of an issue for very closely related sequences such as SARS-CoV-2 or other (but not all) viral data, it may yield the application of such approaches to different types of data more challenging.

2.5.2 Gene Trees

In principle, all placement methods aim to provide the location of a QS on a phylogeny that accurately reflects the underlying pattern of speciation, i. e., the *species tree*. In practice, the reference tree is typically only inferred on a single gene (16S, 18S, ITS, etc.), yielding a *gene tree* which may substantially differ from the species tree, called gene-tree *discordance* (Degnan and Rosenberg, 2009). Alternatively, we may have multiple such gene trees that induce a species tree, and subsequently want to perform query placement onto the species tree via placement onto the constituent gene trees (Sunagawa et al., 2013). Currently, only two placement methods are able to handle such cases: INSTRAL and DEPP. INSTRAL (Rabiee and Mirarab, 2019) performs placement of QSs for a species tree induced by a set of gene trees. It does so by first placing into the individual gene trees using existing ML placement methods, then re-inferring the species tree from the extended gene trees. In contrast to this, DEPP (Jiang et al., 2021, preprint) only considers the problem of discordance between a gene tree and its species tree and attempts to account for this during the placement into the species tree. The approach is based on a model of gene tree discordance learned from the data using deep neural networks that yields an embedding of given sequences into a euclidean space. Incidentally, this makes DEPP the first and so-far only phylogenetic placement method to incorporate machine learning. DEPP then uses the pairwise distances that result from the embedding of both reference and query sequences as input to APPLES, which computes the least-squares placement of the QSs.

2.5.3 Other Use Cases

Some further tools make application-specific usage of placement. The first pertains to the specific case of samples containing sequences from exactly two organisms, and the task of identifying their respective known reference organisms. The tool MISA was developed with this specific use-case in mind (Balaban and Mirarab, 2020).

The second relates to either placing morphological sequences from fossils typically represented by binary characters (presence/absence of a trait) or Ancient DNA (aDNA) sequences. Placing ancient DNA sequences is generally challenging for analysis because of the high degree of degradation due to the age of the DNA molecules, generally shorter read lengths ranging between 50 and 150 base pairs, and post-mortem deamination (Hofreiter et al., 2001). The PATHPHYNDER tool aims to solve this use-case (Martiniano et al., 2022, preprint). Like USHER, PATHPHYNDER operates on nucleotide variants, focusing on single nucleotide polymorphisms. Furthermore, phylogenetic placement has been used for placement of fossils (Berger and Stamatakis, 2010; Bomfleur et al., 2015) using morphological data. This approach uses the maximum likelihood framework to use the signal from mixed morphological (binary) and molecular partitions in the underlying MSA.

Lastly, phylogenetic placement has also been proposed as a way to perform OTU clustering. The HMMUFOTU (Zheng et al., 2018) tool implements this specific use-case, along with automated taxonomic assignment (see also Section "Taxonomic Classification and Functional Analysis"). A unique characteristic in comparison to other placement tools is that HMMUFOTU also performs QS alignment and uses this information to pre-select promising placement locations.

2.6 Workflows Based on Phylogenetic Placement

Over the last decade, several pipelines have been published that use phylogenetic placement tools as their core method, building on it and using its result in various ways.

2.6.1 Automated Analysis Pipelines

One class of placement pipelines focus on simplifying the overall use of placement methods, typically providing the user with the option to use a pre-computed reference tree, obviating the need for manual selection of reference taxa (Stark et al., 2010; Carbone et al., 2016; Douglas et al., 2018; Carbone et al., 2019; Douglas et al., 2020; Erazo et al., 2021; Sempéré et al., 2021). A number of these pipelines also automate the generation of key metrics and downstream analysis steps. Among these pipelines, of particular note is PICRUST2 (Douglas et al., 2018; Douglas et al., 2020), which stands out for accounting for 16S copy number correction, and providing the user with a prediction of the functional content of a sample. Similarly, PAPRICA (Erazo et al., 2021) is a pipeline that computes metabolic pathway predictions for bacterial metagenomic sample data.

2.6.2 Divide-And-Conquer Placement

A further key challenge for existing phylogenetic placement tools is scalability with regards to the size of the reference tree. While more recent methods have shown significant improvements in both the memory footprint and execution time required when placing QSs on reference trees on the order of 10^5 reference taxa (see Section "Distance-Based Placement"), such input sizes remain extremely challenging for ML-based placement methods. A number of workflows have been proposed to scale existing placement methods for this use-case by splitting up the reference tree into smaller subtrees on which phylogenetic placement is then performed, creating a divide-and-conquer approach to phylogenetic placement (Mirarab et al., 2012; Czech et al., 2018; Czech et al., 2020; Koning et al., 2021; Wedell et al., 2021). These approaches vary primarily in how they select subtrees. SEPP (Mirarab et al., 2012) and PPLACERDC (Koning et al., 2021) generate a subtree based on the topology of the reference tree. SEPP is a general boosting technique in particular for highly diverse reference trees (Liu et al., 2012; Mirarab et al., 2012). Further, a multi-level placement approach exists (Czech et al., 2018; Czech et al., 2020), which first places onto a broad RT, and then extracts QSs in pre-selected clades of that RT to place them again onto clade-specific high-resolution RTs. Finally, PPLACER-XR (Wedell et al., 2021) selects a set of neighboring reference branches based on similarity to each query sequence, out of which it creates a subtree. Note that in this case, when decomposing the reference tree differently for every query sequence, scalability with regards to the number of query sequences is severely reduced.

A central promise of placement on very large trees is to simplify the curation and engineering tasks involved in creating a reference tree, as here a typical challenge is to decide which taxa to include in the tree. If placement can instead be performed on a tree encompassing an entire database, the curation challenge is circumvented. However, as another common issue with reference tree generation is the inclusion of overly similar reference sequences resulting in unclear or fuzzy placement signal, divide-and-conquer placement approaches may not be sufficient on their own.

2.6.3 Evaluation of Placement Tools

Lastly, PEWO is an extensible testing framework specifically aimed at benchmarking and comparing different phylogenetic placement softwares (Linard et al., 2020). It includes a wide range of datasets and thus provides an important resource for identifying which placement tool is best suited for specific use-cases by evaluating the accuracy of existing tools, given some dataset. PEWO does so using a pruning-based evaluation procedure, where a subset of leaves is removed from a reference tree. This subset of sequences is subsequently used as input QSs for placement. The accuracy of a placement is calculated as the number of nodes between the best placement location, and the original location of the QS on the reference tree (called the node distance). This basic approach is used for evaluation in most publications that introduce new placement approaches. Note that the node distance measures two sources of error: error introduced by the placement algorithm, and error introduced by the pruning of the reference tree. In contrast to this, the "delta error" used in the evaluation of APPLES measures the additional error introduced through placement, in addition to the error introduced by the process of altering the reference tree through pruning (Metin et al., 2019). This new metric is however not yet included in the PEWO workflow. Nevertheless, the usefulness of a comprehensive and standardized testing framework cannot be emphasized enough, as it substantially facilitates further advancement and standardization in the field and the development of novel methods.

3 VISUALIZATION AND ANALYSIS

As mentioned before, there are two ways to conceptualize phylogenetic placement: 1) as an assignment (or mapping) of individual sequences to the branches of a phylogeny, usually taking the (n-)most likely placement location(s) of each sequence, or 2) as the distribution of all sequences of a sample across the tree, taking their respective abundances and placement probabilities into account. The former is similar to taxonomic assignment, but with full phylogenetic resolution instead of resolution at the taxonomic levels only, while the latter focuses on, e. g., species communities and their diversity as a whole. In the following we provide an overview of analysis methods that make use of such data.

3.1 Abundances and Multiplicities

In both interpretations, an important consideration is whether to take sequence abundances into account. When working with strictly identical sequences, or sequences resulting from some (OTU) clustering, the number of occurrences of each sequence or size of each cluster can be used as additional information for interpreting, e. g., community structure. On the one hand, including their abundances with the placement of each sequence yields information on how prevalent the species of these sequences are; for example, this can provide insight into the key (most abundant) species in environmental samples. On the other hand, dropping abundances and instead considering each sequence once (as a singleton) is more useful for estimating total diversity and taxonomic composition. For example, this way the number of *distinct* sequences can be regarded as a proxy for the number of species that are present in a sample. Whether to include abundances should hence be decided depending on the type of analysis conducted.

In the `jplace` format, these abundances can be stored as the so-called "multiplicity" of each placement (Matsen et al., 2012), in the `"nm"` data field. Unfortunately, the `fasta` (Pearson and Lipman, 1988) and `phylip` (Felsenstein, 1981) formats used as input to placement do not natively support abundance annotations, and current placement tools often do not handle them automatically, meaning that the information can be lost. However, the CHUNKIFY workflow (Czech et al., 2018; Czech et al., 2020) mentioned in Section "Clustering" takes abundances into account and annotates them as multiplicities in the resulting `jplace` file. Furthermore, GAPPA (Czech et al., 2020) offers a command to edit the multiplicities as needed, for example setting them post-hoc to the initial sequence abundance determination.

3.2 Visualization

Prior to more in-depth analyses, a first step in most workflows is a visualization of the immediate results. Following the two interpretations of phylogenetic placement (and hence, depending on the research question at hand), there are several ways to visualize placement results.

First, individual placements can be shown as actual branches attached to the RT, e. g., **Figure 2C**. Typically, only the most likely placement location per sequence is used for this, in order to avoid cluttering of the tree; this hence omits the information about uncertainty. This can be conducted by generating trees from placement results, e. g., in `Newick` format. Tools to this end are GAPPA (Czech et al., 2020) and GUPPY, which is part of PPLACER (Matsen et al., 2010).

This can subsequently be visualized via standard tree viewing tools (for a review, see Czech et al., 2019). Note however that such a visualizations can quickly become overloaded when the number of QSs becomes large.

Second, the LWR distribution of a single sequence can be visualized, to depict the uncertainty in placement across the tree, for example with GGTREE (Yu et al., 2017) and iTOL (Letunic and Bork, 2016; Letunic and Bork, 2019).

Third, the distribution of *all* sequences can be visualized directly on the reference tree, for example as shown in **Figures 2E**, **4A**, taking their per-branch probabilities (and potentially their multiplicities/abundances) into account. This gives an overview of all placements, and can for example reveal important clades that received a high fraction of placements, or indicate whether placements are concentrated in a specific region of the tree. These visualizations can directly be generated by GAPPA (Czech et al., 2020) and ITOL (Letunic and Bork, 2016; Letunic and Bork, 2019); furthermore, GUPPY, can produce tree visualizations in the `phyloXML` format (Han and Zmasek, 2009), which can subsequently be displayed by tree viewer tools such as ARCHAEOPTERYX (Han and Zmasek, 2009).

3.3 Placement Quality and Uncertainty Quantification

An important post-analysis aspect is quality control, both in order to assess the suitability of the RT for the given placed sequences (to, e. g., test for missing reference sequences), and in order to assess the placed sequences themselves. Assuming a 'perfect' reference tree that exactly represents the diversity of the query sequences, the theoretical expectation is that each sequence gets placed onto a leaf of the tree with an LWR close to 1. Ignoring sequencing errors and other technical issues, deviations from this expectation can be due to several issues.

To this end, plotting the histograms or the distribution of the confidences (LWRs) across all placements can be useful, **Figure 4C**. A more involved metric is the so-called Expected Distance between Placement Locations (EDPL, Masten et al., 2010), which for a given sequence represents the uncertainty-weighted average distance between all placement locations of that sequence, or in other words, the sum of distances between locations, weighted by their respective probability, see **Figure 4D**. The EDPL is a measure of how far the likely placement locations of a sequence are spread out across the tree. It hence can distinguish between local and global uncertainty of the placements, that is, between cases where nearby edges constitute equally good placement locations versus cases where the sequence does not have a clear placement position in the tree (Matsen et al., 2010). These metrics can be explored with GAPPA (Czech et al., 2020) and GUPPY (Matsen et al., 2010); see their respective manuals for the available commands.

Examining the distribution of placement statistics, **Figures 4C,D**, or even the values of individual sequences, can help to identify the causes of problematic placements: 1) Sequences that are spread out across a clade with a flat placement distribution might indicate that too many closely related sequences, such as strains, are included in the RT; the EDPL can be used to quantify this. The query sequence is then likely another variant belonging to this subtree. 2) Placements towards inner branches of the RT might hint a hard to place query sequence, or at a lack of reference sequence diversity. This occurs if the (putative) ancestor represented by an inner node of the tree is more closely related to the QSs than the extant representatives included in the RT. This can either be the result of missing taxa in the RT, or even because the diversity of the clade is not fully known yet (also known as incomplete taxon sampling), in which case the QS might have originated from a previously undescribed species. 3) Sequences placed in two distinct clades might indicate technical errors such as the presence of chimeric sequences (Haas et al., 2011). 4) Sequences with elevated placement probability in multiple clades (e. g., placements in more than two subtrees) usually result from more severe issues, such as a total lack of suitable reference sequences for the QS, or a severe misalignment of the QS to the reference. This can for instance occur if metagenomic shotgun data has not been properly filtered, such that the genome region that the QS originated from is not included in the underlying MSA. 5) Lastly, long pendant lengths can also occur if a QS does not fit anywhere in the RT, in particular when the RT contains outgroups, which can cause long branch attraction for placed sequences (Bergsten, 2005).

Quantifying these uncertainties in a meaningful and interpretable way, and distinguishing between their causes, are open research questions. Approaches such as considering the EDPL, flatness of the LWR distribution, pendant lengths relative to the surrounding branch lengths of the RT, might help here, but more work is needed in order to distinguish actual issues from the identification of a new species based on their placement.

3.4 Taxonomic Classification and Functional Analysis

By understanding the taxonomic composition of an environment, questions about its species diversity and richness can be answered. Typical metagenomic data analyses hence often include a taxonomic classification of reads with respect to a database of known sequences (Breitwieser et al., 2019), for example by aggregating relative abundances per taxonomic group. In addition, such a classification based on known data enables to analyze which pathways and functions are present in a sample, and hence to gain insight into the metabolic capabilities of a microbial community.

3.4.1 Preexisting Tools

Many tools exist to these ends: BLAST (Altschul et al., 1990) and other similarity-based methods were among the early methods, but depend on the threshold settings for various parameters (Shah et al., 2019), only provide meaningful results if the reference database contains sequences closely related to the queries (Mahé et al., 2017), and the closest hit does often not represent the most closely related species (Koski and Golding, 2001; Clemente et al., 2011). Thus, the advantages of leveraging the power of phylogenetics for taxonomic assignment have long been recognized (Delsuc and Ranwez, 2020). The classification

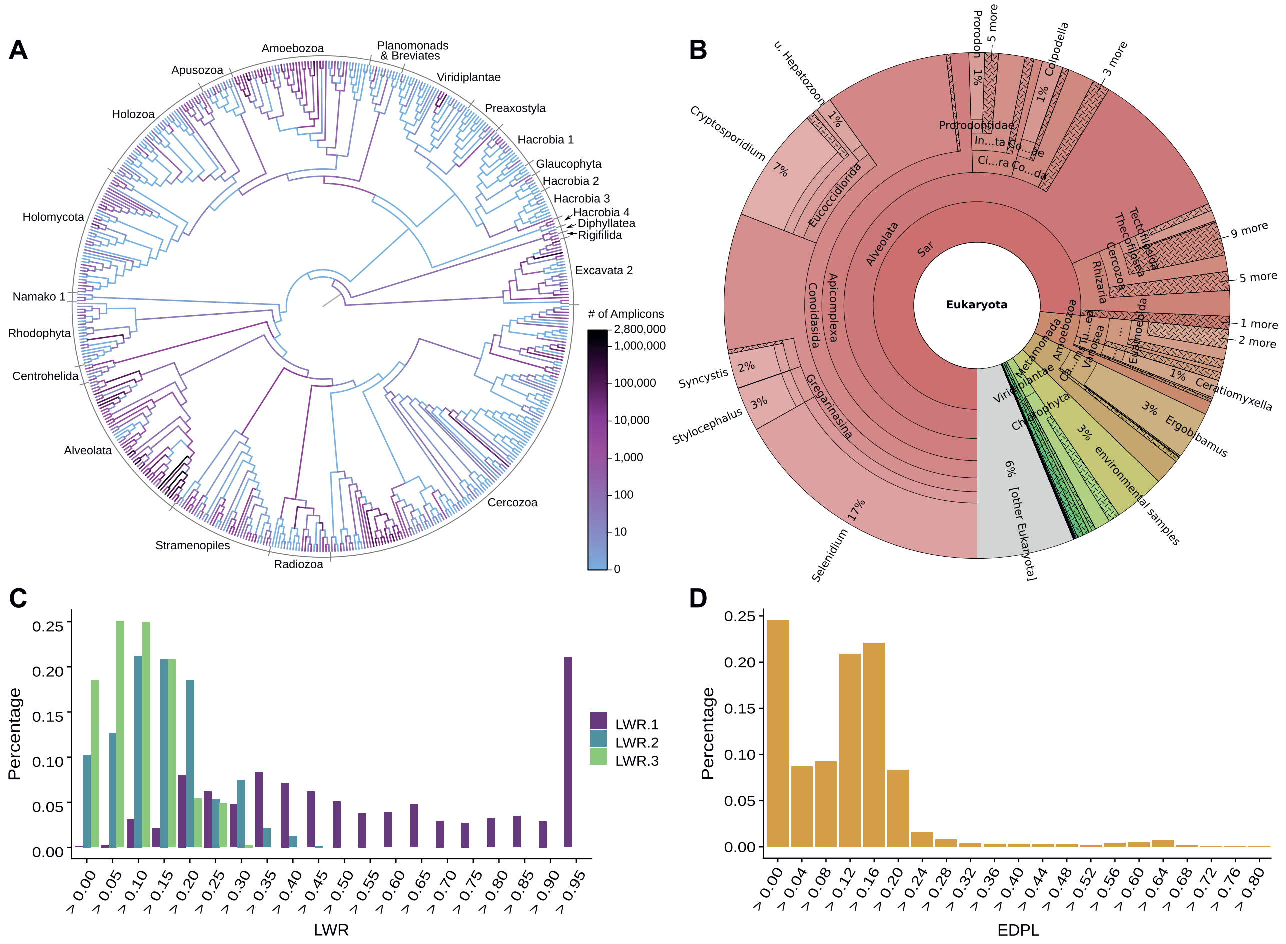


FIGURE 4 | Examination of phylogenetic placement data. Here, we show some techniques for visually inspecting placement data, using an exemplary dataset consisting of 154 soil samples from neotropical rain forests placed on an eukaryotic reference tree (Mahé et al., 2017). **(A)** Heat tree showing the distribution of millions of amplicon reads on the reference tree by summing over the per-branch Likelihood Weight Ratios (LWRs) of all reads. The high abundance of placed reads in the *Alveolata* clade (dark branches in the lower left) visualizes a main finding of the dataset in form of an over-abundance of reads from that clade, shown in the phylogenetic context of the reference tree. Figure adapted from (Mahé et al., 2017). **(B)** Taxonomic assignment of all reads based on the PR^2 (Benson et al., 2009; Guillou et al., 2012) taxonomy. The taxonomy of the reference sequences was used to label each branch of the reference tree by its highest non-conflicting taxonomic path. Then, for each read, the LWRs of its placement locations were accumulated for the branches, creating an overview of taxonomic abundances taking placement confidences into account. The result across all reads is shown here as a Krona plot (Ondov et al., 2011). **(C)** Histogram of the LWRs of the first three most likely placement locations of each read, showing how many of the reads have their first, second, and third most likely placement at each (binned) LWR value. For example, the highest bin of LWR.1 on the right hand side indicates that 20% of the reads have a first (most likely) placement position at or above an LWR of 0.95. That is, these placements have a high LWR and are hence placed with high certainty onto their respective branches. Note that the second most likely placement (LWR.2) can never have an LWR exceeding 1/2 (otherwise, it would be the most likely), the third most likely (LWR.3) not more than 1/3 (otherwise, it would be the second most likely), and so forth. **(D)** Histogram of the Expected Distance between Placement Locations (EDPL), which are computed as the distances (in terms of ML branch path length) between placement locations of a query sequence, weighted by the respective LWR of each location. The EDPL measures how far the placements of a sequence are spread across the branches of the reference tree, and hence how certain the placement in a “neighborhood” of the tree is. Here, most reads have an EDPL below 0.24 branch length units (mean expected number of substitutions per site). This indicates that the reads have most of their likely placements close to one another, within two branches on average, given that the used reference tree has an average branch length of about 0.12.

can be based on *de novo* construction of a phylogeny (Krause et al., 2008; Schreiber et al., 2010), which as mentioned is computationally expensive, and tree topologies might change between samples, yielding downstream analyses and independent comparisons between studies challenging (Boyd et al., 2018). Other tools to investigate the community composition of metagenome datasets via phylogenomic assignment of markers genes are BUSCO (all kingdoms, Simão et al., 2015) and AMPHORA2 (Bacteria and Archaea, Wu and Scott, 2012). These allow relatively fast *de novo* phylogenetic search using several markers simultaneously. Alternatively, dedicated pipelines for 16S metabarcoding data such as QIIME (Caporaso et al., 2010; Bolyen et al., 2019) and MOTHUR (Schloss et al., 2009) are routinely used to conduct taxonomic assignment based on sequence databases and established phylogenies as well as taxonomies; see Section “Sequence Selection” for a list of common databases, and see (López-García et al., 2018; Prodan et al., 2020) for comparisons of such pipelines. Other tools for taxonomic assignment and profiling are available, for example based on k-mers, which often use a fixed taxonomy such as the NCBI taxonomy (Benson et al., 2009; Sayers et al., 2009) to propose an evolutionary context for query sequences. They hence use a taxonomic tree without branch lengths, which can be an advantage when a fully resolved phylogeny is not available. Tools to this end are for example MEGAN (Huson et al., 2007), KRAKEN2 (Wood et al., 2014; Wood et al., 2019), and KAIJU (Menzel et al., 2016), see (Sczyrba et al., 2017; Bremges and McHardy, 2018; Meyer et al., 2019; Ye et al., 2019) for benchmarks and comparisons. However, these approaches are based on sequence similarity and related approaches, and can therefore be incongruent with the true underlying phylogenetic relationships of the sequences under comparison (Smith and Pease, 2017).

3.4.2 Placement-Based Approaches

Phylogenetic placement can be employed to perform an accurate assignment of QSs to taxonomic labels (Czech et al., 2018), with potentially higher resolution than methods based on manually curated taxonomies (Darling et al., 2014; Rajter et al., 2021). This approach leverages models of sequence evolution (Darling et al., 2014), and is hence more accurate than similarity-based methods (von Mering et al., 2007). A further advantage over the above pipelines is the ability to use custom reference trees, thus providing a better context for interpreting the data under study. Incongruencies between the taxonomy and the phylogeny can however hinder the assignment, if they are not resolved (Matsen and Gallagher, 2012). Furthermore, it is important to note that placement-based methods only work when the query sequences are homologous to the available reference data, hence currently limiting the approach to, e. g., short genomes, metabarcoding or filtered metagenomic data.

A simple approach for taxonomic annotation based on placements is to label each branch of the RT by the most descriptive taxonomic path of its descendants, and to assign each QS to these labels based on its placement locations, potentially weighted by LWRs (Czech et al., 2018; Kozlov et al., 2016). This is implemented in GAPPA (Czech et al., 2020), see **Figure 4B** for an example; a similar visualization of the taxonomic assignment of placements can be conducted with BOSSA (Lefeuvre, 2018).

More involved and specialized approaches have also been suggested. PHYLOSIFT (Darling et al., 2014) is a workflow that employs placement for taxonomic classification, using a database of gene families that are particularly well suited for metagenomics. The workflow further includes *Edge PCA* (introduced in Section "Similarity between Samples") to assess community structure across samples, and offers Bayesian hypothesis testing for the presence of phylogenetic lineages. The gene-centric taxonomic profiling tool METANNOTATE (Petrenko et al., 2015) uses a similar approach to identify organisms within a metagenomic sample that perform a function of interest. To this end, it searches shotgun sequences against the NCBI database (Benson et al., 2009; Sayers et al., 2009) first, and then employs placement to classify the reads with respect to genes and pathways of interest. GRAFTM (Boyd et al., 2018) is a tool for phylogenetic classification of genes of interest in large metagenomic datasets. Its primary application is to characterize sample composition using taxonomic marker genes, which can also target specific populations or functions. The abundance profiling methods TIPP (Nguyen et al., 2014) and TIPP2 (Shah et al., 2021) also use marker genes, and use the SEPP (Liu et al., 2012; Mirarab et al., 2012) boosting technique for phylogenetic placement with highly diverse reference trees, which increases classification accuracy when under-represented (novel) genomes are present in the dataset. The more recently introduced TREESAPP tool (Morgan-Lang et al., 2020) uses a similar underlying framework, but improves functional and taxonomic annotation by regressing on the evolutionary distances (branch lengths) of the placed sequences, thereby increasing accuracy and reducing false discovery. Lastly, PHYLOMAGNET (Schön et al., 2019) is a workflow for gene-centric metagenome assembly (MAGs) that can determine the presence of taxa and pathways of interest in large short-read datasets. It allows to explore and pre-screen microbial datasets, in order to select good candidate sets for metagenomic assembly.

3.5 Diversity Estimates

A goal that is intrinsically connected to taxonomic assignment in studies that involve metagenomic and metabarcode sequencing is to quantify the diversity within a sample (called α-diversity) and the diversity between samples (called β-diversity). A plethora of methods exists to quantify the diversity of a set of sequences (for an excellent review, see Tucker et al., 2017). Here, we focus on those approaches that specifically work in conjunction with phylogenetic placement.

Among the α-diversity metrics, Faith's Phylogenetic Diversity (PD) stands out, both for its widespread use in the literature and its direct use of phylogenetic information (Faith, 1992). More recently, a parameterized generalization of the PD was introduced that is able to interpolate between the classical PD and its abundance weighted formulation (McCoy and Matsen, 2013). Notably, this Balance Weighted Phylogenetic Diversity (BWPD) has been implemented to work directly with the results of phylogenetic placement, using the GUPPY `fpd` command (Matsen et al., 2010; Darling et al., 2014).

To our knowledge, the only other method that computes a measure of α-diversity directly from phylogenetic placement results is SCRAPP (Barbera et al., 2020), which also deploys species delimitation methods (Zhang et al., 2013; Kapli et al., 2017). In this method, the connection of phylogenetics to diversity is through the concept of a molecular species (Agapow et al., 2004), and quantifying how many such species are contained within a given sample. To facilitate this, SCRAPP resolves the between-QS phylogenetic relationships, resulting in per-reference-branch trees of those QSs that had their most likely placement on that specific branch. Thus, a byproduct of applying this method is a set of phylogenetic trees of the query sequences.

When the goal is to compute a β-diversity measure, a common choice for non-placement based approaches is the so-called

Unifrac distance (Lozupone and Knight, 2005; Lozupone et al., 2007), which quantifies the relatedness of two communities that are represented by leaves of a shared phylogenetic tree. Interestingly, the weighted version of the Unifrac distance has been shown to be equivalent to the KR-distance (Evans and Matsen, 2012), see Section "Similarity between Samples". As the Unifrac distance is widely used and well understood, this makes the KR-distance a safe choice for calculating between-sample distances, and thus a measure of β-diversity based on phylogenetic placement results.

3.6 Placement Distribution

Depending on the research question at hand, and for larger numbers of QSs, it is often more convenient and easier to interpret to look at the overall placement distribution instead of individually placed sequences. This distribution, as shown in **Figures 2E**, **4A**, summarizes an entire sample (or even multiple samples) by adding up the per-branch probabilities (i. e., LWRs) of each placement location of all sequences in the sample(s), ignoring all branch lengths (distal, proximal, and pendant) of the placements. In this context, the accumulated per-branch probabilities are also called the *edge mass* of a given branch. This terminology is derived from viewing the reference tree as a graph consisting of nodes and edges, and viewing the placements as a mass distribution on that graph. This focuses more on the mathematical aspects of the data, and provides a useful framework for the analysis methods described below.

3.6.1 Normalization of Absolute Abundances

High-throughput metagenomic sequence data are inherently compositional (Li, 2015; Gloor et al., 2017; Quinn et al., 2018), meaning that the total number of reads from HTS (absolute abundances) are mostly a function of available biological material and the specifics of the sequencing process. In other words, the total number of sequences per sample (often also called library size) is insignificant when comparing samples, see (Weiss et al., 2017; Du et al., 2018; Lin and Peddada, 2020) for reviews on this. This implies that sequence abundances are not comparable across samples, and that they can only be interpreted as proportions relative to each another (Calle, 2013; Silverman et al., 2017). However, the PCR amplification process is known to introduce biases (Logares et al., 2014), potentially skewing these proportions. For example, the relative abundances of the final amplicons do not necessarily reflect the original ratio of the input gene regions (Kanagawa, 2013; Li, 2015); this can be problematic in comparative studies. If these characteristics are not considered in analyses of the data (Weiss et al., 2017), spurious statistical results can occur (Aitchison, 1986; Jackson, 1997; Gloor et al., 2016; Tsilimigras and Fodor, 2016); see (Czech, 2020) for further details. For this reason, the estimation of indices such as the species richness is often implemented via so-called *rarefaction* and rarefaction curves (Gotelli and Colwell, 2001), which might however ignore a potentially large amount of the available valid data (McMurdie and Homes, 2014).

Phylogenetic placement of such data hence also needs to take this into account. The total edge masses (e. g., computed as the sum over all LWRs of a sample) are not informative, and merely reflect the total number of placed sequences. A simple strategy, upon which several of the analysis methods introduced below are based, is the normalization of the masses by dividing them by their total sum, effectively turning absolute abundances into relative abundances. This also eliminates the need for rarefaction, as low-abundance sequences only contribute marginally to the data. However, using this approach can still induce compositional artifacts in the data, as the per-branch probabilities (and hence the edge masses per sequence) have to sum to one for all branches of the tree. In other words, it is conceptually not possible to change the relative edge mass on a branch without also affecting edges masses on other branches.

3.6.2 Transformations of Compositional Data

A statistically advantageous way to circumvent these effects, and resulting misinterpretations of compositional placement data, is to transform the data from per-branch values to per-clade values. This way, individual placement masses in the nearby branches of a clade are transformed into a single value for the entire clade, which expresses a measure of difference (called contrast) of the placement masses within the clade versus the masses in the remainder of the tree. This makes such transformations robust against placement uncertainty in a clade (e. g., due to similar reference sequences), implicitly captures the tree topology, and solves the issues of compositional data. From a technical point of view, this transforms the data from a compositional space into an Euclidean coordinate system (Juan and Pawlowsky-Glahn, 2005), where the individual dimensions of a data point are unconstrained and independent of each other. This can be achieved by utilizing the reference tree, whose branches imply bi-partitions of the two clades that are split by each branch (Pawlowsky-Glahn et al., 2015; Silverman et al., 2017). Instead of working with the per-branch placement masses, the accumulated masses on each side of a branch are contrasted against each other. This yields a view of the data that summarizes all placements in the clades implied by each branch. These transformations are, for example, achieved via two methods that in the existing literature have unfortunately confusingly similar names: imbalances and balances (Czech, 2020).

The edge *imbalance* (Matsen and Evans, 2013) is computed on the normalized edge masses of a sample: For each edge, the sum over all masses in the two clades defined by that edge are computed; their difference is then called the *imbalance* of the edge. The edge *balance* (Silverman et al., 2017; Czech and Stamatakis, 2019) is computationally similar, but instead of a difference of sums, it is computed as the (isometric) log-ratio of the geometric means of the masses in each clade; the resulting coordinates are called *balances* (Egozcue et al., 2003; Juan and Pawlowsky-Glahn, 2005; Quinn et al., 2018). Both transformations yield a contrast value for each (inner) branch of the tree, which can then, for example, be used to compare different samples to each other, see Section "Analysis of Multiple Samples". They differ in the details of their statistical properties, but more work is needed to examine the effects of this on placement analyses (Czech, 2020); in practice, both can be (and are) used to avoid compositional artifacts. Alternatively, approaches such as Gamma-Poisson models and their zero-inflated versions (Peng et al., 2016; Weiss et al., 2017), as well as other methods for abundance normalization (Weiss et al., 2017; Du et al., 2018; Lin and Peddada, 2020) can be applied, although future work is needed to establish those in the context of phylogenetic placement.

3.7 Analysis of Multiple Samples

In typical metagenomic and metabarcoding studies, more than one sample is sequenced, e. g., from different locations or points in time of an environment. Furthermore, often per-sample metadata is collected as well, such as the pH-value of the soil or the temperature of the water where a sample was collected. These data allow to infer connections between the species community composition of the samples and environmental features. Given a set of samples (and potentially, metadata variables), an important goal is to understand the community structure (Tyson et al., 2020). To this end, fundamental tasks include measuring their similarity (a *distance* between samples), clustering samples that are similar to each other according to that distance measure, and relating the samples to their environmental variables. To this end, the methods introduced in this section utilize phylogenetic placement, and assume that the sequences from all samples have been placed onto the same underlying reference tree; they are implemented in GAPPA (Czech et al., 2020) and partially in GUPPY (Matsen et al., 2010).

3.7.1 Similarity Between Samples

A simple first data exploration method consists in computing the *Edge Dispersion* (Czech and Stamatakis, 2019) of a set of samples, which detects branches or clades of the tree that exhibit a high heterogeneity across the samples by visualizing a measure of dispersion (such as the variance) of the per-sample placement mass. The method hence identifies branches and clades "of interest", where samples differ in the amount of sequences being placed onto these parts of the tree.

The similarity between the placement distributions of two samples can be measured with the *phylogenetic Kantorovich-Rubinstein* (KR) distance (Evans and Matsen, 2012; Matsen and Evans, 2013), which is an adaptation of the Earth Mover's distance to phylogenetic placement. The KR distance between two samples is a metric that quantifies by *at least* how much the normalized mass distribution of one sample has to be moved across the reference tree to obtain the distribution of the other sample. In other words, it is the minimum work needed to solve the transportation problem between the two distributions (transforming one into the other), and is related to the UniFrac distance (Lozupone and Knight, 2005; Lozupone et al., 2007). The distance is symmetrical, and increases the more mass needs to be moved (that is, the more the abundances per branch and clade differ between the two samples), and the larger the respective moving distance is (that is, the greater the phylogenetic distance along the branches of the tree between the clades is). It is hence an intuitive and phylogenetically informed distance metric for placement data, for example to quantify differences in the species composition of two environments.

Edge Principal Component Analysis (Edge PCA) is a method to detect community structure, which can also be employed for sample ordination and visualization (Darling et al., 2014; Matsen and Evans, 2013). Edge PCA identifies lineages of the RT that explain the greatest extent of variation between the sample communities, and is computed via standard Principal Component Analysis on the per-edge imbalances across all samples. The resulting principal components distinguish samples based on differences in abundances within clades of the reference tree. See for example **Figure 5D**, where each point corresponds to a sample and is colorized according to a metadata variable of the sample, showing that the ordination discriminates samples according to that variable. Furthermore, as the eigenvectors of each principal component correspond to edges of the tree, these can be visualized on the tree (Matsen and Evans, 2013; Czech, 2020), so that those edges and clades of the tree that explain differences between the samples can be identified, e. g., with GUPPY (Matsen et al., 2010) and ARCHAEOPTERYX (Han and Zmasek, 2009), or with GAPPA (Czech et al., 2020). Principal components can also be computed from the balances instead of the imbalances (Czech, 2020).

3.7.2 Clustering of Samples

Given a measure of pairwise distance between samples, a fundamental task consists in clustering, that is, finding groups of samples that are similar according to that measure. *Squash Clustering* (Matsen and Evans, 2013) is a hierarchical agglomerative clustering method for a set of placement samples, and is based on the KR distance. Its results can be visualized as a clustering tree, where terminal nodes represent samples, each inner node represents the cumulative distribution of all samples below that node ("squashed" samples), and distances along the tree edges are KR distances. We show an example in **Figure 5E**, where each sample (terminal node) is colorized according to associated per-sample metadata variables (features measured for each sample), indicating that the clustering (based on the placement distribution) recovers characteristics of the samples based on that metadata variable.

The clustering hierarchy obtained from Squash Clustering grows with the number of samples, which contains a lot of detail, but can be cumbersome to visualize and interpret for large datasets with many samples. *Phylogenetic k-means* clustering and *Imbalance k-means* clustering (Czech and Stamatakis, 2019) are further clustering approaches, which instead yield an assignment of each sample to one of a predefined number of k clusters. Phylogenetic k-means uses the KR distance for determining the cluster assignment of the samples, and hence yields results that are consistent with Squash Clustering, while Imbalance k-means uses edge imbalances, and hence is consistent with results obtained from Edge PCA. Having the choice over the value k can be beneficial to answer specific questions with a known set of categories of samples (e. g., different body locations where samples were obtained from), but is also considered a downside of k-means clustering. Hence, various suggestions exist in the literature to select an appropriate k that reflects the number of "natural" clusters in the data (Thorndike, 1953; Rousseeuw, 1987;

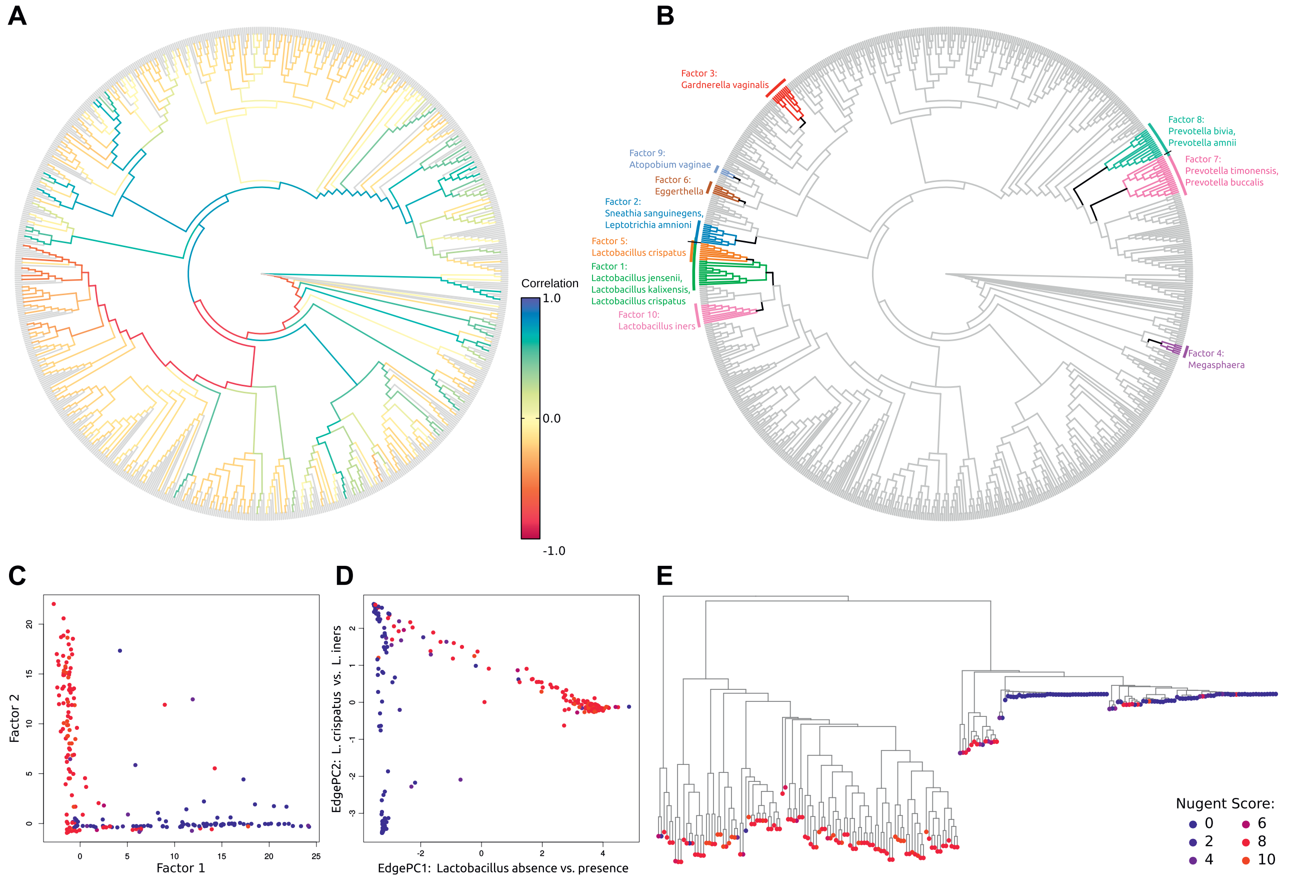


FIGURE 5 | Analyses of phylogenetic placement data. Here, we show several analysis techniques for placement data, which relate multiple samples to each other (e. g., from different locations or points in time) that have been placed on the same underlying reference tree. The example dataset contains 220 vaginal samples of human patients with and without Bacterial Vaginosis (BV), a condition caused by an abnormal vaginal microbiome (Srinivasan et al., 2012), placed on a bacterial tree. The "Nugent" score is an external clinical indicator of the disease (Nugent et al., 1991), which is shown in **(C–E)** as blue (healthy, low score) vs. red colors (severe disease, high score). In healthy patients, two types of *Lactobacilli* dominate the microbiome, while in diseased patients, a diverse mixture of other bacteria take over. All figures are adapted from (Czech and Stamatakis, 2019), for details see (Srinivasan et al., 2012; Matsen and Evans, 2013; Czech and Stamatakis, 2019; Czech, 2020). **(A)** *Edge Correlation* between read abundances in clades of the reference tree (measured via the *imbalance* transformation) and the per-sample Nugent score. This visualization method identifies taxa whose abundances exhibit a relationship with environmental factors. Here, the red path towards the left identifies the *Lactobacillus* clade, that exhibits a strong anti-correlation with the Nugent score (healthy patients with a low score have high abundances in this clade), while blue and green paths show a multitude of clades that correlate with the score (diseased patients with a high score and high abundances in these diverse clades). **(B)** *Placement-Factorization* discretely identifies these clades by splitting up the tree into a number of "factors": Black edges (with colorized clades below them) indicate the first ten factors (groups of taxa, some of them nested) whose differential placement abundances between samples exhibit a strong relationship with the Nugent score. That is, a factor is a clade in which abundances co-vary with metadata (e. g., the Nugent score). Here, these factors are again the *Lactobacillus* clade and a multitude of other clades that are also highlighted in **(A)** by colored paths. **(C)** *Placement-Factorization* can also ordinate samples, by plotting the *balances* (i. e., the abundance contrasts) across the edges identified by factors. Here, the first two factors of **(B)** are shown (each dot represents one sample, colored by its Nugent score), which split healthy and diseased patients. **(D)** *Edge Principal Components Analysis* (EdgePCA) is another ordination method, using PCA on the edge *imbalances*. Here, the first two PC axes are shown, which separate healthy from diseased patients (*Lactobacillus* presence vs. absence) on the first axis, and further distinguish the healthy patients based on the two types of *Lactobacilli* on the second axis. These interpretations of the axes are derived from visualizing the PCA directly on the reference tree, which is another way to show Edge PCA results, see (Matsen and Evans, 2013; Czech, 2020). **(E)** *Squash Clustering* is a hierarchical clustering method, here showing the clustering tree of the samples (not a phylogeny). Tip nodes (leaves) correspond to samples (individual patients), again colorized by their Nugent score, with samples clustered based on similarity of their placement distribution, and vertical distances showing this similarity, measured as the phylogenetic Kantorovich-Rubinstein (KR) distance between samples. Patients with a similar health status are close to each other, in particular the healthy (blue) ones.

Bischof et al., 1999; Pelleg and Moore, 2000; Tibshirani et al., 2001; Hamerly et al., 2004). Visualizing the *cluster centroids* obtained from both methods can further help to interpret results by showing the average distributions of all samples in one of the *k* clusters; see again (Czech, 2020) for details.

3.7.3 Relationship With Environmental Metadata Variables

The above methods only implicitly take metadata into account, e. g., by colorizing their resulting plots according to a variable. Environmental variables can also be incorporated explicitly in phylogenetic placement analysis, to more directly infer the relationships between the species composition of the samples (e. g., in form of abundances per clade) and the environments these communities live in.

The *Edge Correlation* (Czech and Stamatakis, 2019) visualizes parts of the tree where species abundances (as measured by the accumulated probability mass of each sample) exhibit a strong connection with a metadata variable, see **Figure 5A**. It is computed as the per-edge correlation coefficient between the per-sample metadata variable and either the edge masses (highlighting individual edges), or imbalances or balances (highlighting clades) of each sample.

Placement-Factorization (Czech and Stamatakis, 2019; Czech, 2020) is a more involved method. It is an adaption of *PhyloFactorization* (Washburne et al., 2017; Washburne et al., 2019) to phylogenetic placement data. Its goal is to identify branches in the tree along which putative functional traits might have arisen in adaptation to changes in environmental variables. In other words, it can detect clades of the reference tree whose abundances are linked to environmental factors. By "factoring out" the clade with the strongest signal in each step of the algorithm (hence the name of the method), nested dependencies with variables within clades can also be discovered, see **Figure 5B**. This factorization of the tree into nested clades can further be used as an ordination tool to visualize how samples are separated by changes along the factors, and as a dimensionality-reduction tool, see **Figure 5C**. The method assesses the relationship between per-sample metadata features and the balances computed on the samples; by using Generalized Linear Models, it allows to simultaneously incorporate multiple metadata variables of different types, such as numerical values (pH-value, temperature, latitude/longitude, etc), binary values (presence/absence patterns, diseased or not), or categorical values (body site that a sample was taken from).

4 CONCLUSION AND OUTLOOK

In this review we broadly surveyed the concepts, methods, and software tools that constitute and relate to phylogenetic placement. We have also presented guidelines and best practices for many typical use cases, showcased some common misconceptions and pitfalls, and introduced the most prominent downstream analysis methods. Phylogenetic placement is a versatile approach that is particularly applicable in metagenomics (e. g., for metabarcoding data) and broader eDNA-based ecology studies. It allows for the annotation of sequence data with phylogenetic information, and thereby to investigate the taxonomic content, functional capacity, diversity, and interactions of a community of organisms. Further, it allows for comparing samples from multiple spatial and temporal locations, enabling the analysis of community patterns across time and space, as well as their association with environmental metadata variables.

Despite the growing popularity of phylogenetic placement, there are several methodological and usage aspects that will benefit from further developments.

Currently, significant effort is required to create high-quality reference trees. We believe research effort should focus on simplifying this process, potentially through the design of methods that streamline and automate the commonly involved tasks. For example, while there are some metrics that quantify the quality of an inferred phylogenetic tree (Felsenstein, 1985; Dhar and Minin, 2016; Lemoine et al., 2018), there is a lack of metrics to specifically evaluate the suitability of a tree for phylogenetic placement, given some expected input data. Note that the PEWO testing framework (Linard et al., 2020) (see Section "Workflows based on Phylogenetic Placement") represents a first step in this direction.

Ideally, reference trees and alignments should be created by, and shared in, research communities that investigate the same group(s) of organisms. This would not only yield obtaining high-quality reference trees trivial, but would also immensely increase the comparability across studies, as well as their reproducibility. Consequently, we would highly encourage such collaborations, and the public sharing of (perhaps even versioned instances of) gold-standard reference trees. Notably, for some environments, first efforts into this direction have already been undertaken (Berney et al., 2017; Del Campo et al., 2018; Rubinat-Ripoll, 2019; Rajter and Dunthorn, 2021; Rajter et al., 2021).

Furthermore, as mentioned, there is a lack of established methods that evaluate placement quality in a standardized and meaningful way. In particular, robust metrics are missing to distinguish the case where reference sequences of known species are missing from the tree from the case where the placed data actually contains yet undescribed species. A classification based on the LWR and pendant length of the placement locations might offer a solution here.

Lastly, further work is required to connect environmental metadata to the results of phylogenetic placement. Placement-based spatio-temporal methods are of high interest for addressing research questions in ecology and phylogeography. For example, relating geo-locations of samples to their placement could indicate how species communities differ across space, while creating placement time series could show how community compositions develop and change over time.

AUTHOR CONTRIBUTIONS

LC conceived the review and created the figures. LC and PB drafted the manuscript. All authors conducted literature research, and finalized and approved the manuscript.

FUNDING

This work was financially supported by the Carnegie Institution for Science at Stanford, California, United States, the Klaus Tschira Stiftung gGmbH Foundation in Heidelberg, Germany, and the Deutsche Forschungsgemeinschaft (grant DU1319/5-1).

ACKNOWLEDGMENTS

We wish to thank the Reviewers, as well as Guido Grimm, for the detailed and constructive comments that helped to improve this manuscript.

REFERENCES

Agapow, P. M., Bininda-Emonds, O. R., Crandall, K. A., Gittleman, J. L., Mace, G. M., Marshall, J. C., et al. (2004). The Impact of Species Concept on Biodiversity Studies. *Q. Rev. Biol.* 79 (2), 161–179. doi:10.1086/383542

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman and Hall London.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic Local Alignment Search Tool. *J. Mol. Biol.* 215 (3), 403–410. doi:10.1016/S0022-2836(05)80360-2

Amarasinghe, S. L., Su, S., Dong, X., Zappia, L., Ritchie, M. E., and Gouil, Q. (2020). Opportunities and Challenges in Long-Read Sequencing Data Analysis. *Genome Biol.* 21 (1), 30. doi:10.1186/s13059-020-1935-5

Angly, F. E., Dennis, P. G., Skarshewski, A., Vanwonterghem, I., Hugenholtz, P., and Tyson, G. W. (2014). CopyRighter: a Rapid Tool for Improving the Accuracy of Microbial Community Profiles through Lineage-specific Gene Copy Number Correction. *Microbiome* 2 (1), 11. doi:10.1186/2049-2618-2-11

Archie, J., Day, W. H. E., Maddison, W., Meacham, C., Rohlf, F. J., Swofford, D., et al. (1986). *The Newick Tree Format*. Available at: http://evolution.genetics.washington.edu/phylip/newicktree.html.

Arenas, M. (2015). Trends in Substitution Models of Molecular Evolution. *Front. Genet.* 6 (OCT), 319. doi:10.3389/fgene.2015.00319

Auladell, A., Sánchez, P., Sánchez, O., Gasol, J. M., and Ferrera, I. (2019). Long-term Seasonal and Interannual Variability of marine Aerobic Anoxygenic Photoheterotrophic Bacteria. *ISME J.* 13 (138), 1975–1987. doi:10.1038/s41396-019-0401-4

Balaban, M., and Mirarab, S. (2020). Phylogenetic Double Placement of Mixed Samples. *Bioinformatics* 36, i335. doi:10.1093/bioinformatics/btaa489

Balvočiūtė, M., and Huson, D. H. (2017). SILVA, RDP, Greengenes, NCBI and OTT - How Do These Taxonomies Compare? *BMC Genomics* 18 (2), 114. doi:10.1186/s12864-017-3501-4

Barbera, P., Czech, L., Lutteropp, S., and Stamatakis, A. (2020). SCRAPP: A Tool to Assess the Diversity of Microbial Samples from Phylogenetic Placements. *Mol. Ecol. Resour.* 21 (1), 1755–0998. doi:10.1111/1755-0998.13255

Barbera, P., Kozlov, A. M., Czech, L., Morel, B., Darriba, D., Flouri, T., et al. (2018). Massively Parallel Evolutionary Placement of Genetic Sequences. *Syst. Biol* 68 (2), 365–369. doi:10.1093/sysbio/syy054

Bartlett, J. M. S., and Stirling, D. (2003). *A Short History Of the Polymerase Chain Reaction*. PCR Protocols. *Methods Mol. Biol.* 226, 3–6. doi:10.1385/1-59259-384-4:3

Bass, D., Czech, L., Williams, B. A. P., Berney, C., Dunthorn, M., Mahé, F., et al. (2018). Clarifying the Relationships between Microsporidia and Cryptomycota. *J. Eukaryot. Microbiol.* 65 (6), 773–782. doi:10.1111/jeu.12519

Beghini, F., McIver, L. J., Blanco-Míguez, A., Dubois, L., Asnicar, F., Maharjan, S., et al. (2021). Integrating Taxonomic, Functional, and Strain-Level Profiling of Diverse Microbial Communities with bioBakery 3. *eLife* 10. doi:10.7554/elife.65088

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Sayers, E. W. (2009). GenBank *Nucleic Acids Res.* 37, D26–D31. doi:10.1093/nar/gkn723

Berger, S. A., and Stamatakis, A. (2010). "Accuracy of Morphology-Based Phylogenetic Fossil Placement under Maximum Likelihood," in ACS/IEEE International Conference on Computer Systems and Applications AICCSA. doi:10.1109/aiccsa.2010.5586939

Berger, S., and Stamatakis, A. (2012). *PaPaRa 2.0: A Vectorized Algorithm for Probabilistic Phylogeny-Aware Alignment Extension*. Technical report. Heidelberg, Germany: Heidelberg Institute for Theoretical Studies, Heidelberg.

Berger, S. A., Krompass, D., and Stamatakis, A. (2011). Performance, Accuracy, and Web Server for Evolutionary Placement of Short Sequence Reads under Maximum Likelihood. *Syst. Biol.* 60 (3), 291–302. doi:10.1093/sysbio/syr010

Berger, S. A., and Stamatakis, A. (2011). Aligning Short Reads to Reference Alignments and Trees. *Bioinformatics* 27 (15), 2068–2075. doi:10.1093/bioinformatics/btr320

Bergsten, J. (2005). A Review of Long-branch Attraction. *Cladistics* 21 (2), 163–193. doi:10.1111/j.1096-0031.2005.00059.x

Berney, C., Ciuprina, A., Bender, S., Brodie, J., Edgcomb, V., Kim, E., et al. (2017). UniEuk: Time to Speak a Common Language in Protistology!. *J. Eukaryot. Microbiol.* 64 (1), 407–411. doi:10.1111/jeu.12414

Bininda-Emonds, O. R., Brady, S. G., Kim, J., and Sanderson, M. J. (2001). Scaling of Accuracy in Extremely Large Phylogenetic Trees. *Pac. Symp. Biocomput* 547–558, 547–558. doi:10.1142/9789814447362_0053

Bischof, H., Leonardis, A., and Alexander, S. (1999). MDL Principle for Robust Vector Quantisation. *Pattern Anal. Appl.* 2 (1), 59–72. doi:10.1007/s100440050015

Blanke, M., and Morgenstern, B. (2021). App-SpaM: Phylogenetic Placement of Short Reads without Sequence Alignment. *Bioinformatics Adv.* 1 (1), 10. doi:10.1093/bioadv/vbab027

Blaxter, M., Mann, J., Chapman, T., Thomas, F., Whitton, C., Floyd, R., et al. (2005). Defining Operational Taxonomic Units Using DNA Barcode Data. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 360 (1462)–43. doi:10.1098/rstb.2005.1725

Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, C. C., Al-Ghalith, G. A., et al. (2019). Reproducible, Interactive, Scalable and Extensible Microbiome Data Science Using QIIME 2. *Nat. Biotechnol.* 37 (8), 852–857. doi:10.1038/s41587-019-0209-9

Bomfleur, B., Grimm, G. W., and McLoughlin, S. (2015). Osmunda Pulchella Sp. Nov. From the Jurassic of Sweden-reconciling Molecular and Fossil Evidence in the Phylogeny of Modern Royal Ferns (Osmundaceae). *BMC Evol. Biol.* 15 (1), 1–25. doi:10.1186/s12862-015-0400-7

Boyd, J. A., Woodcroft, B. J., and Tyson, G. W. (2018). GraftM: a Tool for Scalable, Phylogenetically Informed Classification of Genes within Metagenomes. *Nucleic Acids Res.* 46 (10), e59. doi:10.1093/nar/gky174

Bray, T. (2018). *The JavaScript Object Notation (JSON) Data Interchange Format, RFC*. Available at: https://tools.ietf.org/html/rfc7159 (Accessed August 14, 2018).

Brady, A., and Salzberg, S. L. (2009). Phymm and PhymmBL: Metagenomic Phylogenetic Classification with Interpolated Markov Models. *Nat. Methods* 6 (9), 673–676. doi:10.1038/nmeth.1358

Breitwieser, F. P., Lu, J., and Salzberg, S. L. (2019). A Review of Methods and Databases for Metagenomic Classification and Assembly. *Brief Bioinform* 20 (4), 1125–1136. doi:10.1093/bib/bbx120

Bremges, A., and McHardy, A. C. (2018). Critical Assessment of Metagenome Interpretation Enters the Second Round. *mSystems* 3 (4). doi:10.1128/mSystems.00103-18

Brown, D. G., and Truszkowski, J. (2012). "LSHPlace: Fast Phylogenetic Placement Using Locality-Sensitive Hashing," in *Biocomputing 2013* (World Scientific). doi:10.1142/9789814447973_0031

Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J., and Holmes, S. P. (2016). DADA2: High-Resolution Sample Inference from Illumina Amplicon Data. *Nat. Methods* 13 (7), 581–583. doi:10.1038/nmeth.3869

Calle, M. L. (2013). Statistical Analysis of Metagenomics Data. *Genomics Inform.* 17 (1), e6. doi:10.5808/GI.2019.17.1.e6

Caporaso, J. G., Kuczynski, J., Stombaugh, J., Bittinger, K., Costello, E. K., Fierer, N., et al. (2010). QIIME Allows Analysis of High-Throughput Community Sequencing Data. *Nat. Methods* 7 (5), 335–336. doi:10.1038/nmeth0510-33510.1038/nmeth.f.303

Carbone, I., White, J. B., Miadlikowska, J., Arnold, A. E., Miller, M. A., Magain, N., et al. (2019). T-BAS Version 2.1: Tree-Based Alignment Selector Toolkit for Evolutionary Placement of DNA Sequences and Viewing Alignments and Specimen Metadata on Curated and Custom Trees. *Microbiol. Resour. Announc* 8 (29). doi:10.1128/mra.00328-19

Carbone, I., White, J. B., Miadlikowska, J., Arnold, A. E., Miller, M. A., Kauff, F., et al. (2016). T-BAS: Tree-Based Alignment Selector Toolkit for Phylogenetic-Based Placement, Alignment Downloads and Metadata Visualization: an Example with the Pezizomycotina Tree of Life. *Bioinformatics*, btw808. doi:10.1093/bioinformatics/btw808

Cardoni, S., Piredda, R., Denk, T., Grimm, G. W., Papageorgiou, A. C., Schulze, E. D., et al. (2022). 5S-IGS rDNA in Wind-Pollinated Trees (Fagus L.) Encapsulates 55 Million Years of Reticulate Evolution and Hybrid Origins of Modern Species. *Plant J.* 109 (4), 909–926. doi:10.1111/tpj.15601

Chatzou, M., Magis, C., Chang, J. M., Kemena, C., Bussotti, G., Erb, I., et al. (2016). *Multiple Sequence Alignment Modeling: Methods and Applications*. doi:10.1093/bib/bbv099

Clare, E. L., Economou, C. K., Bennett, F. J., Dyer, C. E., Adams, K., McRobie, B., et al. (2022). Measuring Biodiversity from DNA in the Air. *Curr. Biol.* 32, 693–700. doi:10.1016/j.cub.2021.11.064

Clemente, J. C., Jansson, J., and Valiente, G. (2011). Flexible Taxonomic Assignment of Ambiguous Sequencing Reads. *BMC Bioinformatics* 12 (1), 8–15. doi:10.1186/1471-2105-12-8

Cole, J. R., Wang, Q., Fish, J. A., Chai, B., McGarrell, D. M., Sun, Y., et al. (2014). Ribosomal Database Project: Data and Tools for High Throughput rRNA Analysis. *Nucleic Acids Res.* 42, D633–D642. doi:10.1093/nar/gkt1244

Collins, R. A., Trauzzi, G., Maltby, K. M., Gibson, T. I., Ratcliffe, F. C., Hallam, J., et al. (2021). Meta-Fish-Lib : A Generalised, Dynamic DNA Reference Library Pipeline for Metabarcoding of Fishes. *J. Fish Biol.* 99 (4), 1446–1454. doi:10.1111/jfb.14852

Curtis, H., Gevers, D., Knight, R., Abubucker, S., Badger, J. H., Chinwalla, A. T., et al. (2012). Structure, Function and Diversity of the Healthy Human Microbiome. *Nature* 486 (7402), 207–214. doi:10.1038/nature11234

Czech, L., Barbera, P., and Stamatakis, A. (2020). Genesis and Gappa: Processing, Analyzing and Visualizing Phylogenetic (Placement) Data. *Bioinformatics* 36 (10), 3263–3265. doi:10.1093/bioinformatics/btaa070

Czech, L., Barbera, P., and Stamatakis, A. (2018). Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement. *Bioinformatics* 35 (7), 1151–1158. doi:10.1093/bioinformatics/bty767

Czech, L., Huerta-Cepas, J., and Stamatakis, A. (2019). A Critical Review on the Use of Support Values in Tree Viewers and Bioinformatics Toolkits. *Mol. Biol. Evol.* 17 (4), 383–384. doi:10.1093/molbev/msx055

Czech, L., and Stamatakis, A. (2019). Scalable Methods for Analyzing and Visualizing Phylogenetic Placement of Metagenomic Samples. *PLOS ONE* 14 (5), e0217050. doi:10.1371/journal.pone.0217050

Czech, L. (2020). *Novel Methods for Analyzing and Visualizing Phylogenetic Placements*. Ph.D. thesis. Karlsruhe, Germany: Karlsruher Institut für Technologie. doi:10.5445/IR/1000105237

Darling, A. E., Jospin, G., Lowe, E., Matsen, F. A., Bik, H. M., and Eisen, J. A. (2014). PhyloSift: Phylogenetic Analysis of Genomes and Metagenomes. *PeerJ* 2, e243. doi:10.7717/peerj.243

Degnan, J. H., and Rosenberg, N. A. (2009). Gene Tree Discordance, Phylogenetic Inference and the Multispecies Coalescent. *Trends Ecol. Evol.* 24 (6), 332–340. doi:10.1016/j.tree.2009.01.009

Deiner, K., Bik, H. M., Mächler, E., Seymour, M., Lacoursière-Roussel, A., Altermatt, F., et al. (2017). Environmental DNA Metabarcoding: Transforming How We Survey Animal and Plant Communities. *Mol. Ecol.* 26 (21), 5872–5895. doi:10.1111/mec.14350

Del Campo, J., Kolisko, M., Boscaro, V., Santoferrara, L. F., Nenarokov, S., Massana, R., et al. (2018). EukRef: Phylogenetic Curation of Ribosomal RNA to Enhance Understanding of Eukaryotic Diversity and Distribution. *Plos Biol.* 16 (9), e2005849–14. doi:10.1371/journal.pbio.2005849

Delsuc, F., and Ranwez, V. (2020). "Accurate Alignment of (Meta)barcoding Data Sets Using MACSE," in *Phylogenetics in the Genomic Era*. Editors C. Scornavacca, F. Delsuc, and N. Galtier. Available at: https://hal.archives-ouvertes.fr/hal-02541199.

Desai, N., Antonopoulos, D., Gilbert, J. A., Glass, E. M., and Meyer, F. (2012). From Genomics to Metagenomics. *Curr. Opin. Biotechnol.* 23 (1), 72–76. doi:10.1016/j.copbio.2011.12.017

DeSantis, T. Z., Hugenholtz, P., Larsen, N., Rojas, M., Brodie, E. L., Keller, K., et al. (2006). Greengenes, a Chimera-Checked 16S rRNA Gene Database and Workbench Compatible with ARB. *Appl. Environ. Microbiol.* 72 (7), 5069–5072. doi:10.1128/AEM.03006-05

Dhar, A., and Minin, V. N. (2016). Maximum Likelihood Phylogenetic Inference. *Encyclopedia Evol. Biol.* 2, 499–506. doi:10.1016/b978-0-12-800049-6.00207-9

Dodsworth, S. (2015). Genome Skimming for Next-Generation Biodiversity Analysis. *Trends Plant Sci.* 20 (9), 525–527. doi:10.1016/j.tplants.2015.06.012

Douglas, C. (2018). *The Application/json Media Type for JavaScript Object Notation (JSON), RFC.* Available at: https://tools.ietf.org/html/rfc4627 (Accessed August 14, 2018).

Douglas, G. M., Maffei, V. J., Zaneveld, J. R., Yurgel, S. N., Brown, J. R., Taylor, C. M., et al. (2020). PICRUSt2 for Prediction of Metagenome Functions. *Nat. Biotechnol.*, 1–5. doi:10.1038/s41587-020-0548-6

Douglas, G. M., Beiko, R. G., and Langille, M. G. I. (2018). "Predicting the Functional Potential of the Microbiome from Marker Genes Using PICRUSt," in *Microbiome Analysis* (Springer), 169–177. doi:10.1007/978-1-4939-8728-3_11

Du, R., An, L., and Fang, Z. (2018). *Performance Evaluation of Normalization Approaches for Metagenomic Compositional Data on Differential Abundance Analysis*. Cham: Springer International Publishing, 329–344. doi:10.1007/978-3-319-99389-8_16

Dunthorn, M., Otto, J., Berger, S. A., Stamatakis, A., Mahé, F., Romac, S., et al. (2014). Placing Environmental Next-Generation Sequencing Amplicons from Microbial Eukaryotes into a Phylogenetic Context. *Mol. Biol. Evol.* 31 (4), 993–1009. doi:10.1093/molbev/msu055

Dupont, A. Ö., Griffiths, R. I., Bell, T., and Bass., D. (2016). Differences in Soil Micro-eukaryotic Communities over Soil pH Gradients Are Strongly Driven by Parasites and Saprotrophs. *Environ. Microbiol.* 18 (6), 2010–2024. doi:10.1111/1462-2920.13220

Eddy, S. R. (1995). Multiple Alignment Using Hidden Markov Models. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* 3, 114–120.

Eddy, S. R. (1998). Profile Hidden Markov Models. *Bioinformatics* 14 (9), 755–763. doi:10.1093/bioinformatics/14.9.755

Edgar, R. C. (2021). MUSCLE V5 Enables Improved Estimates of Phylogenetic Tree Confidence by Ensemble Bootstrapping. *bioRxiv*. doi:10.1101/2021.06.20.449169

Edgar, R. C. (2004). MUSCLE: Multiple Sequence Alignment with High Accuracy and High Throughput. *Nucleic Acids Res.* 32 (5), 1792–1797. doi:10.1093/nar/gkh340

Edgar, R. C. (2010). Search and Clustering Orders of Magnitude Faster Than BLAST. *Bioinformatics* 26 (19), 2460–2461. doi:10.1093/bioinformatics/btq461

Edwards, D. J., and Holt, K. E. (2013). Beginner's Guide to Comparative Bacterial Genome Analysis Using Next-Generation Sequence Data. *Microb. Inform. Exp.* 3 (1), 2. doi:10.1186/2042-5783-3-2

Egozcue, J. J., Pawlowsky-Glahn, V., Mateu-Figueras, G., and Barceló-Vidal, C. (2003). Isometric Logratio Transformations for Compositional Data Analysis. *Math. Geology.* 35 (3), 279–300. doi:10.1023/A:1023818214614

ElRakaiby, M. T., Gamal-Eldin, S., Amin, M. A., and Aziz, R. K. (2019). Hospital Microbiome Variations as Analyzed by High-Throughput Sequencing. *OMICS* 23 (9), 426–438. doi:10.1089/omi.2019.0111

Erazo, N. G., Dutta, A., and Bowman, J. S. (2021). From Microbial Community Structure to Metabolic Inference Using Paprica. *STAR Protoc.* 2 (4), 101005. doi:10.1016/j.xpro.2021.101005

Escobar-Zepeda, A., Vera-Ponce De León, A., and Sanchez-Flores, A. (2015). The Road to Metagenomics: From Microbiology to DNA Sequencing Technologies and Bioinformatics. *Front. Genet.* 6 (348), 1–15. doi:10.3389/fgene.2015.00348

Evans, S. N., and Matsen, F. A. (2012). The Phylogenetic Kantorovich-Rubinstein Metric for Environmental Sequence Samples. *J. R. Stat. Soc. Ser. B Stat Methodol* 74, 569–592. doi:10.1111/j.1467-9868.2011.01018.x

Faith, P. D. (1992). Conservation Evaluation and Phylogenetic Diversity. *Biol. Conservation* 61 (1), 1–10. doi:10.1016/0006-3207(92)91201-3

Felsenstein, J. (1978). Cases in Which Parsimony or Compatibility Methods Will Be Positively Misleading. *Syst. Biol.* 27 (4), 401–410. doi:10.1093/sysbio/27.4.401

Felsenstein, J. (1981). Evolutionary Trees from DNA Sequences: A Maximum Likelihood Approach. *J. Mol. Evol.* 17 (6), 368–376. doi:10.1007/BF01734359

Felsenstein, J. (1985). Confidence Limits on Phylogenies: an Approach Using the Bootstrap. *Evolution* 39 (4), 783–791. doi:10.1111/j.1558-5646.1985.tb00420.x

Felsenstein, J. (2004). *Inferring Phylogenies*. 2nd edition. MA: Sinauer Associates Sunderland. 978-0878931774.

Fu, L., Niu, B., Zhu, Z., Wu, S., and Li, W. (2012). CD-HIT: Accelerated for Clustering the Next-Generation Sequencing Data. *Bioinformatics* 28 (23), 3150–3152. doi:10.1093/bioinformatics/bts565

Giner, C. R., Forn, I., Romac, S., Logares, R., De Vargas, C., and Massana, R. (2016). Environmental Sequencing Provides Reasonable Estimates of the Relative Abundance of Specific Picoeukaryotes. *Appl. Environ. Microbiol.* 82 (15), 4757–4766. doi:10.1128/AEM.00560-16

Gloor, G. B., Macklaim, J. M., Vu, M., and Fernandes, A. D. (2016). Compositional Uncertainty Should Not Be Ignored in High-Throughput Sequencing Data Analysis. *Austrian J. Stat.* 45 (4), 73. doi:10.17713/ajs.v45i4.122

Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., and Egozcue, J. J. (2017). Microbiome Datasets Are Compositional: And This Is Not Optional. *Front. Microbiol.* 8, 2224. doi:10.3389/fmicb.2017.02224

Gohli, J., Bøifot, K. O., Moen, L. V., Pastuszek, P., Skogan, G., Udekwu, K. I., et al. (2019). The Subway Microbiome: Seasonal Dynamics and Direct Comparison of Air and Surface Bacterial Communities. *Microbiome* 7 (1), 1–16. doi:10.1186/s40168-019-0772-9

Goodwin, S., McPherson, J. D., and McCombie, W. R. (2016). Coming of Age: Ten Years of Next-Generation Sequencing Technologies. *Nat. Rev. Genet.* 17 (6), 333–351. doi:10.1038/nrg.2016.49

Gotelli, N. J., and Colwell, R. K. (2001). Quantifying Biodiversity: Procedures and Pitfalls in the Measurement and Comparison of Species Richness. *Ecol. Lett.* 4 (4), 379–391. doi:10.1046/j.1461-0248.2001.00230.x

Guillou, L., Bachar, D., Audic, S., Bass, D., Berney, C., Bittner, L., et al. (2012). The Protist Ribosomal Reference Database (PR2): a Catalog of Unicellular Eukaryote Small Sub-unit rRNA Sequences with Curated Taxonomy. *Nucleic Acids Res.* 41 (D1), D597–D604. doi:10.1093/nar/gks1160

Haas, B. J., Gevers, D., Earl, A. M., Ward, V., Giannoukos, G., Ciulla, D., et al. (2011). Chimeric 16S rRNA Sequence Formation and Detection in Sanger and 454-pyrosequenced PCR Amplicons. *Genome Res.* 21 (3), 494–504. doi:10.1101/gr.112730.110

Hamerly, G., and Elkan, C. (2004). “Learning the K in K-Means,” in *Advances in Neural Information Processing Systems*. Editors S. Thrun, L. K. Saul, and P. B. Schölkopf (MIT Press), 16, 281–288.

Han, M. V., and Zmasek, C. M. (2009). phyloXML: XML for Evolutionary Biology and Comparative Genomics. *BMC Bioinformatics* 10, 356. doi:10.1186/1471-2105-10-356

Hanson, B., Zhou, Y., Bautista, E. J., Urch, B., Speck, M., Silverman, F., et al. (2016). Characterization of the Bacterial and Fungal Microbiome in Indoor Dust and Outdoor Air Samples: a Pilot Study. *Environ. Sci. Process. Impacts* 18 (6), 713–724. doi:10.1039/c5em00639b

Heather, J. M., and Chain, B. (2016). The Sequence of Sequencers: The History of Sequencing DNA. *Genomics* 107 (1), 1–8. doi:10.1016/j.ygeno.2015.11.003

Hebert, P. D., Cywinska, A., Ball, S. L., and deWaard, J. R. (2003). Biological Identifications through DNA Barcodes. *Proc. Biol. Sci.* 270 (1512), 313–321. doi:10.1098/rspb.2002.2218

Hleap, J. S., Littlefair, J. E., Steinke, D., Hebert, P. D. N., and Cristescu, M. E. (2021). Assessment of Current Taxonomic Assignment Strategies for Metabarcoding Eukaryotes. *Mol. Ecol. Resour.* 21 (7), 2190–2203. doi:10.1111/1755-0998.13407

Hofreiter, M., Serre, D., Poinar, H. N., Kuch, M., Pääbo, S., and Ancient, D. N. A. (2001). Ancient DNA. *Nat. Rev. Genet.* 2 (5), 353–359. doi:10.1038/35072071

Holder, M., and Lewis, P. O. (2003). Phylogeny Estimation: Traditional and Bayesian Approaches. *Nat. Rev. Genet.* 4 (4), 275–284. doi:10.1038/nrg1044

Hubert, F., Grimm, G. W., Jousselin, E., Berry, V., Franc, A., and Kremer, A. (2014). Multiple Nuclear Genes Stabilize the Phylogenetic Backbone of the genusQuercus. *Syst. Biodiversity* 12 (4), 405–423. doi:10.1080/14772000.2014.941037

Huelsenbeck, J. P., Ronquist, F., Nielsen, R., and Bollback, J. P. (5550). Bayesian Inference of Phylogeny and its Impact on Evolutionary Biology. *Science* 294, 2310–2314. doi:10.1126/science.1065889

Hugerth, L. W., and Andersson, A. F. (2017). Analysing Microbial Community Composition through Amplicon Sequencing: From Sampling to Hypothesis Testing. *Front. Microbiol.* 8, 1561. doi:10.3389/fmicb.2017.01561

Huson, D. H., Auch, A. F., Qi, J., and Schuster, S. C. (2007). MEGAN Analysis of Metagenomic Data. *Genome Res.* 17 (3), 377–386. doi:10.1101/gr.5969107

Jackson, D. A. (1997). Compositional Data in Community Ecology: The Paradigm or Peril of Proportions? *Ecology* 78 (3), 929–940. doi:10.1890/0012-9658(1997)078[0929:cdicet]2.0.co;2

Jamy, M., Foster, R., Barbera, P., Czech, L., Kozlov, A., Stamatakis, A., et al. (2019). Long-read Metabarcoding of the Eukaryotic rDNA Operon to Phylogenetically and Taxonomically Resolve Environmental Diversity. *Mol. Ecol. Resour.* 20 (2), 429–443. doi:10.1111/1755-0998.13117

Janssen, S., McDonald, D., Gonzalez, A., Navas-Molina, J. A., Jiang, L., Xu, Z. Z., et al. (2018). Phylogenetic Placement of Exact Amplicon Sequences Improves Associations with Clinical Information. *mSystems* 3 (3), e00021–18. doi:10.1128/mSystems.00021-18

Jeong, J., Yun, K., Mun, S., Chung, W.-H., Choi, S.-Y., Nam, Y.-d., et al. (2021). The Effect of Taxonomic Classification by Full-Length 16s rRNA Sequencing with a Synthetic Long-Read Technology. *Sci. Rep.* 11 (1), January. doi:10.1038/s41598-020-80826-9

Ji, Y., Ashton, L., Pedley, S. M., Edwards, D. P., Tang, Y., Nakamura, A., et al. (2013). Reliable, Verifiable and Efficient Monitoring of Biodiversity via Metabarcoding. *Ecol. Lett.* 16 (10), 1245–1257. doi:10.1111/ele.12162

Jiang, Y., Metin, B., Zhu, Q., and Mirarab, S. (2021). *DEPP: Deep Learning Enables Extending Species Trees Using Single Genes*. bioRxiv. doi:10.1101/2021.01.22.427808

Juan, J. E., and Pawlowsky-Glahn, V. (2005). Groups of Parts and Their Balances in Compositional Data Analysis. *Math. Geology.* 37 (7), 795–828. doi:10.1007/s11004-005-7373-9

Jukes, T. H., and Cantor, C. R. (1969). *Mammalian Protein Metabolism. Chapter Evolution of protein molecules*. New York, United States: Academic Press, Inc. 3, 21–132.

Kanagawa, T. (2013). Bias and Artifacts in Multitemplate Polymerase Chain Reactions (PCR). *J. Biosci. Bioeng.* 96 (4), 317–323. doi:10.1016/S1389-1723(03)90130-7

Kapli, P., Lutteropp, S., Zhang, J., Kobert, K., Pavlidis, P., and Stamatakis, A. (2017). Multi-rate Poisson Tree Processes for Single-Locus Species Delimitation under Maximum Likelihood and Markov Chain Monte Carlo. *Bioinformatics* 33 (11), 1630–1638. doi:10.1093/bioinformatics/btx025

Kapli, P., Yang, Z., and Telford, M. J. (2020). Phylogenetic Tree Building in the Genomic Age. *Nat. Rev. Genet.* 21 (7), 428–444. doi:10.1038/s41576-020-0233-0

Karsenti, E., Acinas, S. G., Bork, P., Bowler, C., De Vargas, C., Raes, J., et al. (2011). A Holistic Approach to marine Eco-Systems Biology. *Plos Biol.* 9 (10), e1001177–11. doi:10.1371/journal.pbio.1001177

Katoh, K., and Frith, M. C. (2012). Adding Unaligned Sequences into an Existing Alignment Using MAFFT and LAST. *Bioinformatics* 28 (23), 3144–3146. doi:10.1093/bioinformatics/bts578

Katoh, K., Misawa, K., Kuma, K., and Miyata, T. (2002). MAFFT: a Novel Method for Rapid Multiple Sequence Alignment Based on Fast Fourier Transform. *Nucleic Acids Res.* 30 (14), 3059–3066. doi:10.1093/nar/gkf436

Katz, K., Shutov, O., Lapoint, R., Kimelman, M., Brister, J. R., and O’Sullivan, C. (2022). The Sequence Read Archive: a Decade More of Explosive Growth. *Nucleic Acids Res.* 50 (D1), D387–D390. doi:10.1093/nar/gkab1053

Keck, F., Vasselon, V., Rimet, F., Bouchez, A., and Kahlert, M. (2018). Boosting DNA Metabarcoding for Biomonitoring with Phylogenetic Estimation of Operational Taxonomic Units’ Ecological Profiles. *Mol. Ecol. Resour.* 18 (6), 1299–1309. doi:10.1111/1755-0998.12919

Kembel, S. W., Wu, M., Eisen, J. A., and Green, J. L. (2012). Incorporating 16s Gene Copy Number Information Improves Estimates of Microbial Diversity and Abundance. *Plos Comput. Biol.* 8 (10), e1002743. doi:10.1371/journal.pcbi.1002743

Kemena, C., and Notredame, C. (2009). Upcoming Challenges for Multiple Sequence Alignment Methods in the High-Throughput Era. *Bioinformatics* 25 (19), 2455–2465. doi:10.1093/bioinformatics/btp452

Koning, E., Phillips, M., and Warnow, T. (2021). “pplacerDC: a New Scalable Phylogenetic Placement Method” in *Proceedings of the 12th ACM Conference on Bioinformatics* (Gainesville, Florida: Computational Biology, and Health Informatics), 1–9.

Koski, L. B., and Golding, G. B. (2001). The Closest BLAST Hit Is Often Not the Nearest Neighbor. *J. Mol. Evol.* 52 (6), 540–542. doi:10.1007/s002390010184

Kozlov, A. M., Darriba, D., Flouri, T., Morel, B., and Stamatakis, A. (2019). A Fast, Scalable, and User-Friendly Tool for Maximum Likelihood Phylogenetic Inference *Bioinformatics* 35 (21), 4453–4455. doi:10.1093/bioinformatics/btz305

Kozlov, A. M., Zhang, J., Yilmaz, P., Glöckner, F. O., and Stamatakis, A. (2016). Phylogeny-aware Identification and Correction of Taxonomically Mislabeled Sequences. *Nucleic Acids Res.* 44 (11), 5022–5033. doi:10.1093/nar/gkw396

Krause, L., Diaz, N. N., Goesmann, A., Kelley, S., Nattkemper, T. W., Rohwer, F., et al. (2008). Phylogenetic Classification of Short Environmental DNA Fragments. *Nucleic Acids Res.* 36 (7), 2230–2239. doi:10.1093/nar/gkn038

Kress, W. J., and Erickson, D. L. (2008). DNA Barcodes: Genes, Genomics, and Bioinformatics. *Proc. Natl. Acad. Sci. U S A.* 105 (8), 2761–2762. doi:10.1073/pnas.0800476105

Kuleshov, V., Jiang, C., Zhou, W., Jahanbani, F., Batzoglou, S., and Snyder, M. (2016). Synthetic Long-Read Sequencing Reveals Intraspecies Diversity in the Human Microbiome. *Nat. Biotechnol.* 34 (1), 64–69. doi:10.1038/nbt.3416

Lacoursière-Roussel, A., Côté, G., Leclerc, V., and Bernatchez, L. (2016). Quantifying Relative Fish Abundance with eDNA: a Promising Tool for Fisheries Management. *J. Appl. Ecol.* 53 (4), 1148–1157. doi:10.1111/1365-2664.12598

Langmead, B., and Salzberg, S. L. (2012). Fast Gapped-Read Alignment with Bowtie 2. *Nat. Methods* 9 (4), 357–359. doi:10.1038/nmeth.1923

Lee, Z. M., Bussema, C., and Schmidt, T. M. (2009). rrnDB: Documenting the Number of rRNA and tRNA Genes in Bacteria and Archaea. *Nucleic Acids Res.* 37, D489–D493. doi:10.1093/nar/gkn689

Lefeuvre, P. (2018). *BoSSA: A Bunch of Structure and Sequence Analysis*.

Lemoine, F., Domelevo Entfellner, J.-B., Wilkinson, E., Correia, D., Dávila Felipe, M., De Oliveira, T., et al. (2018). Renewing Felsenstein's Phylogenetic Bootstrap in the Era of Big Data. *Nature* 556 (7702), 452–456. doi:10.1038/s41586-018-0043-0

Letunic, I., and Bork, P. (2016). Interactive Tree of Life (iTOL) V3: an Online Tool for the Display and Annotation of Phylogenetic and Other Trees. *Nucleic Acids Res.* 44 (W1), W242–W245. doi:10.1093/nar/gkw290

Letunic, I., and Bork, P. (2019). Interactive Tree of Life (iTOL) V4: Recent Updates and New Developments. *Nucleic Acids Res.* 47 (W1), W256–W259. doi:10.1093/nar/gkz239

Li, H., and Durbin, R. (2010). Fast and Accurate Long-Read Alignment with Burrows-Wheeler Transform. *Bioinformatics* 26 (5), 589–595. doi:10.1093/bioinformatics/btp698

Li, H., and Durbin, R. (2009). Fast and Accurate Short Read Alignment with Burrows-Wheeler Transform. *Bioinformatics* 25 (14), 1754–1760. doi:10.1093/bioinformatics/btp324

Li, H. (2015). Microbiome, Metagenomics, and High-Dimensional Compositional Data Analysis. *Annu. Rev. Stat. Appl.* 2 (1), 73–94. doi:10.1146/annurev-statistics-010814-020351

Liede-Schumann, S., Grimm, G. W., Nürk, N. M., Potts, A. J., Meve, U., and Hartmann, H. E. K. (2020). Phylogenetic Relationships in the Southern African Genus Drosanthemum (Ruschioideae, Aizoaceae). *PeerJ* 8 (3), e8999. doi:10.7717/peerj.8999

Lin, H., and Peddada, S. D. (2020). Analysis of Microbial Compositions: a Review of Normalization and Differential Abundance Analysis. *NPJ Biofilms Microbiomes* 61 (1), 601–613. doi:10.1038/s41522-020-00160-w

Linard, B., Romashchenko, N., Pardi, F., and Rivals, E. (2020). PEWO: a Collection of Workflows to Benchmark Phylogenetic Placement. *Bioinformatics*. doi:10.1093/bioinformatics/btaa657

Linard, B., Swenson, K., and Pardi, F. (2019). Rapid Alignment-free Phylogenetic Identification of Metagenomic Sequences. *Bioinformatics* 35 (18), 3303–3312. doi:10.1093/bioinformatics/btz068

Lindgreen, S., Adair, K. L., and Gardner, P. P. (2016). An Evaluation of the Accuracy and Speed of Metagenome Analysis Tools. *Sci. Rep.* 6 (1), 19233. doi:10.1038/srep19233

Liu, K., Warnow, T. J., Holder, M. T., Nelesen, S. M., Yu, J., Stamatakis, A. P., et al. (2012). SATe-II: Very Fast and Accurate Simultaneous Estimation of Multiple Sequence Alignments and Phylogenetic Trees. *Syst. Biol.* 61 (1), 90–106. doi:10.1093/sysbio/syr095

Logares, R., Haverkamp, T. H., Kumar, S., Lanzén, A., Nederbragt, A. J., Quince, C., et al. (2012). Environmental Microbiology through the Lens of High-Throughput DNA Sequencing: Synopsis of Current Platforms and Bioinformatics Approaches. *J. Microbiol. Methods* 91 (1), 106–113. doi:10.1016/j.mimet.2012.07.017

Logares, R., Sunagawa, S., Salazar, G., Cornejo-Castillo, F. M., Ferrera, I., Sarmento, H., et al. (2014). Metagenomic 16S rDNA Illumina Tags Are a Powerful Alternative to Amplicon Sequencing to Explore Diversity and Structure of Microbial Communities. *Environ. Microbiol.* 16 (9), 2659–2671. doi:10.1111/1462-2920.12250

López-García, A., Pineda-Quiroga, C., Atxaerandio, R., Adrian, P., Hernández, I., García-Rodríguez, A., et al. (2018). Comparison of Mothur and QIIME for the Analysis of Rumen Microbiota Composition Based on 16S rRNA Amplicon Sequences. *Front. Microbiol.* 9 (DEC), 1–11. doi:10.3389/fmicb.2018.03010

Lorimer, J., Hodgetts, T., Grenyer, R., Greenhough, B., McLeod, C., and Dwyer, A. (2019). Making the Microbiome Public: Participatory Experiments with DNA Sequencing in Domestic Kitchens. *Trans. Inst. Br. Geogr.* 44 (3), 524–541. doi:10.1111/tran.12289

Love, M. I., Hogenesch, J. B., and Irizarry, R. A. (2016). Modeling of RNA-Seq Fragment Sequence Bias Reduces Systematic Errors in Transcript Abundance Estimation. *Nat. Biotechnol.* 34 (12), 1287–1291. doi:10.1038/nbt.3682

Löytynoja, A., Vilella, A. J., and Goldman, N. (2012). Accurate Extension of Multiple Sequence Alignments Using a Phylogeny-Aware Graph Algorithm. *Bioinformatics* 28 (13), 1684–1691. doi:10.1093/bioinformatics/bts198

Lozupone, C., and Knight, R. (2005). UniFrac: a New Phylogenetic Method for Comparing Microbial Communities. *Appl. Environ. Microbiol.* 71 (12), 8228–8235. doi:10.1128/AEM.71.12.8228-8235.2005

Lozupone, C. A., Hamady, M., Kelley, S. T., and Knight, R. (2007). Quantitative and Qualitative Beta Diversity Measures lead to Different Insights into Factors that Structure Microbial Communities. *Appl. Environ. Microbiol.* 73 (5), 1576–1585. doi:10.1128/AEM.01996-06

Mahé, F., de Vargas, C., Bass, D., Czech, L., Stamatakis, A., Lara, E., et al. (2017). Parasites Dominate Hyperdiverse Soil Protist Communities in Neotropical Rainforests. *Nat. Ecol. Evol.* 1 (4), 91. doi:10.1038/s41559-017-0091

Mahé, F., Czech, L., Stamatakis, A., Quince, C., de Vargas, C., Dunthorn, M., et al. (2021). Swarm V3: towards Tera-Scale Amplicon Clustering. *Bioinformatics* 38 (1), 267–269. doi:10.1093/bioinformatics/btab493

Mardis, E. R. (2016). DNA Sequencing Technologies: 2006-2016. *Nat. Protoc.* 12 (2), 213–218. doi:10.1038/nprot.2016.182

Mardis, E. R. (2013). Next-generation Sequencing Platforms. *Annu. Rev. Anal. Chem. (Palo Alto Calif.* 6 (1), 287–303. doi:10.1146/annurev-anchem-062012-092628

Martiniano, R., De Sanctis, B., Hallast, P., and Durbin, R. (2022). Placing Ancient DNA Sequences into Reference Phylogenies. *Mol. Biol. Evol.* 39 (2), msac017. doi:10.1093/molbev/msac017

Matsen, F. A., and Evans, S. N. (2013). Edge Principal Components and Squash Clustering: Using the Special Structure of Phylogenetic Placement Data for Sample Comparison. *PLOS ONE* 8 (3), e56859–17. doi:10.1371/journal.pone.0056859

Matsen, F. A., and Gallagher, A. (2012). Reconciling Taxonomy and Phylogenetic Inference: Formalism and Algorithms for Describing Discord and Inferring Taxonomic Roots. *Algorithms Mol. Biol.* 7 (1), 8. doi:10.1186/1748-7188-7-8

Matsen, F. A., Hoffman, N. G., Gallagher, A., and Stamatakis, A. (2012). A Format for Phylogenetic Placements. *PLoS ONE* 7 (2), e31009–4. doi:10.1371/journal.pone.0031009

Matsen, F. A., Kodner, R. B., and Armbrust, E. V. (2010). Pplacer: Linear Time Maximum-Likelihood and Bayesian Phylogenetic Placement of Sequences onto a Fixed Reference Tree. *BMC Bioinformatics* 11 (1), 538. doi:10.1186/1471-2105-11-538

Matsen, F. A. (2015). Phylogenetics and the Human Microbiome. *Syst. Biol.* 64 (1). doi:10.1093/sysbio/syu053

McCoy, C. O., and Matsen, F. A. (2013). Abundance-weighted Phylogenetic Diversity Measures Distinguish Microbial Community States and Are Robust to Sampling Depth. *PeerJ* 1, e157. doi:10.7717/peerj.157

McDonald, D., Price, M. N., Goodrich, J., Nawrocki, E. P., Desantis, T. Z., Probst, A., et al. (2012). An Improved Greengenes Taxonomy with Explicit Ranks for Ecological and Evolutionary Analyses of Bacteria and Archaea. *ISME J.* 6 (3), 610–618. doi:10.1038/ismej.2011.139

McMurdie, P. J., and Holmes, S. (2014). Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible. *PLoS Comput. Biol.* 10 (4), e1003531. doi:10.1371/journal.pcbi.1003531

Menzel, P., Ng, K. L., and Krogh, A. (2016). Fast and Sensitive Taxonomic Classification for Metagenomics with Kaiju. *Nat. Commun.* 7 (1), 11257–11259. doi:10.1038/ncomms11257

Methé, B. A., Nelson, K. E., Pop, M., Creasy, H. H., Giglio, M. G., Curtis, H., et al. (2012). A Framework for Human Microbiome Research. *Nature* 486 (7402), 215–221. doi:10.1038/nature11209

Metin, B., Jiang, Y., Roush, D., Zhu, Q., and Mirarab, S. (2021). Fast and Accurate Distance-Based Phylogenetic Placement Using divide and Conquer. *Mol. Ecol. Resour.* 22 (3), 1213–1227. doi:10.1111/1755-0998

Metin, B., Sarmashghi, S., and Mirarab, S. (2019). APPLES: Scalable Distance-Based Phylogenetic Placement with or without Alignments. *Syst. Biol.* doi:10.1093/sysbio/syz063/5572672

Meyer, A., Todt, C., Mikkelsen, N. T., and Lieb, B. (2010). Fast Evolving 18S rRNA Sequences from Solenogastres (Mollusca) Resist Standard PCR Amplification and Give New Insights into Mollusk Substitution Rate Heterogeneity. *BMC Evol. Biol.* 110 (1), 70. doi:10.1186/1471-2148-10-70

Meyer, F., Bremges, A., Belmann, P., Janssen, S., McHardy, A. C., and Koslicki, D. (2019). Assessing Taxonomic Metagenome Profilers with OPAL. *Genome Biol.* 20 (1), 51. doi:10.1186/s13059-019-1646-y

Mignardi, M., and Nilsson, M. (2014). Fourth-generation Sequencing in the Cell and the Clinic. *Genome Med.* 6 (4), 31. doi:10.1186/gm548

Mirarab, S., Nguyen, N., and Warnow, T. (2012). "SEPP: SATé-Enabled Phylogenetic Placement," in *Pacific Symposium on Biocomputing* (World Scientific), 247–258. doi:10.1142/9789814366496_0024

Morel, B., Barbera, P., Czech, L., Bettisworth, B., Hübner, L., Lutteropp, S., et al. (2020). Phylogenetic Analysis of SARS-CoV-2 Data Is Difficult. *Mol. Biol. Evol.* 38 (5), 1777–1791. doi:10.1093/molbev/msaa314

Moret, B. M. E., Roshan, U., and Warnow, T. (2002). "Sequence-length Requirements for Phylogenetic Methods," in *Lecture Notes in Computer Science*. Editors R. Guigó and D. Gusfield (Berlin, Heidelberg: Springer Berlin Heidelberg), 2452, 343–356. 3540442111. doi:10.1007/3-540-45784-4_26

Morgan, J. L., Darling, A. E., and Eisen, J. A. (2010). Metagenomic Sequencing of an In Vitro-simulated Microbial Community. *PLoS ONE* 5 (4), e10209–10. doi:10.1371/journal.pone.0010209

Morgan-Lang, C., McLaughlin, R., Armstrong, Z., Zhang, G., Chan, K., and Hallam, S. J. (2020). TreeSAPP: the Tree-Based Sensitive and Accurate Phylogenetic Profiler. *Bioinformatics* 36 (18), 4706–4713. doi:10.1093/bioinformatics/btaa588

Mühlemann, B., Vinner, L., Margaryan, A., Wilhelmson, H., De La Fuente Castro, C., Allentoft, M. E., et al. (2020). Diverse variola Virus (Smallpox) Strains Were Widespread in Northern Europe in the Viking Age. *Science* 369 (6502). doi:10.1126/science.aaw8977

Muir, P., Li, S., Lou, S., Wang, D., Spakowicz, D. J., Salichos, L., et al. (2016). Erratum to: The Real Cost of Sequencing: Scaling Computation to Keep Pace with Data Generation. *Genome Biol.* 17 (1), 78–79. doi:10.1186/s13059-016-0961-9

Nguyen, L. T., Schmidt, H. A., Von Haeseler, A., and Minh, B. Q. (2015). IQ-TREE: a Fast and Effective Stochastic Algorithm for Estimating Maximum-Likelihood Phylogenies. *Mol. Biol. Evol.* 32 (1), 268–274. doi:10.1093/molbev/msu300

Nguyen, N. P., Mirarab, S., Liu, B., Pop, M., and Warnow, T. (2014). TIPP: Taxonomic Identification and Phylogenetic Profiling. *Bioinformatics* 30 (24), 3548–3555. doi:10.1093/bioinformatics/btu721

Niedringhaus, T. P., Milanova, D., Kerby, M. B., Snyder, M. P., and Barron, A. E. (2011). Landscape of Next-Generation Sequencing Technologies. *Anal. Chem.* 83 (12), 4327–4341. doi:10.1021/ac2010857

Notredame, C., Higgins, D. G., and Heringa, J. (2000). T-coffee: a Novel Method for Fast and Accurate Multiple Sequence Alignment. *J. Mol. Biol.* 302 (1), 205–217. doi:10.1006/jmbi.2000.4042

Nugent, R. P., Krohn, M. A., and Hillier, S. L. (1991). Reliability of Diagnosing Bacterial Vaginosis Is Improved by a Standardized Method of Gram Stain Interpretation. *J. Clin. Microbiol.* 29 (2), 297–301. doi:10.1128/JCM.29.2.297-301.1991

Ondov, B. D., Bergman, N. H., and Phillippy, A. M. (2011). Interactive Metagenomic Visualization in a Web Browser. *BMC Bioinformatics* 12 (1), 385. doi:10.1186/1471-2105-12-385

Oulas, A., Pavloudi, C., Polymenakou, P., Pavlopoulos, G. A., Papanikolaou, N., Kotoulas, G., et al. (2015). Metagenomics: Tools and Insights for Analyzing Next-Generation Sequencing Data Derived from Biodiversity Studies. *Bioinform Biol. Insights* 9 (75–88), 75–88. doi:10.4137/BBI.S12462

Pareek, C. S., Smoczynski, R., and Tretyn, A. (2011). Sequencing Technologies and Genome Sequencing. *J. Appl. Genet.* 52 (4), 413–435. doi:10.1007/s13353-011-0057-x

Parks, D. H., Rinke, C., Chuvochina, M., Chaumeil, P. A., Woodcroft, B. J., Evans, P. N., et al. (2017). Recovery of Nearly 8,000 Metagenome-Assembled Genomes Substantially Expands the Tree of Life. *Nat. Microbiol.* 2 (11), 1533–1542. doi:10.1038/s41564-017-0012-7

Pawlowsky-Glahn, V., Egozcue, J. J., and Tolosana-Delgado, R. (2015). *Modeling and Analysis of Compositional Data*. Chichester, UK: John Wiley & Sons.

Peabody, M. A., Van Rossum, T., Lo, R., and Brinkman, F. S. (2015). Evaluation of Shotgun Metagenomics Sequence Classification Methods Using In Silico and *In Vitro* Simulated Communities. *BMC Bioinformatics* 16, 363. doi:10.1186/s12859-015-0788-5

Pearson, W. R., and Lipman, D. J. (1988). Improved Tools for Biological Sequence Comparison. *Proc. Natl. Acad. Sci. U S A.* 85 (8), 2444–2448. doi:10.1073/pnas.85.8.2444

Pelleg, D., and Moore, A. W. (2000). X-means: Extending K-Means with Efficient Estimation of the Number of Clusters. *ICML* 1, 727–734.

Peng, X., Li, G., and Liu, Z. (2016). Zero-Inflated Beta Regression for Differential Abundance Analysis with Metagenomics Data. *J. Comput. Biol.* 23 (2), 102. doi:10.1089/cmb.2015.0157

Pereira-Flores, E., Glöckner, F. O., and Fernandez-Guerra, A. (2019). Fast and Accurate Average Genome Size and 16s rRNA Gene Average Copy Number Computation in Metagenomic Data. *BMC Bioinformatics* 20 (1), 453. doi:10.1186/s12859-019-3031-y

Pervez, M. T., Babar, M. E., Nadeem, A., Aslam, M., AwanAwan, A. R., Aslam, N., et al. (2014). Evaluating the Accuracy and Efficiency of Multiple Sequence Alignment Methods. *Evol. Bioinform Online* 10, 205–217. doi:10.4137/EBO.S19199

Petrenko, P., Lobb, B., Kurtz, D. A., Neufeld, J. D., and Doxey, A. C. (2015). MetAnnotate: Function-specific Taxonomic Profiling and Comparison of Metagenomes. *BMC Biol.* 13 (1), 92. doi:10.1186/s12915-015-0195-4

Pettersson, E., Lundeberg, J., and Ahmadian, A. (2009). Generations of Sequencing Technologies. *Genomics* 93 (2), 105–111. doi:10.1016/j.ygeno.2008.10.003

Piredda, R., Grimm, G. W., Schulze, E. D., Denk, T., and Simeone, M. C. (2021). High-throughput Sequencing of 5S-IGS in oaks: Exploring Intragenomic Variation and Algorithms to Recognize Target Species in Pure and Mixed Samples. *Mol. Ecol. Resour.* 21 (2), 495–510. doi:10.1111/1755-0998.13264

Price, M. N., Dehal, P. S., and Arkin, A. P. (2010). FastTree 2-approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE* 5 (3), e9490. doi:10.1371/journal.pone.0009490

Prodan, A., Tremaroli, V., Brolin, H., Zwinderman, A. H., Nieuwdorp, M., and Levin, E. (2020). Comparing Bioinformatic Pipelines for Microbial 16S rRNA Amplicon Sequencing. *PLoS ONE* 15 (1), e0227434–19. doi:10.1371/journal.pone.0227434

Pruesse, E., Quast, C., Knittel, K., Fuchs, B. M., Ludwig, W., Peplies, J., et al. (2007). SILVA: a Comprehensive Online Resource for Quality Checked and Aligned Ribosomal RNA Sequence Data Compatible with ARB. *Nucleic Acids Res.* 35 (21), 7188–7196. doi:10.1093/nar/gkm864

Quast, C., Pruesse, E., Yilmaz, P., Gerken, J., Schweer, T., Yarza, P., et al. (2013). The SILVA Ribosomal RNA Gene Database Project: Improved Data Processing and Web-Based Tools. *Nucleic Acids Res.* 41 (D1), D590–D596. doi:10.1093/nar/gks1219

Quinn, T. P., Erb, I., Richardson, M. F., and Crowley, T. M. (2018). Understanding Sequencing Data as Compositions: an Outlook and Review. *Bioinformatics* 34 (16), 2870–2878. doi:10.1093/bioinformatics/bty175

Rabiee, M., and Mirarab, S. (2019). INSTRAL: Discordance-Aware Phylogenetic Placement Using Quartet Scores. *Syst. Biol.* 69 (2), 384–391. doi:10.1093/sysbio/syz045

Rajter, Ľ., and Dunthorn, M. (2021). Ciliate SSU-rDNA Reference Alignments and Trees for Phylogenetic Placements of Metabarcoding Data. *Metabarcoding and Metagenomics* 5, e69602. doi:10.3897/mbmg.5.69602

Rajter, Ľ., Ewers, I., Graupner, N., Vďačný, P., and Dunthorn, M. (2021). Colpodean Ciliate Phylogeny and Reference Alignments for Phylogenetic Placements. *Eur. J. Protistol* 77, 125747. doi:10.1016/j.ejop.2020.125747

Ren, R., Sun, Y., Zhao, Y., Geiser, D., Ma, H., and Zhou, X. (2016). Phylogenetic Resolution of Deep Eukaryotic and Fungal Relationships Using Highly Conserved Low-Copy Nuclear Genes. *Genome Biol. Evol.* 8 (9), 2683–2701. doi:10.1093/gbe/evw196

Reuter, J. A., Spacek, D. V., and Snyder, M. P. (2015). High-Throughput Sequencing Technologies. *Mol. Cel* 58 (4), 586–597. doi:10.1016/j.molcel.2015.05.004

Ritter, C. D., Dunthorn, M., Anslan, S., de Lima, V. X., Tedersoo, L., Nilsson, R. H., et al. (2020). Advancing Biodiversity Assessments with Environmental DNA: Long-Read Technologies Help Reveal the Drivers of Amazonian Fungal Diversity. *Ecol. Evol.* 10 (14), 7509–7524. doi:10.1002/ece3.6477

Rognes, T., Flouri, T., Nichols, B., Quince, C., and Mahé, F. (2016). VSEARCH: a Versatile Open Source Tool for Metagenomics. *PeerJ* 4, e2584. doi:10.7717/peerj.2584

Ronquist, F. (2004). Bayesian Inference of Character Evolution. *Trends Ecol. Evol.* 19 (9), 475–481. doi:10.1016/j.tree.2004.07.002

Rousseeuw, P. J. (1987). Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *J. Comput. Appl. Math.* 20, 53–65. doi:10.1016/0377-0427(87)90125-7

Rubinat-Ripoll, L. (2019). *Lrubinat/Photoreft: A 16s Rdna Reference Tree Representing the Main Groups of Picophototrophic Eukaryotes and Prokaryotes*. Available at: https://zenodo.org/record/3476953.

Ruppert, K. M., Kline, R. J., and Rahman, M. S. (2019). Past, Present, and Future Perspectives of Environmental Dna (edna) Metabarcoding: A Systematic Review in Methods, Monitoring, and Applications of Global edna. *Glob. Ecol. Conservation* 17, e00547. doi:10.1016/j.gecco.2019.e00547

Saitou, N., and Nei, M. (1987). The Neighbor-Joining Method: A New Method for Reconstructing Phylogenetic Trees. *Mol. Biol. Evol.* 4 (4), 406–425. doi:10.1093/oxfordjournals.molbev.a040454

Sankoff, D. (1975). Minimal Mutation Trees of Sequences. *SIAM J. Appl. Math.* 28 (1), 35–42. doi:10.1137/0128004

Savolainen, V., Cowan, R. S., Vogler, A. P., Roderick, G. K., and Lane, R. (2005). Towards Writing the Encyclopedia of Life: An Introduction to DNA Barcoding. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 360 (1462), 1805–1811. doi:10.1098/rstb.2005.1730

Sayers, E. W., Barrett, T., Benson, D. A., Bryant, S. H., Canese, K., Chetvernin, V., et al. (2009). Database Resources of the National Center for Biotechnology Information. *Nucleic Acids Res.* 37, D5–D15. doi:10.1093/nar/gkn741

Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, E. B., et al. (2009). Introducing Mothur: Open-Source, Platform-independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Appl. Environ. Microbiol.* 75 (23), 7537–7541. doi:10.1128/AEM.01541-09

Schön, M. E., Eme, L., and Ettema, T. J. G. (2019). PhyloMagnet: Fast and Accurate Screening of Short-Read Meta-Omics Data Using Gene-Centric Phylogenetics. *Bioinformatics* 36 (6), 1718–1724. doi:10.1093/bioinformatics/btz799

Schreiber, F., Gumrich, P., Daniel, R., and Meinicke, P. (2010). Treephyler: Fast Taxonomic Profiling of Metagenomes. *Bioinformatics* 26 (7), 960–961. doi:10.1093/bioinformatics/btq070

Sczyrba, A., Hofmann, P., Belmann, P., Koslicki, D., Janssen, S., Dröge, J., et al. (2017). Critical Assessment of Metagenome Interpretation-A Benchmark of Metagenomics Software. *Nat. Methods* 14 (11), 1063–1071. doi:10.1038/nmeth.4458

Segata, N., Waldron, L., Ballarini, A., Narasimhan, V., Jousson, O., and Huttenhower, C. (2012). Metagenomic Microbial Community Profiling Using Unique Clade-specific Marker Genes. *Nat. Methods* 9 (8), 811–814. doi:10.1038/nmeth.2066

Sempéré, G., Pétel, A., Abbé, M., Lefeuvre, P., Roumagnac, P., Mahé, F., et al. (2021). metaXplor: an Interactive Viral and Microbial Metagenomic Data Manager. *GigaScience* 10 (2), January. doi:10.1093/gigascience/giab001

Shah, N., Molloy, E. K., Pop, M., and Warnow, T. (2021). TIPP2: Metagenomic Taxonomic Profiling Using Phylogenetic Markers. *Bioinformatics*. doi:10.1093/bioinformatics/btab023

Shah, N., Nute, M. G., Warnow, T., and Pop, M. (2019). Misunderstood Parameter of NCBI BLAST Impacts the Correctness of Bioinformatics Workflows. *Bioinformatics*. doi:10.1093/bioinformatics/bty833

Sharon, I., Kertesz, M., Hug, L. A., Pushkarev, D., Blauwkamp, T. A., Castelle, C. J., et al. (2015). Accurate, Multi-Kb Reads Resolve Complex Populations and Detect Rare Microorganisms. *Genome Res.* 25 (4), 534–543. doi:10.1101/gr.183012.114

Silverman, J. D., Bloom, R. J., Jiang, S., Durand, H. K., Dallow, E., Mukherjee, S., et al. (2021). Measuring and Mitigating PCR Bias in Microbiota Datasets. *Plos Comput. Biol.* 17 (7), e1009113. doi:10.1371/journal.pcbi.1009113

Silverman, J. D., Washburne, A. D., Mukherjee, S., and Lawrence, A. D. (2017). A Phylogenetic Transform Enhances Analysis of Compositional Microbiota Data. *eLife* 6, e21887. doi:10.7554/eLife.21887

Simão, F. A., Waterhouse, R. M., Ioannidis, P., Kriventseva, E. V., and Zdobnov, E. M. (2015). BUSCO: Assessing Genome Assembly and Annotation Completeness with Single-Copy Orthologs. *Bioinformatics* 31 (19), 3210–3212. doi:10.1093/bioinformatics/btv351

Smith, S. A., and Pease, J. B. (2017). Heterogeneous Molecular Processes Among the Causes of How Sequence Similarity Scores Can Fail to Recapitulate Phylogeny. *Brief Bioinform* 18 (3), 451–457. doi:10.1093/bib/bbw034

Srinivasan, S., Hoffman, N. G., Morgan, M. T., Matsen, F. A., Fiedler, T. L., Hall, R. W., et al. (2012). Bacterial Communities in Women with Bacterial Vaginosis: High Resolution Phylogenetic Analyses Reveal Relationships of Microbiota to Clinical Criteria. *PLOS ONE* 7 (6), e37818. doi:10.1371/journal.pone.0037818

Stamatakis, A. (2014). RAxML Version 8: a Tool for Phylogenetic Analysis and post-analysis of Large Phylogenies. *Bioinformatics* 30 (9), 1312–1313. doi:10.1093/bioinformatics/btu033

Stark, M., Berger, S. A., Stamatakis, A., and von Mering, C. (2010). MLTreeMap-accurate Maximum Likelihood Placement of Environmental DNA Sequences into Taxonomic and Functional Reference Phylogenies. *BMC Genomics* 11 (1), 461. doi:10.1186/1471-2164-11-461

Strimmer, K., and Rambaut, A. (2002). Inferring Confidence Sets of Possibly Misspecified Gene Trees. *Proc. Biol. Sci.* 269, 137–142. doi:10.1098/rspb.2001.1862

Sunagawa, S., Mende, D. R., Zeller, G., Izquierdo-Carrasco, F., Berger, S. A., Kultima, J. R., et al. (2013). Metagenomic Species Profiling Using Universal Phylogenetic Marker Genes. *Nat. Methods* 10 (12), 1196–1199. doi:10.1038/nmeth.2693

Temperton, B., Giovannoni, S. J., and Metagenomics, G. (2012). Metagenomics: Microbial Diversity through a Scratched Lens. *Curr. Opin. Microbiol.* 15 (5), 605–612. doi:10.1016/j.mib.2012.07.001

Thomas, T., Gilbert, J., and Meyer, F. (2012). Metagenomics - a Guide from Sampling to Data Analysis. *Microb. Inform. Exp.* 2 (1), 3. doi:10.1186/2042-5783-2-3

Thorndike, R. L. (1953). Who Belongs in the Family? *Psychometrika* 18 (4), 267–276. doi:10.1007/bf02289263

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the Number of Clusters in a Data Set via the gap Statistic. *J. R. Stat. Soc. Ser. B (Statistical Methodology)* 63 (2), 411–423. doi:10.1111/1467-9868.00293

Truong, D. T., Franzosa, E. A., Tickle, T. L., Scholz, M., Weingart, G., Pasolli, E., et al. (2015). MetaPhlAn2 for Enhanced Metagenomic Taxonomic Profiling. *Nat. Methods* 12 (10), 902–903. doi:10.1038/nmeth.3589

Tsilimigras, M. C. B., and Fodor, A. A. (2016). Compositional Data Analysis of the Microbiome: Fundamentals, Tools, and Challenges. *Ann. Epidemiol.* 26 (5), 330–335. doi:10.1016/j.annepidem.2016.03.002

Tucker, C. M., Cadotte, M. W., Carvalho, S. B., Davies, T. J., Ferrier, S., Fritz, S. A., et al. (2017). A Guide to Phylogenetic Metrics for Conservation, Community Ecology and Macroecology. *Biol. Rev. Camb Philos. Soc.* 92 (2), 698–715. doi:10.1111/brv.12252

Turakhia, Y., Thornlow, B., Hinrichs, A. S., De Maio, N., Gozashti, L., Lanfear, R., et al. (2021). Ultrafast Sample Placement on Existing tRees (UShER) Enables Real-Time Phylogenetics for the SARS-CoV-2 Pandemic. *Nat. Genet.* 53 (6), 809–816. doi:10.1038/s41588-021-00862-7

Tyson, G. W., Chapman, J., Hugenholtz, P., Allen, E. E., Ram, R. J., Richardson, P. M., et al. (2020). Community Structure and Metabolism through Reconstruction of Microbial Genomes from the Environment. *Nature* 428, 37–43. doi:10.1038/nature02340

van Dijk, E. L., Auger, H., Jaszczyszyn, Y., and Thermes, C. (2014). Ten Years of Next-Generation Sequencing Technology. *Trends Genet.* 30 (9), 418–426. doi:10.1016/j.tig.2014.07.001

von Mering, C., Hugenholtz, P., Raes, J., Tringe, S. G., Doerks, T., Jensen, L. J., et al. (2007). Quantitative Phylogenetic Assessment of Microbial Communities in Diverse Environments. *Science* 315 (5815), 1126–1130. doi:10.1126/science.1133420

Wang, L. G., Lam, T. T., Xu, S., Dai, Z., Zhou, L., Feng, T., et al. (2020). Treeio: An R Package for Phylogenetic Tree Input and Output with Richly Annotated and Associated Data. *Mol. Biol. Evol.* 37 (2), 599–603. doi:10.1093/molbev/msz240

Wang, Q., Garrity, G. M., Tiedje, J. M., and Cole, J. R. (2007). Naive Bayesian Classifier for Rapid Assignment of rRNA Sequences into the New Bacterial Taxonomy. *Appl. Environ. Microbiol.* 73 (16), 5261–5267. doi:10.1128/AEM.00062-07

Wang, W. L., Xu, S. Y., Ren, Z. G., Tao, L., Jiang, J. W., and Zheng, S. S. (2015). Application of Metagenomics in the Human Gut Microbiome. *World J. Gastroenterol.* 21 (3), 803–814. doi:10.3748/wjg.v21.i3.803

Washburne, A. D., Silverman, J. D., Leff, J. W., Dominic, J., Bennett, J. L. D., Mukherjee, S., et al. (2017). Phylogenetic Factorization of Compositional Data Yields Lineage-Level Associations in Microbiome Datasets. *PeerJ* 5, e2969. doi:10.7717/peerj.2969

Washburne, A. D., Silverman, J. D., Morton, J. T., Becker, D. J., Crowley, D., Mukherjee, S., et al. (2019). Phylofactorization: a Graph Partitioning Algorithm to Identify Phylogenetic Scales of Ecological Data. *Ecol. Monogr.* 89 (2), e01353. doi:10.1002/ecm.1353

Wedell, E., Cai, Y., and Warnow, T. (2021). "Scalable and Accurate Phylogenetic Placement Using Pplacer-XR," in *International Conference on Algorithms for Computational Biology* (Springer), 94–105. doi:10.1007/978-3-030-74432-8_7

Weisburg, W. G., Barns, S. M., Pelletier, D. A., and Lane, D. J. (1991). 16S Ribosomal DNA Amplification for Phylogenetic Study. *J. Bacteriol.* 173 (2), 697–703. doi:10.1128/jb.173.2.697-703.1991

Weiss, S., Xu, Z. Z., Peddada, S., Amir, A., Bittinger, K., Gonzalez, A., et al. (2017). Normalization and Microbial Differential Abundance Strategies Depend upon Data Characteristics. *Microbiome* 5 (1), 27. doi:10.1186/s40168-017-0237-y

Westcott, S. L., and Schloss, P. D. (2015). De Novo clustering Methods Outperform Reference-Based Methods for Assigning 16S rRNA Gene Sequences to Operational Taxonomic Units. *PeerJ* 3 (12), e1487. doi:10.7717/peerj.1487

Woese, C. R., and Fox, G. E. (1977). Phylogenetic Structure of the Prokaryotic Domain: the Primary Kingdoms. *Proc. Natl. Acad. Sci. U S A.* 74 (11), 5088–5090. doi:10.1073/pnas.74.11.5088

Woese, C. R., Kandler, O., and Wheelis, M. L. (1990). Towards a Natural System of Organisms: Proposal for the Domains Archaea, Bacteria, and Eucarya. *Proc. Natl. Acad. Sci. U S A.* 87 (12), 4576–4579. doi:10.1073/pnas.87.12.4576

Wood, D. E., Lu, J., and Langmead, B. (2019). Improved Metagenomic Analysis with Kraken 2. *Genome Biol.* 20 (1), 1–13. doi:10.1186/s13059-019-1891-0

Wood, D. E., Salzberg, S. L., Heidelberg, J., Halpern, A., Rusch, D., Eisen, J., et al. (2014). Kraken: Ultrafast Metagenomic Sequence Classification Using Exact Alignments. *Genome Biol.* 15 (3), R46. doi:10.1186/gb-2014-15-3-r46

Wu, M., and Scott, A. J. (2012). Phylogenomic Analysis of Bacterial and Archaeal Sequences with AMPHORA2. *Bioinformatics* 28 (7), 1033–1034. doi:10.1093/bioinformatics/bts079

Yang, Z. (2006). *Computational Molecular Evolution*. Oxford University Press.

Ye, S. H., Siddle, K. J., Park, D. J., and Sabeti, P. C. (2019). Benchmarking Metagenomics Tools for Taxonomic Classification. *Cell* 178 (4), 779–794. doi:10.1016/j.cell.2019.07.010

Yilmaz, P., Parfrey, L. W., Yarza, P., Gerken, J., Pruesse, E., Quast, C., et al. (2014). The SILVA and "All-Species Living Tree Project (LTP)" Taxonomic Frameworks. *Nucleic Acids Res.* 42 (D1), D643–D648. doi:10.1093/nar/gkt1209

Yu, G., Smith, D. K., Zhu, H., Guan, Y., and Lam, T. T. Y. (2017). Ggtree : an R Package for Visualization and Annotation of Phylogenetic Trees with Their Covariates and Other Associated Data. *Methods Ecol. Evol.* 8 (1), 28–36. doi:10.1111/2041-210X.12628

Zhang, J., Kapli, P., Pavlidis, P., and Stamatakis, A. (2013). A General Species Delimitation Method with Applications to Phylogenetic Placements. *Bioinformatics* 29 (22), 2869–2876. doi:10.1093/bioinformatics/btt499

Zheng, Q., Bartow-McKenney, C., Meisel, J. S., and Grice, E. A. (2018). HmmUFOtu: An HMM and Phylogenetic Placement Based Ultra-fast Taxonomic Assignment and OTU Picking Tool for Microbiome Amplicon Sequencing Studies. *Genome Biol.* 19 (1), 82. doi:10.1186/s13059-018-1450-0

Zhou, X., Shen, X. X., Hittinger, C. T., and Rokas, A. (2018). Evaluating Fast Maximum Likelihood-Based Phylogenetic Programs Using Empirical Phylogenomic Data Sets. *Mol. Biol. Evol.* 35 (2), 486–503. doi:10.1093/molbev/msx302

Zou, Q., Lin, G., Jiang, X., Liu, X., and Zeng, X. (2020). Sequence Clustering in Bioinformatics: an Empirical Study. *Brief. Bioinform.* 21 (1), 1–10. doi:10.1093/bib/bby090

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Czech, Stamatakis, Dunthorn and Barbera. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 22 June 2022
doi: 10.3389/fbinf.2022.861505

Metascan: METabolic Analysis, SCreening and ANnotation of Metagenomes

*Geert Cremers, Mike S. M. Jetten, Huub J. M. Op den Camp and Sebastian Lücker**

Department of Microbiology, RIBES, Radboud University, Nijmegen, Netherlands

Large scale next generation metagenomic sequencing of complex environmental samples paves the way for detailed analysis of nutrient cycles in ecosystems. For such an analysis, large scale unequivocal annotation is a prerequisite, which however is increasingly hampered by growing databases and analysis time. Hereto, we created a hidden Markov model (HMM) database by clustering proteins according to their KEGG indexing. HMM profiles for key genes of specific metabolic pathways and nutrient cycles were organized in subsets to be able to analyze each important elemental cycle separately. An important motivation behind the clustered database was to enable a high degree of resolution for annotation, while decreasing database size and analysis time. Here, we present Metascan, a new tool that can fully annotate and analyze deeply sequenced samples with an average analysis time of 11 min per genome for a publicly available dataset containing 2,537 genomes, and 1.1 min per genome for nutrient cycle analysis of the same sample. Metascan easily detected general proteins like cytochromes and ferredoxins, and additional *pmoCAB* operons were identified that were overlooked in previous analyses. For a mock community, the BEACON (F1) score was 0.72–0.93 compared to the information in NCBI GenBank. In combination with the accompanying database, Metascan provides a fast and useful annotation and analysis tool, as demonstrated by our proof-of-principle analysis of a complex mock community metagenome.

Keywords: metagenomics, metabolism, annotation, microbiology, ecology

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Shaman Narayanasamy,
University of Luxembourg,
Luxembourg
Zhichao Zhou,
The University of Hong Kong, Hong
Kong SAR, China

****Correspondence:***
Sebastian Lücker
s.luecker@science.ru.nl

Specialty section:
This article was submitted to
Genomic Analysis,
a section of the journal
Frontiers in Bioinformatics

Received: *24 January 2022*
Accepted: *30 May 2022*
Published: *22 June 2022*

Citation:
Cremers G, Jetten MSM,
Op den Camp HJ and Lücker S (2022)
Metascan: METabolic Analysis,
SCreening and ANnotation
of Metagenomes.
Front. Bioinform. 2:861505.
doi: 10.3389/fbinf.2022.861505

INTRODUCTION

Alongside the advances in DNA sequencing, genome annotation has come a long way. Metagenomic sequence data are becoming available at increasing rates, making accurate and fast (automated) analysis tools even more important. Through the advancements of sequencing technologies, a single isolated bacterium prior to sequencing is not a requirement anymore, leading to an increase in the sequencing of metagenomes. This, in turn, leads to new challenges in annotation. It is common for metagenomes to be binned prior to annotation into metagenome-assembled genomes (MAGs). Especially when samples are (ultra-)deep sequenced, the number of MAGs per sample can reach thousands of near-complete genomes (Anantharaman et al., 2016). Not only do all these MAGs need to be annotated individually, which is time and effort consuming, there is also the greater ecological question of how the metabolic processes in the original sample relate to one another.

Additionally, there is the problem of protein ortho- and paralogs, which is especially prevalent when metagenomes lack enough sequencing depth for binning. Genes in a single genome are often distinct enough for a meaningful annotation, especially since for small genomes direct comparison like BLAST analysis (Altschul et al., 1990) to a database is still feasible. However, using BLAST on complex metagenomes is too computationally intense and time-consuming, and this will increase in the future, as databases keep growing every day (Evanko, 2009). Therefore, a faster, indirect comparison is preferred like the use of hidden Markov models (HMM), where annotation is based on matching amino acid patterns rather than whole gene or protein sequences. However, these patterns are very similar for ortho- and paralogs that have similar evolutionary origins (Jensen, 2001), which makes HMM databases with high resolution a necessity to achieve optimal annotations. Automated annotation is often dividing the process in single, specific functions like gene-calling, ribosomal RNA gene identification, and gene annotation. The results of the single analyses are subsequently combined in so called wrapper-scripts. For bacterial genomes, Prokka (Seemann, 2014) is probably the most well-known and fastest pipeline used at the moment. In recent years, scripts have been published that are able to annotate multiple genomes simultaneously, often by using well established databases like PFAM (Mistry et al., 2021), KOFAM (Aramaki et al., 2020), and TIGRFAM (Haft et al., 2013). Examples of these are METABOLIC (Zhou et al., 2019), DRAM (Shaffer et al., 2020), and eggNOG-mapper v2 (here-after eggNOG) (Cantalapiedra et al., 2021).

Here, we report on the construction of a new database by first clustering proteins for each KO number of the KEGG pathway database (Kanehisa and Goto, 2000) involved in central metabolic functions and subsequently building HMM profiles for each cluster. Key genes of major metabolic pathways were organized in pathway-specific individual databases (subsets), based on the grouping of Anantharaman et al. (2016). These databases together with a modified version of Prokka were then used for a gene-centric annotation and analysis of a mock community and previously published (meta-)genomes, either for all MAGs separately, or the unbinned assembly.

MATERIALS AND METHODS

Database Creation

For the creation of the database, all KO numbers from the KEGG database that are part of metabolic pathways ("09100 Metabolism"; https://www.genome.jp/brite/ko00001) were collected and linked to Uniprot entries through LINKDB (https://www.genome.jp/linkdb). For KO numbers with more than three entries, the entries were downloaded from the TrEMBL UniProt database (release 2018–09) (Bateman, 2019) and converted into multi-FASTA files. The sequences were filtered on length by calculating the average sequence length for each KO number, after which sequences longer than 150% and shorter than 60% of the average sequence length were discarded. If a set consisted of less than three sequences after length filtering, the unfiltered set was used.

For sequence de-replication, sets containing more than three entries were clustered (nearest neighbor) using Linclust from the MMSeq2.0 package (settings: -v 0 --kmer-per-seq 160 --min-seq-id 0.5 --similarity-type 1 --sub-mat blosum80. out --cluster-mode 2 --cov-mode 0 -c 0.7) (Steinegger and Söding, 2018). For each KO-number, clusters with less than three sequences were combined into 1 cluster. If less than three unique sequences were left after de-replication, the entire KO number was discarded. Subsequently, all resulting sequences for each KO number cluster were aligned individually using mafft v7 (settings: --quiet --anysymbol) (Katoh and Standley, 2013) and HMM profiles were created using hmmbuild (default settings) (Eddy, 2011).

Subsets with key genes for each metabolic pathway were created automatically based on KEGG classification ("09102 Energy metabolism") and manually curated where possible (**Supplementary Data S2**) based on the functional classification described in Anantharaman et al. (2016). HMM profiles for hydrogenases were created by downloading FASTA files for each hydrogenase group from the HydDB website (Søndergaard et al., 2016) followed by HMM profile creation as described above.

Metascan

Metascan expects a folder containing one or more DNA sequence files in FASTA format, where each file represents either an unbinned assembly (metagenome contigs) or a single MAG. When analyzing a complete unbinned metagenome, Metascan will generate an overview of all metabolic pathways and nutrient cycles. If the metagenome was binned, providing all MAGs allows annotation of each MAG. When using MAGs as input, the unbinned sequences (and, if applicable, small contigs discarded after size-filtering) are expected to be included as one or multiple separate bins, since a full gene-centric analysis of a metagenome is also dependent on the unbinned fraction of the microbial population that may exist in the sample.

Procedure

The core process starts with gene calling by Prodigal (Hyatt et al., 2010) (**Figure 1**). Per default, Metascan runs a few additional analyses that can be excluded if a fast overview of the nutrient cycles present in the ecosystem is desired. Before annotation, a ribosomal RNA gene search is performed by either Barnnap (https://github.com/tseemann/barrnap) or RNAmmer (Lagesen et al., 2007). The recovered rRNA gene sequences are compared against a local NCBI nr database using BLASTN (Sayers et al., 2019). Subsequent gene annotation is performed using hmmsearch (Eddy, 2011) against each of the seven subsets of the key genes representing important nutrient cycles [Nitrogen, Methane, Carbon fixation, Hydrogenases, C1 (methylotrophy) molecules, Sulfur, and Oxidative phosphorylation; **Table 2**] and one miscellaneous subset of metal cycling. After annotation of the key genes, the remaining open reading frames (ORFs) are annotated using the HMM profiles of the remaining metabolic genes. If the metagenome was previously binned and abundance was estimated, this data can be entered in a separate TSV file.

For a full annotation of MAGs, the option—prokka is available. This Prokka legacy option provides tRNA search Aragorn (Laslett and Canback, 2004), ncRNA scan Infernal

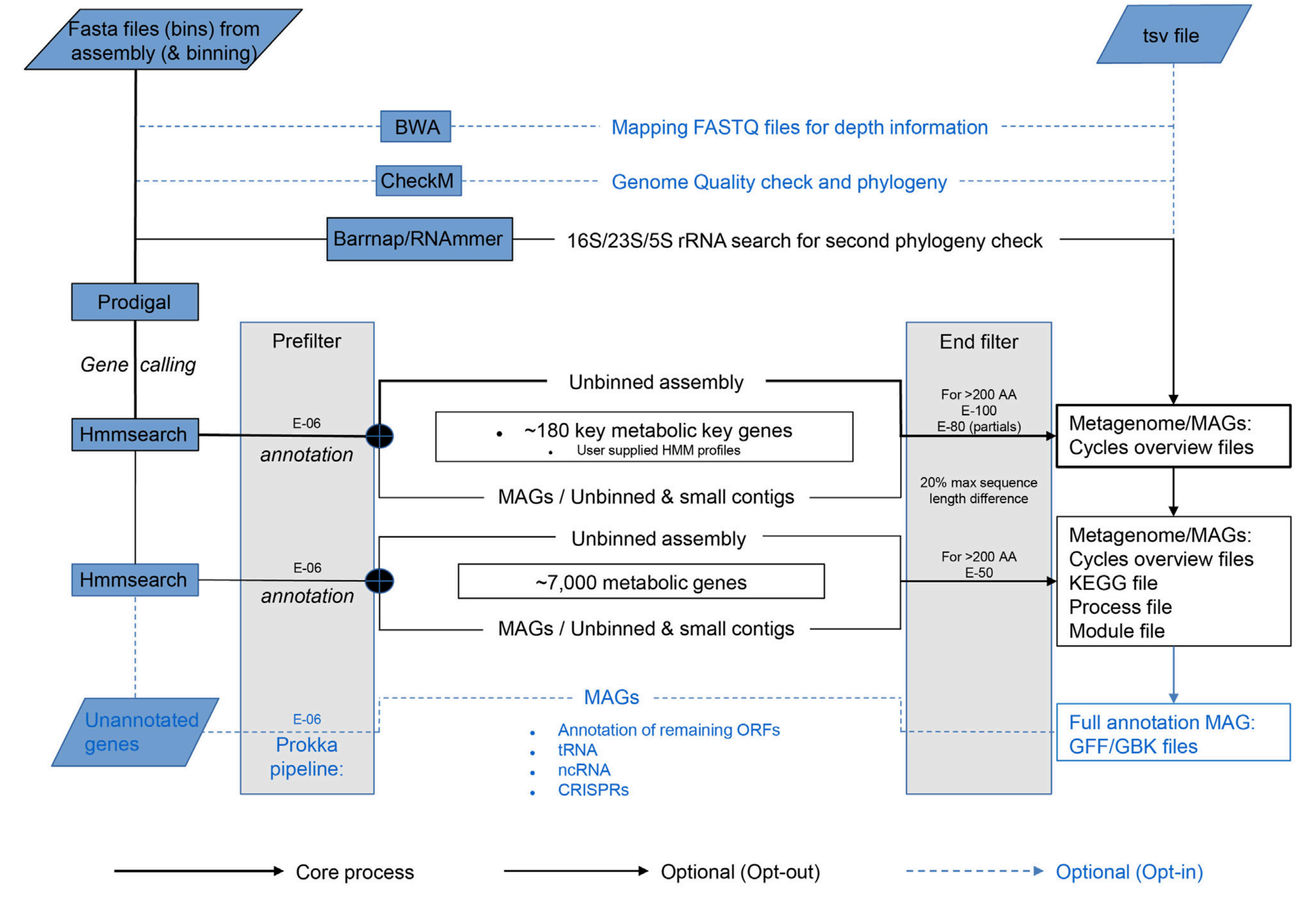

Fasta files (bins) from assembly (& binning)
tsv file
BWA
Mapping FASTQ files for depth information
CheckM
Genome Quality check and phylogeny
Barrnap/RNAmmer
16S/23S/5S rRNA search for second phylogeny check
Prodigal
Gene calling
Prefilter
End filter
Unbinned assembly
Hmmsearch
E-06
annotation
~180 key metabolic key genes
User supplied HMM profiles
For >200 AA
E-100
E-80 (partials)
Metagenome/MAGs: Cycles overview files
MAGs / Unbinned & small contigs
20% max sequence length difference
Unbinned assembly
Hmmsearch
E-06
annotation
~7,000 metabolic genes
For >200 AA
E-50
Metagenome/MAGs: Cycles overview files
KEGG file
Process file
Module file
MAGs / Unbinned & small contigs
MAGs
Unannotated genes
E-06
Prokka pipeline:
Annotation of remaining ORFs
tRNA
ncRNA
CRISPRs
Full annotation MAG: GFF/GBK files
Core process
Optional (Opt-out)
Optional (Opt-in)

FIGURE 1 | Schematic overview of the Metascan pipeline. The bold black lines represent the core analyses of the program. The thin black lines show the default Metascan pipeline when no options are opted out. The blue dotted lines indicate the options that can be invoked through the command line for full annotations.

(Nawrocki and Eddy, 2013), and CRISPR scan Minced (Bland et al., 2007), exactly as Prokka (Seemann, 2014) would. It also annotates the remaining unidentified ORFs using BLASTP and the Prokka internal database. These options are also available individually.

Bin Size

Metascan uses bin size in two different ways. First, for optimized gene calling, Prodigal has a single genome or metagenome mode. Thus, Metascan must determine whether the bin can be considered as single trustworthy MAG. Since the largest known bacterial genome is currently a little under 14.8 Mbp (Han et al., 2013), the maximum size for a bin to be considered a single prokaryotic genome is 15 Mbp. Anything larger is regarded as metagenomic by Metascan. Furthermore, for Prodigal the lower limit of the bin size is set at 0.5 Mbp, as this is the minimum Prodigal requires for gene-calling in single mode. Thus, bins smaller than 0.5 Mbp and larger than 15 Mbp are processed in meta mode. Prodigal in Metascan is also set to predict partial genes at the ends of contigs, as these are expected to be abundant in a metagenome. Secondly, the maximum bin size is also used to limit runtime by preventing time-consuming analyses like tRNA, ncRNA, CRISPR, and BLAST searches against small and unbinned contigs, as well as the unbinned metagenome.

E-values

Like bin size, the e-value settings are important for the final outcome. Three different e-values are implemented in the Metascan workflow (**Figure 1**). The first and lowest e-value serves as a prefilter for HMM results to reduce the amount of working data. Here, E-06 is the highest score corresponding to the lowest protein identity allowed by Metascan, and this e-value is also used by all other first-level analyses. Next, Metascan differentiates between the application of the full metabolic dataset or the key gene set only. If only the smaller key genes databases are applied, the stringency is set to a more stringent setting of E-100 to exclude large numbers of false positives. When including the larger metabolic database, the stringency is lowered to E-50 because the risk for false positives is reduced by the probable presence of genes with higher similarity in the database, and a lower e-value here is useful to avoid false negatives. Simultaneously, the program applies a filter on size difference of ≥20% (by default) between target (as calculated by Hmmbuild during HMM construction) and query sequence to remove hits that clearly differ in size, but which contain similar sequence motifs. After all databases are queried, the hit with the highest bit-score is selected for each ORF.

For small proteins (<200 amino acids), the only e-value considered is the prefilter. Short sequences are not long enough to build up enough bit-score, resulting in large e-values even when the similarity is high. Since this will also include incomplete partial genes missing a start or stop codon, the hits are selected on size difference between target and query of maximum 30%. If desired, all target e-values can be set manually. As a final option, the program also accepts user generated HMM profiles, both as single input or in combination with the existing databases.

Output

For each analysis, an overview file is produced that contains the number of hits for each gene of each nutrient cycle and the number of bins/MAGs harboring these genes, alongside their relative abundance (**Supplementary File S1.1**). For both genes and organisms, the absolute and relative coverage is provided, if applicable. A Krona (Ondov et al., 2011) HTML file is produced for visual reference. A TSV file is generated containing all protein hits for easy retrieval of proteins of interest. Finally, two more TSV files are created, containing the genes for each process and metabolic module, as used by KEGG mapper (Kanehisa and Sato, 2020) on the genome.jp website. The process file can be used to manually create a cycle diagram using the provided blank cycle diagram (**Supplementary File S1.2**).

For each bin, an overview file is produced with the number of hits for each gene and phylogenetic information if applicable. A file containing hits of all the detected KEGG numbers is created, which can be entered into KEGG mapper for further analysis. Two files containing hits against the database and statistics like bitscore and output from hmmsearch are retained as well. One file contains all possible hits, the other file is an overview of all the highest scoring hits. Furthermore a few additional files are created, including a file containing all ribosomal RNA genes and a tab-separated file with annotated genes for easy retrieval. Finally, a few FASTA, statistical, log and GenBank files are created, similar to standard Prokka output (**Supplementary File S1.1**).

Validation

Mock Community

For eight different microorganisms, representing different metabolic traits, the genomes (**Table 1**) were downloaded and fully annotated using Metascan four ways: as separate genome bins or as a single simulated metagenome and using either only key-genes or the whole metabolic set (**Table 2**). The mock metagenome was simulated using CAMISIM (Fritz et al., 2019) on all eight genomes (default settings). Both the simulated metagenome and the eight genomes were also analyzed using METABOLIC (default settings), DRAM (default settings), Prokka and eggNOG (hmmer method and default settings).

To obtain an accurate list of key genes present in these genomes, each KO number in the metabolic core dataset was cross-referenced with the KO numbers present in KEGG for those organisms. For unclear or missing results, additional BLAST checks and manual searches in the NCBI GenBank files were performed. Since no golden standards exist for the used organisms, the GenBank files generated by Metascan, Prokka (default settings) and eggNOG were compared to the GenBank files from NCBI using BEACON (Kalkatawi et al., 2015) with an offset of 2%. METABOLIC did not create files that could be converted into Genbank files. DRAM created Genbank files, but no annotation was present. Therefore, both programs could not be included in the BEACON comparison.

The BEACON scores were found to be identical to F1 scores (Van Rijsbergen, 1977) and we consequently report the BEACON scores as F1 scores for the comparison of the different annotations (**Supplementary Data S3**).

Metagenome Analysis

2537 MAGs and the accompanying coverage data from the study by Anantharaman et al. (2016) were downloaded from ggKbase (https://ggkbase.berkeley.edu/2500-curated-genomes/organisms/). The key gene as well as full metabolic analyses were performed on the binned and unbinned genomes (**Table 2**). Both the binned and unbinned datasets were furthermore analyzed using METABOLIC (default settings) and DRAM (default settings). EggNOG accepted only a single FASTA file, and thus only the unbinned dataset was analyzed. The results of the Metascan analyses and the original study were manually compared by analyzing the statistics for the various nutrient cycles.

Computing Platform

All analyses were performed using 12 cores except for DRAM (10) on a server with one 32 core Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60 GHz and 227 G RAM.

Code and Data Availability Statement

Metascan can be obtained from https://github.com/gcremers/metascan, the required databases from Zenodo.org (https://doi.org/10.5281/zenodo.6365663).

RESULTS

Database Creation

For the creation of the HMM database, 7,788 unique KO numbers associated with metabolic pathways were identified from file ko00000. keg (7 May 2018; renamed in KEGG to ko00001. keg in recent versions). When connecting these to proteins deposited in UniProt, 876 KO numbers had less than 3 UniProt entries available and were therefore excluded. Sequences from the remaining 6,912 KO numbers were downloaded from the UniProtKB/TrEMBL database, converted to FASTA format, and subjected to dereplication and length filtering (60%–150% of the mean length for each set). After dereplication, 46 sequence sets were discarded because a limited amount (<3) of unique sequences was left for alignment. Five unfiltered sets were retained as the length

TABLE 1 | Genomes used in the mock community of this study.

Organism	Size (bp)	Topology	Accession number	Metabolism
Methanosarcina acetivorans str. C2A	5,751,492	Circular	AE010299	Methanogen
Nitrosomonas eutropha C91	2,781,824	Circular + plasmids	CP000450	Autotrophic ammonia-oxidizer
Paracoccus denitrificans PD1222	5,236,194	Circular + plasmids	CP000489	Denitrifier and methylotroph
Escherichia coli str. K-12 substr. MG1655	4,641,652	Circular	NC_000913	Heterotroph
Candidatus Methylomirabilis oxyfera	2,752,854	Circular	FP565575	Denitrifying methanotroph
Nitrospira moscoviensis strain NSP M-1	4,589,485	Circular	NZ_CP011801	Autotrophic nitrite-oxidizer
Methylacidiphilum fumariolicum SolV	2,476,671	Circular	NZ_LM997411	Nitrogen fixing methanotroph
Candidatus Kuenenia stuttgartiensis MBR1	4,406,153	Circular	NZ_LT934425	Anammox

TABLE 2 | Overview of different analysis options, analysis times and properties per dataset. Time is the total analysis time. Pathways indicate whether the results are ordered by ecological pathways and processes in the output. Abundance shows the option to include depth values into the analysis and GBK indicates the state of the Genbank file that is created by the program.

Dataset		Metascan key genes	Metascan Full annotation	DRAM	eggNOG	METABOLIC	Prokka
8 genomes	Time	01 h 01	4 h 46	3 h 26	2 days 7 h 02	0 h 39	0 h 16
	Pathways	Yes	Yes	Individual	no	Individual	No
	Abundance	NA	NA	NA	NA	NA	NA
	GBK	full	full	No genes	No RNA	No genes	full
Simulated meta-genome	Time	1 h 08	2 h 54	1 h 06	2 days 09 h 18	0 h 50	0 h 10
	Pathways	yes	yes	yes	no	yes	No
	Abundance	NA	NA	NA	NA	NA	NA
	GBK	limited	limited	No genes	No RNA	No genes	full
2,537 genomes	Time	2 days 22 h 29	19 days 08 h 21	34 days 13 h 2	NP	3 days 11 h 01	NP
	Pathways	Yes	Yes	Individual		Individual	
	Abundance	Yes	Yes	No		no	
	GBK	full	full	No genes		No genes	
Unbinned meta-genome	Time	1 day 23 h 06	12 days 09 h 57	36 days 23 h 42[a]	Over 44 days[b]	3 days 17 h 28	NP
	Pathways	Yes	Yes	Yes		yes	
	Abundance	NA	NA	NA		NA	
	GBK	limited	limited	No genes		No genes	

NP, not performed; NA, not applicable.

[a] *Program crashed and was manually resumed, missing one step in the process.*

[b] *The program run for over 44 days and was manually stopped.*

TABLE 3 | Number of genes per subset (cycle) and the number of corresponding HMM profiles.

#KO	Cycles	#HMM profiles
38	Hydrogenases	38
25	C1 molecules	319
34	Carbon fixation	643
12	Methane	32
14	Miscellaneous	213
38	Nitrogen	557
14	Oxygen	556
40	Sulfur	650
6,739	Non-key genes	114,157
6,916	Total	**117,127**

filtering step would have dropped the available sequences below three. In total, this left a final of 6,866 KO numbers available for alignment and HMM building.

After manually adding missing entries, subsets for each nutrient cycle were manually created (**Table 3**). For each key gene in a nutrient cycle, entries were manually checked and completed for lesser studied genes like hydrazine synthase. Finally, 38 profiles were calculated for hydrogenases by aligning sequences taken from HydDB (Søndergaard et al., 2016) for each (sub-)category.

Mock Community

We used DRAM, METABOLIC, eggNOG, and Prokka to analyze the original eight genomes and the CAMISIM simulated metagenome. We also used Metascan to analyze the eight genomes of the mock community using four different input and analysis settings (**Table 2**). Analysis times ranged from 16 min for all eight genomes (Prokka) to 2 days and 7 h for eggNOG; for the simulated metagenome this was from 10 min (Prokka) versus 1 day and 10 h for eggNOG. Metascan and Prokka both provided full GenBank files for further analysis, whereas eggNOG provided a GenBank files without RNAs. DRAM and METABOLIC did not include the annotation within the GFF file, which meant a meaningful GenBank file could not be constructed.

TABLE 4 | Number of genes retrieved from the GenBank files of the mock community and four different Metascan analyses, ordered by cycle. Percentages state the percentage relative to the total number of genes recovered from the GenBank files.

	Number of genes				
Nutrient cycle	**GBK**	**Unbinned, key genes**	**Binned, key genes**	**Unbinned, full**	**Binned, full**
Sulfur	65	70	77	67	71
Hydrogen	15	19	22	10	*12*
Methane	25	24	24	25	25
Nitrogen	117	108	117	114	127
Oxidative phosphorylation	53	55	60	54	59
C1	68	95	108	72	79
Carbon fixation	85	123	154	87	118
Miscellaneous	19	26	31	18	18
All	**447**	**520 (116.3%)**	**593 (132.7%)**	**447 (100.0%)**	**509 (113.9%)**

GBK: GenBank file from NCBI, Key genes: Analysis using only the key genes as reference. Full: Analysis using all metabolic genes as reference. Unbinned: simulated metagenome generated by CAMISIM., Binned: separate genomes from NCBI.

Runtimes

When testing the mock community, we first needed to identify all genes belonging to the different nutrient cycle within the NCBI entries for each microorganism. This proved not to be straightforward, since in GenBank the annotations are not stored with these cycles in mind. We thus created the individual nutrient cycling profiles of the reference organisms by manually mining KO numbers from their annotations in KEGG and GenBank for metabolic key genes and compared these to the Metascan output. For a complete annotation of all eight genome bins including all ~7,000 metabolic genes, the analysis took 4 h and 46 min, with an average of 35.6 min per genome bin. On the same system, it took 2 h and 54 min for the simulated metagenome, with the exclusion of several steps (tRNA, ncRNA, CRISPR detection, and BLASTP) in the process due to bin size. The key genes only analyses took 68 min for the simulated metagenome and 1 h and 1 min for the binned genomes.

Gene-Centric Annotation

The manual key genes mining of the mock community against NCBI and KEGG yielded a total of 447 key genes for all eight genomes, with the Nitrogen cycle being the most abundant (117 genes) and enzymes involved in hydrogen metabolism the least (nine genes; **Table 4**).

Overall, the total amount of key genes recovered from the mock community by Metascan varied from 133% (binned and key genes only) to 100% (simulated and all metabolic genes) compared to the GenBank annotations. Among the cycles, Hydrogen (67%–147%), C1 (methylotrophy; 106%–159%), Carbon fixation (102%–181%), and Miscellaneous (95%–163%) have the largest variability, whereas Sulfur (103%–119%), Methane (96%–100%), Nitrogen (92%–109%), and Oxidative phosphorylation (102%–113%) showed better congruency with the GenBank annotation. As could be expected, the analyses that used all metabolic genes from the KEGG dataset are more comparable to the GenBank annotations than the analyses using only key genes. Binning the mock metagenome into genome bins did not influence these results much.

When looking into the data in more detail (**Supplementary Data S4**), it became apparent that the majority of differences was caused by a few specific types of proteins, mainly ferredoxins, and cytochromes. *Cbb$_3$*-type cytochrome *c* oxidase subunit III (K00406) was found 5 and 14 times by Metascan in the simulated metagenome full metabolic and binned key genes-only analyses, respectively, vs. three times in the GenBank annotations. A similar pattern was observed for the cytochrome b_{556}-containing formate dehydrogenase subunit gamma (FdoI, K00127; 17 and 6 vs. 5), the Fe-S subunits of anaerobic carbon-monoxide dehydrogenase (CooF, K00196; 30 and 11 vs. 2) and arsenate oxidase (AoxA, K08355; 9 and 0 vs. 0). Another example is the Fe-S-containing beta subunit of formate dehydrogenase (FdoH and K00124), where both binned (19) and simulated metagenome key genes-only (15) Metascan analyses yielded a surplus of positive hits. However, BLASTP analysis of these proteins against the NCBI database identified 13 of them as NADH-quinone oxidoreductase subunit NuoF. Manual inspection of the input data (K00124) used to generate the FdoH HMM profiles (**Supplementary File S1.3**) showed that several entries in these protein clusters are labeled as NuoF, indicating either misannotated entries or unspecificity within this database entry.

Another group of gene annotations that deviated from the GenBank entries entailed group 4 Ni-Fe hydrogenases. Here, in the key genes-only annotation Metascan found seven proteins in addition to those predicted in NCBI. However, all seven proteins were apparently corresponding to NuoC or NuoD subunits of NADH dehydrogenase complexes and not true hydrogenases, as they also were lacking the catalytic Ni-binding motif, despite e-values of 0.0 to 9E-161 in the HydDB database search.

Genome-Centric Annotation of Metagenome-Assembled Genomes

Besides the broad metabolic overview that Metascan provides on the metagenome level, an additional useful feature is the possibility for parallel single genome annotations during the analysis, which allows for immediate downstream analysis of genomic potential for any given MAG. For comparison of single

TABLE 5 | BEACON (F1) scores comparisons of the GenBank files created by Prokka, Metascan, eggNOG, and NCBI for all eight genomes.

	NCBI[a]			Metascan[b]		
Genbank	**Metascan**	**eggNOG**	**Prokka**	**NCBI**	**eggNOG**	**Prokka**
E. coli	0.91	0.90	0.91	0.91	0.99	1
M. fumariolicum SolV	0.84	0.83	0.84	0.84	0.99	0.99
Candidatus K. stuttgartiensis	0.80	0.79	0.80	0.80	0.99	1
N. eutropha	0.83	0.82	0.83	0.83	0.99	0.99
Candidatus M. oxyfera	0.81	0.80	0.81	0.81	0.99	1
M. acetivorans	0.72	0.73	0.72	0.72	0.99	0.99
N. moscoviensis	0.80	0.79	0.80	0.80	0.99	0.99
P. denitrificans	0.87	0.47	0.87	0.87	0.55	1

[a] *F1 score compared to the Genbank files from NCBI.*
[b] *F1 scores compared to the Metascan annotation.*

TABLE 6 | Direct and detailed comparison of the GenBank files from NCBI and Metascan. The differences in the grey area are related to the NCBI reference.

Gene calls	M. a	N. e	P. d	E. c	cM. o	N. m	M. f	cK. s
Detected identical	2,960	2,193	4,324	3,988	2,294	3,400	1,875	3,089
Detected similar	472	75	196	97	106	213	73	177
Unique to NCBI	1,118	379	653	452	742	896	400	833
Unique to Metascan	1,514	490	656	330	361	933	348	825
ΔrRNA	−1	0	0	0	0	−1	0	−1
ΔtRNA	57	0	2	2	0	2	1	0
ΔncRNA	0	−3	−2	−72	0	−2	−2	−3
Δframeshift/Pseudo	0	−343	−213	−86	−2	−109	−151	−281
ΔFunctional genes	−1,409	−559	−998	−1,052	−1,438	−648	−379	−797
Total Reference	4,550	2,687	5,173	4,537	3,142	4,509	2,348	4,099
Total Metascan	4,946	2,758	5,176	4,415	2,757	4,546	2,296	4,091

M.a = M. acetivorans, *N. e* = N. eutropha, *P.d* = P. denitrificans, *E. c* = E. coli, *cM.o* = "Candidatus *M. oxyera*", *N.m* = N. moscoviensis, *M. f* = M. fumariolicum SolV, *cK.s* = "Candidatus *K. stuttgartiensis*". *Δ+, Metascan annotated more genes; Δ−, metascan annotated less.*

genome annotations, we used BEACON to compare the annotations produced by Prokka, Metascan, and eggNOG for each genome used in the mock community to the GenBank files from NCBI with an offset of 2% (**Table 5**, **Supplementary Data S5**). BEACON (F1) scores range from 0.90–0.91 for *E. coli* to 0.72–0.73 *M. acetivorans*. The results are very similar for all three methods for all organisms, except for the eggNOG annotation of *P. denitrificans* (0.47), which strongly deviated from Prokka and Metascan (0.87). When comparing Metascan to the different approaches, eggNOG, and Prokka F1 scores range from 0.99 to 1, except for *P. denitrificans* (eggNOG, 0.55). The similarity scores to the NCBI annotations again range from 0.72 (*M. acetivorans*) to 0.91 (*E. coli*). These results show that Metascan, eggNOG, and Prokka annotations are very similar to each other and that all three equally differ from the NCBI GenBank files.

In-Depth Comparison Metascan vs. NCBI

Compared to Metascan annotations, the number of genes with function annotations in NCBI GenBank was higher for all samples (**Table 6**). This was caused by the higher number of (conserved) hypothetical proteins in the Metascan/Prokka annotations, as these programs use a conservative annotation regime. Annotations containing words like "conserved" and "containing" are labeled hypothetical, as there is no definitive known function for these proteins. As a result, there are more hypothetical proteins in the Metascan annotations and thus a lower degree of genes with assigned apparent functions.

For two organisms there was a larger difference in the amount of ORFs called by GenBank compared to the other two methods. The first was *M. acetivorans*, for which 4550 ORFs were predicted by GenBank and 4,946 by Metascan, which is a difference of 8% (396 ORFs). However, visualizing the ORFs of *M. acetivorans* in Artemis (Carver et al., 2012) (**Supplementary File S1.4**) indicated the presence of amber stop-codons (TAG) within several genes in the NCBI GenBank annotation. The substitution of a TAG stop codon by a sense codon is a codon usage variation which has been described in some microorganisms and ciliates (Tourancheau et al., 1995). As a matter of fact, the usage of the unusual amino acid pyrrolysine has first been described in a paper by Heinemann et al. (2009). When re-analyzing the genome with Metascan using a translation table that does not use TAG as stop codon like table 25, a more intuitive layout of the ORFs appeared, as well as a gene count that is closer to the GenBank file (4,631). BLASTx analysis of a few of these ORFs against the NCBI *nr* database showed that they had full length hits against database entries, which had either amino acid X or O (pyrrolysine) at the position of the stop codon in the query sequence (**Supplementary File S1.4**).

Contrastingly, in the annotation of *M. oxyfera* Metascan predicted 2757 ORFs, which are 385 less than in the GenBank file (3,142; 13% difference). When comparing the two analyses through Artemis, it becomes apparent that the NCBI GenBank file contains more small proteins (<200 amino acids) than the Metascan GenBank file. The reason for this could be the threshold setting (1E-06) for small proteins to be considered a true protein within Metascan.

Noteworthy are the 57 tRNAs in *M. acetivorans* found by Metascan that were not present in the GenBank entry. This exemplifies that also GenBank files are far from perfect, as was discussed before (De Simone et al., 2020). However, Metascan had difficulties in identifying pseudo-genes (up to 343 genes in *Nitrosomonas eutropha*) and ncRNAs (up to 72 in *E. coli*).

Metagenome Analysis

For a metagenome analysis, 2,537 genomes from a large-scale metagenomic study of aquifer sediments (Anantharaman et al., 2016) were downloaded from ggKbase (https://ggkbase-help.berkeley.edu) together with a pre-parsed file containing the average coverage depth for each bin. The per-genome key gene analysis for all 2,537 genomes in this dataset took almost three full days to complete, with an average of 1.7 min per genome. In the full analysis using all metabolic genes, it took the script about 19.5 days, corresponding to an average of 11 min per genome. The key gene analysis of the unbinned metagenome (i.e., the combined bins) was finished in just under 2 days, which would equate to 1.1 min per genome (**Table 2**).

Similar to the mock community analyses, the formate dehydrogenase iron-sulfur-containing beta (K00124; 369, 1018, 973, and 951 hits in the Full Annotation (FA), Binned Key gene (BK), Unbinned Key gene (UK), and Full Unbinned (FU) analyses, respectively) and gamma subunits (K00127; 126 FA, 1413 BK, 1381 UK, and 454 FU), and the anaerobic carbon-monoxide dehydrogenase iron-sulfur subunit CooF (K00196; 654 FA, 1862 BK, 833 UK, and 766 FU) showed clear differences in gene counts. Furthermore, malyl-CoA ligase frequencies were overestimated in the key gene analyses. BLAST analysis of these indicated that the misannotated genes were actually succinyl-CoA ligases, a gene not included in the key gene set but present in the large metabolic set.

Metascan vs. Reference

A direct comparison between the analyses from Anantharaman et al. (2016) and Metascan is hampered by different choices made during analyses, like which genes to include in the key gene set and how to define the nutrient cycles. However, a few things became apparent (**Table 7**; **Supplementary File S1.5**). For instance, when focusing on methylotrophy Metascan identified 82 enzymes related to the pyrroloquinoline quinone (PQQ)-dependent methanol dehydrogenases (MDH) in the binned key gene analysis, which were not reported in the original analysis. After curating the retrieved set for (nearly) full length genes, a tree was constructed (Felsenstein, 1985; Saitou and Nei, 1987; Jones et al., 1992; Kumar et al., 2016), revealing that most of these proteins are PQQ-dependent alcohol dehydrogenases from largely uncharacterized lineages within this protein family (**Supplementary File S1.6**). Anantharaman et al. (2016) found one organism (Burkholderiales bacterium RIFCSPLOWO2_12_67_14) putatively involved in methane oxidation, based on the presence of the genes encoding the particulate methane monooxygenase (*pmoCAB*). In the key genes-only analysis, Metascan found five *pmoB,* and one *pmoC* gene hits that could also be confirmed using BLAST. In the full metabolic annotation, Metascan found additional six *pmoA* and five *pmoC* genes. In total, these genes were divided over four species from the order Burkholderiales. Thus, besides the earlier mentioned species, the dataset contained three previously unrecognized Burkholderiales bacteria encoding particulate methane monooxygenase. From those three, two MAGs contained two complete *pmoCAB* operons and one was predicted to only encode *pmoA* and *pmoC.* However, a BLAST search on the gene directly downstream revealed that *pmoCA* is followed by an unrecognized *pmoB* in this organism as well. Based on the coverage of the four species containing the *pmoCAB* genes, methanotrophy is found in ca. 0.6% of the entire sample and 0.16% of the total number of organisms, and methylotrophy constitutes 0.82% and 0.84%, respectively. Correspondingly, malyl-CoA lyase (*mcl*), a marker gene for the serine pathway in methanotrophy and methylotrophy, had a total abundance of 1.7% and was detected in 0.1% of all organisms. While these findings expand the number of putative methane oxidizers present, it still indicates that methane oxidation is of minor importance in this aquifer ecosystem.

On the contrary, a process in the nitrogen cycle that appears to be over-predicted by Metascan is nitrate reduction to ammonium (both assimilatory and dissimilatory), which is mainly caused by large numbers of misannotated small subunits of the two main enzyme systems catalyzing nitrite reduction (*nirD* and *nrfH*). BLAST analyses showed that besides true *nirD* these genes encode diverse ferredoxins, Rieske 2Fe-2S proteins and dioxygenases.

Metascan vs. METABOLIC and DRAM

The eggNOG analysis ran for over 44 days and was expected to run for over a year at 5 h per genome, therefore the analysis was not included into the metagenome analysis in this paper. METABOLIC and DRAM reported the results as lists of identified genes per genome and did not provide a combined overview of all analyzed genomes. However, for DRAM an overview could be created from the available data. The binned analysis took 31 days and 13 h, 12 days longer than Metascan. The unbinned analysis ran for 36 days and 23 h, after which it crashed due to memory issues during the creation of the GFF files. Nevertheless, the distillation of the annotation was possible with the annotation files that were produced so far. Strikingly, both DRAM analyses were nearly identical and can thus be reported as one (unbinned; **Supplementary Data S6**). In METABOLIC, the binned analysis ran for 3 days and 17 h, the unbinned analysis for 3 days and 11 h. As METABOLIC did not provide a full overview of the combined genomes only the unbinned dataset was used for comparison. Both METABOLIC and DRAM reported the results in KEGG numbers, which were used for making the comparisons.

Table 8 summarizes the annotation results, reporting the maximum number any single protein assigned to the respective process was detected, or the sum of all detected hydrogenases in the case of hydrogen metabolism. Overall, annotations are

TABLE 7 | |Results from the Anantharaman et al. (2016) study and Metascan binned key gene analysis. Groundwater and sediment sample annotations were taken are from Anantharaman et al. (2016).

	Groundwater		Sediment		Metascan	
	N# org	**%O-Depth[a]**	**N# org**	**%O-Depth[a]**	**N# org**	**%O-Depth[a]**
Carbon Cycle						
Carbon fixation	186	12	186	30	1022	38
Methanogenesis	0	0	0	0	0	0
Methanotrophy	0	0	0	0	5	<1
Methylotrophy	NA	<1	NA	<1	51	3
Hydrogen oxidation	356	22	356	45	400	14
Sulfur Cycle						
Sulfate reduction	21	<1	21	2	165	9
Sulfite reduction	21	<1	21	<1	724	32
Thiosulfate oxidation	77	7	77	9	199	10
Thiosulfate reduction	53	2	53	6	361	17
sulfite oxidation	51	3	51	8	83	6
sulfide oxidation	208	17	208	29	371	18
sulfur oxidation	157	13	157	14	2	<1
sulfur reduction	223	16	223	23	194	12
Nitrogen cycle						
Nitrogen fixation	54	3	54	1	87	5
Anammox	11	2	11	1	22	<1
ammonia oxidation	0	0	0	0	14	<1
Nitrite oxidation	85	8	85	15	265[a]	14[a]
DNRA	108	12	108	13	499[b]	22[b]
Denitrification						
Nitrate reduction	212	15	212	18	265[a]	14[a]
Nitrite reduction	150	23	150	21	159	7
Nitric oxide reduction	109	6	109	11	168	10
Nitrous oxide reduction	56	3	56	4	98	6

[a] *%O-depth is the percentage of the organisms that can perform the process in absolute numbers (depth). For instance, 12% of every single bacteria/archaea can perform Carbon Fixation in Groundwater.*

[b] *The HMMs, in Metascan cannot distinguish between nitrate reductases and nitrite oxidoreductases.*

[c] *These are the numbers for the small subunit NirD. Large subunit NirB has N# 151 and 10% O-depth.*

similar for all three methods, with a few exceptions. Most notably, DRAM did neither detect any methanol dehydrogenases, nor anaerobic ammonium oxidation (anammox). The high number for MDHs in the other methods is likely an overestimation, which was confirmed by BLAST analysis that indicated that the number of MDHs is more in line with the predicted methanotrophy genes (15–17 MDHs). DRAM also did not report several sulfur cycling processes. For thiosulfate oxidation, METABOLIC detects the largest number of genes (320 vs. ∼130 in Metascan and DRAM), but the lowest for thiosulfate reduction (133). Here, Metascan reports much higher numbers (684 and 1,372), followed by DRAM (234). However, these numbers especially for Metascan appear to be an overestimation, as they are only based on the detection of PhsA. In contrast, PhsB was detected 385 and 226 times by Metascan, and PhsC even only 0 and 25 times, However, none of these genes was included in METABOLIC or DRAM, hampering a comparison between methods.

Finally, when comparing the two Metascan analyses with each other, it becomes apparent that the number of genes predicted in the full annotation is higher for almost all cycles, likely due to the higher e-value (E-50 vs. E-100) used in the full annotation.

DISCUSSION

Database Construction

In this study, we present Metascan, a new tool for analysis of the metabolic potential of complex microbial communities. We developed this tool to enable researchers to obtain a fast but detailed and reliable overview of the main nutrient cycle reactions encoded by complex microbial communities in large environmental metagenomic datasets. This functionality currently is lacking in most annotation tools, which mainly focus on genome-centric analyses and rarely structure their output to give an overview of the biogeochemical nutrient cycles being catalyzed in the investigated environment. Moreover, the currently available databases used for similarity search-based annotations are too large to allow fast annotations of complete metagenomes, too unstructured to yield an overview of the nutrient cycles taking place, or, in the case of well-curated databases, also too small to offer the required resolution especially for environmental communities rich in uncultured and understudied microorganisms. We thus constructed a novel HMM-based database that not only allowed fast and accurate gene- or genome-centric annotation of complex metagenomes, but also categorized the identified protein-coding genes according to the relevant nutrient cycles.

TABLE 8 | Results from Metascan (unbinned), Metabolic (unbinned), and DRAM (unbinned) analyses of the Anantharaman metagenome (2016).

	Metascan		METABOLIC	Dram
	key	full		
Carbon Cycle	#hits[a]	#hits[a]	#hits[a]	#hits[a]
Carbon fixation	1578	2776	1707	1686
Methanogenesis	0	0	0	0
Methanotrophy	6	8	5	6
Methylotrophy	99[b]	294[b]	66	0
Hydrogen formation[c]	557	545	471	
Hydrogen oxidation[d]	1370	2596	537	2008[e]
Sulfur Cycle				
Sulfate reduction	193	480[f]	127	124
Sulfite reduction	449	718	378	388
Thiosulfate oxidation	133	195	320	124
Thiosulfate reduction	684[g]	1372[g]	133	234
sulfite oxidation	152	317	45	
sulfide oxidation	491	877	587	
sulfur oxidation	2	3	2	
sulfur reduction	451	681	276	
Nitrogen cycle				
Nitrogen fixation	103	208	102	87
Anammox	53	90	60	0
Ammonia oxidation	6	8	6	6
Nitrite oxidation	294	537	162	198
DNRA	670	578	290	198
Denitrification				
Nitrate reduction	294	537	148	198
Nitrite reduction	168	358	201	195
Nitric oxide reduction	181	303	340	194
Nitrous oxide reduction	98	39	96	96

[a]*Reporting the maximum number any single protein assigned to the respective process was detected.*

[b]*Combined XoxF, MxaF (both EC:1.1.2.7) and NDMA-dependent MDH (EC:1.1.99.37).*

[c]*Sum of all Fe-Fe hydrogenases.*

[d]*Sum of all Ni-Fe hydrogenases.*

[e]*Sum of all hydrogenases detected, as there is no distinction between Ni-Fe and Fe-Fe hydrogenases in DRAM.*

[f]*Inflated by AsrB, otherwise 338.*

[g]*Inflated by PhsA, otherwise 385 and 226, respectively, based on PhsB detection.*

Metagenome Analysis

A direct comparison of different annotation tools is hampered by the choices made during the analysis and the reporting of the results. Genes with multiple subunits can be reported as present when all, some or just one subunit is present. Some processes are part of two pathways (e.g., carbon fixation in methanol metabolism), and some cycles are represented by multiple pathways (carbon fixation). Obviously, different choices have a direct impact on the results. For instance, some protein complexes with multiple subunits like the anaerobic sulfate reductase (ASR) consist of subunits rarely detected in the Metascan full annotation (AsrA and ArsC, both detected five times) and others that are likely overpredicted (AsrB, detected 480 times).

In general, Metascan reached a similar level of precision as the GenBank reference annotation, although it tended to overpredict certain functions. This was especially prevalent for annotations of cytochromes and ferredoxins, which are very common proteins in nature and participate in a wide range of metabolic reactions, not seldom with overlap and interchangeability in function. To this extent, while both cytochromes and ferredoxins contain conserved domains that can easily be recognized through bioinformatics, a large set of well annotated reference proteins is required to ensure their exact annotation. However, this level of resolution is not present in most databases, and many automatically annotated genomes contain mis-annotated genes or lack proper annotation altogether. These errors then are propagated through different databases, consequently leading to a reduced reliability of annotation also in conventional tools (Schnoes et al., 2009). An example of this is K00124 (FdoH) in the UniProtKB/TrEMBL database, where either 1) the UniProtKB/TrEMBL dataset is heavily misannotated and many true NuoF are wrongly categorized under K00124, 2) the protein entries identified by BLASTP in the GenBank database are wrongly annotated as NuoF and in fact are true FdoHs, which then would also indicate that in the GenBank files from the mock community NuoFs are underrepresented, or 3) these subunits belong to distinct protein complexes participating in different pathways but are too similar to be distinguished by HMM searches. While the last option seems plausible here, since NuoF and FdoH both are Fe-S proteins with a common evolutionary history (Oh and Bowien, 1998), this issue mainly appears to be caused by the propagation of misannotations in the public databases (Schnoes et al., 2009), as especially many formate dehydrogenase beta subunit genes appear to be deposited as NuoF in GenBank. Similarly, for the overestimated carbon monoxide dehydrogenase iron-sulfur subunit CooF (K00196), the raw data gathered from UniProtKB/TrEMBL contained mostly unnamed ferredoxins, which corresponds to a large part of the obtained false positives in our analyses.

Another factor hampering correct functional annotation can be overlapping functionality of enzymes. For instance, malyl- and succinyl-CoA ligases react with two structurally quite similar substrates, as both malate and succinate are small four-carbon dicarboxylic acids. Since both proteins furthermore catalyze the same type of reaction, they are structurally very similar with respect to their conserved regions, which is also reflected in the fact that succinyl-CoA ligase is able to use malate as alternative substrate (Nolte et al., 2014). Consequently, when using a small database as in our key genes-only analysis that contained only the malyl-CoA ligase, E-values for hits against succinyl-CoA ligases are small enough to be considered significant, leading to the observed overestimation of malyl-CoA ligases. For this particular case, this could largely be resolved by adding the succinyl-CoA ligase to the core gene set representing the citric acid. In general, this showcases the necessity of using databases with good resolution, but it also highlights the underlying intrinsic problem of annotating complex microbial communities, where the genes of novel microorganisms might be so distinct that an automatic differentiation between such similar functions is not possible.

Despite these imperfections in our HMM database, annotations with Metascan achieved a level of precision comparable to other annotation tools, but at a greatly reduced analysis time. In general, it is becoming increasingly challenging to obtain fast and reliable annotations due to the rapid growth of reference databases and the increasing size and sequencing depth of metagenomic samples to be analyzed. Thus, methods that reduce the reference dataset by clustering entries into subsets represented by HMM profiles are promising developments to overcome this hurdle, especially when considering that Metascan reached a high precision despite the drawbacks of the uncurated input database.

As indicated above, it became apparent during the development of this tool that we needed to construct a database that not only allows fast and accurate annotation of gene functions, but also categorizes the output according to the major nutrient cycles, which required a novel approach to build and structure this database.

Further Considerations

Here, we opted for a proof-of-concept approach, based on clustering proteins deposited in the UniProtKB/TrEMBL database. This database is by no means perfect since many protein entries in TrEMBL are not correctly annotated or incomplete, and herein lies the major point of improvement of our HMM database. The ideal input dataset would be manually curated like for instance the UniProtKB/SwissProt database, which will vastly increase the correctness for annotation. However, such well-curated databases are not yet suitable as many KO numbers are represented by less than three entries, which is the minimal number of sequences needed to create a HMM profile. A solution to circumvent this limitation would be a top-down approach, starting from a well curated database and subsequently adding missing HMM profiles using entries from other, less-well curated data sources.

Another possibility to improve the reliability of annotation is by employing a more stringent trimming and clustering algorithm when building the HMM database. However, while creating a database with stricter clustering rules will increase correctness, this will be at the expense of a longer analysis time. Lastly, the proteins in our database were clustered based on similarity, but if clustering instead was achieved by means of phylogenetic trees, this would provide additional information not only about evolutionary descent, but also about the exact function of proteins belonging to large and diverse enzyme families. However, this comes with its own set of difficulties and is not a trivial matter.

In the future, similar HMM subsets as developed here for nutrient cycling metabolic pathways could be constructed for non-metabolic pathways for a more complete genomic annotation. This will however greatly increase the runtime of the script, which would mean the need for a heavier computational infrastructure. For virus detection, a database of viral genes could be constructed in a similar way as presented here. Furthermore, the same procedure might be applicable for cell loci-specific proteins (e.g., cell wall or S-layer spanning), as these often share stretches of conserved amino-acids. In combination with RNA-seq, our HMM-based annotation approach would not only detect metabolic potential, but also actual activity of the overall cycles.

All things considered, we feel that Metascan can be of great help in mapping the important nutrient cycling pathways in an ecosystem by reducing and simplifying the input databases without compromising accuracy.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://www.ncbi.nlm.nih.gov/bioproject/PRJNA288027 (NCBI BioProject PRJNA288027) or https://ggkbase.berkeley.edu/2500-curated-genomes/organisms (ggKbase).

AUTHOR CONTRIBUTIONS

GC, MJ, HO, and SL contributed to conception and design of the study. GC built and organized the database and performed data analysis. GC wrote the first draft of the manuscript. GC, HO, and SL wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

FUNDING

This work was partly funded by the Netherlands Organisation for Scientific Research [NWO; Gravitation Grant No. 024.002.002 (SIAM) and VIDI grant No. 016.Vidi.189.050] and the European Research Council ERC Advanced Grant No. 669371 (Volcano) and ERC Synergy Grant No. 854088 (MARIX).

ACKNOWLEDGMENTS

We are grateful to our colleagues at the Department of Microbiology for helpful discussions and beta testing.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.861505/full#supplementary-material

REFERENCES

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic Local Alignment Search Tool. *J. Mol. Biol.* 215, 403–410. doi:10.1016/S0022-2836(05)80360-2

Anantharaman, K., Brown, C. T., Hug, L. A., Sharon, I., Castelle, C. J., Probst, A. J., et al. (2016). Thousands of Microbial Genomes Shed Light on Interconnected Biogeochemical Processes in an Aquifer System. *Nat. Commun.* 7, 13219–13311. doi:10.1038/ncomms13219

Aramaki, T., Blanc-Mathieu, R., Endo, H., Ohkubo, K., Kanehisa, M., Goto, S., et al. (2020). KofamKOALA: KEGG Ortholog Assignment Based on Profile HMM and Adaptive Score Threshold. *Bioinformatics* 36, 2251–2252. doi:10.1093/BIOINFORMATICS/BTZ859

Bateman, A. (2019). UniProt: A Worldwide Hub of Protein Knowledge. *Nucleic Acids Res.* 47, D506–D515. doi:10.1093/nar/gky1049

Bland, C., Ramsey, T. L., Sabree, F., Lowe, M., Brown, K., Kyrpides, N. C., et al. (2007). CRISPR Recognition Tool (CRT): A Tool for Automatic Detection of Clustered Regularly Interspaced Palindromic Repeats. *BMC Bioinforma.* 8, 209. doi:10.1186/1471-2105-8-209

Cantalapiedra, C. P., Hernández-Plaza, A., Letunic, I., Bork, P., and Huerta-Cepas, J. (2021). eggNOG-mapper V2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. *Mol. Biol. Evol.* 38, 5825–5829. doi:10.1093/MOLBEV/MSAB293

Carver, T., Harris, S. R., Berriman, M., Parkhill, J., and McQuillan, J. A. (2012). Artemis: An Integrated Platform for Visualization and Analysis of High-Throughput Sequence-Based Experimental Data. *Bioinformatics* 28, 464–469. doi:10.1093/bioinformatics/btr703

De Simone, G., Pasquadibisceglie, A., Proietto, R., Polticelli, F., Aime, S., J M Op den Camp, H. H., et al. (2020). Contaminations in (Meta)genome Data: An Open Issue for the Scientific Community. *IUBMB Life* 72, 698–705. doi:10.1002/iub.2216

Eddy, S. R. (2011). Accelerated Profile HMM Searches. *PLoS Comput. Biol.* 7, e1002195. doi:10.1371/journal.pcbi.1002195

Evanko, D. (2009). Metagenomics versus Moore's Law. *Nat. Methods* 6, 623. doi:10.1038/nmeth0909-623

Felsenstein, J. (1985). Confidence Limits on Phylogenies: An Approach Using the Bootstrap. *Evolution* 39, 783–791. doi:10.1111/j.1558-5646.1985.tb00420.x

Fritz, A., Hofmann, P., Majda, S., Dahms, E., Dröge, J., Fiedler, J., et al. (2019). CAMISIM: Simulating Metagenomes and Microbial Communities. *Microbiome* 7, 17–12. doi:10.1186/S40168-019-0633-6/FIGURES/5

Haft, D. H., Selengut, J. D., Richter, R. A., Harkins, D., Basu, M. K., and Beck, E. (2013). TIGRFAMs and Genome Properties in 2013. *Nucleic Acids Res.* 41, D387–D395. doi:10.1093/nar/gks1234

Han, K., Li, Z. F., Peng, R., Zhu, L. P., Zhou, T., Wang, L. G., et al. (2013). Extraordinary Expansion of a Sorangium Cellulosum Genome from an Alkaline Milieu. *Sci. Rep.* 3, 2101. doi:10.1038/srep02101

Heinemann, I. U., O'Donoghue, P., Madinger, C., Benner, J., Randau, L., Noren, C. J., et al. (2009). The Appearance of Pyrrolysine in tRNAHis Guanylyltransferase by Neutral Evolution. *Proc. Natl. Acad. Sci. U. S. A.* 106, 21103–21108. doi:10.1073/pnas.0912072106

Hyatt, D., Chen, G. L., LoCascio, P. F., Land, M. L., Larimer, F. W., and Hauser, L. J. (2010). Prodigal: Prokaryotic Gene Recognition and Translation Initiation Site Identification. *BMC Bioinforma.* 11, 119. doi:10.1186/1471-2105-11-119

Jensen, R. A. (2001). Orthologs and Paralogs - We Need to Get it Right. *Genome Biol.* 2, INTERACTIONS1002. doi:10.1186/gb-2001-2-8-interactions1002

Jones, D. T., Taylor, W. R., and Thornton, J. M. (1992). The Rapid Generation of Mutation Data Matrices from Protein Sequences. *Comput. Appl. Biosci.* 8, 275–282. doi:10.1093/bioinformatics/8.3.275

Kalkatawi, M., Alam, I., and Bajic, V. B. (2015). BEACON: Automated Tool for Bacterial GEnome Annotation ComparisON. *BMC Genomics* 16, 616. doi:10.1186/s12864-015-1826-4

Kanehisa, M., and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* 28, 27–30. doi:10.1093/nar/28.1.27

Kanehisa, M., and Sato, Y. (2020). KEGG Mapper for Inferring Cellular Functions from Protein Sequences. *Protein Sci.* 29, 28–35. doi:10.1002/pro.3711

Katoh, K., and Standley, D. M. (2013). MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability. *Mol. Biol. Evol.* 30, 772–780. doi:10.1093/molbev/mst010

Kumar, S., Stecher, G., and Tamura, K. (2016). MEGA7: Molecular Evolutionary Genetics Analysis Version 7.0 for Bigger Datasets. *Mol. Biol. Evol.* 33, 1870–1874. doi:10.1093/molbev/msw054

Lagesen, K., Hallin, P., Rødland, E. A., Staerfeldt, H. H., Rognes, T., and Ussery, D. W. (2007). RNAmmer: Consistent and Rapid Annotation of Ribosomal RNA Genes. *Nucleic Acids Res.* 35, 3100–3108. doi:10.1093/nar/gkm160

Laslett, D., and Canback, B. (2004). ARAGORN, a Program to Detect tRNA Genes and tmRNA Genes in Nucleotide Sequences. *Nucleic Acids Res.* 32, 11–16. doi:10.1093/nar/gkh152

Mistry, J., Chuguransky, S., Williams, L., Qureshi, M., Salazar, G. A., Sonnhammer, E. L. L., et al. (2021). Pfam: The Protein Families Database in 2021. *Nucleic Acids Res.* 49, D412–D419. doi:10.1093/NAR/GKAA913

Nawrocki, E. P., and Eddy, S. R. (2013). Infernal 1.1: 100-fold Faster RNA Homology Searches. *Bioinformatics* 29, 2933–2935. doi:10.1093/bioinformatics/btt509

Nolte, J. C., Schürmann, M., Schepers, C. L., Vogel, E., Wübbeler, J. H., and Steinbüchel, A. (2014). Novel Characteristics of Succinate Coenzyme a (Succinate-coa) Ligases: Conversion of Malate to Malyl-Coa and Coa-Thioester Formation of Succinate Analogues *In Vitro*. *Appl. Environ. Microbiol.* 80, 166–176. doi:10.1128/AEM.03075-13

Oh, J. I., and Bowien, B. (1998). Structural Analysis of the Fds Operon Encoding the NAD+-linked Formate Dehydrogenase of Ralstonia Eutropha. *J. Biol. Chem.* 273, 26349–26360. doi:10.1074/jbc.273.41.26349

Ondov, B. D., Bergman, N. H., and Phillippy, A. M. (2011). Interactive Metagenomic Visualization in a Web Browser. *BMC Bioinforma.* 12, 385. doi:10.1186/1471-2105-12-385

Saitou, N., and Nei, M. (1987). The Neighbor-Joining Method: a New Method for Reconstructing Phylogenetic Trees. *Mol. Biol. Evol.* 4, 406–425. doi:10.1093/oxfordjournals.molbev.a040454

Sayers, E. W., Beck, J., Brister, J. R., Bolton, E. E., Canese, K., Comeau, D. C., et al. (2019). Database Resources of the National Center for Biotechnology Information. *Nucleic Acids Res.* 48, D9–D16. doi:10.1093/nar/gkz899

Schnoes, A. M., Brown, S. D., Dodevski, I., and Babbitt, P. C. (2009). Annotation Error in Public Databases: Misannotation of Molecular Function in Enzyme Superfamilies. *PLoS Comput. Biol.* 5, e1000605. doi:10.1371/journal.pcbi.1000605

Seemann, T. (2014). Prokka: Rapid Prokaryotic Genome Annotation. *Bioinformatics* 30, 2068–2069. doi:10.1093/bioinformatics/btu153

Shaffer, M., Borton, M. A., McGivern, B. B., Zayed, A. A., La Rosa, S. L., Solden, L. M., et al. (2020). DRAM for Distilling Microbial Metabolism to Automate the Curation of Microbiome Function. *Nucleic Acids Res.* 48, 8883–8900. doi:10.1093/NAR/GKAA621

Søndergaard, D., Pedersen, C. N. S., and Greening, C. (2016). HydDB: A Web Tool for Hydrogenase Classification and Analysis. *Sci. Rep.* 6, 1–8. doi:10.1038/srep34212

Steinegger, M., and Söding, J. (2018). Clustering Huge Protein Sequence Sets in Linear Time. *Nat. Commun.* 9, 2542–2548. doi:10.1038/s41467-018-04964-5

Tourancheau, A. B., Tsao, N., Klobutcher, L. A., Pearlman, R. E., and Adoutte, A. (1995). Genetic Code Deviations in the Ciliates: Evidence for Multiple and Independent Events. *EMBO J.* 14, 3262–3267. doi:10.1002/j.1460-2075.1995.tb07329.x

Van Rijsbergen, C. J. (1977). A Theoretical Basis for the Use of Co-occurrence Data in Information Retrieval. *J. Documentation* 33, 106–119. doi:10.1108/eb026637

Zhou, Z., Tran, P. Q., Breister, A. M., Liu, Y., Kieft, K., Cowley, E. S., et al. (2019). METABOLIC: High-Throughput Profiling of Microbial Genomes for Functional Traits, Biogeochemistry and Community-Scale Functional Networks. *Microbiome* 10, 761643. doi:10.1101/761643

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Cremers, Jetten, Op den Camp and Lücker. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 24 June 2022
doi: 10.3389/fbinf.2022.871256

Deep Learning Encoding for Rapid Sequence Identification on Microbiome Data

Jacob Borgman, Karen Stark, Jeremy Carson and Loren Hauser*

Department of Data Science, Digital Infuzion, Inc., Gaithersburg, MD, United States

We present a novel approach for rapidly identifying sequences that leverages the representational power of Deep Learning techniques and is applied to the analysis of microbiome data. The method involves the creation of a latent sequence space, training a convolutional neural network to rapidly identify sequences by mapping them into that space, and we leverage the novel encoded latent space for denoising to correct sequencing errors. Using mock bacterial communities of known composition, we show that this approach achieves single nucleotide resolution, generating results for sequence identification and abundance estimation that match the best available microbiome algorithms in terms of accuracy while vastly increasing the speed of accurate processing. We further show the ability of this approach to support phenotypic prediction at the sample level on an experimental data set for which the ground truth for sequence identities and abundances is unknown, but the expected phenotypes of the samples are definitive. Moreover, this approach offers a potential solution for the analysis of data from other types of experiments that currently rely on computationally intensive sequence identification.

Keywords: deep learning, microbiome, convolutional neural networks, rapid sequence identification, encoding, embedding, denoising

OPEN ACCESS

Edited by:
Nikos Kyrpides,
Joint Genome Institute, Berkeley Lab (DOE), United States

Reviewed by:
Yu Li,
King Abdullah University of Science and Technology, Saudi Arabia
Kentaro Tomii,
National Institute of Advanced Industrial Science and Technology, Japan

****Correspondence:***
Loren Hauser
lorenh@digitalinfuzion.com

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *08 February 2022*
Accepted: *30 May 2022*
Published: *24 June 2022*

Citation:
Borgman J, Stark K, Carson J and Hauser L (2022) Deep Learning Encoding for Rapid Sequence Identification on Microbiome Data. Front. Bioinform. 2:871256. doi: 10.3389/fbinf.2022.871256

INTRODUCTION

The identification of known sequences and of new variants related to known sequences has been foundational to biological science over decades. The original Smith-Waterman algorithm (Smith and Waterman, 1981) identified the most optimal alignments between sequences but was computationally demanding and therefore slow. BLAST was first introduced in 1990 (Altschul et al., 1990) as a more rapid approximation and has evolved to its current form (Camacho et al., 2009) as the main workhorse for sequence identification. The use of k-mers (Edgar, 2004) has also become a widely used method for faster rapid approximations based on string searches and counts. Because of the large numbers of reads found in many experimental microbiome samples and the frequency with which bacteria contain multiple copies of the 16S gene many times with single-base variation, there is a need for a solution that can further reduce computational demands on sequence identification while simultaneously providing single-base resolution of sequence variation. Moreover, improved methodology for identification of which single-base variants in a microbiome sample represent sequencing errors and which are likely to be true biological sequence variants would assist in obtaining accurate abundance results.

A search of PubMed using the term “Microbiome” generates over 100,000 listings and a graph showing exponential growth over the last 10 years. The Human Microbiome Project (https://portal.

hmpdacc.org/), which contains just 18 microbiome studies, contains over 30,000 samples. Most published microbiome studies contain small number of samples, and therefore, their statistical resolving power is low. In order to increase the resolving power of studies on a specific subject, larger studies containing many thousands of samples are desirable, and the capability to combine multiple studies for meta-analysis would be useful. In either case, this means that thousands of samples would need to be processed with speed and accuracy using a single set of analysis tools. Reduction in the computational burdens of using these tools would promote the ability of more researchers to conduct studies with larger samples.

The output of commonly used microbiome tools falls into two general categories: operational taxon units (OTUs) that group together closely related strains into higher level taxonomic units and amplicon sequence variants (ASVs) that strive to achieve the base pair level accuracy required to bring taxonomic identification to the strain level. The microbiome analysis pipeline tool QIIME2 (Bokulich et al., 2018) uses the widely adopted VSEARCH algorithm (Rognes et al., 2016) for microbiome data analysis at the OTU level, while also permitting optional selection of a broader range of algorithms. VSEARCH uses a k-mer-based approach to speed sequence identification and error resolution, originally inspired by USEARCH (Edgar, 2010). USEARCH evolved to include UPARSE (Edgar, 2013) for OTU analysis using default 97% identity for clustering. Many clustering methods including mothur-average (Schloss et al., 2009), UPARSE, and UCLUST (Edgar, 2010) are benchmarked and compared in Kopylova et al. (2016) and applied to test microbiota data sets at the OTU level.

OTU methods intentionally speed analysis by settling for higher level taxonomic resolution and that is frequently sufficient for phenotypic studies. ASV methods take on the additional challenge of trying to achieve finer-grained taxonomic resolution by distinguishing sequence variation that is due to errors in the sequencing process from true biological sequence variants. Among ASV methods, the DADA2 microbiome analysis tool (Callahan et al., 2016) uses a probabilistic model to identify amplicon sequence variants (ASVs) with high sequence fidelity and has been chosen by multiple comparative studies as having the highest biological resolution for differentiating closely related and/or low abundance strains (Nearing et al., 2018; Caruso et al., 2019; Prodan et al., 2020). The UNOISE3 algorithm (Edgar, 2016) uses a kmer-based approach to sequence identification and error correction to produce ASVs. UCLUST, UPARSE, VSEARCH, and UNOISE3 allow for pooling all samples or clustering sequence reads for each sample individually for error correction. DADA2 uses a subset of samples to learn its error profile and then applies this error model to one sample at a time. The Deblur algorithm (Amir et al., 2017) operates on each sample separately for clustering to identify ASVs. Caruso et al. (2019) found DADA2 and UNOISE to be preferable for maximizing detection of true community members but note Deblur may be more appropriate for minimizing detection of spurious ASVs. UNOISE has been shown to have significantly higher speed (Nearing et al., 2018) than DADA2. Performance benchmarks and detailed comparison of the algorithmic similarities and differences among the VSEARCH, DADA2 and UNOISE3 algorithms is given in Tremblay and Yergeau (2019) and among DADA2, UNOISE3 and Deblur in Nearing et al. (2018).

Microbiomics is an ideal field for applying recent advances in machine learning that may offer speed advantages in combination with high accuracy when there is sufficient training data available. There is a large quantity of publicly shared microbiome data, with countless studies revealing the pivotal role microbial populations play in establishing and maintaining healthy conditions within diverse set of ecosystems, including the human body. The gut microbiome alone has been implicated in bone and brain development, obesity, diabetes, autoimmune conditions, autism, cardiovascular disease, metabolic disorders, inflammatory bowel disorders, and drug response (Cho and Blaser, 2012; Jandhyala et al., 2015; Levy et al., 2017; Thursby and Juge, 2017; Barlow et al., 2018; Gilbert et al., 2018). The presence or absence of certain bacterial populations are often directly linked to these medical conditions. Effective tools for characterizing healthy versus unhealthy microbial populations with resolution as close to the strain level as possible have an important impact on biological discovery, potentially leading to new diagnostics and treatments. Soil and plant microbiomes are also subjects of active research, where the same tools can be applied to determine microbial composition and lead to valuable interventions.

Unprecedented levels of accuracy in other fields have been achieved by the expansion of machine learning through the development of Deep Learning algorithms. In 2012, the convolutional network AlexNet created a sensation with its dramatic improvement demonstrated in an established computer vision competition using the ImageNet challenge data (Krizhevsky et al., 2012). Since then, image based neural networks have continued to evolve, both in terms of architecture and training strategies, from recurrent neural networks to the now widely applied Transformer (Vaswani et al., 2017) design. Aside from computer vision, these algorithms have revolutionized other important areas such as speech and text recognition and have created headline news with vast AI improvements in specialized domains such as board games [e.g., AlphaGo (Silver et al., 2016)] and protein folding [AlphaFold (Jumper et al., 2021)].

Deep learning algorithms have been applied to classify the phenotype of microbiome and metagenome samples. Asgari et al. (2018) showed that deep learning can outperform random forest classifiers and support vector machines for phenotypic prediction from 16S data when the number of samples is large. Zhao et al. (2021) use kmer embeddings and convolutional neural networks, recurrent neural networks, and attention mechanisms to predict taxonomic classifications and sample-associated attributes of whole microbiome data at the level of a read. They use additional methods such as voting to determine the phenotype of each sample from the deep-learning-predicted phenotype of the reads. This enables the predictor to consider many thousands of read sequences and it achieves accuracy at phenotypic prediction comparable to existing methods. An early application of deep recursive neural networks to metagenomic data did not show much improvement over other methods for metagenomic classification but the ability to learn hierarchical representations of a data set that is produced could be useful (Ditzler et al., 2015). Furusawa et al. (2021) chose an unusual approach to perform image analysis on Gram-stained fecal samples to classify their microbiome state with a deep convolutional neural network. Although the prediction success was low for fecal state, particularly on samples from adjacent time points, it had more success in predicting quantitative changes in microbial abundances. García-Jiménez et al. (2021) explored the creation of deep latent spaces for prediction of the ecological composition of a microbiome sample using minimal sequencing features and incorporating sample environmental metadata such as rainfall and plant age. These methods are generally not intended to produce exact microbial composition based on rigorous sequence variant identification or optimal abundance estimates at the level of ASVs. They focus more on the ability to accurately categorize the sample as a whole with respect to a relevant phenotype (i.e., a population characteristic of interest). However, for a deeper understanding of the microbial population, the population dynamics and the ability to approach the mechanisms by which the microbiome exerts its influence, an accurate analysis of its composition at the true sequence variation level provides more scientific insight than a phenotypic classifier.

Sequence identification in the closely related field of metagenomics is an area where deep learning algorithms are beginning to be applied. The Seeker tool (Auslander et al., 2020) addresses the challenging problem of detecting bacteriophage in metagenomic sequence data since bacteriophage evolve rapidly, quickly losing sequence similarity to known bacteriophage. It uses Long Short-Term Memory (LSTM) networks, a type of Recurrent Neural Network (RNN), trained on bacteriophage and bacteria sequences to detect subtle differences in sequence usage and is able to predict which sequences in the metagenome are bacteriophage and which are from bacteria, even when homology to known bacteriophage is very low. Virfinder2 (Ren et al., 2020) is a convolutional neural network (CNN) that learned to predict viral sequences by training on the differences between prokaryotic and viral DNA sequences. Its success does not rely on known sequence homologies or use of pre-defined features such as kmers. While using more traditional machine learning and not deep learning, the VirSorter2 algorithm (Guo et al., 2021) has demonstrated considerable success in identifying both RNA and DNA viruses within metagenomic samples. It relies on a collection of known viral motifs and annotations that are used as input features to a set of random forest classifiers each trained on a major viral group. Its modular design allows for easy updates as known viral diversity grows. While each of these tools is successful at sequence analysis and appropriate for metagenomics, their algorithmic approaches are not readily adapted for microbiome analysis that relies on 16S amplicon sequences from a single bacterial gene and then attempts to identify the population of bacteria represented by those sequences.

Herein, we describe a deep learning approach to finding ASVs and obtaining their abundance estimates on sample sequencing data obtained from mock communities of known bacterial composition. We will show that using a latent sequence space based on all known bacterial V4 sequences from the 16S gene and using a Convolutional Neural Network algorithm trained to map V4 sequences obtained from experimental data into this space will match or better the accuracy of the best available open source microbiome tools in significantly shorter computational time. A denoising method that starts with clustering of experimental sequence data in the V4 16S latent sequence space achieves accurate abundance estimates. Although motivated by the desire for improved sensitivity and accurate abundance estimation of the microbial community, we demonstrate that the output still supports phenotypic prediction by comparison to previously published results for four data sets where the exact microbial composition is unknown, but the phenotypes of the samples are unambiguous.

This approach may be extensible to other types of experimental sequence data in addition to microbiome where single nucleotide resolution, correction of likely sequencing errors and accurate abundance estimation are desirable. We refer to this mathematical approach as Deep Learning Encoding for Rapid Sequence Identification (DERSI).

MATERIALS AND METHODS

In order to identify a method for both rapid and accurate identification of ASVs from microbiome experiments that use 16S sequence, a series of steps were performed. The steps were used on data from microbiome sequence analysis using the V4 region of the 16S gene.

1. The first step was to create a 10-dimensional latent space that encoded the distances among all known bacterial and archaeal V4 sequences. An overview of this step is presented in **Figure 1**. A copy of the Silva rRNA database (Yilmaz et al., 2014) version 132 that contained alignments for known 16S sequences was supplemented with sequences from GenBank. The Silva database consisted of a set of more than 200,000 samples of known 16S sequences placed into alignment with each other. Due to gaps and insertions, ~50,000 possible nucleic acid positions are present in this alignment matrix for the full-length 16S gene. The alignment matrix exhibits extreme sparsity and any manipulation of it rapidly becomes computationally infeasible. The number of features was therefore reduced by eliminating nucleic acid base positions that were present in less than 0.1% of sequences. The resulting matrix was given to the Mothur (Schloss et al., 2009) software package as a template and each of the 16S sequences from GenBank that were not in that release of Silva were aligned to the template by Mothur tool using default parameters. The resulting alignment matrix was then trimmed to the V4 region resulting in 320 nucleic acid positions (features) and 117,161 unique V4 sequences (samples).

To create the latent sequence space for V4, the pairwise distances among each of the aligned V4 sequences in the V4 matrix (D) must be accurately reflected in the corresponding distances among those samples in the latent space (d). First, each V4 sequence was converted to a one hot vector, and then the distance to each of the other V4 sequences was calculated based

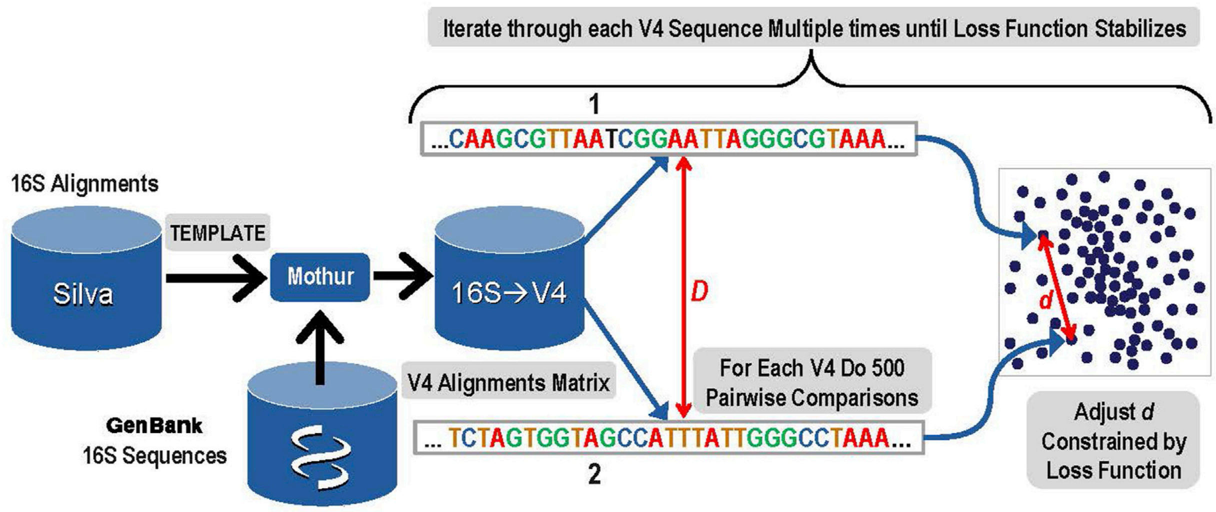


FIGURE 1 | Overview of Step 1: The construction of a latent space that represents known V4 sequences and their alignment distances with reduced dimensionality while maintaining a high degree of accuracy.

on the alignments and the data was sorted by this distance. The distance (D) reflects the distance between the aligned sequences and its detailed calculation is presented in the **Supplementary Material Choosing Proper Distance Metrics**. Since it is not computationally feasible to iteratively train the latent space while rigorously adjusting all of the embedded latent space distances (d) for every possible pair during every iteration, a sampling method was used. In order to optimize the latent space to distinguish single base pair variation, the 300 sequences most similar to each V4 sequence based on the distance (D) were included in the sampling. In addition to those 300 nearest sequences to the current V4 sequence, 200 more were chosen for comparison to the current sequence by sampling from the remaining 116,860 potential pairs using a dilation formula that biases toward the most similar remaining sequences with the most distantly related sequences receiving the lowest sampling representation. The intention is to provide sufficient accuracy to distinguish closely related sequences that may differ by as little as a single base using the embedding distance d.

The mathematical details for the sampling method are given here. For each known V4 sequence:

1. Convert the sequence to a one-hot vector, calculate and sort by the distances from this known sequence to all other V4 sequences, selecting the n sequences with smallest distance to this known sequence
2. Gather additional sampling from the remaining m–n V4 sequences using an exponential schedule such that the ith sample is at position $n * f^{i}$ in the sorted samples, where the dilation factor f is found by solving the relation:

$$N = n \times f^{m-n} \; and \; f = e^{\ln\left(\frac{N}{n}\right)/(m-n)}$$

Where in this instance:

N = 117,161 unique sequences for the V4 region
m = 500 total number of samplings per sequence
n = 300 number of nearest neighbors included.

After completing these samplings for the distance comparisons that will be used to ensure the constructed latent space reflects the actual sequence distances, the next step was to actually construct the space. Within the latent space, each of the 117,161 V4 sequences was represented as a 10-dimensional vector. Initial values for each vector were filled at random using a centered Gaussian with sigma = 10. Determining the accurate placement for each unique V4 sequence vector in the latent space was done during an iterative gradient descent training by adjusting the distances (d) among pairwise sequences within the 10-dimensional latent space to closely match the distances calculated for each of the sampled 500 distance comparisons in the original sequence matrix (D). Thus, for each V4 sequence, a total of 500 pairwise comparisons were used in each iteration of the gradient descent training to construct the latent space.

In mathematical terms, given the input sequence space S and the embedding space E, we seek a mapping $f : S \rightarrow E$, such that for every $x, y \in S$ and $f(x), f(y) \in E$, we obtain $D(x, y) \approx d(f(x), f(y))$, where D and d are distance functions in the original sequence and embedding spaces, respectively. To ensure that d corresponds to D, we used a loss function that favors nearest neighbors. The form of this loss to be used during gradient descent training is $L = (1 - \frac{d}{D+\epsilon})^2 = (\frac{d-D}{D+\epsilon})^2$. In addition to the accuracy promoted by the sampling approach described above, this loss function will also encourage high resolution for close sequences (small values of D) for the facilitated detection of single nucleotide base changes, while permitting lower resolution between highly divergent sequences. The $\epsilon \sim 1$ regularizes the loss for vanishing phylogenetic distances.

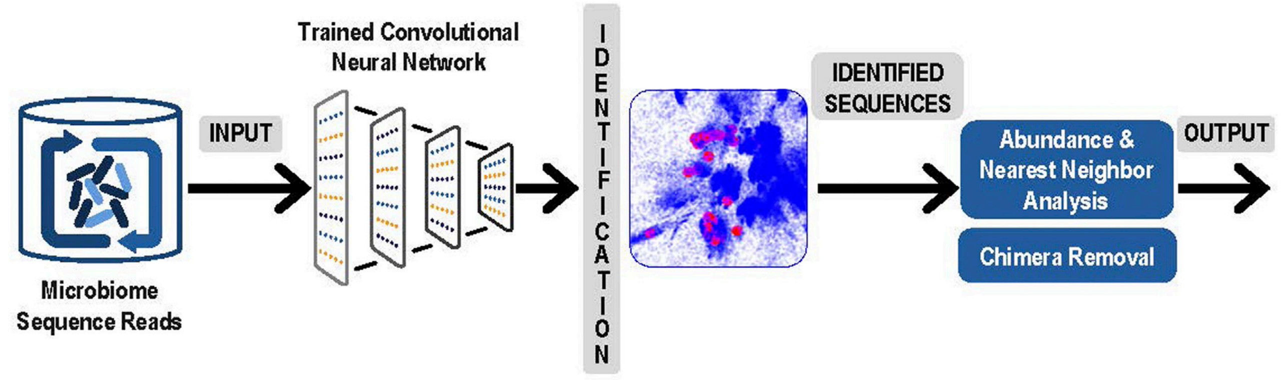

Microbiome Sequence Reads
INPUT
Trained Convolutional Neural Network
IDENTIFICATION
IDENTIFIED SEQUENCES
Abundance & Nearest Neighbor Analysis
Chimera Removal
OUTPUT

FIGURE 2 | Analysis of Experimental Sequence Data: Paired reads from microbiome data were input to the trained neural network for identification. Resulting clusters were analyzed for abundance and correction of likely sequencing errors resulting in output of each unique ASV and its corresponding abundances.

The gradient descent training iteratively continued adjusting distances within the space until the changes to the average loss function with each iteration fell below five significant digits. Finally, we note that the metadata was carried through the process since each Silva/Genbank sequence in the matrix was a known V4 sequence, so each 10-dimensional vector used in the training was associated with a known V4 and its taxonomic identity. Additional mathematical and algorithmic details for the calculation of *D* and for the Gradient Descent and its subfunctions are given in the **Supplementary Material**.

2. The second step was to train a deep learning algorithm so that it could take any V4 sequence and map it into the previously built latent space. A convolutional neural network was trained using the 117,161 known unique bacterial V4 sequences and their corresponding 10-dimensional vectors in the sequence space. This V4 encoder was a subtype of convolutional neural networks that is fully convolutional (e.g., Long et al., 2015; Maggiori et al., 2016) sometimes also referred to as a fully convolutional network (FCN) with a total of 90 layers, 32 of which were convolutional. A max-pooling layer was inserted after the first two convolutions to reduce the size of the network and encourage translational invariance for spatial motifs. All convolutional layers (except the last) were followed by batch normalization to stabilize training, and a ReLU activation. The final convolution produced the 10-dimensional vector encoding that matched the target 10-dimensional vector for each input sequence. We show a spreadsheet with all 90 layers in the **Supplementary Material**. As for all trained deep learning algorithms, the trained neural network can generalize and produce 10-dimensional vectors even for sequences not included in the training set, in this case, if new and previously unknown V4 sequences are found experimentally. We trained this CNN encoder within the PyTorch framework using an Adam optimizer with learning rate = 0.0001, a loss function that was simple Euclidean distance between the 10-dimensional output and the precomputed 10-dimensional embeddings. Each training batch of 200 training samples were randomly sampled from the 117,161 V4 SG dataset, training with a total of 50,000 batches.

3. The next step was to use the latent space and the trained convolutional neural network to identify and measure the abundance of the sequences obtained from microbiome experiments. Sequence obtained from paired reads from a microbiome sample were presented to the trained convolutional neural network and mapped into the correct position in the latent space. Note that this now comprised a rapid classification process that was accomplished without any explicit pairwise alignment of the sequences from the microbiome experiment. After each sequence was mapped, the result was a collection of sequences from the microbiome experiment represented as clusters in the latent sequence space. In **Figure 2**, below we present an overview of this step, and of the next and final step in the process, the denoising.

4. The final step was denoising by examining the experimental microbiome data for possible sequencing errors and finalizing the number of V4 ASVs and their abundance. To separate actual sequence variants from sequencing errors, our denoising process began with analyzing the relative abundances of closely related sequences. Reads with sequence that occurred only once in the microbiome sample were eliminated. For each remaining sequence, a determination was made whether to consider it a valid unique bacterial sequence variant or if it likely originated as a sequencing error from a more abundant “parent” sequence. To identify candidate parent reads, a fast Nearest Neighbor search was done in the latent space using NanoFlann (https://github.com/jlblancoc/nanoflann). A maximum of 20 nearest neighbors were selected that were within at least a 15 bp radius in the latent space and that were at least 20-fold more abundant than the sequence under consideration. For each of these parent candidates the edit distance was computed using Edlib package (Šošic and Šikic, 2017) and only candidates with less than 1 bp difference per 64 bases (98.5% match) were retained. If a sequence had no candidate parents after this process, it became an identified V4 ASV. Otherwise, remaining candidate parents were sorted by edit distance and the closest was selected as the likely parent (in event of a tie, the more abundant was favored). The child sequence was now considered to be a likely sequencing error originating from the parent sequence. Once the process was

completed, the parent-child relationships were traversed until a sequence was identified that had no parent. Each sequence that had no parent was then considered an identified V4 ASV and its children and any grandchildren were then included into its abundance count.

Standard chimera removal was accomplished using the VSEARCH package uchime_denovo command. Since the latent space training in the previous step had been conducted with known V4 sequences, and the phylogenetic metadata from the Silva/Genbank sequence matrix was carried through for each 10-dimensional vector during that latent space training, the 10-dimensional vector for each resulting ASV in the latent space was readily associated with phylogenetic information for its exact or closest-associated known V4.

Analysis on Mock Community Data Sets

Two sets of experimental microbiome data for mock communities were then analyzed using these methods. First, the data were processed as in step 3 above and mapped by the trained convolutional neural network. Next, they were processed through step 4 for denoising and finalization of the ASVs. These data sets were chosen from the mockrobiota resource (Bokulich et al., 2016; http://caporaso-lab.github.io/mockrobiota/) for samples that were created by sequencing bacterial mixtures of known abundance and composition in order to rigorously assess the performance of the method by using data for which the results to be obtained are known. Mock 16 (Schirmer et al., 2015) was selected to determine the robustness of the method for identifying ASVs on a complex mixture of known composition, containing 49 bacterial and 10 archaeal species. Mock 12 (Callahan et al., 2016) was chosen to examine the method across extreme variation in concentration across the bacterial mixture. The Mock 23 (Gohl et al., 2016) was chosen to give additional statistical basis for abundance estimation and average speed measurements on a relatively small and less complex data set. Since it does not exceed the Mock 16 for diversity or the Mock 12 for concentration range, the analysis in addition to speed is shown in the **Supplementary Tables S5, S6**. Each ASV identified for each data set was validated by BLAST to confirm DERSI's taxonomic identification.

In most cases, the bacterial genomes contain more than one copy of the 16S gene, and these additional copies may be identical or varied in their V4 region sequences. In order to provide a high level of rigor to the expected sequences and their abundances, we therefore deemed it necessary in our analysis to calculate the expected number of V4 sequences by looking at the genomes of each bacterium and to adjust the expected number of ASVs for each bacterium. In some cases, the V4 sequences were identical between two different bacterial genomes, and the number of expected ASVs was accordingly reduced. In a few cases, full bacterial genomes were not yet available and best estimates were made based on numbers and sequences of closely related bacterial genomes. The expected numbers of ASVs and OTUs depends upon knowing how many variants are present in each genome. And while copy number does not affect the number of ASVs, it does affect the expected abundance measures. Therefore, both the number of expected ASVs and their expected % of the total composition were adjusted to reflect the genomic composition of the mock community.

While these mock community data sets are intentional compositions, previous work on these data sets has also demonstrated the presence of unintentional contaminants (e.g., Callahan et al., 2016; Edgar, 2016; Nearing et al., 2018). Calculating precision and recall for this type of data requires determining exactly how a useful number can be rigorously generated, given that unintended contaminants are present in these data. A prior review of microbiome algorithms chose to calculate precision based on perfect matches to a reference sequence considered true positives versus noisy (less than 100%) matches to known sequence considered false positives (Caruso et al., 2019), and we have based our approach largely upon this. A second method was also presented that considered all unexpected sequences to be false positives, and therefore was likely to confound the accuracy of the experimental protocol and its susceptibility to contamination with the accuracy of the algorithms.

At very low concentrations it becomes very challenging to assess false positives versus minor contaminants, and VSEARCH has been previously shown to identify large numbers of such sequences. Given the difficulty in assessing whether this shows exceptional sensitivity to low abundance contaminants or a severely elevated false positive rate, we chose to apply several filters to the sequences that do not have a 100% match to a known V4. We eliminated ASV/OTUs: 1) with less than 92% identity to the closest known V4 sequence; 2) or with less than 0.01% abundance and less than 99% identity to the closest known sequence; 3) or with less than 0.001% abundance and less than 100% identity. At this level of stringency, those V4 sequences found at very low concentrations are highly likely to be false positives since at 99% identity they will be only one or two bases different from a known V4 sequence.

Of the remaining ASVs/OTUs those identified at 100% to a known sequence are considered to be true positives whether intentionally added to the mock community or not. The complete set of sequences to be used for calculating recall numbers was the union of all such unique V4s with a 100% match to a known sequence found by all four compared algorithms.

Since we have conducted a genomic analysis to identify expected values for each of the V4 sequences from the intentionally input bacterial genomes, we are able to make a comparison of the identified values to the expected values. A Bhattacharyya coefficient (Bhattacharyya, 1943) was computed over the abundances detected for both mockrobiota data sets to give a measure of the overall accuracies of the abundance estimates made by each algorithm. The Bhattacharyya coefficient provides a divergence measure between two multinomial populations and so is suitable to describe the differences between the population of expected sequence abundance values for the input mock community with the second multinomial population being the values of those sequence abundance values reported by the algorithm being tested.

Data Sets for Speed Comparison

The Mock 23 was included to give additional statistical basis for average speed measurements on a relatively small and less complex data set. We also included the Mock 12 and Mock 16 data, and these show increasing size.

Finally, to assess the speed performance on much larger data sets than afforded by these mock communities, we selected the first 1,072 samples from the Goodrich et al. (2016) microbiome data set. Since this is a human biologically identified data set the exact expected composition of the bacterial community and associated abundances are not known and therefore a full analysis against expected values was not performed, it was only used for the speed comparisons.

Data Sets for Phenotypic Analysis

The motivation for the development of the DERSI method was greater accuracy in the determination of ASVs combined with high speed. However, to demonstrate that this approach also supports phenotypic analysis, we chose several previously published data sets for which to compare to published results for experimental data sets where the exact composition of the bacterial mixture is not known but the correct phenotype for the overall sample is known.

We selected two microbiome datasets that used 16S V4 sequence from an analysis of the effects of water decontamination method and choice of bedding material on the fecal microbiome of mice (Bidot et al., 2018). We selected the first data set of fecal microbiome samples for mice in which both groups use corncob bedding but one group was given autoclaved water and the other group water purified by reverse osmosis. The second data set of fecal microbiome data was from mice who were given either paper bedding and water purified by reverse osmosis or corncob bedding and water purified by autoclaving.

The third set of microbiome data was chosen from Mezzasalma et al. (2018) comprising multiple microbiome samples from three grape cultivars grown in the same vineyard. Each sample was taken from a different vine and consisted of consisted of a small bunch of grapes. The cultivar was the phenotype to be predicted. The original V3–V4 reads were trimmed to obtain the V4 sequencing data from this experiment.

A fourth set of microbiome data for phenotypic analysis was taken from a study of chicken ceca transplantation (Glendinning et al., 2022). Microbiome samples from two ceca obtained from donor chickens of the Roslin broiler breed were transplanted into chickens of the Ross broiler breed. Additional chickens received sham transplantations with saline as controls. Subsequently, the microbiome samples from the transplant recipient chickens, the two donor ceca and the controls were sampled and sequenced using the V4 region of the 16S.

For all phenotype data sets, the V4 sequence reads were input to the DERSI trained convolutional neural network. The output data was then normalized using a trimmed mean and taking the logarithm, followed by the denoising process described in step 4 above. PCA was then used to map each normalized sample into a 3D space for comparison to published results.

Benchmarks

The analysis of the diverse Mock 16 and the extreme concentration variation Mock 12 data sets was benchmarked against three widely used methods for microbiome analysis: an OTU method, VSEARCH (Rognes et al., 2016) and two ASV method DADA2 (Callahan et al., 2016) and UNOISE3 (Edgar, 2016). VSEARCH was chosen since it has been widely used for OTU analysis, has been benchmarked against other algorithms and is included in the QIIME2 microbiome pipeline. DADA2 has been widely recognized as the most sensitive method for detection of ASVs in multiple benchmarks and is also included in the QIIME2 pipeline where it can be optionally selected. UNOISE3 is private source software with a freeware 32-bit executable and has been shown to give near comparable results to DADA2, sometimes with greater specificity. VSEARCH and DADA2 were run using QIIME2, and the 32-bit UNOISE3 software for Linux was downloaded from https://drive5.com/usearch/download.html as part of the overall USEARCH package.

Speed for all four methods was measured on the same System76 Oryx Pro Laptop using a Linux operating system (Ubuntu 20.04). Multiple steps were included in the speed measurements, including preprocessing and dereplication, identification of ASVs/OTUs, denoising to correct potential sequence errors, chimera removal and abundance calculations.

Parameters for Each Algorithm for Analysis and Speed Comparisons

These are the details of the steps and parameters used for comparison of the algorithms for the mock community analysis and the speed comparisons. Primers were removed from the mock community data sets Mock 16 and Mock 23 using multiple sequence alignment against our expanded Silva database using mothur. Primers were not present in Mock 12. Pooling of samples is an option for some algorithms, however, we did not pool samples for these benchmarks in order to be close to DADA2's process and ensure a fair comparison.

DADA2 was run as included with its particular preprocessing methods and defaults in QIIME2 except for forward and reverse quality trimming. The only parameters changed were the forward and reverse truncation, determined by inspecting Q values for each data set (all other parameters were left at their defaults):

```
mock12 -p-trunc-len-f 180 -p-trunc-len-r 140
mock16 and mock 23 -p-trunc-len-f 200 -p-trunc-len-r 180
Goodrich study -p-trunc-len-f 200 -p-trunc-len-r 140.
```

For VSEARCH, UNOISE3 and DERSI, after removing primers, we performed the identical merging and quality filtering so they would each receive the same input. This was accomplished using the vsearch command --fastq_mergepairs to merge pairs, with the following settings: -fastq_ascii 33; --fastq_minlen 180; --fastq_minovlen 20; --fastq_maxdiffs 12; --fastq_qmin 0; --fastq_qminout 0'; --fastq_qmax 41; --fastq_qmaxout 41; --fasta_width 0; --fastq_maxns 0. We then used the vsearch command --fastq_filter to quality filter,

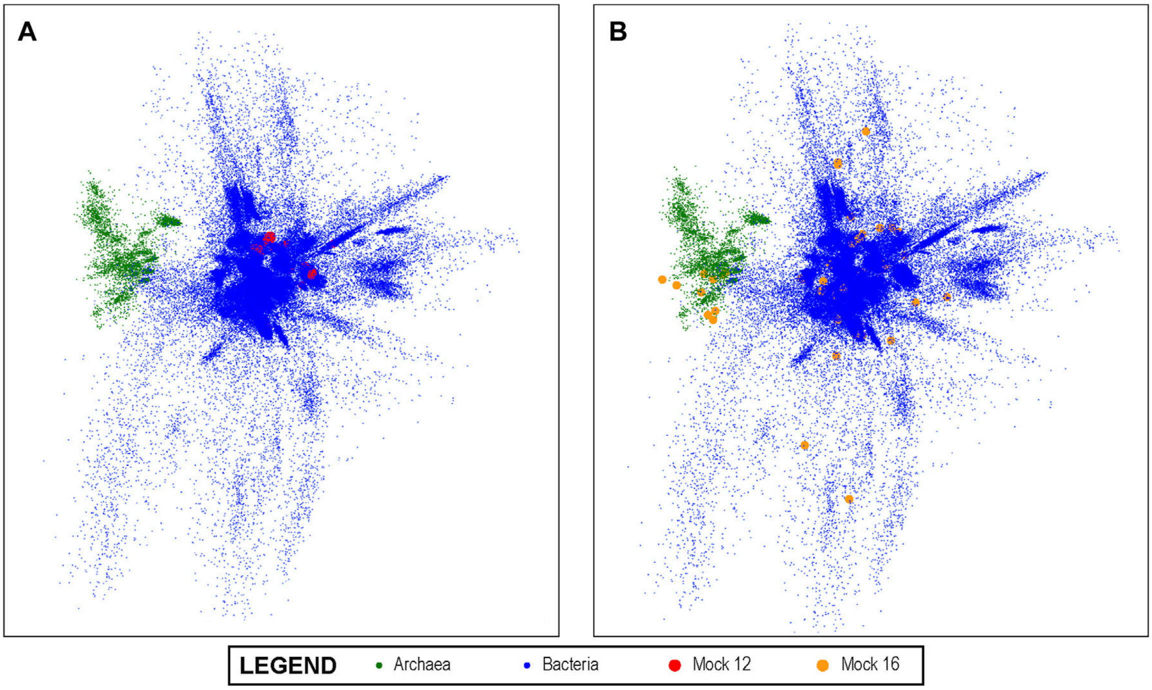


FIGURE 3 | The 10-dimensional V4 Latent Space: The latent space of all known bacterial V4 sequences projected into three dimensions using PCA for visualization; the Archeal V4s clearly separate from other V4s. In **(A)** we show the projection of the mock 12 community sequences as mapped by our V4 encoders onto the overall bacterial latent space. It can be visually observed that the mock 12consists of a relatively small number and not especially diverse set of bacteria. In **(B)**, we projected the highly diverse mock 16 community as mapped by the V4 encoder and this much greater diversity is readily observed. This demonstrates the ability of the method to produce results that can offer informative visualizations.

with following settings: --fastq_maxee 1.0; --fastq_minlen 225; --fastq_maxlen 275; --fastq_maxns 0; --fasta_width 0.

The above two steps were common to VSEARCH, UNOISE3 and DERSI to ensure that each received the identical input to dereplication. Each algorithm used its own dereplication method but VSEARCH, USEARCH (UNOISE3) and DERSI's dereplications are equivalent. All three algorithms dropped singletons. Reads were dereplicated using the vsearch/usearch command: --fastx_uniques --minuniquesize 2. OTUs/ASVs abundances were produced for VSEARCH with the following commands: vsearch --cluster_size {} --id 0.97 and then vsearch --uchime_denovo. For UNOISE3 the USEARCH commands: usearch -unoise3 (all default settings; this does error correction) usearch -otutab (this constructs abundances querying original sequences including singetons). For DERSI, dereplicated reads were encoded by the neural network, and these embeddings were then used for error correction as described in step 4 above. Chimera removal used vearch --uchime_denovo.

RESULTS

Latent Space Creation and Use

Training during Step 1 of the method continued until the loss function achieved an average value of 0.1101 and the variation at each iteration fell below five significant digits. The resultant latent space is a dense, structured, 10-dimensional point cloud for all known V4 sequences that reflects their aligned distances from each other to a very high degree of accuracy.

A visualization of this space is shown in **Figure 3** using PCA to project the 10-dimensional space into three dimensions. In the visualization, each dot represents a unique V4 sequence, and its proximity to other dots accurately reflects their sequence similarity. It can be seen that there are distinguishable groups of closely related sequences. Archaea, for example, are shown in green and are clearly more closely related to each other than to other V4 sequences, as would be expected from molecular phylogeny. Since the latent space created by this method lends itself to this type of visual representation, it also enables the results of the analysis of experimental data sets that are mapped into this space by the trained convolutional neural network to be projected onto this overall visualization. We show this in **Figure 3A** for the Mock 12 data set and in **Figure 3B** for the Mock 16 data set. It can be visually observed that the Mock 16 represents a highly diverse set with members distributed widely over known bacterial genome space. In contrast, the Mock 12 data set that consisted primarily of Bacterioidies and Firmicutes, shows a much more compact distribution.

Analysis of the Mock-16 Data Set

For the initial evaluation and testing of our new V4 deep learning encoding approach DERSI, as well as for comparisons to the other

TABLE 1 | Algorithm performance on complex bacterial mixture in the mock 16 data.

V4 variants	DERSI	DADA2	UNOISE3	VSEARCH
Added to Mock 16 Found/expected	60/63	59/63	60/63	48/63
Contaminants (>0.001%) Found/expected	21/22	21/22	19/22	17/22
False Positives (>0.01%) Found	1	6	1	3
Precision/Recall	99/95	93/94	99/93	96/76

widely used identification and quantitation algorithms DADA2 UNOISE3 and VSEARCH, we chose the Mock-16 data set because of its significant phylogenetic breadth, as it contains 59 species, 10 of which are Archaea. Although DNA from each of the 59 species was added in equal amounts, bacterial species vary in the copy numbers of the 16S gene and may have sequence variation among those copies for the V4 regions. Some species also share identical V4 regions. We have therefore calculated the expected number of unique V4s in the mock community as 63, details are shown in **Supplementary Material**.

A comparison of the output from analyzing the Mock-16 data set with DERSI, DADA2, UNOISE3 and VSEARCH is shown in **Table 1**. As detailed in the *Materials and Methods* for the calculation of precision and recall, some sequences identified at very low concentrations and with no exact matches to a known sequence were eliminated from consideration since it cannot be determined if they are false positives or represent actual contaminants of novel sequences in very low amounts.

Each ASV identified was validated by BLAST to confirm DERSI's taxonomic identification, and all taxonomy was found to be accurate. In our **Supplementary Material**, we provide the details of how each unique ASV identified maps to a bacterial species or group.

The output from DERSI, UNOISE3 and DADA2 shows virtually the same ASVs until the read count is below 31 reads per ASV. This is ~0.006% of a very large read count for a single sample (~520,000 reads). Below this level each algorithm finds a slightly different set of ASVs. DERSI finds 14, DADA2 finds 12, VSEARCH finds 11 and UNOISE3 finds 12, with DERSI retaining a slightly higher fidelity at these lower levels. We also note that in some cases with this set of experimental data the sequencing method itself failed due to no/low productivity of certain primer sets resulting in lack of detection by all three algorithms. It has been previously noted that the primers used for the V4 region do not amplify all V4s with equal efficiency resulting in some ASVs or OTUs that were not found in the sequence data (Allaband et al., 2019).

VSEARCH tends to combine very closely related ASVs into a single OTU, which can obscure the presence of closely related species, for example, it has put *Chlorobium phaeobacteroides strain DSM 266, Chlorobium phaeovibrioides DSM 265 and Chlorobium limicola strain DSM 245* into a single OTU (**Supplementary Material**). This is in keeping with the VSEARCH algorithm intended to predict at an OTU level rather than at the finer-grained prediction of ASV methods.

Row 2 of the table contains ASVs that were not intentionally included in the mock community but are detected and have 100% match to a known bacterial sequence. These are not closely related to the original input organisms and should not be considered false positives, but likely arise from inadvertent contamination. Many of these have been previously described (Callahan et al., 2016) in the original analysis of the data set. We have therefore included them in the precision and recall analysis. We note that our algorithm DERSI does slightly better at detecting such potential contamination.

Overall, VSEARCH performs considerably less well than DERSI. While UNOISE3 and DADA2 perform relatively well, DERSI has the best precision and recall of the four algorithms on the mock 16 data set.

Analysis of the Mock-12 Data Set

For the second major test of our new algorithm, we analyzed the data set listed as Mockrobiota Mock-12 since it has a 5-log unit variation in the input abundances (see **Table 2** first column). As was true for Mock 16, the exact expected abundance may vary from the input percentage of the bacteria due to multiple copies within a genome and sequence variation in these multiple copies (see **Supplementary Table S3** for details). Three of the five genomes in the most abundant two categories and two of the genomes in the lowest abundance category have multiple V4 regions. A number of the species input still do not have a complete genome in GenBank and in those cases the copy number was estimated based on closely related genomes. We show the resulting likely number of input V4s in **Table 2** as the expected count. The read abundance data does not vary significantly from the expected abundance based on this approximation (for a full list of expected abundances for each bacterial species, see the **Supplementary Material**). Each ASV identified was validated by BLAST to confirm DERSI's taxonomic identification, and the taxonomy provided was found to be correct.

The analysis of the four algorithms follows similar outcomes to that seen in the Mock-16 analysis. DERSI and UNOISE3 create an identical list of ASVs until the read count is below 17 or 0.0012%, at which point DERSI performs significantly better. Whereas VSEARCH tends to combine closely related V4s into single OTUs (more details in **Supplementary Material**). DERSI and UNOISE3 find two genomes, one (*Bacteriodes fragilis*) at the 0.01–0.1% abundance category and one genome (*Eubacterium rectale DSM 17629*) in the second lowest abundance category that DADA2 folded into another ASV. UNOISE3 misses all, while DADA2 misses four of the genomes in the lowest abundance category. DERSI and VSEARCH find seven of the 13 of the lowest abundance ASVs or OTUs in the input data set; note that none of the algorithms find any of the other six, they appear to be missing from the set of reads.

In rows 7 and 8 of **Table 2**, we present a summary of sequences identified in the mock community that were not expected based on the intended bacterial inputs but match a known V4 at 100% identity, and therefore likely represent contaminants. These

TABLE 2 | Algorithm performance across extreme abundance variation in the mock 12 data.

V4 variants	DERSI found/expected	DADA2 found/expected	UNOISE3 found/expected	VSEARCH found/expected
Added at >10%	2/2	2/2	2/2	2/2
Added at 1–10%	7/7	7/7	7/7	3/7
Added at 0.1–1%	4/4	4/4	4/4	4/4
Added at 0.01–0.1%	4/4	3/4	4/4	2/4
Added at 0.001–0.01%	4/4	3/4	4/4	3/4
Added at 0.0001–0.001%	7/13	4/13	0/13	7/13
Contaminant at 0.001–0.01%	2/2	2/2	2/2	2/2
Contaminant at 0.0001–0.001%	10/10	2/10	0/10	8/10
False positive at 0.01–0.1%	0	7	1	1
False positive at 0.001–0.01%	0	1	1	1
Precision/recall	100/87	77/67.5	92/50	94/67

TABLE 3 | Bhattacharyya coefficient comparing abundance estimates to expected values.

Data set	DERSI	DADA2	UNOISE3	VSEARCH
mock 12	99.78	99.61	99.79	87.77
mock 16	96.24	96.00	96.29	91.12
mock 23	98.86	98.45	98.75	98.75

TABLE 4 | Speed in Seconds of Each Algorithm on four data sets.

Data set	Sequence reads	DERSI	DADA2	UNOISE3	VSEARCH
mock23	329,358	13	316	15	5
mock16	592,231	22	427	60	11
mock12	2,040,485	48	813	93	29
Goodrich	467,643,460	7,548	12,387	21,080	847

appear only at the two lowest concentrations. All four algorithms find the two most abundant contaminants (*Enterococcus hirae* and *Anaerostipes caccae*). Whereas, DERSI finds 10, VSEARCH 8, DADA2 two and UNOISE3 0 of the low level contaminants.

As shown in rows 9 and 10 of **Table 2**, DERSI identifies 0, UNOISE3 2, VSEARCH 2, and DADA2 eight ASVs or OTUs that have no known match at 100% and meet the criteria for false positives.

The calculated precision and recall show that DERSI has the highest precision and the highest recall of the four algorithms. In fact, DERSI achieves 100% precision results on the Mock 12 for ASV identification, and for recall, DERSI misses only six of the lowest abundance V4s that appear to be actually missing from the input sequence data.

Abundance Analysis

In **Table 3**, we present a summary of the accuracy of all four algorithms in correctly identifying the known abundance across the full mock 12, 16 and mock 23 data sets. Expected values were created by examining both copy numbers of the 16S gene, and whether the bacterial genomes contained multiple copies, some of which may be variant within the V4 region.

To rigorously compare the results for each algorithm to the expect values we applied the Bhattacharyya coefficient. The Bhattacharyya coefficients computed for each algorithm compare the abundances to the expected values for the input mix of bacteria over the entire data set; higher scores are better. Details of abundances for each individual bacterium in each data set can be found in **Supplementary Material**. VSEARCH does not perform as well as the other algorithms, likely due to the OTU approach to grouping. The accurate performance of DERSI is nearly identical to UNOISE3 exceeding UNOISE3 by only 0.02 overall, an amount that is statistically insignificant. DERSI does very slightly better compared to DADA2 in reproducing expected abundances, but there are only very slight variations among the three algorithms.

Benchmark for Speed

We compared our algorithm, DERSI, to DADA2, UNOISE3 and VSEARCH on four data sets using the same laptop and present the results in **Table 4**. The size of each data set is given by the number of V4 sequence reads. One of the known advantages of the OTU algorithm VSEARCH is its speed, and indeed VSEARCH shows the best performance across all data sets. Of the ASV methods, our algorithm DERSI was the most rapid. We note that UNOISE3 is initially faster than DADA2 but loses this advantage for the largest data set. The UNOISE3 denoising algorithm itself is a very rapid step but outputs only a list of ASVs without abundances. The step to determine abundances in the USEARCH package is much slower than the error correction but is included in the measure since DERSI outputs both a list of ASVs and their abundances as do DADA2 and VSEARCH. We conclude that DERSI offers a speed advantage across a broad range of data set sizes.

Phenotypic Prediction

Since DERSI was designed and optimized for accuracy in identifying ASVs and their abundance, it is desirable to show that this approach is still able to support phenotypic prediction that relies on all the data for each sample as a whole. There are no phenotypes associated with mock communities so to examine the effectiveness of DERSI on experimental data of unknown bacterial composition, but known sample phenotype, we reanalyzed four published data sets to compare phenotypic predictions to published results. All results are shown as

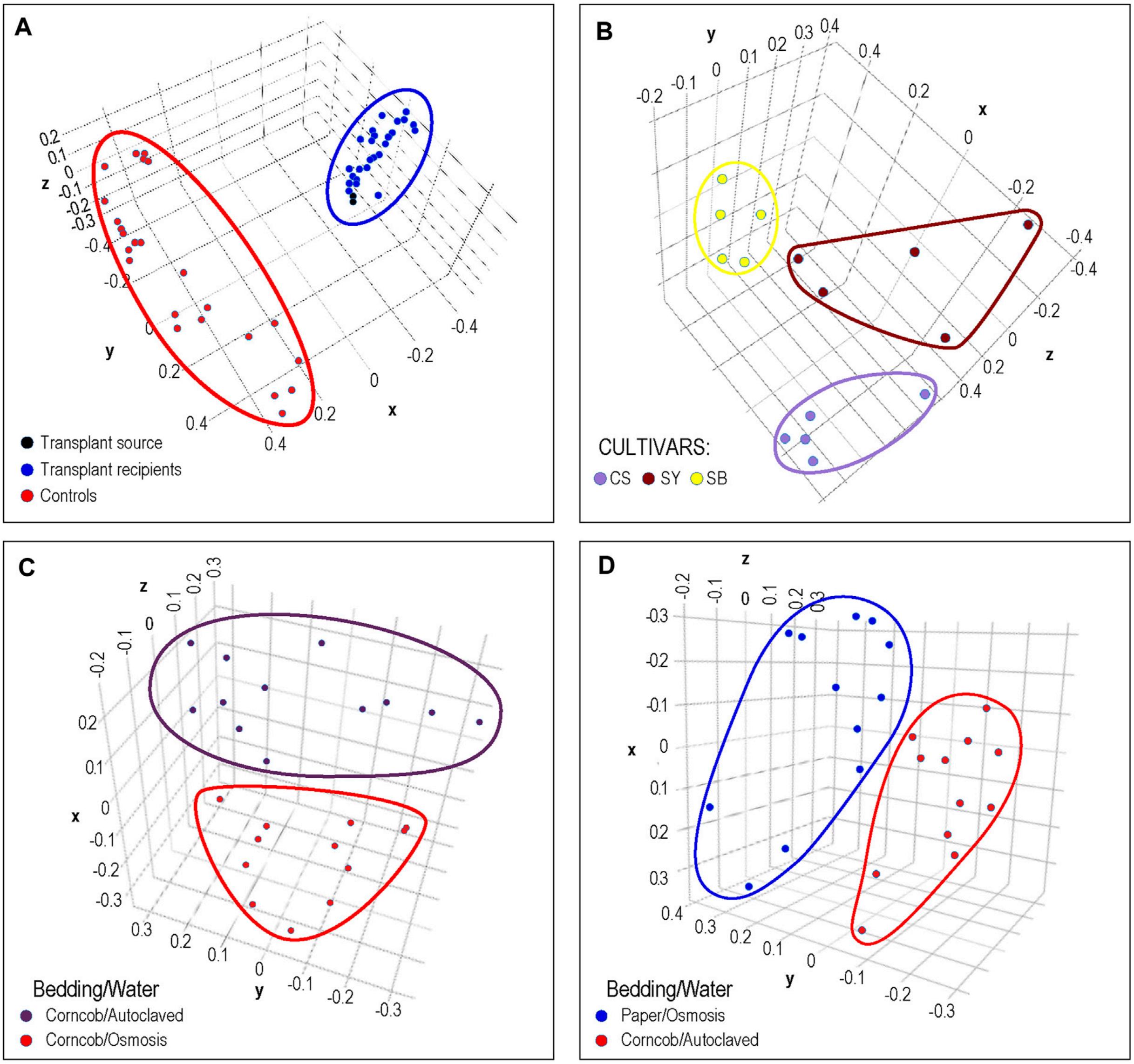


FIGURE 4 | Phenotypic Analysis: Each dot represents a sample from previously published microbiome data that was processed by DERSI into a set of ASVs. These were then normalized and plotted using PCA. The axes represent the first three principal components. In **(A)**, we the results of a chicken cecal microbiome transplantation experiment. Clear separation was achieved between controls and the transplant recipients whose microbiomes cluster with their donors. In **(B)**, we show a grape microbiome experiment; each cultivar sample was a small bunch of grapes collected from the Alpine Italian Vineyard. Each cultivar is linearly separable, exceeding the results in the original publication in which two of the three cultivars overlapped. In **(C)**, we show the impact of choice of water purification method on mouse microbiomes, the two groups of microbiomes are separable. In **(D)** we show the separation of microbiomes of mice using paper bedding with osmosis purified water in blue, to microbiomes of mice using corncob bedding and autoclaved water. Although the separation is relatively narrow, the distance between groups does exceed that of the original publication. In all of these cases, DERSI's output matched or exceeded that of previously published results showing that the method does support phenotypic analysis.

scatter plots using the first three principal components produced by PCA.

In **Figure 4A**, the results of cecal transplants between breeds of chickens are presented. One group (shown in red) represents sham transplants using saline solution and are the same breed as the transplant recipients. The other group shows two cecal transplant donor microbiomes in black from a different chicken breed than sham and actual donors, and the transplant recipients in blue. It can clearly be seen that the microbiome of the recipients is very similar to the donor microbiomes with which they group. This matches the results shown in the original publication.

In **Figure 4B**, we show the normalized results from DERSI on the grape cultivar microbiome data (Mezzasalma et al., 2018). The microbiomes from the three cultivars are linearly separable. Our analysis shows that our increased accuracy for sequence variants and abundance clearly also supports effective phenotype prediction since it readily separates the microbiomes for all samples from the three different cultivars in the Italian Alpine Vineyard, whereas in the initial published analysis only one cultivar (Sauvignon Blanc, yellow) could be cleanly separated from the other two.

In **Figures 4C,D**, we show the results of phenotypic prediction on two microbiome datasets from an analysis of the effects of water decontamination method and choice of bedding material on mice (Bidot et al., 2018). In C, a PCA of the microbiome composition for two phenotypic groups, mice who shared the same bedding type but whose water was purified either by osmosis or by autoclaving. The two water phenotypes clearly assort from each other and are represented by two distinct groups. This meets or slightly exceeds the separation shown in the original publication. Similarly, in the panel at right we show mice who used paper bedding with osmosis purified water compared to the microbiomes of mice who used corncob bedding and autoclaved water. The separation between the two groups appears to surpass that in Bidot et al. (2018).

Taken together, all four results support the conclusion that the accuracy for sequence variants and abundance shown for DERSI also supports quality phenotypic analysis that can match or exceed published results.

DISCUSSION

To our knowledge, this is the first demonstration of the use of latent space with a deep learning algorithm to make sequence identifications, and moreover to be able to distinguish closely-related sequence variations with single base accuracy. The first step is a gradient descent training to form the latent space. This 10-dimensional space is an embedding of all of the sequences that reflects their phylogenetic distance and is far more information-rich than a typical two-dimensional phylogenetic tree while still having reduced dimensionality. It becomes a reference component of the method and is not repeated when microbiome samples are being analyzed. This training need only be redone when the collection of known V4 sequences has grown to the extent that the reference space needs to be refreshed.

The process of training the convolutional neural network similarly creates a tool that becomes a stable part of the method and is not repeated with each analysis run. Any machine learning effort can be divided into two phases: 1) a design and construction phase using training data and 2) a deployment phase for predicting on new data. The first phase is where the permanent structure and elements of the tool are decided. For DADA2, VSEARCH, and UNOISE3, the equivalent process consists of algorithmic structure but without any pre-trained results; they must establish transitions probabilities (DADA2) or k-mer features (VSEARCH, UNOISE3) with every analysis run. In the example of a neural network, including our convolutional neural network, an architecture is chosen by a human before any processing of data and then the network weights are fixed by the training procedure. This construction phase is done only once for DERSI. Thereafter the NN runs in its deployment phase, and it is standard procedure to assess neural network speed and performance including only its inference on new data, as we have done on the four data sets for the benchmark. While there is additional time devoted to the original development of the tool, for the runs on data, DERSI was clearly the most rapid algorithm.

Some run time choices may also affect speed. We provide end to end processing time for speed tests, from fastq files to OTU/ASV abundance output, since this is the time a user will experience running these algorithms. To be consistent with DADA2, we employed denoising on a sample-by-sample basis for VSEARCH, USEARCH UNOISE3, and DERSI (rather than the much faster pooling of all reads into a single “super sample”). The sample-by-sample approach helps preserve ASVs that might otherwise be folded into close and more abundant variants. On the other hand, besides speed, the pooling approach does have the benefit of suppressing false positives (along with some true positives just mentioned), on average elevating signal over noise. Some advantages and disadvantage of pooling of samples and sample-by-sample analysis are further discussed by Edgar, (2016). The choice can largely be experimentally driven. Both DERSI and UNOISE3/USEARCH could utilize the pooling of samples instead of the single sample approach we have used for the benchmarks here, and that would likely greatly enhance the speed of both relative to the other algorithms.

We also note that our DERSI process is single threaded. The other three algorithms are implemented in their software as multi-threaded, so that much of their process can run in parallel. There is no algorithmic barrier to multi-threading the DERSI algorithm and that would also further enhance its speed.

Programming language choice and operating system may also impact speed. Marrizoni et al. (2020) compared several microbiome pipelines on two different operating systems and found some differences in actual results among versions of the same pipeline available for Mac OS and Linux. Deep learning algorithms are also able to readily leverage GPUs which are fast relative to CPUs, but it is unlikely that the others used in our comparison could do so to great advantage.

Our convolutional network maps each sequence to the 10-dimensional space that has been previously optimized to capture both global and local phylogenetic sequence structure associated with a large rRNA V4 database. Even without further potential enhancements, it is this approach that enables the analysis of each V4 data set to be accomplished with excellent speed, while still providing the best available accuracy.

We are also unaware of any deep learning algorithms being integrated into methods for sequencing error correction, particularly removing sequencing noise while resolving true genomic variations. While our error correction method bears some similarity to the algorithmic approach of UNOISE3 the major difference lies in the fact that clustering to find nearest neighbors, and to seed potential ASVs occurs in the 10-dimensional latent space leveraging the locations in that space that have been assigned to each sequence by the trained CNN. The analogous step in UNOISE3 (and in VSEARCH) uses kmers to find nearest neighbors.

At the time that many mock communities were added to the Mockrobiota resource, not all bacterial species used had full genomic sequence, and a few of the bacteria used still lack complete genomic sequence. Since most bacteria have multiple copies of the 16S gene, we adjusted our expected abundances using not merely the percent of the bacteria that was used in creating the mock community, but also how the genomic copy number and the number of variants affect the expected abundance. In the few cases where the genomes are still not fully known, we used closest relatives to approximate the genome copies. While previous studies compared the abundances found among different algorithms, we did not find any prior work that utilized the genomic information about copy number and number of variants for the genomes intentionally added to the mock community. We also note that we did not attempt to correct the expected abundance percentages given that each mock community appeared to have some bacterial contaminants. There is no accurate way to know the true abundance of the contaminant. Our use of the Bhattacharyya coefficient to compare expected abundances of the intentionally added sequences to the abundances found for them by each algorithm would be expected to slightly lower the scores of all algorithms due to contaminants but would have impacted all of them equivalently so the comparisons would be expected to be valid. Since most of the algorithms performed quite well at abundance estimation, the impact of this appears to be quite small. Moreover, despite the differences among the algorithms in precision and recall, the three ASV methods are all near equally good at abundance estimation, likely because most of the differences occurred at the lowest concentration levels that would least impact the coefficient over the entire data set. While the coefficient showed a slim and likely statistically insignificant advantage for DERSI over the others, the results do clearly demonstrate that DERSI is able to at least match the best available algorithms for accurate abundance estimation.

The dimensionality of the latent space was chosen empirically by starting with three and increasing. We found 10 dimensions achieved high precision and recall, good abundance recovery and equaled or matched the best available current methods for the analysis of the microbiome mock communities. In the future if applying the method to longer sequences, it might become necessary to use a higher dimensionality for the latent space, potentially incurring somewhat higher computational overhead in the one-time training for the embedding of all know sequences of the chosen length, but should not have much impact on the mapping into the space that is a rapid step using the trained CNN that occurs when using the method on microbiome data sets.

Since Zhao et al. intended to improve phenotypic prediction without an intermediate prediction of ASVs, they leveraged deep learning to classify each individual sequence read for its likelihood to belong to a phenotypic class. Our approach was fundamentally different, although our encoder has a convolutional architecture, it is usedfor mapping sequence reads into a latent space that has reduced dimensions. The reduced dimensions of the latent space enables computational efficiencies. Our output is analogous to the separately trained word embeddings that have been a critical ingredient supporting recent advances in natural language processing (NLP). These word embeddings serve as compact representations of word usage that encode the contents of a document while reducing dimensionality and are the input for larger neural network such as BERT (Devlin et al., 2019). Our sequence embeddings are analogous in that they are trained to faithfully represent a biological sequence (instead of a word or phrase). This paper focuses primary on the quality of those embeddings, as judged by their usefulness in recovering true biological sequences in mock communities. In the future, for phenotype studies, it would be possible to develop even more powerful neural networks that leverage these embeddings further for phenotypic classification, just as BERT leverages its word embeddings to classify documents. In the current work, we have demonstrated that a simple purely linear network (i.e., Principle Component Analysis) on the output ASVs for each sample is sufficient to recover the phenotypic structure of the samples and obtains at least equivalent or slightly improved results compared to previously published work.

Moreover, creating a latent space using these methods for the full 16S sequence should enable a 16S latent space to be used with multiple convolutional neural networks, each trained to map a different variable region of the 16S gene into the full 16S latent space. This would offer a significant advancement in the ability to directly compare microbiome studies conducted by sequencing different variable regions of the 16S gene and enabling more informative meta-analysis of the underlying biology, although some caution would be warranted due to technical differences in the amplification of diverse sequences (e.g., Bukin et al., 2019; Darwish et al., 2021). As full-length 16S sequencing becomes more economical and accurate, a convolutional neural network could also be trained on the full length rather than just the variable regions.

In fact, the method should be generalizable to many types of experiments that rely on sequence identification in addition to microbiome analysis, making this a promising area for future research to fully explore the applicability of these methods to additional biological studies. For example, rather than a latent space intended for microbiome analysis, one could be created from all known sequences for any particular protein or enzyme family and applied to proteomics data. In addition, metagenomic analysis is currently very computationally burdensome and accuracy is challenging for low abundance organisms. Potentially DERSI could shift that burden away from the individual metagenomic experiments and onto the one-time creation of a very large latent space and the training of multiple deep learning algorithms. For these more demanding analyses, reduction even in the one-time computational demands of initially creating the latent space could be managed by judicious choice of the metagenomic challenge to address. For example, rather than full genomes, the system could be applied to the phylogenetic classification of metagenomic samples by training a number of individual neural nets on each of a subset of the 92 core bacterial genes identified by the UBCG pipeline (Na et al., 2018).

In conclusion, the current work demonstrates that our Deep Learning for Rapid Sequence Identification (DERSI) algorithm that combines a latent sequence space with a deep learning encoder can match or better the precision and recall of existing widely accepted methods for microbiome analysis while performing at greater speed. Potential exists for further enhancing the speed of the algorithm, and of generalizing the method to more types of data including metagenomics and proteomics.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: The Mock 12, Mock 16 and Mock 23 datasets can be found at http://caporaso-lab.github.io/mockrobiota/. The grape microbiome data sets can be found in the EBI metagenomics portal (https://www.ebi.ac.uk/metagenomics/) under the accession code PRJEB25720 (ERP107664). The remaining microbiome data sets are also from EBI and can be found at that following links: Chicken: PRJEB46338 https://www.ebi.ac.uk/ena/browser/view/PRJEB46338?show=reads Mouse: PRJNA453789 https://www.ebi.ac.uk/ena/browser/view/PRJNA453789?show=reads Twins: PRJEB13747 https://www.ebi.ac.uk/ena/browser/view/PRJEB13747?show=reads.

AUTHOR CONTRIBUTIONS

LH conceived the initial project idea. JB provided the mathematical insights, overall mathematical approach and algorithmic development. LH and JB designed the experiments, analyzed the experimental output and validated the results. KS provided data science expertise, review of analysis, management and feedback on scientific results and prioritization. JC provided technical and software support and resolution for technical issues. JB, LH and KS wrote, reviewed and finalized the manuscript.

FUNDING

All funding for this project was provided by Digital Infuzion, Inc. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article or the decision to submit it for publication.

ACKNOWLEDGMENTS

We recognize Hemant Virkar, President and CEO of Digital Infuzion, Inc. and Nancy Nurthen (formerly at Digital Infuzion, Inc.) for their support of this project. We thank Amy Buck for graphics work and Bonnie Werner for technical assistance with the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.871256/full#supplementary-material

REFERENCES

Allaband, C., McDonald, D., Vázquez-Baeza, Y., Minich, J. J., Tripathi, A., Brenner, D. A., et al. (2019). Microbiome 101: Studying, Analyzing, and Interpreting Gut Microbiome Data for Clinicians. *Clin. Gastroenterol. Hepatol.* 17 (2), 218–230. Epub 2018 Sep 18. PMID: 30240894; PMCID: PMC6391518. doi:10.1016/j.cgh.2018.09.017

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic Local Alignment Search Tool. *J. Mol. Biol.* 215 (3), 403–410. PMID: 2231712. doi:10.1016/S0022-2836(05)80360-2

Amir, A., McDonald, D., Navas-Molina, J. A., Kopylova, E., Morton, J. T., Zech Xu, Z., et al. (2017). Deblur Rapidly Resolves Single-Nucleotide Community Sequence Patterns. *mSystems* 2 (2), e00191–16. PMID: 28289731; PMCID: PMC5340863. doi:10.1128/mSystems.00191-16

Asgari, E., Garakani, K., McHardy, A. C., and Mofrad, M. R. K. (2018). MicroPheno: Predicting Environments and Host Phenotypes from 16S rRNA Gene Sequencing Using a K-Mer Based Representation of Shallow Sub-samples. *Bioinformatics* 34 (13), i32–i42. Erratum in: Bioinformatics. 2019 Mar 15;35(6):1082. PMID: 29950008; PMCID: PMC6022683. doi:10.1093/bioinformatics/bty296

Auslander, N., Gussow, A. B., Benler, S., Wolf, Y. I., and Koonin, E. V. (2020). Seeker: Alignment-free Identification of Bacteriophage Genomes by Deep Learning. *Nucleic Acids Res.* 48 (21), e121. PMID: 33045744; PMCID: PMC7708075. doi:10.1093/nar/gkaa856

Barlow, G., Lin, E. A., and Mathur., R. (2018). "An Overview of the Roles of the Gut Microbiome in Obesity and Diabetes," in *Nutritional and Therapeutic Interventions for Diabetes and Metabolic Syndrom*. Second Edition, 65–91.

Bhattacharyya, A. (1943). On a Measure of Divergence between Two Statistical Populations Defined by Their Probability Distributions. *Bull. Calcutta Math. Soc.* 35, 99–109.

Bidot, W. A., Ericsson, A. C., and Franklin, C. L. (2018). Effects of Water Decontamination Methods and Bedding Material on the Gut Microbiota. *PLoS One* 13 (10), e0198305. PMID: 30359379; PMCID: PMC6201873. doi:10.1371/journal.pone.0198305

Bokulich, N. A., Kaehler, B. D., Rideout, J. R., Dillon, M., Bolyen, E., Knight, R., et al. (2018). Optimizing Taxonomic Classification of Marker-Gene Amplicon Sequences with QIIME 2's Q2-Feature-Classifier Plugin. *Microbiome* 6, 90. doi:10.1186/s40168-018-0470-z

Bokulich, N. A., Rideout, J. R., Mercurio, W. G., Shiffer, A., Wolfe, B., Maurice, C. F., et al. (2016). Mockrobiota: a Public Resource for Microbiome Bioinformatics Benchmarking. *mSystems* 1 (5), e00062–16. doi:10.1128/mSystems.00062-16

Bukin, Y. S., Galachyants, Y. P., Morozov, I. V., Bukin, S. V., Zakharenko, A. S., and Zemskaya, T. I. (2019). The Effect of 16S rRNA Region Choice on Bacterial Community Metabarcoding Results. *Sci. Data* 6, 190007. Sci Data. 2022 Mar 17;9(1):94. PMID: 30720800; PMCID: PMC6362892. doi:10.1038/sdata.2019.7

Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J., and Holmes, S. P. (2016). DADA2: High-Resolution Sample Inference from Illumina Amplicon Data. *Nat. Methods* 13, 581–583. doi:10.1038/nmeth.3869

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., et al. (2009). BLAST+: Architecture and Applications. *BMC Bioinforma.* 10, 421. doi:10.1186/1471-2105-10-421

Caruso, V., Song, X., Asquith, M., and Karstens, L. (2019). Performance of Microbiome Sequence Inference Methods in Environments with Varying

Biomass. *mSystems* 4 (1), e00163–18. PMID: 30801029; PMCID: PMC6381225. doi:10.1128/mSystems.00163-18

Cho, I., and Blaser, M. J. (2012). The Human Microbiome: at the Interface of Health and Disease. *Nat. Rev. Genet.* 13 (4), 260–270. PMID: 22411464: PMCID: PMC3418802.G. doi:10.1038/nrg3182

Darwish, N., Shao, J., Schreier, L. L., and Proszkowiec-Weglarz, M. (2021). Choice of 16S Ribosomal RNA Primers Affects the Microbiome Analysis in Chicken Ceca. *Sci. Rep.* 11 (1), 11848. PMID: 34088939; PMCID: PMC8178357. doi:10.1038/s41598-021-91387-w

Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). "Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding,". Long and Short Papers in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA (Association for Computational Linguistics), 4171–4186.Vol. 1

Ditzler, G., Polikar, R., and Rosen, G. (2015). Multi-Layer and Recursive Neural Networks for Metagenomic Classification. *IEEE Trans. Nanobiosci.* 14 (6), 608–616. doi:10.1109/TNB.2015.2461219

Edgar, R. C. (2004). Local Homology Recognition and Distance Measures in Linear Time Using Compressed Amino Acid Alphabets. *Nucleic Acids Res.* 32 (1), 380–385. PMID: 14729922; PMCID: PMC373290. doi:10.1093/nar/gkh180

Edgar, R. C. (2010). Search and Clustering Orders of Magnitude Faster Than BLAST. *Bioinformatics* 26 (19), 2460–2461. Epub 2010 Aug 12. PMID: 20709691. doi:10.1093/bioinformatics/btq461

Edgar, R. C. (2013). UPARSE: Highly Accurate OTU Sequences from Microbial Amplicon Reads. *Nat. Methods* 10 (10), 996–998. Epub 2013 Aug 18. PMID: 23955772. doi:10.1038/nmeth.2604

Edgar, R. C. (2016). UNOISE2: Improved Error-Correction for Illumina 16S and ITS Amplicon Sequencing. *bioRxiv*, 081257. doi:10.1101/081257

Furusawa, C., Tanabe, K., Ishii, C., Kagata, N., Tomita, M., and Fukuda, S. (2021). Decoding Gut Microbiota by Imaging Analysis of Fecal Samples. *iScience* 24 (12), 103481. doi:10.1016/j.isci.2021.103481

García-Jiménez, B., Muñoz, J., Cabello, S., Medina, J., and Wilkinson, M. D. (2021). Predicting Microbiomes through a Deep Latent Space. *Bioinformatics* 37 (10), 1444–1451. PMID: 33289510; PMCID: PMC8208755. doi:10.1093/bioinformatics/btaa971

Gilbert, J. A., Blaser, M. J., Caporaso, J. G., Jansson, J. K., Lynch, S. V., and Knight, R. (2018). Current Understanding of the Human Microbiome. *Nat. Med.* 24 (4), 392–400. PMID: 29634682; PMCID: PMC7043356. doi:10.1038/nm.4517

Glendinning, L., Chintoan-Uta, C., Stevens, M. P., and Watson, M. (2022). Effect of Cecal Microbiota Transplantation between Different Broiler Breeds on the Chick Flora in the First Week of Life. *Poult. Sci.* 101 (2), 101624. Epub 2021 Nov 28. PMID: 34936955; PMCID: PMC8704443. doi:10.1016/j.psj.2021.101624

Gohl, D. M., Vangay, P., Garbe, J., MacLean, A., Hauge, A., Becker, A., et al. (2016). Systematic Improvement of Amplicon Marker Gene Methods for Increased Accuracy in Microbiome Studies. *Nat. Biotechnol.* 34 (9), 942–949. Epub 2016 Jul 25. PMID: 27454739. doi:10.1038/nbt.3601

Goodrich, J. K., Davenport, E. R., Beaumont, M., Jackson, M. A., Knight, R., Ober, C., et al. (2016). Genetic Determinants of the Gut Microbiome in UK Twins. *Cell Host Microbe* 19 (5), 731–743. PMID: 27173935; PMCID: PMC4915943. doi:10.1016/j.chom.2016.04.017

Guo, J., Bolduc, B., Zayed, A. A., Varsani, A., Dominguez-Huerta, G., Delmont, T. O., et al. (2021). VirSorter2: A Multi-Classifier, Expert-Guided Approach to Detect Diverse DNA and RNA Viruses. *Microbiome.* 9 (1), 37. doi:10.1186/s40168-020-00990-y

Jandhyala, S. M., Talukdar, R., Subramanyam, C., Vuyyuru, H., Sasikala, M., and Nageshwar Reddy, D. (2015). Role of the Normal Gut Microbiota. *World J. Gastroenterol.* 21 (29), 8787–8803. PMID: 26269668; PMCID: PMC4528021. doi:10.3748/wjg.v21.i29.8787

Kopylova, E., Navas-Molina, J. A., Mercier, C., Xu, Z. Z., Mahé, F., He, Y., et al. (2016). Open-Source Sequence Clustering Methods Improve the State of the Art. *mSystems* 1 (1), e00003–15. PMID: 27822515; PMCID: PMC5069751. doi:10.1128/mSystems.00003-15

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* 25, 1097–1105. doi:10.5555/2999134.2999257

Levy, M., Kolodziejczyk, A. A., Thaiss, C. A., and Elinav, E. (2017). Dysbiosis and the Immune System. *Nat. Rev. Immunol.* 17 (4), 219–232. Epub 2017 Mar 6. PMID: 28260787. doi:10.1038/nri.2017.7

Long, J., Shelhamer, E., and Darrell, T. (2015). "Fully Convolutional Networks for Semantic Segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, 7-12 June 2015, 3431–3440. doi:10.1109/cvpr.2015.7298965

Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. (2016). "Fully Convolutional Neural Networks for Remote Sensing Image Classification," in 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 5071–5074. doi:10.1109/IGARSS.2016.7730322

Marizzoni, M., Gurry, T., Provasi, S., Greub, G., Lopizzo, N., Ribaldi, F., et al. (2020). Comparison of Bioinformatics Pipelines and Operating Systems for the Analyses of 16S rRNA Gene Amplicon Sequences in Human Fecal Samples. *Front. Microbiol.* 11, 1262. PMID: 32636817; PMCID: PMC7318847. doi:10.3389/fmicb.2020.01262

Mezzasalma, V., Sandionigi, A., Guzzetti, L., Galimberti, A., Grando, M. S., Tardaguila, J., et al. (2018). Geographical and Cultivar Features Differentiate Grape Microbiota in Northern Italy and Spain Vineyards. *Front. Microbiol.* 9, 946. doi:10.3389/fmicb.2018.00946

Na, S. I., Kim, Y. O., Yoon, S. H., Ha, S. M., Baek, I., and Chun, J. (2018). UBCG: Up-To-Date Bacterial Core Gene Set and Pipeline for Phylogenomic Tree Reconstruction. *J. Microbiol.* 56 (4), 280–285. Epub 2018 Feb 28. PMID: 29492869. doi:10.1007/s12275-018-8014-6

Nearing, J. T., Douglas, G. M., Comeau, A. M., and Langille, M. G. I. (2018). Denoising the Denoisers: an Independent Evaluation of Microbiome Sequence Error-Correction Approaches. *PeerJ* 6, e5364. PMID: 30123705; PMCID: PMC6087418. doi:10.7717/peerj.5364

Prodan, A., Tremaroli, V., Brolin, H., Zwinderman, A. H., Nieuwdorp, M., and Levin, E. (2020). Comparing Bioinformatic Pipelines for Microbial 16S rRNA Amplicon Sequencing. *PLoS One* 15 (1), e0227434. PMID: 31945086; PMCID: PMC6964864. doi:10.1371/journal.pone.0227434

Ren, J., Song, K., Deng, C., Ahlgren, N. A., Fuhrman, J. A., Li, Y., et al. (2020). Identifying Viruses From Metagenomic Data Using Deep Learning. *Quant. Biol.* 8 (1), 64–77. doi:10.1007/s40484-019-0187-4

Rognes, T., Flouri, T., Nichols, B., Quince, C., and Mahé, F. (2016). VSEARCH: a Versatile Open-Source Tool for Metagenomics. *PeerJ* 4, e2584. doi:10.7717/peerj.2584

Schirmer, M., Ijaz, U. Z., D'Amore, R., Hall, N., Sloan, W. T., and Quince, C. (2015). Insight into Biases and Sequencing Errors for Amplicon Sequencing with the Llumina MiSeq Platform. *Nucleic Acids Res.* 43, e37. doi:10.1093/nar/gku1341

Schloss, P. D., Westcott, S. L., Ryabin, T., Hall, J. R., Hartmann, M., Hollister, E. B., et al. (2009). Introducing Mothur: Open-Source, Platform-independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Appl. Environ. Microbiol.* 75, 7537–7541. doi:10.1128/AEM.01541-09

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., et al. (2016). Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature* 529 (7587), 484–489. PMID: 26819042. doi:10.1038/nature16961

Smith, T. F., and Waterman, M. S. (1981). Identification of Common Molecular Subsequences. *J. Mol. Biol.* 147 (1), 195–197. PMID: 7265238. doi:10.1016/0022-2836(81)90087-5

Šošic, M., and Šikic, M. (2017). Edlib: a C/C ++ Library for Fast, Exact Sequence Alignment Using Edit Distance. *Bioinformatics* 33 (9), 1394–1395. PMID: 28453688; PMCID: PMC5408825. doi:10.1093/bioinformatics/btw753

Thursby, E., and Juge, N. (2017). Introduction to the Human Gut Microbiota. *Biochem. J.* 474 (11), 1823–1836. PMID: 28512250; PMCID: PMC5433529. doi:10.1042/BCJ20160510

Tremblay, J., and Yergeau, E. (2019). Systematic Processing of Ribosomal RNA Gene Amplicon Sequencing Data. *Gigascience.* 8 (12), giz146. doi:10.1093/gigascience/giz146

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention Is All You Need. *Adv. neural Inf. Process. Syst.* 30, 6000–6010. doi:10.48550/ARXIV.1706.03762

Yilmaz, P., Parfrey, L. W., Yarza, P., Gerken, J., Pruesse, E., Quast, C., et al. (2014). The SILVA and "All-species Living Tree Project (LTP)" Taxonomic Frameworks. *Nucleic Acids Res.* 42, D643–D648. doi:10.1093/nar/gkt1209

Zhao, Z., Woloszynek, S., Agbavor, F., Mell, J. C., Sokhansanj, B. A., and Rosen, G. L. (2021). Learning, Visualizing and Exploring 16S rRNA Structure Using an Attention-Based Deep Neural Network. *PLoS Comput. Biol.* 17 (9), e1009345. PMID: 34550967; PMCID: PMC8496832. doi:10.1371/journal.pcbi.1009345

Conflict of Interest: JB, KS, JC and LH were employed by Digital Infuzion, Inc.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Borgman, Stark, Carson and Hauser. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

METHODS
published: 22 July 2022
doi: 10.3389/fbinf.2022.918853

microTrait: A Toolset for a Trait-Based Representation of Microbial Genomes

Ulas Karaoz[1]* and *Eoin L. Brodie*[1,2]

[1]Earth and Environmental Sciences, Lawrence Berkeley National Laboratory, Berkeley, CA, United States, [2]Department of Environmental Science, Policy and Management, University of California, Berkeley, CA, United States

Remote sensing approaches have revolutionized the study of macroorganisms, allowing theories of population and community ecology to be tested across increasingly larger scales without much compromise in resolution of biological complexity. In microbial ecology, our remote window into the ecology of microorganisms is through the lens of genome sequencing. For microbial organisms, recent evidence from genomes recovered from metagenomic samples corroborate a highly complex view of their metabolic diversity and other associated traits which map into high physiological complexity. Regardless, during the first decades of this *omics* era, microbial ecological research has primarily focused on taxa and functional genes as ecological units, favoring breadth of coverage over resolution of biological complexity manifested as physiological diversity. Recently, the rate at which provisional draft genomes are generated has increased substantially, giving new insights into ecological processes and interactions. From a genotype perspective, the wide availability of genome-centric data requires new data synthesis approaches that place organismal genomes center stage in the study of environmental roles and functional performance. Extraction of ecologically relevant traits from microbial genomes will be essential to the future of microbial ecological research. Here, we present *microTrait*, a computational pipeline that infers and distills ecologically relevant traits from microbial genome sequences. *microTrait* maps a genome sequence into a trait space, including discrete and continuous traits, as well as simple and composite. Traits are inferred from genes and pathways representing energetic, resource acquisition, and stress tolerance mechanisms, while genome-wide signatures are used to infer composite, or life history, traits of microorganisms. This approach is extensible to any microbial habitat, although we provide initial examples of this approach with reference to soil microbiomes.

Keywords: functional traits, functional guilds, ecological strategy, trait-based model, profile hidden markov model, microbial genome, fitness traits, trait inference workflow

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Bruno Koshin Vázquez Iha,
University of São Paulo, Brazil
Phil B. Pope,
Norwegian University of Life Sciences,
Norway

****Correspondence:***
Ulas Karaoz
ukaraoz@lbl.gov

Specialty section:
This article was submitted to
Genomic Analysis,
a section of the journal
Frontiers in Bioinformatics

Received: *12 April 2022*
Accepted: *20 June 2022*
Published: *22 July 2022*

Citation:
Karaoz U and Brodie EL (2022)
microTrait: A Toolset for a Trait-Based
Representation of Microbial Genomes.
Front. Bioinform. 2:918853.
doi: 10.3389/fbinf.2022.918853

IMPORTANCE

The rapid adoption of high-throughput microbial sequencing is leading to accumulation of microbial genomes at an ever-increasing rate. These genomes represent instances from not only isolated microbes but also microbial populations in their native environmental context as metagenome-assembled genomes (MAGs) or single-cell amplified genomes (SAGs). We believe that an ability to efficiently predict ecological traits directly from primary sequence data is a necessary interface between microbial *omics* information and trait-based microbial ecology, and success here will significantly advance our ability to uncover generalizable features of microbiomes and their environmental context. To streamline the process of going from genome sequences to putative ecological traits, we developed *microTrait,* a set of tools to efficiently discover and distill the trait-based representation of a microbial genome.

INTRODUCTION

Linking microbiome structure and dynamics to ecosystem functioning globally in a predictive way and in face of global change has been a long-standing goal of microbial ecology (Finlay et al., 1997; Prosser et al., 2007; Van Der Heijden et al., 2008; Todd-Brown et al., 2012; Bier, Bernhardt et al., 2015). Efforts towards this goal traditionally included taxon-centric measurement approaches (Thompson et al., 2017; Ramirez et al., 2018) (Madin et al., 2020). Genetic, physiological, and ecological characterization of cultured isolates provided links between specific taxa and ecosystem processes like contributions to elemental and nutrient cycles, and biomass production. With the commoditization of high-throughput sequencing of taxonomic marker sequences, much effort in taxon-centric approaches shifted to extrapolating what is learned from representative isolates in the lab to their phylogenetic nearest neighbors detected with environmental community sequencing (Langille et al., 2013; Asshauer et al., 2015). Such approaches to infer functional groups via phylogenetic markers inherently assume strong phylogenetic conservation of microbial traits. Furthermore, without any whole-genome data, they are limited to taxa with cultured isolates.

Microbial-biogeochemical models are crucial tools in linking microbiome dynamics, environmental responses, and ecosystem processes across scales. Wide-spread availability of taxon-centric microbial measurements have naturally popularized taxon-centric models including few species or functional groups dominant at the local scale of interest. The upward scalability of such models would be limited given the fact that no single taxa would dominate at larger scales and with a limited number of parameter sets, the model would have poor adaptive capability both across scales and environmental conditions. Moreover, trying to approach the complexity of real systems at larger scales by adding more taxa or functional groups lead to increasingly complex models with a continuous demand for more parameters. Given these limitations of taxon-centric approaches in modeling the diversity and activity of microbes globally and with changing environmental conditions, trait-based representation of microbes is becoming increasingly popular.

Trait-based approaches represent an intermediate approach to modeling complex populations while also preserving key mechanistic properties that determine fitness in dynamic systems. The trait-based framework represents microbes with traits that can be summarized by few parameters and that are constrained by environmentally-dependent trade-offs. These approaches were developed in the field of plant ecology (Westoby and Wright 2006; Ackerly and Cornwell 2007), and have more recently been applied within microbial ecology at various scales, including global oceans and terrestrial environments (Follows et al., 2007; Allison 2012; Bouskill et al., 2012). The main underlying assumption is that combination of traits determines physiological performance which influences individual fitness and life history evolution. By abandoning the taxon concept, the trait-based framework strives to achieve a succinct description of the microbial communities with few essential communities, avoiding the complexity trap of taxon-centric modeling approaches. The challenge with this approach is to identify the key properties or traits of members of microbial communities and how these traits are regulated or trade-off against other traits, and to use this information to parameterize or constrain the functional potential of the modeled communities.

Traits may be identified through *'omic* approaches (e.g. potential to produce or the detected activity of an extracellular enzyme, the genes for a specific metabolic pathway, the genomic capacity to replicate rapidly etc) or through physiological studies (e.g. enzyme, substrate uptake or growth kinetics, cell surface area, biomass stoichiometry, composition of storage pools etc.) or they may be inferred by manipulation experiments such as stable-isotope tracing with substrates at various concentrations to determine relative affinities. The paradigm shift from a taxa-to a trait-centric representation of microbiomes is partly stimulated by the wide-use of *omic* technologies to illuminate the functional potential of environmental microbial communities and their interactions with each other, higher organisms, and their environment (Sharon and Banfield 2013; Anantharaman et al., 2016; Gupta et al., 2016; Sangwan et al., 2016; Woodcroft et al., 2018). In particular, focusing on genome rather genes as ecological units makes the incorporation of many concepts from ecological and evolutionary theory into models possible therefore increase the value of the *omic* data for trait-based modeling (Prosser 2015). The rate at which isolate genomes, single-cell assembled genomes (SAGs) and metagenome-assembled genomes (MAGs) are being generated provide an unprecedented resource to study patterns in fitness trait conservation, trait linkage (i.e. co-occurrence patterns of traits within ecological units), trait trade-offs, and trait-environment relationships across scales. This continuous stream of microbial genomes necessitates development of computational tools that can efficiently and robustly extract potential traits from genome sequences.

Currently, the methods used to infer functional traits from genome sequences include 1) pairwise sequence alignments and database search (Shaffer et al., 2020), 2) statistical learning methods (Feldbauer et al., 2015; Weimann et al., 2016), and 3) phylogenetic inference (Goberna and Verdu 2016). Homologous inference from sequence alignments with tools like BLAST (Altschul et al., 1990), USearch (Edgar 2010), or DIAMOND (Buchfink et al., 2015) have large memory requirements and long run times, which makes these methods challenging to scale for a typical user to thousands of genome sequences. In addition, for the detection of remote homologs, the sensitivity of alignment-based methods is lower than the profile methods (Brenner et al., 1998). Statistical learning methods to predict microbial traits depend on the availability of extensive training sets to establish genotype-phenotype relationships. Such data exist only for a very limited set of core phenotypes and therefore the resulting models, while they can be highly accurate, offer a narrow view of the microbial trait space (Yabuuchi 2001; Ruan 2013). Phylogeny-

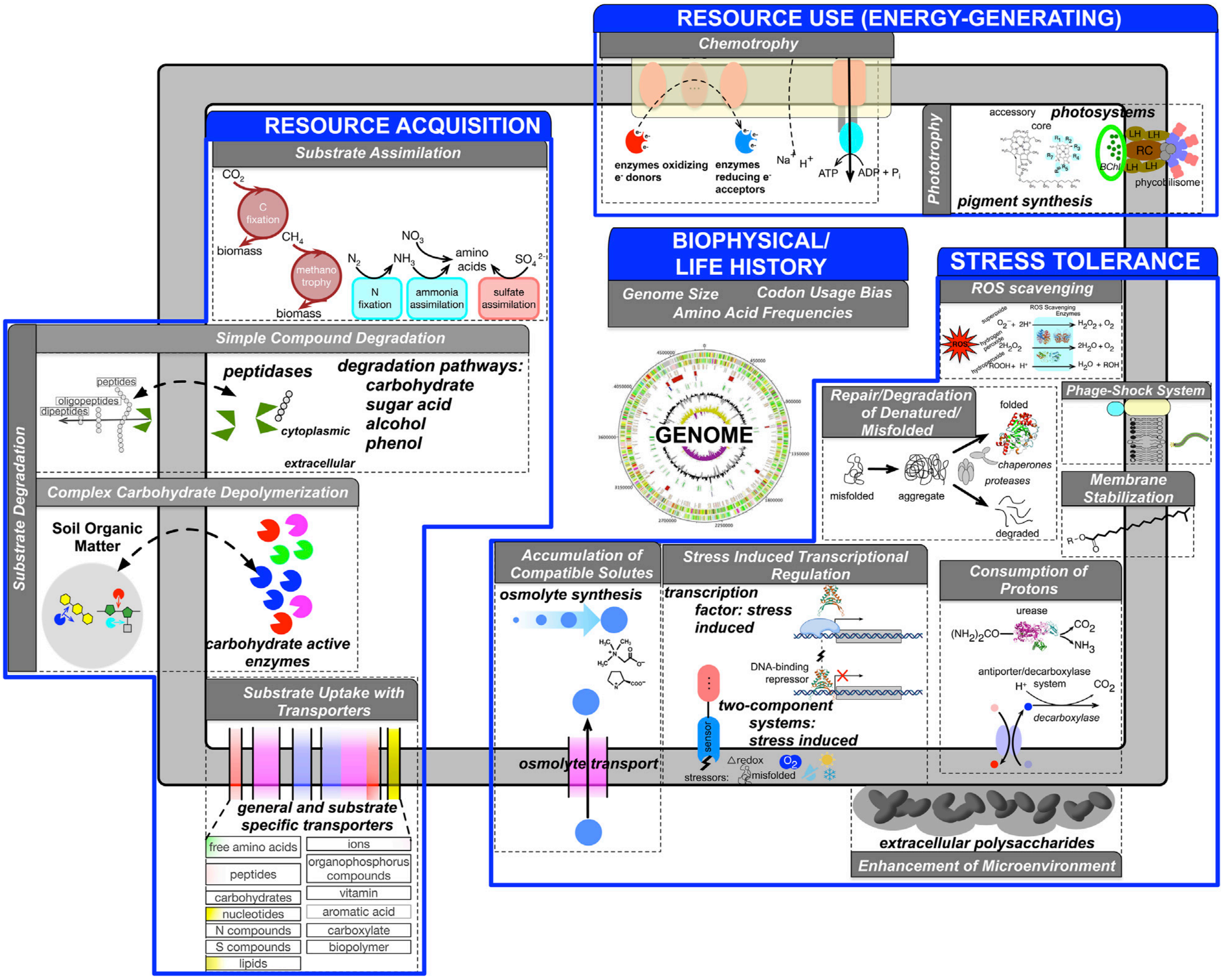


FIGURE 1 | Conceptual overview of genome-derivable traits (gray boxes) underlying ecological strategies (blue boxes) represented in *microTrait* based on literature surveys. For each trait, genomic features are indicated. **Supplementary Table S1** provides full details for the *microTrait* hierarchy. **Supplementary Table S8** lists references for genomic features underlying ecological traits.

based methods predict missing trait values of new genomes based on the traits of their evolutionary relatives. While phylogenetic conservatism of certain traits has been documented for bacteria and archaea, prokaryotic traits of ecological relevance have overall weak phylogenetic signal (Martiny et al., 2013). In addition, as the bulk of the current information on phenotypes are centered around organisms of biotechnological and medical interest, the accuracy of the phylogenetic trait prediction remains low (Goberna and Verdu 2016).

To fill this need, we developed an R package, *microTrait,* that provides a conceptual framework and associated pipelines to translate a microbial genome into a suite of potential fitness traits. *microTrait* maps a genome sequence into a hierarchical trait space that covers energetic, resource acquisition, stress tolerance, and life history traits that underlie microbial strategies describing environmental microbes (Malik et al., 2020). Our pipeline makes use of literature-supported *omics* markers defining trait-based microbial strategies to quantify trait profiles for microbial genomes. Given a genome sequence, individual gene markers are detected with a model-based approach using a new HMM database of protein families. The models have been trained with protein sequences that represent sequence diversity from genomes and metagenomes and their accuracy measured independently with KEGG orthology database. The traits are inferred from gene markers based on their presence/absence patterns and presented in a hierarchical manner.

RESULTS

Microbial Traits With Genomic Basis

The overarching goal of our approach is to reduce the dimensionality and complexity of the genomic information such that a genome is represented as a feature vector where individual features represent one or more aspects of an ecological strategy (Lajoie and Kembel 2019). Microbial traits span a wide range of phenotypic, ecological, and metabolic characteristics. The choice of specific traits and their representational granularity depend on the research question of interest. We first review the genome based traits inferred by *microTrait,* rationalize their choice primarily following the frameworks proposed by (Green et al., 2008) and more recently (Malik et al., 2020) (**Figure 1**).

At the very fundamental level, our approach takes as input a genome sequence and maps it to a trait space in a computationally scalable way. Here we adopt a microbial counterpart of the widely used definition of "functional traits" for macroorganisms as measurable characteristics that "impact fitness of an organism via its effect on growth, reproduction, or survival" at the individual level (Violle et al., 2007; Violle et al., 2014). Unlike for macroorganisms, measuring traits at the individual microbe level in complex communities is currently not feasible, although single-cell imaging and '*omic* technologies are beginning to expand our understanding of population heterogeneity at these native scales (Wang and Bodovitz 2010; Bock et al., 2016). Genomes have recently been proposed as the ecological units (Prosser 2015; Turaev and Rattei 2016) at which genome-inferred traits should be measured. Advances in DNA sequencing and computational protocols has led to a more or less continuous stream of provisional genomes not only from cultured isolates but also from single-cells (SAGs) and metagenomes (MAGs) (Sharon and Banfield 2013). Though as an ecological unit, the resolution represented by MAGs may not currently match its counterpart for macroorganisms, possibly representing mosaics and distorting or masking intra-population differences, they nevertheless provide an unprecedented window into complex microbiomes and provide especially valuable insights into the physiology and metabolism of uncultivated organisms in their natural environments. As such, a genome-centric lens to traits allows scaling of organism level traits to communities (through incorporation of genome abundances) and therefore at larger scale as well as studying trait linkage across ecologically relevant units.

We identified genomic features that can be mapped to microbial ecological strategies, conceptualized under four dimensions (**Figure 1**) organized as a hierarchy ("*microTrait* hierarchy": **Supplementary Table S1**). Within each strategy, the trait information is organized as a hierarchy whose leaf nodes map to specific genome derived features. **Supplementary Table S8** lists the full list of references that establish the links between each genome derived feature and the ecological strategy at the most granular level. Here we give an overview of the traits for each ecological strategy:

Resource Acquisition Traits

A tremendous variety of substrates ranging from simple inorganic ions to complex organic molecules serve as resources for microbes. Microbes have adapted a suite of concrete strategies with genomic basis to be competitive in a wide range of environments with spatiotemporally variable resource profiles. Many microorganisms have the potential to produce exoenzymes that can disassemble complex resources (substrate degradation), which can then be acquired through uptake (substrate uptake) via membrane transporters (Berntsson et al., 2010; Arnosti 2011; Zimmerman et al., 2013; Arnostil et al., 2014; Courty and Wipf 2016; Bergauer et al., 2018). Thus, one aspect of resource acquisition strategy concerns the investment in both the number and diversity of exoenzymes and membrane transporters a microbe would maintain in a microbial genome. Substrate uptake is linked to substrate assimilation traits that determine the capacity for assimilation of inorganic compounds.

Resource Use (Energy Generating) Traits

Redox reactions underlie all biological energy metabolism and redox chemistry provides an organizing principle to connect microscale to global scale processes (Falkowski et al., 2008; Ramirez-Flandes et al., 2019). Genes whose protein products catalyze redox reactions, their coupling to energy conservation, and their genomic organization determine the basis for microbial metabolic strategies. Historically, in the pre-genomic era, single metabolic traits were evaluated in isolation to define "metabolic functional groups" but genomic data has underlined the tremendous metabolic flexibility of microbes (Anantharaman et al., 2016). As a result, classical enumerations of microbial metabolism are not sufficient to represent the linkage of metabolic traits. Representation of microbes as a suite of energy metabolism traits provides a more complete picture and a data driven definition of metabolic guilds.

Stress Tolerance Traits

Stress may be induced by physical, chemical, or biological conditions that adversely affect microbial growth and survival. Microbes that use stress tolerance strategies respond to a variety of stressors using several physiological and evolutionary mechanisms. Though the specific stress response depends on the particular suboptimal conditions, common traits with genomic underpinnings have been broadly identified (General Stress Tolerance Traits). These include increasing the concentration of some molecular chaperones (stress proteins/heat-shock proteins) to combat biomolecular damage in response to stress. This is a universal feature across all domains of life but the relative importance of genetic (i.e., diversity and gene copy number) or regulatory (transcriptional, translational, and post-translational) processes under different stressors is less clear (Feder and Hofmann 1999; Hecker and Volker 2001; Yu et al., 2015).

Genomic bases of microbial traits that underlie stress tolerance to specific physiochemical and chemical factors have also been identified: 1) Temperature stress: a suite of heat shock genes serving as chaperones and proteases are involved in the protection, repair, and degradation of denatured/misfolded proteins. Response to cold shock involves adaptation of the membrane via an increase in the proportion of unsaturated fatty acids and activation of chaperone cold shock proteins to restore mRNA functionality. 2) Desiccation, osmotic, salt stress: Known molecular strategies to tolerate drought and freezing include production or uptake of osmolytes like trehalose and glycine betaine to reduce water potential and maintain hydration or synthesis of extracellular polymeric substances (Csonka 1989; Ko et al., 1994; Mindock et al., 2001; Costa et al., 2018). 3) Oxidative stress: The response to oxidative stress is a complex one that involves the coordinated regulation of many genes most critically involving enzymes that scavenge reactive oxygen species. The activation of such regulons requires

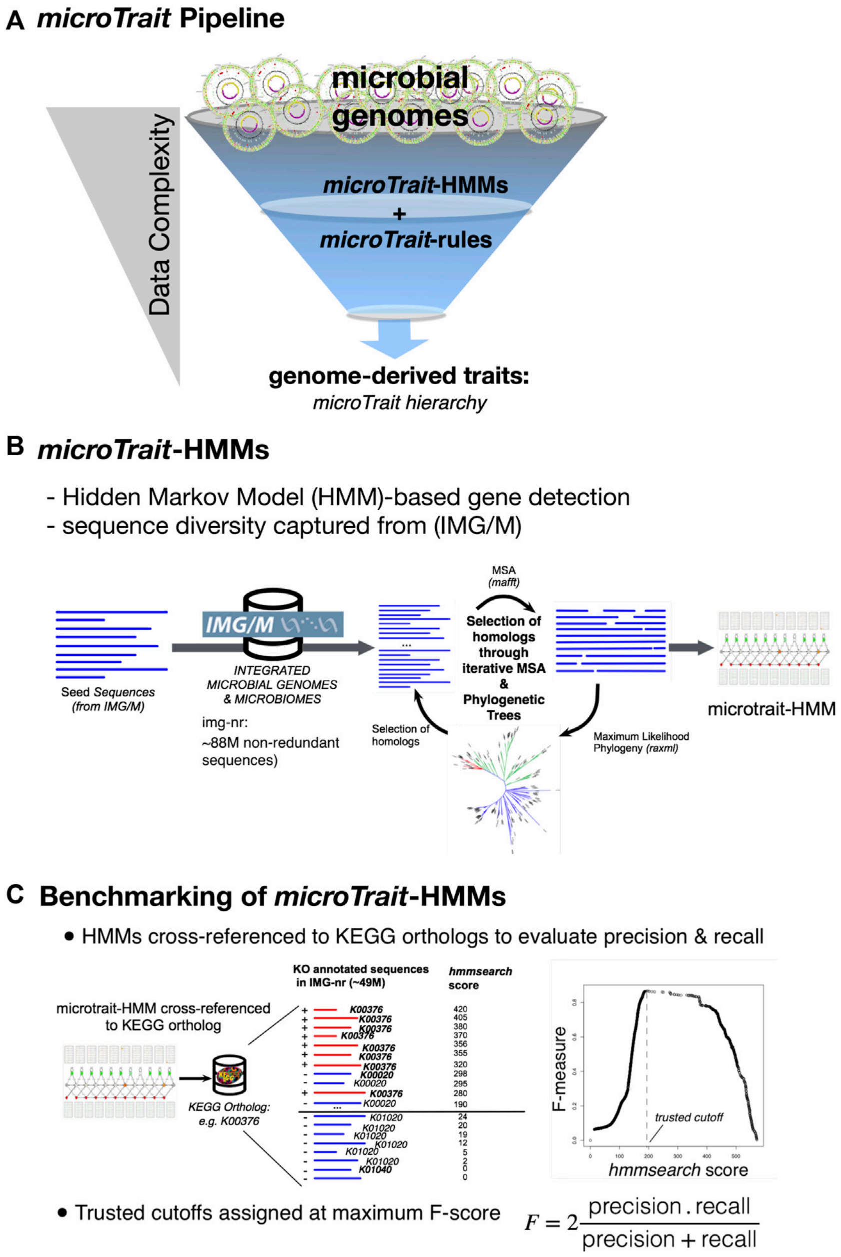

A microTrait Pipeline
microbial genomes
Data Complexity
microTrait-HMMs
+
microTrait-rules
genome-derived traits:
microTrait hierarchy
B microTrait-HMMs
- Hidden Markov Model (HMM)-based gene detection
- sequence diversity captured from (IMG/M)
Seed Sequences (from IMG/M)
IMG/M
INTEGRATED MICROBIAL GENOMES & MICROBIOMES
img-nr: ~88M non-redundant sequences)
Selection of homologs
MSA (mafft)
Selection of homologs through iterative MSA & Phylogenetic Trees
Maximum Likelihood Phylogeny (raxml)
microtrait-HMM
C Benchmarking of microTrait-HMMs
• HMMs cross-referenced to KEGG orthologs to evaluate precision & recall
microtrait-HMM cross-referenced to KEGG ortholog
KEGG Ortholog: e.g. K00376
KO annotated sequences in IMG-nr (~49M)
hmmsearch score
F-measure
trusted cutoff
hmmsearch score
• Trusted cutoffs assigned at maximum F-score
F = 2 precision . recall / precision + recall

FIGURE 2 | Overview of *microTrait*. **(A)** *microTrait* pipeline consists of a library of gene-level Hidden Markov Models (*microTrait*-HMMs) for detection of genome features and logical rules (*microTrait*-rules) that map these features to traits. The output from the pipeline are trait matrices (genomes × traits) at different granularities corresponding the levels of the *microTrait* hierarchy. **(B)** Workflow for construction of *microTrait*-HMMs. Each HMM models the diversity of sequences from IMG/M at gene-level. **(C)** Benchmarking of *microTrait*-HMMs. The trusted cutoffs for *microTrait*-HMMs were determined through cross-references to KEGG orthologs (whenever available).

redox sensing (two-component redox sensors and redox-sensitive TFs). 4) pH stress: Similarly to general, oxidative, and temperature stress, molecular mechanisms for protection from acid stress include investment in chaperones, proteases and the ability to sense and respond to redox conditions through two-component systems and TFs. Unique mechanisms for maintenance of intracellular pH include the consumption and extrusion of intracellular protons by acid-inducible amino acid decarboxylase-antiporter and urease systems, and the enzymatic conversion of unsaturated fatty acids into cyclopropane fatty acids.

Life History Traits

Ecological and evolutionary processes leave their signatures in overall microbial genome content and organization. A key dimension of any ecological strategy is growth. Optimal growth characteristics of microbes are key to understand how the key traits regarding resource acquisition, resource use, and stress tolerance are realized to adapt to a particular environmental niche. Traits that concern these characteristics are classified as life history traits. Codon usage bias and ribosomal RNA (rRNA) operon copy number are linked to maximum growth rate, a life history trait constraining all other functional traits (Weider et al., 2005; Vieira-Silva and Rocha 2010; Weissman et al., 2021). Another key life history trait closely linked to the overall genomic adaptation is optimal growth temperature (OGT). Temperature is a master regulator of enzyme activity and overall cell machinery. A combination of quantifiable proteome-wide features predictable from genome sequences allows OGT to be hypothesized solely from genomic sequence (Zeldovich et al., 2007; Sauer and Wang 2019).

microTrait Pipeline

The computational pipeline to infer traits from primary genome sequences has two major components (**Figure 2A**): 1) a database of gene HMMs (*microTrait-HMM*) to model the diversity of protein families based on sequences from genomes and metagenomes with independently established accuracy to detect genetic loci (**Figure 2B** and **Supplementary Table S2**), 2) a set of rules (*microTrait* rules) encoded in predicate logic to infer traits from presence and absence of the set of loci modeled in *microTrait-HMM* (**Supplementary Table S4**). The model-based detection of genetic loci ensures decreased run-times and interoperability across datasets (given model and scoring cutoff). The rule-based framework to infer traits from primary features gives the user the flexibility for redefinition and refinement.

Cross References to External Databases From *microTrait*-HMM

The statistical models in *microTrait-HMM* reflect the most recent sequence diversity from both cultured and uncultured microbes and therefore should have improved accuracy over existing methods to detect genes underlying traits covered in *microTrait*. To ensure interoperability of the *microTrait* pipeline with the existing HMM databases and relevant sequence database resources, for each gene model we provide database cross references to KEGG (Kanehisa and Goto 2000), Transporter Classification Database (Saier et al., 2016), and Enzyme nomenclature database (through EC numbers) (1999).

Performance of Gene HMMs and Assignment of Trusted-Cutoffs

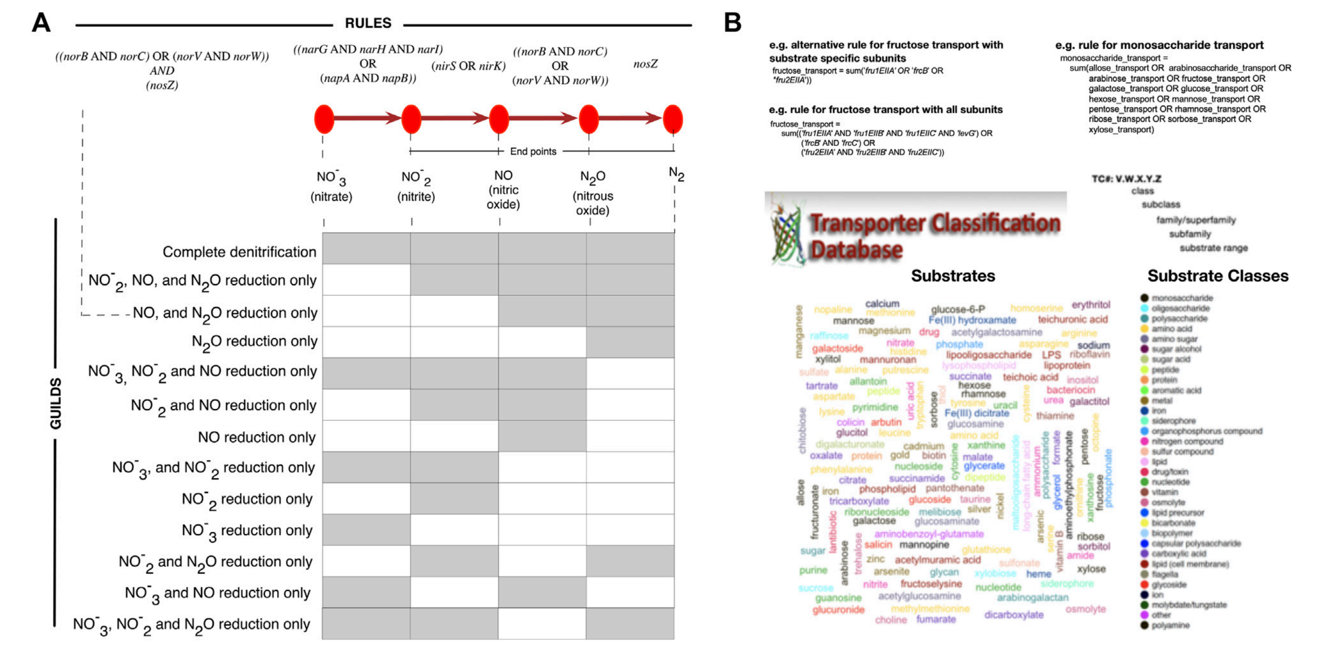
We assessed the performance of each *microTrait-HMM* by first determining the corresponding orthologous group (KO number) in KEGG orthologs database (when the loci was represented in KEGG) (**Figure 2C**). A test dataset for the gene model in question was built by using IMG/M sequences labeled with the determined KO number ("true positives") and the remaining KO numbers ("true negative"). IMG/M database was scanned with the profile HMM using HMMER/hmmsearch. F-scores (harmonic mean of precision and recall) were calculated as a function of "hmmsearch scores" based on the test dataset with R using ROCR package (Sing et al., 2005). The smallest score that maximizes F-scores was assigned as the trusted cutoff. **Supplementary Table S3** summarizes the performance of each model in *microTrait*-HMM. Overall, at the determined trusted cutoffs, the overwhelming majority of *microTrait*-HMMs (94.2%-1,686 out of 1790 HMMs) had high sensitivity (≥75%) and low FPR (false positive rate), with 92% of HMMs having an F-score >=0.8 (**Supplementary Figure S1**).

microTrait Pipeline: Derivation of Traits From Genome Sequences

The input to *microTrait* is a genome sequence (.fa) or the corresponding protein coding genes (.faa) in FASTA format. When genomic rather than protein coding gene sequences are supplied, Prodigal is used to predict open reading frames (Hyatt et al., 2010). For each genome, protein sequences are scanned against *microTrait-HMM* with HMMER/hmmsearch to generate a count table for the detected gene models. Binary and continuous traits are assigned using the count table and predefined logical rules mapping the presence/absence of genes(s) or other rules to specific traits (**Figure 3**). The rules can be edited by the users within the R package. Their role is twofold: On one hand they allow modifications in the way some binary traits can be defined (for instance based on one or more proteins in a large complex, or one or more steps in a pathway) giving the user flexibility. They can also be used to increase detection sensitivity for provisional or lower quality genomes (i.e., SAGs and MAGs).

Modular Trait Definitions With Predicate Logic

microTrait uses Boolean algebra to map protein family content into traits through *microTrait* rules (**Supplementary Table S5**). In this framework, each protein family is a Boolean variable (i.e. equals 1 if detected, 0 otherwise) whose value is determined by the output of the corresponding *microTrait*-HMM. The traits are represented by rules whose arguments are one or more protein families, other rules, or a combination of these. Conceptually, the rules map to representations of protein complexes with multiple subunits or a series of enzyme catalyzed reactions that transform one molecular species into another. While the standard package comes with a predefined set of rules, the rules themselves and the mapping of rules to traits are modular and can be modified by the user. As an example, consider denitrification traits (**Figure 3A**). The canonical denitrification pathways, excluding accessory and regulatory proteins, involve 4 protein complexes (NarGHI: the inner membrane-bound nitrate reductase; NapAB: the periplasmic nitrate reductase; NorBC, NorVW: nitric oxide reductases) and 3 proteins (NirS, NirK: nitrite reductases; NosZ: nitrous oxide reductase). Together, these are represented by 12 protein families (italicized gene names in **Figure 3A**) and the four individual enzymatic steps are represented by 4 rules. From these rules, several denitrification traits corresponding to individual functional guilds can be defined.



e.g. alternative rule for fructose transport with substrate specific subunits

```
fructose_transport = sum('fru1EIIA' OR 'frcB' OR
"fru2EIIA'))
```

e.g. rule for fructose transport with all subunits

```
fructose_transport =
  sum(('fru1EIIA' AND 'fru1EIIB' AND 'fru1EIIC' AND 'levG') OR
      ('frcB' AND 'frcC') OR
      ('fru2EIIA' AND 'fru2EIIB' AND 'fru2EIIC'))
```

e.g. rule for monosaccharide transport

```
monosaccharide_transport =
  sum(allose_transport OR  arabinosaccharide_transport OR
      arabinose_transport OR fructose_transport OR
      galactose_transport OR glucose_transport OR
      hexose_transport OR mannose_transport OR
      pentose_transport OR rhamnose_transport OR
      ribose_transport OR sorbose_transport OR
      xylose_transport)
```

FIGURE 3 | Trait inference with *microTrait* rules. microTrait rules use simple boolean logic to map presence/absence of *microTrait*-HMMs (italicized) to traits. The reconfigurability of the rules makes the exploration of the effect of different trait definitions on the microbial guilds possible and therefore enables a flexible microbial trait extraction pipeline. Examples for trait definitions from rules for **(A)** denitrification traits. Rule-based inference allows flexible definition of traits, for example by end products of denitrification. **(B)** substrate uptake. *microTrait* represents substrate uptake traits using the range substrates documented in TCDB (Transported Classification Database) (shown as word cloud colored by substrate class). Traits relevant to the uptake of substrates (example for monosaccharides) can be defined in a hierarchical manner with rules defined from other rules and *microTrait*-HMMs.

For transporters and polymer specific extracellular enzymes, we compiled a list of the experimentally reported substrates of each enzyme using the Transporter Classification Database (TCDB) (Saier et al., 2016) and the Database of carbohydrate-active enzymes (dbCAN) (Yin et al., 2012). We then classified each reported substrate into broad substrate classes (**Figure 3B** and **Supplementary Table S6**). The relevant rules for transporters and extracellular enzymes let the user quantify the number of protein complexes with a given substrate or substrate class.

A challenge in assigning traits to genomes based on the protein family signatures is the modularity of the underlying pathways. This modularity might be truly reflecting the genomic variation within a set of isolates, MAGs or SAGs but also be an apparent manifestation of incomplete and noisy genomic information. Starting with genomic sequences, *microTrait* allows the investigation of this modularity across a set of genomes. The resulting information can be used by the user to define custom logical rules to assign traits based on the protein family content.

Comparing *microTrait* With a Taxonomy-Based Inference of Microbial Functional Groups

Linking taxonomic classification with function is a commonly used method to infer microbial traits. Faprotax is a manually curated database that maps taxa to functional groups based on the physiological studies for the cultured representatives of these taxa (Louca et al., 2016). The taxonomic resolution is typically at species or genus level but can also be less specific (i.e. family or higher). Using a large collection of isolate genomes from environmental ecosystems (refer to Materials and Methods for construction of the genome collection) and literature references for functional affiliations based on taxonomic names in Faprotax (**Supplementary Table S11**), we have quantified the extent to which *microTrait*-rules recovered the validated culturable taxa for different microbial functional groups. For each functional group, we first matched the taxonomic names from literature, primarily genus/species names but also extending to higher ranks for certain functional groups, to canonical NCBI taxonomic names. All available genomes from environmental ecosystems with the respective taxonomic affiliation were considered as a "positive" for that functional group according to the Faprotax approach (**Supplementary Table S12**). We have then tested how many of these assumed Faprotax positives the *microTrait* pipeline was able to recall solely based on the functional trait predictions from genomes. In addition, for each functional group, we have also evaluated the specificity of genome-based calls based on the assumption that all negatives via the Faprotax taxonomic affiliation were "true negatives" (**Supplementary Table S13**).

Among 41 functional groups, 29 had a recall rate over 70%. Functional groups for which *microTrait* had low recall rates included anammox (0 *microTrait*+ genomes out of 7

Faprotax+ genomes; 0/7), dark iron oxidation (10/16), iron respiration (19/86), aerobic nitrite oxidation (6/13), chlorate reducers (3/6), dark sulfide oxidation (49/93), anoxygenic photoautotrophy Fe oxidizing (9/16), dark sulfur oxidation (71/124), sulfur respiration (82/139), thiosulfate respiration (88/145). A close examination of the taxonomic identity of the genomes "missed" by *microTrait* suggested a variety of explanations for the functional groups with poor recall.

A primary advantage of inferring microbial traits directly from genomic sequences rather than by taxonomic names is the ability to resolve diversity (species or strain level), which increases the prediction accuracy. We have observed that for many functional groups defined in Faprotax, the genomes that were assigned to the taxonomic clades lacked the required genetic repertoire for the metabolic function in question. Some prominent examples are for the "anammox" and "dark iron oxidation". For anammox, among the diversity of taxa (genus and species), only *P. mendocina* had corresponding genomes in the isolate set (n = 7) and none of those had the genomic features for anammox suggesting that this is a strain specific trait for *P. mendocina.* Similarly, for dark iron oxidation, genome features suggested that the trait can be strain specific. Among 15 *R. palustris* and 2 *M. ferrooxydans,* a limited number (9 and 1 genome respectively) was genome-supported to carry the trait. There were also cases where the genomic evidence suggested that trait conservation was limited to deep taxonomic levels so a taxonomic inference at genus or family level would have impacted the accuracy of Faprotax method. For instance, methanotrophy is associated with Methylocystaceae (family) and Methylocapsa (genus) yet the trait was specific to subfamily/ subgenus. Among 7 Methylocystaceae genera with genome representatives, 2 genera (Methylocystis and Methylosinus) had genome support for the trait. Similarly, 2 out 3 Methylocapsa species with genomes had evidence for the trait.

It should be noted that, there were also cases for which the absence of the genomic signal reflected limited knowledge for the genetic underpinnings of the trait. A typical example was for iron respiration, a trait for which current evidence suggests that electron transport for iron reduction proceeds in a different and unknown mechanism in acidophiles compared with *Ferrimonas* and *Shewanella* (Malik et al. 2018). Another example was for chlorate reduction, a process whose genomic trait sits in a region prone to horizontal transfer (Clark et al., 2013) which impacts the accuracy of a gene-level profile HMM approach. Overall, these disagreements between taxonomic and genome-based approaches suggests that, a genomic feature-based approach such as *microTrait* increases prediction accuracy and precision, even when one considers single traits (such as functional groups).

High-Throughput Extraction of Microbial Traits from Genomes with *microTrait*

As an example of scalable extraction of traits from genomes, we applied *microTrait* to publicly available isolate genomes and MAGs. The datasets we used included 1) isolate genomes from environmental ecosystems from IMG/M (n = 6,157), 2) MAGs from an aquifer system (n = 2,545) (Anantharaman et al., 2016), 3) MAGs from a thawing permafrost (n = 1,530) (Woodcroft et al., 2018), 4) MAGs from hydrothermal sediments (n = 666) (Dombrowski et al., 2018), and 5) MAGs from publicly available metagenome samples, referred to as Uncultivated Bacteria and Archaea Dataset (UBA) (n = 7,902) (Parks et al., 2017). This compendium of datasets (genome compendium) resulted in a total number of 20,062 genomes.

We tested *microTrait* on a machine with a 2.3 GHz 16-core Intel Xeon Processor E5-2,698. When run using a single core, with a single genome processed using that core, *microTrait* processed that genome in 3.94 ± 2.59 min, with an average of 1.11 min/Mb of genome sequence (**Supplementary Figure S2**). From these, we predict that *microTrait* can process an average microbial genome of size 4 Mb in approximately 4.5 min. In all runs, the memory footprint of *microTrait* was not larger than 60 MB. In a multiprocessor compute environment, *microTrait* is easily parallelizable using a typical data-level parallelization scheme (for instance using R's *parallel* package (distributed as part of R-core)) mapping genomes to separate logical processors. In our tests, when run in a 64 processor compute node, the processing of the compendium of 20,062 genomes (total size = 47.9 Gb) took 12.47 h.

microTrait Trait Matrix

When applied to multiple genomes, *microTrait* outputs a trait matrix of "genomes x traits" with three types of qualitative variables. Binary trait variables are calculated as presence/ absence of a specific functional capacity and span 1) energy generation via specific electron acceptors/donors, 2) capacity to degrade, assimilate, or acquire specific substrates. Continuous trait variables are of two groups. The first group of continuous traits are calculated starting from counts of specific functional capacities in the genome and span 1) acquisition of chemical classes of substrates with transporters or via extracellular breakdown, 2) investment in extracellular polysaccharides and osmolytes. For each genome, the counts are normalized by genome size. The second group represent life history traits and include 1) minimum generation time (unit: h^{-1}) predicted based on indices of codon-usage bias in ribosomal protein genes (a proxy for highly expressed genes) (Vieira-Silva and Rocha 2010) (Weissman et al., 2021), 2) optimal growth temperature (unit: °C) predicted from a suite of features derived from the nucleotide and protein sequences of the genome (Sauer and Wang 2019).

Refinement of Functional Guilds Using *microTrait*

To exemplify the use of *microTrait* in refining functional guilds, we explored how denitrifier guilds can be defined based on the genomic distribution of denitrification traits in the isolate genomes from our compendium of genomes. Denitrification is a key biologically catalyzed process by which nitrogen available to plants is transformed to the atmospheric nitrogen pool as gaseous forms of nitrogen as molecular N_2 or as an oxide of N. Denitrification occurs as a step-wise reduction of nitrogen oxides with gaseous products. Four reductases are involved in the denitrification, NAR, NIR, NOR and N2OR, sequentially catalyzing the reductions of NO3 - → NO2 - → NO →N2O →N2. Several previous studies reported both genomic and phenotypic evidence for truncated versions of the denitrification pathway but a global genomic analysis is not currently available (Sanford et al., 2012; Jones et al., 2014; Lycus et al., 2017; Liu et al., 2018; Gao et al., 2019).

We used the *microTrait* pipeline to explore all of the publicly available environmental genomes from the IMG/M database (**Supplementary Table S9**). This resulted in a "genomes X rules" matrix specifying for each genome whether each of the rules was asserted as TRUE or FALSE. The matrix was subset to rules underlying denitrification traits and the genomes were clustered based on their denitrification trait profiles. The clustering gave 13 denitrification-associated functional guilds, with 58.3% of the screened genomes involved in at least one denitrification-related process (**Supplementary Figure S3**). Only, a small proportion of these had the genomic capacity to perform complete denitrification to N2. Overall, the guilds correspond to generation of the same end products from different starting nitrogen compounds (e.g. guilds 1–4, 5–7, and 8–9 generating N_2, N_2O, and NO respectively), or multiple end products with missing steps (e.g. guilds 11–13). The default trait matrix in *microTrait* defines denitrification traits by the end products of denitrification (**Supplementary Table S7**) yet the workflow of going from genomic features to traits via *microTrait* rules makes redefinition of traits possible.

Testing Trait Dimensionality of Microbial Genomes from a Given Ecosystem

microTrait hierarchy maps a microbial genome to a high-dimensional space of putative functional traits of ecological relevance. In trait-based ecological modeling, trait selection is of central importance not only for biological but also for computational, statistical, and practical reasons (Lajoie and Kembel 2019). In our conceptualization of the relevant traits for terrestrial ecosystems, the set of selected traits are assumed to approximate the intrinsic (i.e. true underlying but unobserved) dimensionality of microbial traits. Unlike for plants for which accumulated evidence suggests that the intrinsic dimensionality of functional trait space is low (Laughlin 2014), the intrinsic dimensionality of the trait space of microbes in specific ecosystems remains largely unknown. However, we can assume that if the selected trait proxies are largely independent of each other then, taken jointly, they should represent the underlying functional differences, and improve our ability to explain and predict microbial distributions.

To investigate whether the selected traits in *microTrait* are largely independent, we used an extensive dataset of genomes of microbes isolated from terrestrial ecosystems to study the correlation structure of their *microTrait* profiles. The trait matrix (at granularity 3) for a total of 4,116 genomes of organisms isolated from terrestrial environments (ST9) was computed using *microTrait.* A non-parametric rank-order correlation metric was used to estimate the degree of relatedness between all trait pairs, visualized as a correlation matrix and reordered to elucidate the potential hidden structure and pattern in the matrix (**Figure 4A**).

Overall, the bulk of the correlations were weak ($|\rho| < 0.3$) suggesting that *microTrait* trait dimensions map to largely independent traits (**Figure 4B**). On the extremes, strong positive correlations would be indicative of redundancy of trait dimensions while negative correlations would be indicative of underlying tradeoffs for the ecosystem in question. Few strongly positively correlated blocks corresponded to phototrophic resource use traits linking the variety of phototrophic pigments and photosystems.

Dimensionality Reduction with Guild-Centric Analysis of Microbial Genomes With *microTrait*

Metagenomics allow the recovery of the genomes of all detectable members of an ecosystem along extensive spatiotemporal gradients. The genomes then provide support for co-occurrence of ecologically relevant traits of the members that together underlie the ecosystem function. A typical genome-centric microbiome study involves the analysis of hundreds to thousands of genomes leading to trait matrices of high genomic dimensionality. This high dimensionality poses a particular problem for statistical analyses (Johnstone and Titterington 2009). Further, when attempting to leverage the information from these genomes for downstream modeling applications, there is both a practical need and discovery opportunities in quantify and reducing this dimensionality in a tractable manner. Organizing microbial members of an ecosystem community into "putative guilds" can reduce the dimensionality of a metagenomic dataset and hypothesize the functional niche of community members and computationally explore their interactions independently of their taxonomic origin. Here, using the soil ecosystem as an example, we show how to define microbial guilds in a data-driven manner using *microTrait.*

Given a set of genomes representing a habitat, *microTrait* can be used to discover and define functional guilds, parameterize the defined guilds with life history traits (minimum doubling time and optimal growth temperature), and reduce the dimensionality of the trait space in a quantifiable way. **Figure 5** outlines the guild-centric pipeline starting with a trait matrix leading to the definition and characterization of the microbial guilds. Since *microTrait* encompasses both continuous and binary traits, the similarity between genomes are measured using a distance metric suitable for mixed data types (Wishart 2003) (see Methods). The resulting distance matrix (genomes x genomes) is clustered with unsupervised hierarchical clustering, visualized with trait presence/absence (i.e., treating continuous traits as binary variables), and annotated with the distribution of life history traits and trait prevalence across the dataset (**Figure 5A**). Quantifying relationships between genomes based on their trait profiles gives the opportunity to dynamically define guilds in a data-driven way for any dataset. The proportion of inter-guild variance explained can then be quantified as a function of the number of guilds (**Figure 5B**). A larger number of guilds corresponds to a smaller information loss at the expense of greater complexity for downstream applications. The user

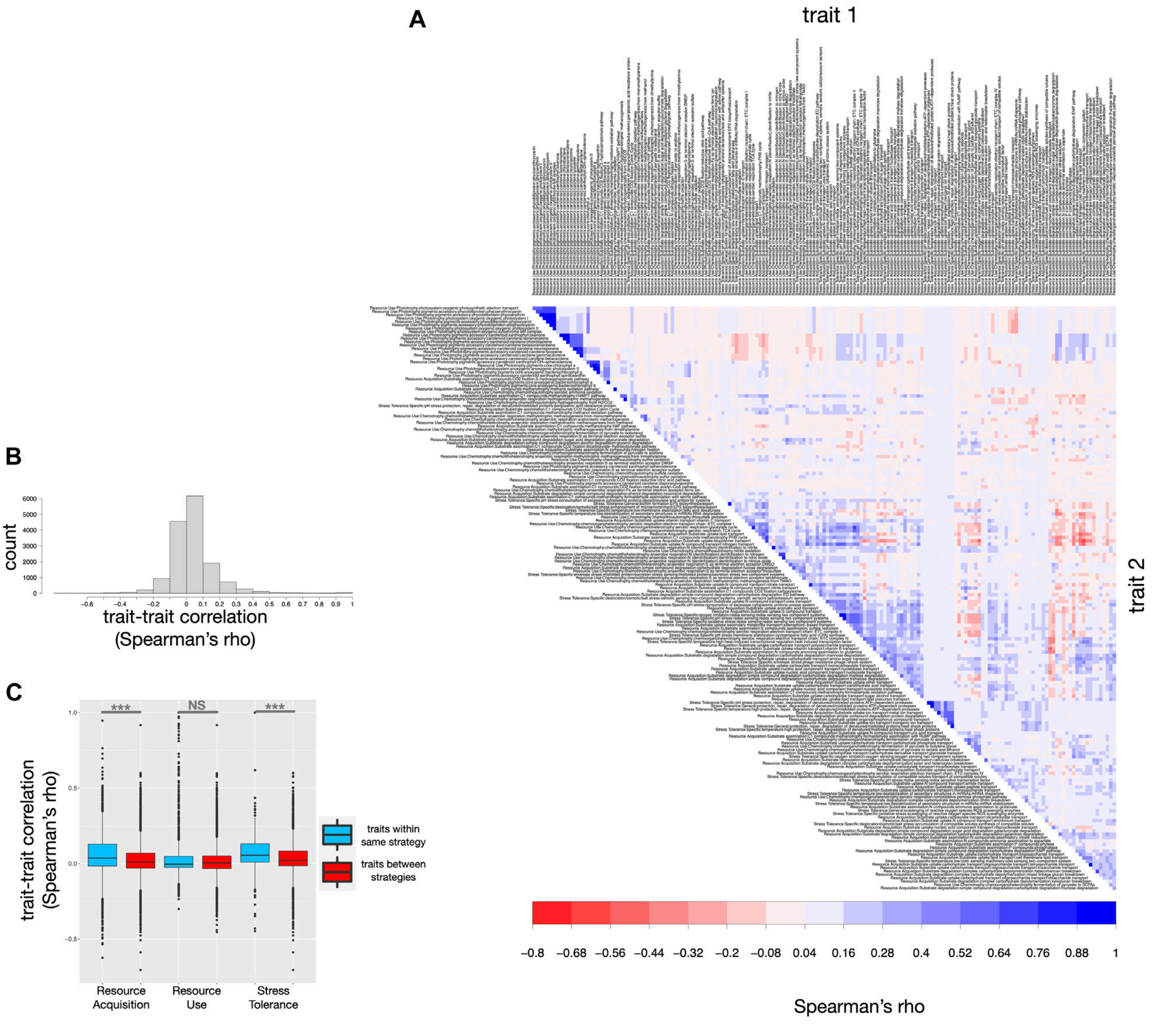


FIGURE 4 | Correlation matrix for *microTrait* defined traits. The strength of the correlation (Spearman's rho) is represented by the color intensity (positive: blue, negative: red). Left upper panel: the distribution of trait-to-trait correlation values, left lower panel: comparison of the distribution of trait-to-trait correlations within and between ecological strategies.

decides here where to operate along the curve depending on the shape (rate of change in steepness with increasing guilds) and the application of interest. Once determined, the guilds can be defined which results in a list of guilds, each representing a number of genomes and the joint distribution of traits captured by them. It is often useful to examine the distribution of the number of genomes that underlies each guild as on average the within-guild trait variance would be higher for guilds supported by a smaller number of genomes. The user can filter the guilds by number of genomes to generate a dataset that represents guild profiles, that is a fingerprint of the co-occurrence of traits for each guild and the within-guild distribution of life history traits (**Figure 5C** and ST 16).

We applied the *microTrait* data-driven guild-definition pipeline to soil isolate genomes from IMG (3,430 genomes with GOLD Ecosystem Type = "Soil OR Rhizoplane OR Rhizosphere OR Root"). All traits except "anaerobic ammonia oxidation (anammox)" were detected at least once in the dataset resulting in a trait matrix of dimensionality 3,430 genomes X 190 traits. To date no pure culture isolates of anammox organisms have been obtained (Jetten et al., 2005). Clustering analysis indicated that a total of 196 guilds captured 70% of the inter-guild variance, with 16 guilds supported by at least 50 genomes. Comparison of the trait profiles across guilds elucidates the differentiating trait features of a set of guilds with respect to other guilds.

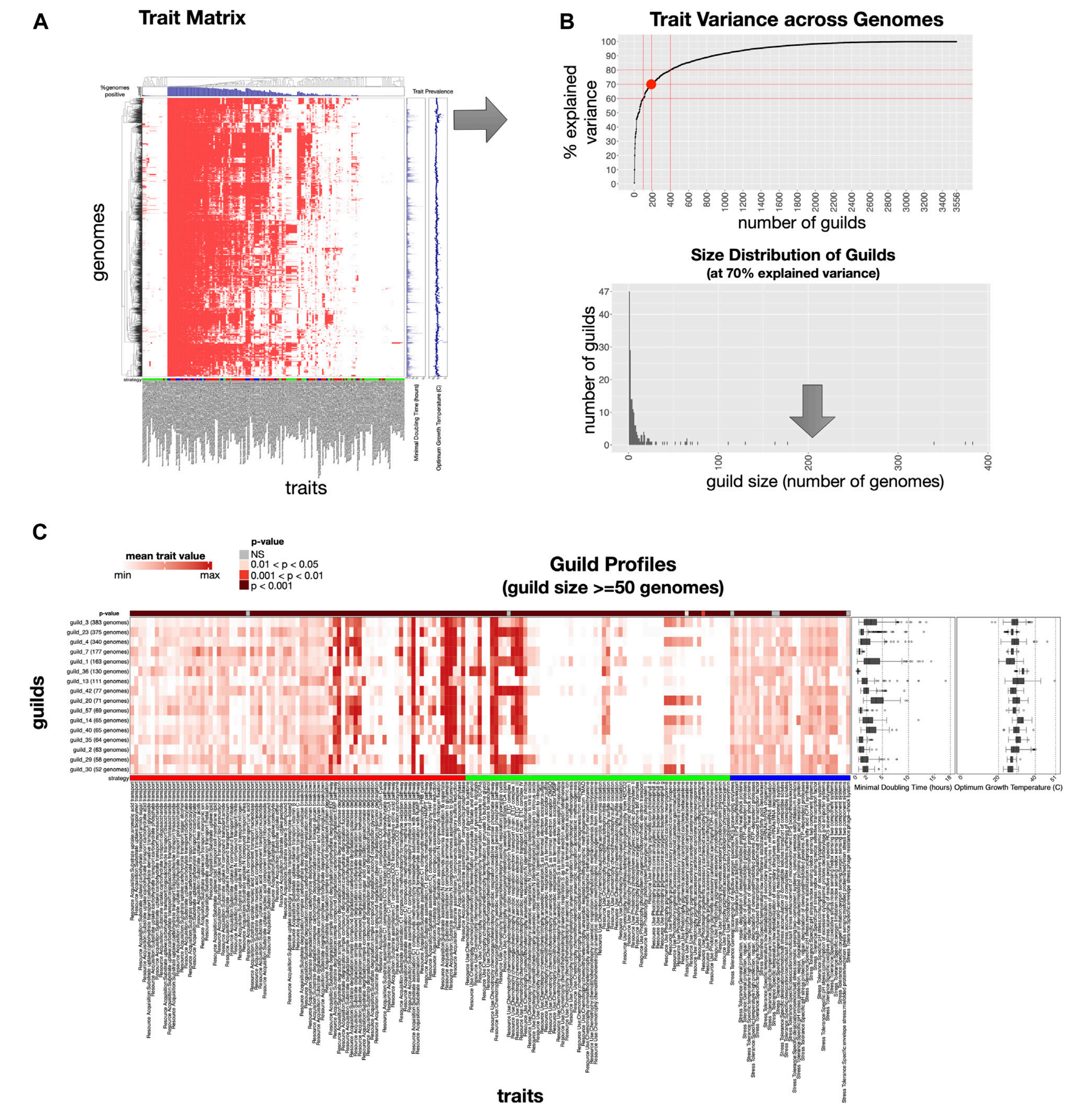


FIGURE 5 | Primary use cases and graphical outputs of *microTrait* workflow. **(A)** Trait matrix provides clustering of a set of input genomes using trait profiles from *microTrait* outputs based on a distance metric taking into account mixed data types (i.e. for binary and count traits). Heatmap visualization use presence (red)/absence (white) of traits, with trait prevalence (% genomes positive) shown at the top panel. Life history traits (minimal doubling time and optimum growth temperature) are overlaid on the right panel in continuous scale. **(B)** Trait variance across genomes based on the genome clustering is quantified as a function of the number of guilds using analysis of variance using distance matrices. Guilds can be defined either at a fixed number of guilds or based on percent explained within-guild variance, which results in a size (number of supporting genomes) distribution of guilds. **(C)** Visualization of trait profiles for the defined guilds (guilds × traits), with mean trait values visualized across a color scale. Traits are ordered by ecological strategies (red: resource acquisition, green: resource use, blue: stress tolerance). For each trait, top panel shows the statistical significance of comparison of mean trait values across guilds. The distribution of life history traits are shown on the right side panels.

For example, the top three guilds supported by the highest numbers of genomes (guild 3, guild 23, and guild 4; 383, 375, and 340 genomes respectively) were each enriched in specific traits under resource acquisition and resource use strategies (ST16). Guild 23 compared to guild 3, and 4 was marked by enrichment of the ability to assimilate simple C compounds, use 2 C compounds in the absence of glucose via glyoxylate cycle, uptake a variety of N compounds (elemental N and urea) as well aromatic acids and biopolymers, and fix elemental nitrogen for biomass. On the other hand, compared to guild 23, guild 3, and 4 represent a different strategy for incorporation of N compounds into biomass through assimilatory nitrate reduction and a unique ability to assimilate P compounds. Notably, although all three guilds were enriched in the capacity to utilize glucose, guilds 23 and guilds 3, and 4 differed in their preferred glycolytic pathways (canonical Embden-Meyerhoff-Parnass (EMP) pathway in guilds 3, and 4 vs. less common Entner–Doudoroff (ED) pathway in guild 23) reflecting differing preferences in balancing production of ATP (energy yield) and cost of protein synthesis to achieve maximum fitness (Flamholz et al., 2013). Across these three guilds (3, 23, and 4) differences in enrichment for stress tolerance mechanisms were not apparent, however, other guilds did display enrichment in specific stress tolerance strategies. For instance, among all the guilds supported by at least 50 genomes, guilds 7 and 14 were uniquely enriched in traits for desiccation and pH stress tolerance respectively.

DISCUSSION

Genome sequencing, from a data perspective, now provides a primary window into the traits that regulate fitness and function across Earth's microbiomes. Genomes are increasingly recognized as a fundamental unit in the study of microorganisms, however, the integration of this information is required to understand how such genome units relate to ecologically coherent behavior. Exploration of feedbacks between microorganisms and their environments requires numerical modeling approaches, and the assimilation of genomic information has substantially lagged its generation. This assimilation of microbiome information into numerical models in an automated fashion remains a significant challenge as microbial communities are ultra-diverse, physiologically plastic, and dynamically adaptive. Trait-based approaches to microbial ecology provide a framework to represent microbial diversity in a way that facilitates prediction, integration and generalization (Lajoie and Kembel 2019) and the rate at which isolate and metagenome-assembled genomes are being generated provide an unprecedented resource to explore patterns in microbial trait conservation and linkage. The resulting information can be used to initialize and parameterize mechanistic trait-based models spanning a scale of complexities to explore the drivers of patterns in the distribution and co-occurrence of microbial traits. With *microTrait*, our goal was to provide an extendable toolset and computational pipeline to infer microbial traits from genomic data and show how the resulting information can be used to define microbial guilds with varying parameters.

Our approach to infer ecological traits from genomic data couples profile search methods with reconfigurable simple predicate logic. This coupling provides important advantages for deriving microbial traits from large numbers of phylogenetically diverse microbial genomes. Profile methods represent information across a family of evolutionarily related sequences from a multiple sequence alignment and increase sensitivity by incorporating position-specific information into a model. Moreover, the set of sequences from which gene-level *microTrait-HMMs* have been trained were selected from an extensive sequence database (IMG/M (Chen et al., 2019)) that not only includes genomes of cultured isolates but also MAGs and SAGs, the majority of which had been derived from environmental samples. Given that the bulk of the stream of incoming genomes from new studies is expected from MAGs with higher phylogenetic diversity compared to isolate genomes, the ability to detect remote homologs underlying microbial traits and explore sequence diversity from environmental samples is critical to increase the accuracy of trait prediction. With future releases of IMG, new sequences can be incorporated into multiple sequence alignments and consecutively *microTrait-HMM*s can be updated.

To benchmark and determine the score thresholds for each gene-level *microTrait-HMM*, we used the corresponding genes from the corresponding KO (KEGG Orthology) group. While this approach makes a systematic assessment of model accuracy possible by balancing model precision and recall, it should be noted that the computed thresholds may be overly strict for certain applications. Sequences in the KO database correspond to a highly curated set of sequences with a limited phylogenetic scope, this may lead to high precision and low recall with respect to the true labels especially for phylogenetically divergent or novel genomes not well represented in KEGG (Jaffe et al., 2020). Since the true orthologs for the underlying protein families are not known but can only be inferred, the accuracy of the model can only be estimated using independent labels such as those from KEGG. For applications where a higher recall at the expense of a lower precision is desired, it would be desirable to lower the HMM cutoff thresholds depending on the user input. We leave the implementation of such modifications for future work.

In this work, we focused on mechanistically well-studied traits whose genetic underpinnings have previously been documented and which can be conceptualized as Boolean rules. In addition to extraction of microbial traits with a rule-based system, further opportunities exist for unsupervised discovery of traits. For example, genomes with metadata labels determined experimentally or through text-mining (Alneberg et al., 2020) (Brbic et al., 2016) indicating the ecological niches of the organisms can be leveraged for exploring the genetic basis of organismal adaptation. Statistical modeling of the organismal niche and inference based on domain or gene content would be the classical approach towards this (Zhalnina et al., 2018; Ceja-Navarro et al., 2019). In addition, the exponential increase in the availability of high-quality MAGs with rich metadata will make feasible machine learning approaches that focus on prediction rather than explainability using a much larger number of features also feasible (Drouin et al., 2019).

Despite the increasing availability of genomic and physiological data of microbes, the adoption of trait-based approaches in microbial ecology is relatively recent. Unlike plants and animals, working definitions of microbial traits and conceptual frameworks to define functional guilds from these are lacking. The large diversity of microbial lifestyles manifest as a large number of potential traits some of which might be unobserved. Even with thousands of diverse genomes, the high-dimensionality of the potential trait space poses a challenge to define functional guilds for microbes. Here we adopted an operational definition of microbial guild as "groups consisting of diverse microorganisms with similar traits" based on a synthesis of a relatively small number of master traits that define microbial lifestyles. Depending on the specific analysis goals, a user might want to fine tune the granularity at which traits are defined (e.g., selection of different pathway endpoints as in denitrification or transporter/enzyme substrate classification). In *microTrait*, the reconfigurability of the rules makes the exploration of the effect of different trait definitions on the microbial guilds possible and therefore enables a flexible microbial trait extraction pipeline.

Finally, a trait-based microbial ecology framework has the potential to integrate ecological and genomic data. For this promise to be achieved however, the availability of metadata on the provenance and biogeochemical/ecological identification of the underlying biological samples is essential. Environmental metadata give essential context for genome data but current isolation of metadata resources (GOLD (Mukherjee et al., 2019) and NCBI's BioSample (Barrett et al., 2012)) and lack of rich ontological and data standards hinder interoperability and reusability. Reusability of metadata is further hampered by inability to download metadata in bulk. Even within a single resource with a relatively consistent data schema, the fill rates for the existent terms are very low leading to existence of a large number of genomes without any usable metadata. For example, within 162,711 bacterial and archaeal GOLD genomes (accessed on 04/2021), only 17% had the Ecosystem field (GOLD: Study Fields: Ecosystem) completed with one of the three categories (Environmental, Engineered, or Host). Among the Environmental genomes, only ~41% (7,868 genomes) had even the broadest ecosystem classification completed (GOLD: Study Fields: Ecosystem Category) leaving an overwhelming majority of genomes unusable. For a trait-based framework to fulfill its full potential in elucidating microbial trait-environment relationships, significant community efforts towards higher quality metadata standards and metadata enrichment such as that led by National Microbiome Data Collaborative (NMDC, https://microbiomedata.org/) towards higher quality metadata standards and metadata enrichment will be much needed.

METHODS

Implementation

microTrait is implemented in R. Besides R-base functions, it depends on R packages dplyr, tidyr, tidyverse, readr (Wickham, 2019; Hadley et al., 2018; Wickham et al., 2019; Wickham and Henry, 2019) for efficient data access, manipulation and storage, doMC (Weston and Calaway 2015) to implement multicore functionality. *microTrait* is available from https://github.com/ukaraoz/microtrait.

Construction of a Gene HMM Database of Protein Families (*microTrait-HMM*)

We constructed an HMM database that model gene loci underlying functional traits (called *microTrait-HMM*) based on archaeal and bacterial sequence diversity from 1) genomes of cultured organisms, 2) single cell genomes, 3) metagenome-assembled genomes, and 4) metagenomes from environmental, host associated and engineered microbiome samples. For each gene loci, a profile HMM was trained as follows. Seed protein sequences were collected from the non-redundant IMG/M database (img_core_v400) based on "EC Number", "Gene Symbol", and "IMG Term and Synonym" (Chen et al., 2019). Multiple sequences alignments (MSA) were generated from the seed sequences using MAFFT with an accuracy-oriented parameter set (--maxiterate 1,000 --localpair--anysymbol) (Katoh et al., 2005). Profile HMMs were built with HMMER/hmmbuild (Eddy 2008). We call the set of HMMs *microTrait-HMM* (**Supplementary Table S2**). All seed sequences, MSAs, and profile HMMs are available at https://github.com/ukaraoz/microtrait-hmm.

Estimation of Life History Traits (Minimal Doubling Time and Optimum Growth Temperature)

To estimate minimal doubling time from genome-wide codon usage bias, *microTrait* uses gRodon R package (Weissman et al., 2021) using multiple linear regression models trained on the dataset of maximal growth rates compiled by Vieira-Silva and Rocha (Vieira-Silva and Rocha 2010). Optimum growth temperature is estimated with the multiple linear regression models based on the same features of tRNA and 16S rRNA genes, ORFs and translated ORFs determined by Sauer and Wang (Sauer and Wang 2019), but reimplementing their python pipeline in R as part of the *microTrait* package itself to increase computational efficiency.

Inference of Guilds

Ecological guilds were inferred from *microTrait* trait matrix with variance partitioning and clustering analysis. Trait values for "count traits" were normalized by genome size to express them as "per base-pair genomic investments". The normalized trait matrix was used to calculate genome-to-genome distances using Wishart distance metric for mixed variable data (Wishart 2003) as implemented in R kmed package. Wishart distance is similar to the Gower distance (Gower 1971) for mixed variable data but applies a variance weight rather than a range for the numerical variables and uses a squared distance component. The resulting distance matrix was used to cluster genomes using hierarchical clustering with complete linkage. Next, we quantified variance in the genome to genome distances as a function of the number of defined guilds. We first cut the tree from hierarchical clustering into clusters ranging from 2 clusters to the total number of genomes in the dataset. Then, for each cut that corresponds to a given number of clusters, we quantified the variance in the distance matrix using cluster identity as a source of variation (using adonis2 in R vegan package) and plotted the resulting coefficient of determination (R^2) as a function of the number of clusters. This allows the user the option to pick the number of guilds capturing a given level of trait variance across the dataset, and vice versa. Given a threshold for a trait variance or a number of guilds, we then assign each genome to a guild based on the corresponding tree cut from hierarchical clustering. Finally, we visualize the trait profiles for the defined guilds using trait positivity as a metric.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article **Supplementary Material**, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

microTrait was conceived by UK and EB. UK developed the code, performed the computational analyses and wrote the original draft of the manuscript. EB contributed to the writing, review, and editing of the manuscript.

FUNDING

This work was supported by the Watershed Function Science Focus Area, and the Belowground Biogeochemistry Science Focus Areas at Lawrence Berkeley National Laboratory, funded by the US Department of Energy, Office of Science, Office of Biological and Environmental Research, Environmental System Science program under Award No. DE-AC02-05CH11231.

ACKNOWLEDGMENTS

This manuscript benefited from discussions with organizers and participants at the US National Institute for Mathematical and Biological Synthesis (NIMBioS) Pan-microbial Trait Ecology Workshop June 14–16 2017 at the University of Tennessee, Knoxville, for which we are grateful.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.918853/full#supplementary-material

Supplementary Figure S1 | Performance of *microTrait*-HMMs with respect to cross-reference to KEGG orthologous families (KO). Each point corresponds to a gene-level HMM with the estimated sensitivity (true positive rate) and specificity (as false positive rate or 1-specificity) corresponding to the scoring threshold that maximizes F-score. The inset shows the cumulative distribution for the maximum F-scores.

Supplementary Figure S2 | *microTrait* runtimes. Distribution of running times for isolate and metagenome-assembled genome sets normalized for genome size (measured as time (minutes) per Mb of sequence). Each point in the distribution corresponds to a genome. The normalized running times depend on the genome content, with more HMM hits requiring longer processing.

Supplementary Figure S3 | Refinement of functional guilds using *microTrait*.

Supplementary Figure S4 | Example *microTrait* trait matrix for soil isolate genomes as in **Figure 5A**, in high resolution.

Supplementary Table S1 | *microTrait* hierarchy. Hierarchical mapping of genome-derived features into ecological function of increasing granularity in *microTrait*. *microTrait* hierarchy is an unbalanced hierarchy with 3 levels, with certain leaves spanning all 3 levels. References supporting the inference of traits from genome derived features are given in **Supplementary Table S8**.

Supplementary Table S2 | *microTrait* HMMs. List of gene-level HMMs underlying *microTrait* pipeline (“*microTrait*-HMMs”), with cross-references (“dbxref”) to KEGG, EC, and Transporter Classification Database.

Supplementary Table S3 | Evaluation of *microTrait* HMMs. Performance of *microTrait*-HMMs with respect to cross-reference to KEGG orthologous families (KO). For each model, the model score maximizing F-score for the corresponding KO is used as a trusted cutoff.

Supplementary Table S4 | *microTrait* rules. Each *microTrait* rule is a boolean expression for presence/absence of *microTrait* HMMs or other *microTrait* rules.

Supplementary Table S5 | Mapping of *microTrait* rules to the *microTrait* hierarchy. *microTrait* traits are either of type binary or count. Count traits can be counted by themselves or by their substrate (microtrait_rule-type = “count_by_substrate”) in case of transporters. Refer to ST6 for the mapping between substrates and the *microTrait* hierarchy.

Supplementary Table S6 | Classification of substrates for substrate uptake and degradation by chemical class.

Supplementary Table S7 | *microTrait* traits by strategy, type (i.e. binary, count), and granularity.

Supplementary Table S8 | References for genome-derived features underlying ecological traits.

Supplementary Table S9 | Selected GOLD genomes of organisms isolated from aquatic or terrestrial environments. Environmental isolate genomes (GOLD_organisms:Cultured == “Yes” AND GOLD_organisms:Ecosystem == “Environmental”) from GOLD database (https://gold.jgi.doe.gov/) were selected and filtered using ecosystem category and sample collection site (GOLD_organisms:Ecosystem Category == “Aquatic OR Terrestrial” OR GOLD_organisms:Sample Collection Site (MIGS-13) == “soil OR sediment OR rhizosphere”).

Supplementary Table S10 | Taxonomic breakdown of selected GOLD genomes.

Supplementary Table S11 | Mapping between taxa and functional groups based on Faprotax database. Faprotax (Functional Annotation of Prokaryotic Taxa) (http://www.loucalab.com/archive/FAPROTAX/lib/php/index.php?section=Download) is a database that maps prokaryotic clades (e.g. class, order, family, genus, species) to metabolic functions. For comparison with *microTrait* rules for the same metabolic functions, we resolved the listed taxa names to standard names, which are listed in this table (column: taxa).

Supplementary Table S12 | Mapping of Faprotax taxa name to the NCBI taxa name.

Supplementary Table S13 | Functional group assignments with Faprotax and *microTrait*. Each GOLD genome was assigned to a Faprotax functional group by taxonomy (i.e. based on Faprotax database as in ST11) and by *microTrait* (i.e based on genome sequence).

Supplementary Table S14 | Evaluation of *microTrait* traits (genome-based) with respect to Faprotax functional groups (taxonomic name based). For each functional group, validity of *microTrait* predictions is evaluated based on Faprotax classifications (T: number of *microTrait* predicted positive genomes, N: number of *microTrait* predicted negative genomes, TP: number of true positive genomes, TN: number of true negative genomes, FP: number of false positive genomes, FN: number of false negative genomes, TPR: true positive rate, TNR: true negative rate).

Supplementary Table S15 | Correlations between traits. Spearman's rank correlation coefficient between pairs of traits.

Supplementary Table S16 | Guild trait profile matrix. Trait profiles (*microTrait* granularity 3) for defined guilds as mean trait values.

Supplementary Table S17 | Guild taxonomic profiles. Taxonomic profiles for defined guilds as relative abundance of genome taxonomy (phylum, class, order, family, genus).

REFERENCES

Ackerly, D. D., and Cornwell, W. K. (2007). A Trait-Based Approach to Community Assembly: Partitioning of Species Trait Values into within- and Among-Community Components. *Ecol. Lett.* 10 (2), 135–145. doi:10.1111/j.1461-0248.2006.01006.x

Allison, S. D. (2012). A Trait-Based Approach for Modelling Microbial Litter Decomposition. *Ecol. Lett.* 15 (9), 1058–1070. doi:10.1111/j.1461-0248.2012.01807.x

Alneberg, J., Bennke, C., Beier, S., Bunse, C., Quince, C., Ininbergs, K., et al. (2020). Ecosystem-wide Metagenomic Binning Enables Prediction of Ecological Niches from Genomes. *Commun. Biol.* 3 (1), 119. doi:10.1038/s42003-020-0856-x

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic Local Alignment Search Tool. *J. Mol. Biol.* 215 (3), 403–410. doi:10.1016/S0022-2836(05)80360-2

Anantharaman, K., Brown, C. T., Hug, L. A., Sharon, I., Castelle, C. J., Probst, A. J., et al. (2016). Thousands of Microbial Genomes Shed Light on Interconnected Biogeochemical Processes in an Aquifer System. *Nat. Commun.* 7, 13219. doi:10.1038/ncomms13219

Arnosti, C. (2011). Microbial Extracellular Enzymes and the Marine Carbon Cycle. *Ann. Rev. Mar. Sci.* 3, 401–425. doi:10.1146/annurev-marine-120709-142731

Arnosti, C., Bell, C., Moorhead, D. L., Sinsabaugh, R. L., Steen, A. D., Stromberger, M., et al. (2014). Extracellular Enzymes in Terrestrial, Freshwater, and Marine Environments: Perspectives on System Variability and Common Research Needs. *Biogeochemistry* 117 (1), 5–21. doi:10.1007/s10533-013-9906-5

Asshauer, K. P., Wemheuer, B., Daniel, R., and Meinicke, P. (2015). Tax4Fun: Predicting Functional Profiles from Metagenomic 16S rRNA Data. *Bioinformatics* 31 (17), 2882–2884. doi:10.1093/bioinformatics/btv287

Author Anonymous (1999). IUPAC-IUBMB Joint Commission on Biochemical Nomenclature (JCBN) and Nomenclature Committee of IUBMB (NC-IUBMB), Newsletter 1999. *Eur. J. Biochem.* 264 (2), 607–609. doi:10.1046/j.1432-1327.1999.news99.x

Barrett, T., Clark, K., Gevorgyan, R., Gorelenkov, V., Gribov, E., Karsch-Mizrachi, I., et al. (2012). BioProject and BioSample Databases at NCBI: Facilitating Capture and Organization of Metadata. *Nucleic Acids Res.* 40, D57–D63. Database issue). doi:10.1093/nar/gkr1163

Bergauer, K., Fernandez-Guerra, A., Garcia, J. A. L., Sprenger, R. R., Stepanauskas, R., Pachiadaki, M. G., et al. (2018). Organic Matter Processing by Microbial Communities throughout the Atlantic Water Column as Revealed by Metaproteomics. *Proc. Natl. Acad. Sci. U. S. A.* 115 (3), E400–E408. doi:10.1073/pnas.1708779115

Berntsson, R. P., Smits, S. H., Schmitt, L., Slotboom, D. J., and Poolman, B. (2010). A Structural Classification of Substrate-Binding Proteins. *FEBS Lett.* 584 (12), 2606–2617. doi:10.1016/j.febslet.2010.04.043

Bier, R. L., Bernhardt, E. S., Boot, C. M., Graham, E. B., Hall, E. K., Lennon, J. T., et al. (2015). Linking Microbial Community Structure and Microbial Processes: an Empirical and Conceptual Overview. *FEMS Microbiol. Ecol.* 91 (10). doi:10.1093/femsec/fiv113

Bock, C., Farlik, M., and Sheffield, N. C. (2016). Multi-Omics of Single Cells: Strategies and Applications. *Trends Biotechnol.* 34 (8), 605–608. doi:10.1016/j.tibtech.2016.04.004

Bouskill, N. J., Tang, J., Riley, W. J., and Brodie, E. L. (2012). Trait-based Representation of Biological Nitrification: Model Development, Testing, and Predicted Community Composition. *Front. Microbiol.* 3, 364. doi:10.3389/fmicb.2012.00364

Brbic, M., Piskorec, M., Vidulin, V., Krisko, A., Smuc, T., and Supek, F. (2016). The Landscape of Microbial Phenotypic Traits and Associated Genes. *Nucleic Acids Res.* 44 (21), 10074–10090.

Brenner, S. E., Chothia, C., and Hubbard, T. J. (1998). Assessing Sequence Comparison Methods with Reliable Structurally Identified Distant Evolutionary Relationships. *Proc. Natl. Acad. Sci. U. S. A.* 95 (11), 6073–6078. doi:10.1073/pnas.95.11.6073

Buchfink, B., Xie, C., and Huson, D. H. (2015). Fast and Sensitive Protein Alignment Using DIAMOND. *Nat. Methods* 12 (1), 59–60. doi:10.1038/nmeth.3176

Ceja-Navarro, J. A., Karaoz, U., Bill, M., Hao, Z., White, R. A., 3rd, Arellano, A., et al. (2019). Gut Anatomical Properties and Microbial Functional Assembly Promote Lignocellulose Deconstruction and Colony Subsistence of a Wood-Feeding Beetle. *Nat. Microbiol.* 4 (5), 864–875. doi:10.1038/s41564-019-0384-y

Chen, I. A., Chu, K., Palaniappan, K., Pillay, M., Ratner, A., Huang, J., et al. (2019). IMG/M v.5.0: an Integrated Data Management and Comparative Analysis System for Microbial Genomes and Microbiomes. *Nucleic Acids Res.* 47 (D1), D666–D677. doi:10.1093/nar/gky901

Clark, I. C., Melnyk, R. A., Engelbrektson, A., and Coates, J. D. (2013). Structure and Evolution of Chlorate Reduction Composite Transposons. *mBio* 4 (4). doi:10.1128/mBio.00379-13

Costa, O. Y. A., Raaijmakers, J. M., and Kuramae, E. E. (2018). Microbial Extracellular Polymeric Substances: Ecological Function and Impact on Soil Aggregation. *Front. Microbiol.* 9, 1636. doi:10.3389/fmicb.2018.01636

Courty, P. E., and Wipf, D. (2016). Editorial: Transport in Plant Microbe Interactions. *Front. Plant Sci.* 7, 809. doi:10.3389/fpls.2016.00809

Csonka, L. N. (1989). Physiological and Genetic Responses of Bacteria to Osmotic Stress. *Microbiol. Rev.* 53 (1), 121–147. doi:10.1128/mr.53.1.121-147.1989

Dombrowski, N., Teske, A. P., and Baker, B. J. (2018). Expansive Microbial Metabolic Versatility and Biodiversity in Dynamic Guaymas Basin Hydrothermal Sediments. *Nat. Commun.* 9 (1), 4999. doi:10.1038/s41467-018-07418-0

Drouin, A., Letarte, G., Raymond, F., Marchand, M., Corbeil, J., and Laviolette, F. (2019). Interpretable Genotype-To-Phenotype Classifiers with Performance Guarantees. *Sci. Rep.* 9 (1), 4071. doi:10.1038/s41598-019-40561-2

Eddy, S. R. (2008). A Probabilistic Model of Local Sequence Alignment that Simplifies Statistical Significance Estimation. *PLoS Comput. Biol.* 4 (5), e1000069. doi:10.1371/journal.pcbi.1000069

Edgar, R. C. (2010). Search and Clustering Orders of Magnitude Faster Than BLAST. *Bioinformatics* 26 (19), 2460–2461. doi:10.1093/bioinformatics/btq461

Falkowski, P. G., Fenchel, T., and Delong, E. F. (2008). The Microbial Engines that Drive Earth's Biogeochemical Cycles. *Science* 320 (5879), 1034–1039. doi:10.1126/science.1153213

Feder, M. E., and Hofmann, G. E. (1999). Heat-shock Proteins, Molecular Chaperones, and the Stress Response: Evolutionary and Ecological Physiology. *Annu. Rev. Physiol.* 61, 243–282. doi:10.1146/annurev.physiol.61.1.243

Feldbauer, R., Schulz, F., Horn, M., and Rattei, T. (2015). Prediction of Microbial Phenotypes Based on Comparative Genomics. *BMC Bioinforma.* 16 (Suppl. 14), S1. doi:10.1186/1471-2105-16-S14-S1

Finlay, B. J., Maberly, S. C., and Cooper, J. I. (1997). Microbial Diversity and Ecosystem Function. *Oikos* 80 (2), 209–213. doi:10.2307/3546587

Flamholz, A., Noor, E., Bar-Even, A., Liebermeister, W., and Milo, R. (2013). Glycolytic Strategy as a Tradeoff between Energy Yield and Protein Cost. *Proc. Natl. Acad. Sci. U. S. A.* 110 (24), 10039–10044. doi:10.1073/pnas.1215283110

Follows, M. J., Dutkiewicz, S., Grant, S., and Chisholm, S. W. (2007). Emergent Biogeography of Microbial Communities in a Model Ocean. *Science* 315 (5820), 1843–1846. doi:10.1126/science.1138544

Gao, H., Mao, Y., Zhao, X., Liu, W. T., Zhang, T., and Wells, G. (2019). Genome-centric Metagenomics Resolves Microbial Diversity and Prevalent Truncated Denitrification Pathways in a Denitrifying PAO-Enriched Bioprocess. *Water Res.* 155, 275–287. doi:10.1016/j.watres.2019.02.020

Goberna, M., and Verdú, M. (2016). Predicting Microbial Traits with Phylogenies. *ISME J.* 10 (4), 959–967. doi:10.1038/ismej.2015.171

Gower, J. C. (1971). A General Coefficient of Similarity and Some of its Properties. *Biometrics* 27 (4), 857–871. doi:10.2307/2528823

Green, J. L., Bohannan, B. J., and Whitaker, R. J. (2008). Microbial Biogeography: from Taxonomy to Traits. *Science* 320 (5879), 1039–1043. doi:10.1126/science.1153475

Gupta, A., Kumar, S., Prasoodanan, V. P., Harish, K., Sharma, A. K., and Sharma, V. K. (2016). Reconstruction of Bacterial and Viral Genomes from Multiple Metagenomes. *Front. Microbiol.* 7, 469. doi:10.3389/fmicb.2016.00469

Hadley, W., Hester, J., and Francois, R. (2018). *Readr: Read Rectangular Text Data.*

Hecker, M., and Völker, U. (2001). General Stress Response of Bacillus Subtilis and Other Bacteria. *Adv. Microb. Physiol.* 44, 35–91. doi:10.1016/s0065-2911(01)44011-2

Hyatt, D., Chen, G. L., Locascio, P. F., Land, M. L., Larimer, F. W., and Hauser, L. J. (2010). Prodigal: Prokaryotic Gene Recognition and Translation Initiation Site Identification. *BMC Bioinforma.* 11, 119. doi:10.1186/1471-2105-11-119

Jaffe, A. L., Castelle, C. J., Matheus Carnevali, P. B., Gribaldo, S., and Banfield, J. F. (2020). The Rise of Diversity in Metabolic Platforms across the Candidate Phyla Radiation. *BMC Biol.* 18 (1), 69. doi:10.1186/s12915-020-00804-5

Jetten, M., Schmid, M., van de Pas-Schoonen, K., Sinninghe Damsté, J., and Strous, M. (2005). Anammox Organisms: Enrichment, Cultivation, and Environmental Analysis. *Methods Enzymol.* 397, 34–57. doi:10.1016/S0076-6879(05)97003-1

Johnstone, I. M., and Titterington, D. M. (2009). Statistical Challenges of High-Dimensional Data. *Philos. Trans. A Math. Phys. Eng. Sci.* 367, 4237–4253. doi:10.1098/rsta.2009.0159

Jones, C. M., Spor, A., Brennan, F. P., Breuil, M.-C., Bru, D., Lemanceau, P., et al. (2014). Recently Identified Microbial Guild Mediates Soil N2O Sink Capacity. *Nat. Clim. Change* 4 (9), 801–805. doi:10.1038/nclimate2301

Kanehisa, M., and Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.* 28 (1), 27–30. doi:10.1093/nar/28.1.27

Katoh, K., Kuma, K., Miyata, T., and Toh, H. (2005). Improvement in the Accuracy of Multiple Sequence Alignment Program MAFFT. *Genome Inf.* 16 (1), 22–33.

Ko, R., Smith, L. T., and Smith, G. M. (1994). Glycine Betaine Confers Enhanced Osmotolerance and Cryotolerance on Listeria Monocytogenes. *J. Bacteriol.* 176 (2), 426–431. doi:10.1128/jb.176.2.426-431.1994

Lajoie, G., and Kembel, S. W. (2019). Making the Most of Trait-Based Approaches for Microbial Ecology. *Trends Microbiol.* 27 (10), 814–823. doi:10.1016/j.tim.2019.06.003

Langille, M. G., Zaneveld, J., Caporaso, J. G., McDonald, D., Knights, D., Reyes, J. A., et al. (2013). Predictive Functional Profiling of Microbial Communities Using 16S rRNA Marker Gene Sequences. *Nat. Biotechnol.* 31 (9), 814–821. doi:10.1038/nbt.2676

Laughlin, D. C. (2014). The Intrinsic Dimensionality of Plant Traits and its Relevance to Community Assembly. *J. Ecol.* 102 (1), 186–193. doi:10.1111/1365-2745.12187

Liu, S., Chen, Q., Ma, T., Wang, M., and Ni, J. (2018). Genomic Insights into Metabolic Potentials of Two Simultaneous Aerobic Denitrification and Phosphorus Removal Bacteria, Achromobacter Sp. GAD3 and Agrobacterium Sp. LAD9. *FEMS Microbiol. Ecol.* 94 (4). doi:10.1093/femsec/fiy020

Louca, S., Parfrey, L. W., and Doebeli, M. (2016). Decoupling Function and Taxonomy in the Global Ocean Microbiome. *Science* 353 (6305), 1272–1277. doi:10.1126/science.aaf4507

Lycus, P., Lovise Bøthun, K., Bergaust, L., Peele Shapleigh, J., Reier Bakken, L., and Frostegård, Å. (2017). Phenotypic and Genotypic Richness of Denitrifiers Revealed by a Novel Isolation Strategy. *ISME J.* 11 (10), 2219–2232. doi:10.1038/ismej.2017.82

Madin, J. S., Nielsen, D. A., Brbic, M., Corkrey, R., Danko, D., Edwards, K., et al. (2020). A Synthesis of Bacterial and Archaeal Phenotypic Trait Data. *Sci. Data* 7 (1), 170. doi:10.1038/s41597-020-0497-4

Malik, A. A., Martiny, J. B. H., Brodie, E. L., Martiny, A. C., Treseder, K. K., and Allison, S. D. (2020). Defining Trait-Based Microbial Strategies with Consequences for Soil Carbon Cycling under Climate Change. *ISME J.* 14 (1), 1–9. doi:10.1038/s41396-019-0510-0

Malik, A. A., Martiny, J. B. H., Brodie, E. L., Martiny, A. C., Treseder, K. K., and Allison, S. D. (2018). Defining Trait-Based Microbial Strategies with Consequences for Soil Carbon Cycling under Climate Change. *bioRxiv*, 445866.

Martiny, A. C., Treseder, K., and Pusch, G. (2013). Phylogenetic Conservatism of Functional Traits in Microorganisms. *ISME J.* 7 (4), 830–838. doi:10.1038/ismej.2012.160

Mindock, C. A., Petrova, M. A., and Hollingswort, R. I. (2001). Re-evaluation of Osmotic Effects as a General Adaptative Strategy for Bacteria in Sub-freezing Conditions. *Biophys. Chem.* 89 (1), 13–24. doi:10.1016/s0301-4622(00)00214-3

Mukherjee, S., Stamatis, D., Bertsch, J., Ovchinnikova, G., Katta, H. Y., Mojica, A., et al. (2019). Genomes OnLine Database (GOLD) v.7: Updates and New Features. *Nucleic Acids Res.* 47 (D1), D649–D659. doi:10.1093/nar/gky977

Parks, D. H., Rinke, C., Chuvochina, M., Chaumeil, P. A., Woodcroft, B. J., Evans, P. N., et al. (2017). Recovery of Nearly 8,000 Metagenome-Assembled Genomes Substantially Expands the Tree of Life. *Nat. Microbiol.* 2 (11), 1533–1542. doi:10.1038/s41564-017-0012-7

Prosser, J. I., Bohannan, B. J., Curtis, T. P., Ellis, R. J., Firestone, M. K., Freckleton, R. P., et al. (2007). The Role of Ecological Theory in Microbial Ecology. *Nat. Rev. Microbiol.* 5 (5), 384–392. doi:10.1038/nrmicro1643

Prosser, J. I. (2015). Dispersing Misconceptions and Identifying Opportunities for the Use of 'omics' in Soil Microbial Ecology. *Nat. Rev. Microbiol.* 13 (7), 439–446. doi:10.1038/nrmicro3468

Ramirez, K. S., Knight, C. G., de Hollander, M., Brearley, F. Q., Constantinides, B., Cotton, A., et al. (2018). Detecting Macroecological Patterns in Bacterial Communities across Independent Studies of Global Soils. *Nat. Microbiol.* 3 (2), 189–196. doi:10.1038/s41564-017-0062-x

Ramírez-Flandes, S., González, B. O., and Ulloa, O. (2019). Redox Traits Characterize the Organization of Global Microbial Communities. *Proc. Natl. Acad. Sci. U. S. A.* 116 (9), 3630–3635. doi:10.1073/pnas.1817554116

Ruan, J. (2013). Bergey's Manual of Systematic Bacteriology (Second Edition) Volume 5 and the Study of Actinomycetes Systematic in China. *Wei Sheng Wu Xue Bao* 53 (6), 521–530.

Saier, M. H., Jr., Reddy, V. S., Tsu, B. V., Ahmed, M. S., Li, C., and Moreno-Hagelsieb, G. (2016). The Transporter Classification Database (TCDB): Recent Advances. *Nucleic Acids Res.* 44 (D1), D372–D379. doi:10.1093/nar/gkv1103

Sanford, R. A., Wagner, D. D., Wu, Q., Chee-Sanford, J. C., Thomas, S. H., Cruz-García, C., et al. (2012). Unexpected Nondenitrifier Nitrous Oxide Reductase Gene Diversity and Abundance in Soils. *Proc. Natl. Acad. Sci. U. S. A.* 109 (48), 19709–19714. doi:10.1073/pnas.1211238109

Sangwan, N., Xia, F., and Gilbert, J. A. (2016). Recovering Complete and Draft Population Genomes from Metagenome Datasets. *Microbiome* 4, 8. doi:10.1186/s40168-016-0154-5

Sauer, D. B., and Wang, D. N. (2019). Predicting the Optimal Growth Temperatures of Prokaryotes Using Only Genome Derived Features. *Bioinformatics* 35 (18), 3224–3231. doi:10.1093/bioinformatics/btz059

Shaffer, M., Borton, M. A., McGivern, B. B., Zayed, A. A., La Rosa, S. L., Solden, L. M., et al. (2020). DRAM for Distilling Microbial Metabolism to Automate the Curation of Microbiome Function. *bioRxiv* 48 (16). 8883–8900. doi:10.1093/nar/gkaa621

Sharon, I., and Banfield, J. F. (2013). Microbiology. Genomes from Metagenomics. *Science* 342 (6162), 1057–1058. doi:10.1126/science.1247023

Sing, T., Sander, O., Beerenwinkel, N., and Lengauer, T. (2005). ROCR: Visualizing Classifier Performance in R. *Bioinformatics* 21 (20), 3940–3941. doi:10.1093/bioinformatics/bti623

Thompson, L. R., Sanders, J. G., McDonald, D., Amir, A., Ladau, J., Locey, K. J., et al. (2017). A Communal Catalogue Reveals Earth's Multiscale Microbial Diversity. *Nature* 551 (7681), 457–463. doi:10.1038/nature24621

Todd-Brown, K. E. O., Hopkins, F. M., Kivlin, S. N., Talbot, J. M., and Allison, S. D. (2012). A Framework for Representing Microbial Decomposition in Coupled Climate Models. *Biogeochemistry* 109 (1), 19–33. doi:10.1007/s10533-011-9635-6

Turaev, D., and Rattei, T. (2016). High Definition for Systems Biology of Microbial Communities: Metagenomics Gets Genome-Centric and Strain-Resolved. *Curr. Opin. Biotechnol.* 39, 174–181. doi:10.1016/j.copbio.2016.04.011

Van Der Heijden, M. G., Bardgett, R. D., and Van Straalen, N. M. (2008). The Unseen Majority: Soil Microbes as Drivers of Plant Diversity and Productivity in Terrestrial Ecosystems. *Ecol. Lett.* 11 (3), 296–310. doi:10.1111/j.1461-0248.2007.01139.x

Vieira-Silva, S., and Rocha, E. P. (2010). The Systemic Imprint of Growth and its Uses in Ecological (Meta)genomics. *PLoS Genet.* 6 (1), e1000808. doi:10.1371/journal.pgen.1000808

Violle, C., Reich, P. B., Pacala, S. W., Enquist, B. J., and Kattge, J. (2014). The Emergence and Promise of Functional Biogeography. *Proc. Natl. Acad. Sci. U. S. A.* 111 (38), 13690–13696. doi:10.1073/pnas.1415442111

Violle, C., Navas, M.-L., Vile, D., Kazakou, E., Fortunel, C., Hummel, I., et al. (2007). Let the Concept of Trait Be Functional!. *Oikos* 116, 882–892. doi:10.1111/j.0030-1299.2007.15559.x

Wang, D., and Bodovitz, S. (2010). Single Cell Analysis: the New Frontier in 'omics'. *Trends Biotechnol.* 28 (6), 281–290. doi:10.1016/j.tibtech.2010.03.002

Weider, L. J., Elser, J. J., Crease, T. J., Mateos, M., Cotner, J. B., and Markow, T. A. (2005). The Functional Significance of Ribosomal (R)DNA Variation: Impacts on the Evolutionary Ecology of Organisms. *Annu. Rev. Ecol. Evol. Syst.* 36 (1), 219–242. doi:10.1146/annurev.ecolsys.36.102003.152620

Weimann, A., Mooren, K., Frank, J., Pope, P. B., Bremges, A., and McHardy, A. C. (2016). From Genomes to Phenotypes: Traitar, the Microbial Trait Analyzer. *mSystems* 1 (6). doi:10.1128/mSystems.00101-16

Weissman, J. L., Hou, S., and Fuhrman, J. A. (2021). Estimating Maximal Microbial Growth Rates from Cultures, Metagenomes, and Single Cells via Codon Usage Patterns. *Proc. Natl. Acad. Sci. U. S. A.* 118 (12). doi:10.1073/pnas.2016810118

Westoby, M., and Wright, I. J. (2006). Land-plant Ecology on the Basis of Functional Traits. *Trends Ecol. Evol.* 21 (5), 261–268. doi:10.1016/j.tree.2006.02.004

Weston, S., and Calaway, R. (2015). *doMC: Foreach Parallel Adaptor for 'parallel'.*

Wickham, H. (2019). Welcome to the Tidyverse. *J. Open Source Softw.* 4 (43), 1686. doi:10.21105/joss.01686

Wickham, H., and Henry, L. (2019). *Tidyr: Easily Tidy Data with 'spread()' and 'gather()' Functions*.

Wickham, H., Francois, R., Henry, L., and Müller, K. (2019). *Dplyr: A Grammar of Data Manipulation*.

Wishart, D. (2003). *K-Means Clustering with Outlier Detection, Mixed Variables and Missing ValuesExploratory Data Analysis in Empirical Research Studies in Classification, Data Analysis, and Knowledge Organization*. Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-55721-7_23

Woodcroft, B. J., Singleton, C. M., Boyd, J. A., Evans, P. N., Emerson, J. B., Zayed, A. A. F., et al. (2018). Genome-centric View of Carbon Processing in Thawing Permafrost. *Nature* 560 (7716), 49–54. doi:10.1038/s41586-018-0338-1

Yabuuchi, E. (2001). Current Topics on Classification and Nomenclature of Bacteria. 7. Taxonomic Outline of Archeae and Bacteria in the Second Edition of Bergey's Manual of Systematic Bacteriology. *Kansenshogaku Zasshi* 75 (8), 653–655. doi:10.11150/kansenshogakuzasshi1970.75.653

Yin, Y., Mao, X., Yang, J., Chen, X., Mao, F., and Xu, Y. (2012). dbCAN: a Web Resource for Automated Carbohydrate-Active Enzyme Annotation. *Nucleic Acids Res.* 40, W445–W451. Web Server issue. doi:10.1093/nar/gks479

Yu, A., Li, P., Tang, T., Wang, J., Chen, Y., and Liu, L. (2015). Roles of Hsp70s in Stress Responses of Microorganisms, Plants, and Animals. *Biomed. Res. Int.*, 510319. doi:10.1155/2015/510319

Zeldovich, K. B., Berezovsky, I. N., and Shakhnovich, E. I. (2007). Protein and DNA Sequence Determinants of Thermophilic Adaptation. *PLoS Comput. Biol.* 3 (1), e5. doi:10.1371/journal.pcbi.0030005

Zhalnina, K., Louie, K. B., Hao, Z., Mansoori, N., da Rocha, U. N., Shi, S., et al. (2018). Dynamic Root Exudate Chemistry and Microbial Substrate Preferences Drive Patterns in Rhizosphere Microbial Community Assembly. *Nat. Microbiol.* 3 (4), 470–480. doi:10.1038/s41564-018-0129-3

Zimmerman, A. E., Martiny, A. C., and Allison, S. D. (2013). Microdiversity of Extracellular Enzyme Genes Among Sequenced Prokaryotic Genomes. *ISME J.* 7 (6), 1187–1199. doi:10.1038/ismej.2012.176

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Karaoz and Brodie. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ORIGINAL RESEARCH
published: 09 August 2022
doi: 10.3389/fbinf.2022.866902

Cronos: A Machine Learning Pipeline for Description and Predictive Modeling of Microbial Communities Over Time

Aristeidis Litos[1,2], *Evangelia Intze*[3], *Pavlos Pavlidis*[2] *and Ilias Lagkouvardos*[2,4]*

[1]School of Medicine, University of Crete, Heraklion, Greece, [2]Institute of Computer Science, Foundation of Research and Technology, Heraklion, Greece, [3]School of Science and Technology, Hellenic Open University, Patras, Greece, [4]Core Facility Microbiome—ZIEL Institute for Food and Health, Technical University of Munich, Freising, Germany

Microbial time-series analysis, typically, examines the abundances of individual taxa over time and attempts to assign etiology to observed patterns. This approach assumes homogeneous groups in terms of profiles and response to external effectors. These assumptions are not always fulfilled, especially in complex natural systems, like the microbiome of the human gut. It is actually established that humans with otherwise the same demographic or dietary backgrounds can have distinct microbial profiles. We suggest an alternative approach to the analysis of microbial time-series, based on the following premises: 1) microbial communities are organized in distinct clusters of similar composition at any time point, 2) these intrinsic subsets of communities could have different responses to the same external effects, and 3) the fate of the communities is largely deterministic given the same external conditions. Therefore, tracking the transition of communities, rather than individual taxa, across these states, can enhance our understanding of the ecological processes and allow the prediction of future states, by incorporating applied effects. We implement these ideas into Cronos, an analytical pipeline written in R. Cronos' inputs are a microbial composition table (e.g., OTU table), their phylogenetic relations as a tree, and the associated metadata. Cronos detects the intrinsic microbial profile clusters on all time points, describes them in terms of composition, and records the transitions between them. Cluster assignments, combined with the provided metadata, are used to model the transitions and predict samples' fate under various effects. We applied Cronos to available data from growing infants' gut microbiomes, and we observe two distinct trajectories corresponding to breastfed and formula-fed infants that eventually converge to profiles resembling those of mature individuals. Cronos is freely available at https://github.com/Lagkouvardos/Cronos.

Keywords: microbial profiles, microbiome, machine learning, De novo clustering, microbial communities, infant gut maturation, multinomial logistic regression, time-series

OPEN ACCESS

Edited by:
Joao Carlos Setubal,
University of São Paulo, Brazil

Reviewed by:
Aditya Mishra,
University of Texas MD Anderson Cancer Center, United States
Julia Pavan Soler,
University of São Paulo, Brazil

Correspondence:
Ilias Lagkouvardos
ilias.lagkouvardos@tum.de

Specialty section:
This article was submitted to Genomic Analysis, a section of the journal Frontiers in Bioinformatics

Received: *31 January 2022*
Accepted: *15 June 2022*
Published: *09 August 2022*

Citation:
Litos A, Intze E, Pavlidis P and Lagkouvardos I (2022) Cronos: A Machine Learning Pipeline for Description and Predictive Modeling of Microbial Communities Over Time. Front. Bioinform. 2:866902. doi: 10.3389/fbinf.2022.866902

1 INTRODUCTION

Advances in sequencing technologies allowed the investigation of diverse environments in terms of bacterial community structure as standardized practice (Mukherjee et al., 2021). Studies of microbial communities over time are steadily gaining in popularity compared with the majority of studies, in which a single time point is investigated, allowing for a further understanding of community dynamics.

Microbial communities consist of multiple species entangled in complex interactions that affect their individual behavior, overall system dynamics, and environmental niche properties (Stubbendieck et al., 2016). Internal phenomena include direct interactions, such as mutualism (Morris et al., 2013) or competition (Stubbendieck et al., 2016) and indirect interactions, such as quorum sensing (Miller and Bassler, 2001). Internal interactions in combination with external factors, such as antibiotics (Iizumi et al., 2017), infants' birth mode, or diet (Kim et al., 2019), affect the individual bacteria behavior and shape the environment landscape (Tan et al., 2021). Therefore, a complete understanding of microbial systems can only be achieved by studying the overall microbial communities rather than each microbial organism in isolation.

Time-series analysis of abundance and co-occurrence of microbes have been investigated mainly via traditional statistical methods (Chaffron et al., 2010; Steele et al., 2011). Several bioinformatic tools for bacterial time-series analysis have been developed, exploiting the increasing data availability. These tools, along with other studies, focus mainly on single or specific taxa and their relative abundance over time (Vergin et al., 2013; Sharon et al., 2013; Xia et al., 2011; Ki et al., 2018; Zhang et al., 2019). However, those approaches inherit the limitations and assumptions of the statistical methods used. Relying on experimental design labels may mask distinct patterns or structures in each group and therefore misinterpret the microbial community trajectories. Often, abundance values for a given group of samples at a time point can exhibit multiple modes implying the existence of more than one underlying distribution. Comparing values among time points with statistical methods relying on means or ranks is not appropriate for multimodal datasets.

In the first 2 years of life, the gut microbiome is subjected to many compositional changes (Bäckhed et al., 2015; Stewart et al., 2018). The procedure toward the adult microbiome is often called maturation (Mesa et al., 2020). Evidence suggests an association between infant gut bacteria and diet (Pannaraj et al., 2017; Jiang et al., 2018; Camacho-Morales et al., 2021), the way the infant was delivered (Jakobsson et al., 2014), antibiotic usage (Korpela et al., 2020; Lemas et al., 2016), maternal body mass index (Soderborg et al., 2018), or even environmental factors (Sugino et al., 2021). Alterations of the human gut microbiome during the maturation procedure motivate the analysis of microbiome profiles using time-series approaches.

In this study, we propose a novel framework for microbial community time-series data analysis. Embedded in an R-based tool, Cronos, is based on the following premises and concepts. Intrinsic microbial community structures within a time point are shaped due to specific attractor states (Estrela et al., 2022; Goldford et al., 2018). These states can be identified by unsupervised machine learning techniques. Microbial communities' evolution can be explored by capturing transitions among attractor states over time. We developed an implementation of this concept in Cronos software. Cronos applies machine learning techniques to analyze complete microbial profiles over time and describe the attractor states (Costea et al., 2018). Our software explores the microbial community profile evolution by capturing transitions among clusters over time. As a consequence, it is able to predict future community structure states. Cronos is freely available, as an open-source code at https://github.com/Lagkouvardos/Cronos.

2 MATERIALS AND METHODS

Cronos is an R script that performs the tasks of 1) dividing and labeling the samples based on the time points, 2) calculating the pairwise UniFrac distances among the samples at every time point, 3) performing *de novo* clustering of the samples profiles, 4) calculating and visualizing the taxonomic representation of clusters, 5) applying Markovian property test, 6) transition modeling based on given metadata, and 7) predicting future states.

Cronos functions rely on R packages ade4, dplyr, GUniFrac, phangorn, cluster, fpc, markovchain, spgs, caret, nnet, gtools, mclust, igraph, and network, which Cronos installs automatically if required, along with all of their dependencies. Cronos requires three files as inputs. A table of microbial profiles (e.g., OTU or ASV abundance tables), a mapping file containing information about the time points and the corresponding metadata of the samples, and a phylogenetic tree of all taxa in the profiles table.

2.1 *De novo* Clustering, Evaluation, and Validation

Cronos calculates the GUniFrac, a beta-diversity distance metric variant (Chen et al., 2012) of the UniFrac distance methods (Lozupone and Knight, 2005), for each pair of samples at every time point, using the phylogenetic tree input, to create a dissimilarity matrix. Then, *de novo* clustering is performed via the partitioning around medoid (PAM) method (Schubert and Rousseeuw, 2021; Costea et al., 2018). Cronos assesses the optimal number of clusters via the Calinski–Harabasz index.

Cronos applies a brute force method to select the optimal number of clusters at every time point. Clustering via PAM is performed using as the number of clusters (k) all the numbers between two and nine. Due to computational constraints, the maximum number of clusters was set to nine. The optimal number of clusters is assessed using the Calinski–Harabasz index (Calinski and Harabasz, 1974) also known as the variance ratio criterion, from the fpc R package. The Calinski–Harabasz index is translated into the ratio of the sum of between clusters dispersion to intercluster dispersion. Higher Calinski–Harabasz index values indicate better clustering performance.

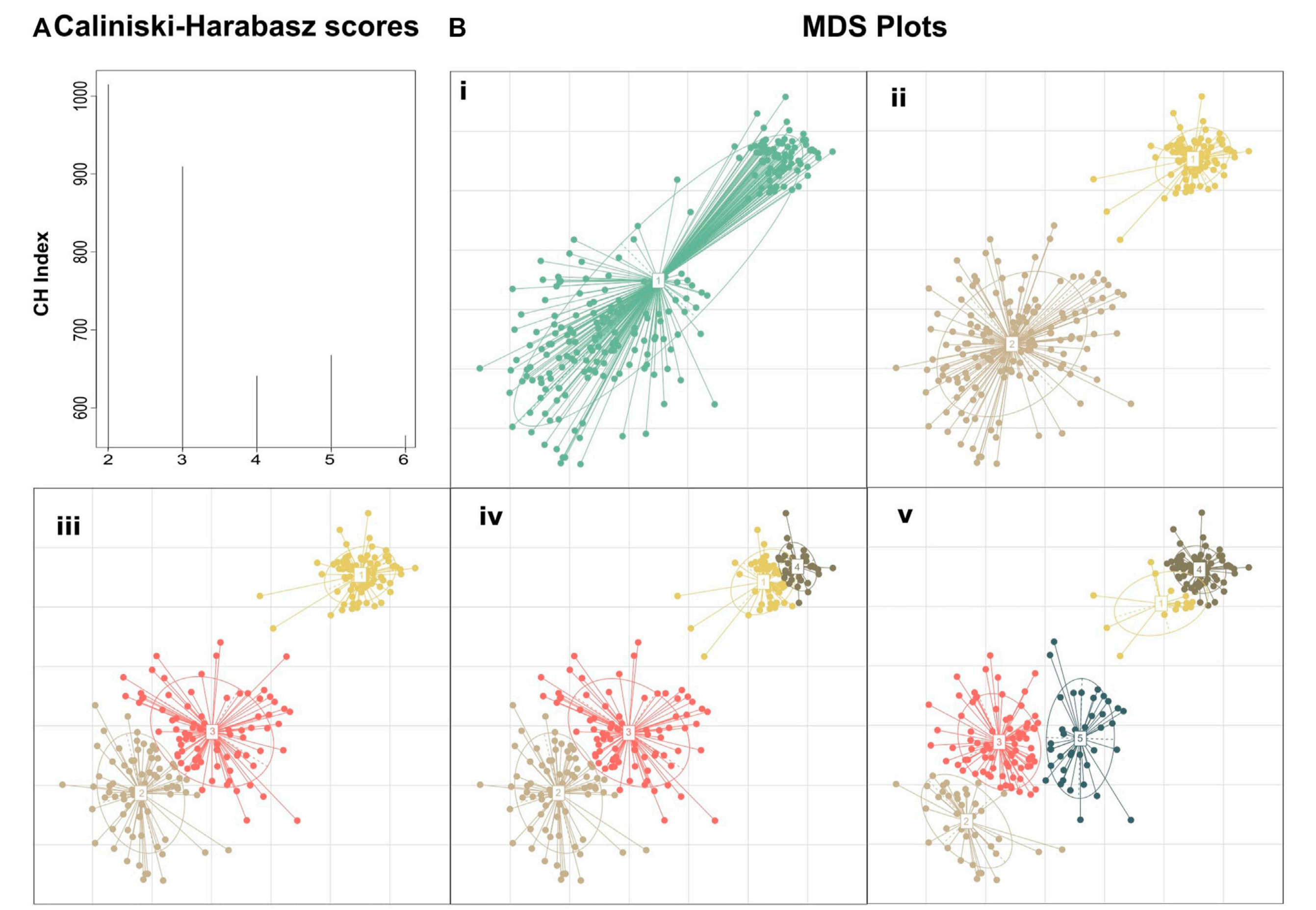


FIGURE 1 | Clustering performed on a manually created dataset demonstrating the added insight on the structure achieved by selecting the number of clusters based on clustering quality drop rather than simply using the clustering with the maximum score. **(A)** Calinski–Harabasz indexes calculated for PAM clustering with k ranging from 2 to 6. Although k = 2 shows the highest score suggesting the existence of two primary groupings in the dataset, k = 3 also fits well into the dataset, revealing the composite nature of the second cluster. Further subclustering results in a large drop in the clustering quality suggesting that the dataset does not fit well to the number of clusters **(B)** MDS plots of the dataset at different *de novo* clustering levels (i) MDS plot of the unclustered dataset, (ii–v) PAM clustering of the dataset for k = 2–5.

Calinski–Harabasz (s) index is calculated as

$$s = \left(\frac{tr\,(Bk)}{tr\,(Wk)} * \frac{n-k}{k-1}\right) \tag{1}$$

where n is the sample size divided into k clusters, tr (Bk) is the trace of the between cluster dispersion matrix, and tr (Wk) is the trace of the within-cluster dispersion matrix defined by

$$Wk = \sum_{p=1}^{k} \sum_{x \in C_p} \left(x - C_p\right)\left(x - C_p\right)^T \tag{2}$$

$$Bk = \sum_{p=1}^{k} n_p \left(C_p - C_E\right)\left(C_p - C_E\right)^T \tag{3}$$

where C_p is the set of points in cluster p, C_E the center of cluster E, and n_p the number of points in cluster p.

In order to achieve high clustering resolution but avoid overclustering, we determined the optimal number of clusters based on two rules: The maximum consecutive Calinski–Harabasz score difference and the difference between the absolute maximum of Calinski–Harabasz scores and the one with the highest difference. Such an approach, empirically, demonstrated both high clustering resolution and avoided meaningless overclustering.

First, we calculate the Calinski–Harabasz indexes for two to nine clusters. Second, we calculate the difference between Calinski–Harabasz indexes for every two consecutive numbers of clusters and select the highest. Third, we calculate the difference in Calinski–Harabasz scores between the preselected and the absolute maximum of CH scores.

$$k = \begin{cases} argmax\,(S_k) & if\, maxS_k - maxS_{argmax\,(S_k - S_{k+1})} & \geq |\max\,(S_k - S_{k+1})| \\ argmax\,(S_k - S_{k+1}) & if\, maxS_k - maxS_{argmax\,(S_k - S_{k+1})} & < |\max\,(S_k - S_{k+1})| \end{cases} \tag{4}$$

The optimal number of clusters is selected as the absolute maximum of Calinski–Harabasz scores if $maxS_k - maxS_{argmax(S_k - S_{k+1})} \geq |max(S_k - S_{k+1})|$ or the preselected k $maxS_k - maxS_{argmax(S_k - S_{k+1})} < |max(S_k - S_{k+1})|$.

The motivation behind this approach is that if we rely only on the maximum CH score, we will detect just a crude clustering of the data, overlooking, thus, any fine data clustering (**Figure 1**). By assessing the value of k by the Eq (4), we will obtain the highest possible resolution on a given time point (any further refinement will diminish the clustering quality) while keeping the CH score of clustering close to the absolute maximum score. To highlight this approach we created a hypothetical dataset manually derived from three Gaussian distributions with standard deviations of 0.1, 0.4, and 0.6 and means (6.5,6.5), (3,3), and (4,4), respectively. The absolute maximum Calinski–Harabasz value indicates that the optimal number of clusters for this dataset is 2, even though we manufactured the dataset from three different Gaussian distributions (**Figure 1**).

Since PAM clustering will divide the dataset into at least two groups even when data contain no clusters, Cronos also performs a validity check of clustering. To address this issue, we apply a Bayesian information criterion (BIC)-based methodology to evaluate whether k clusters (k > 1) are better than a scenario with no clusters for each time point. We apply Gaussian mixture model (GMM) clustering with 1 and the optimal number k of clusters as components, using the mclust R package. To compare the two clustering outcomes from GMM, the BIC score was calculated using the same R package.

2.2 Transition Analysis

Clustering at each timepoint results in the characterization of samples over time. To further understand the evolution of the microbiome profiles, Cronos primarily checks for the Markovian property of the transitions of clusters from each time point to the next. A transition acquires the Markovian property when it depends only on the current state and not on any previous one. A custom test was created to verify the first-order Markovian assumption (i.e., future state does not depend on the exact previous one but only the current) among the transitions of all samples based on the verifyMarkovProperty test of markovchain R package. The test examines all successive triplets of time points, in terms of states–cluster assignments. Let $x_1, x_2, \ldots, x_N$ be a set of observations with N the optimal number of clusters selected and n_{ijk} is the number of times t ($1 \leq t \leq N-2$) such that $x_t = i$, $x_{t+1} = j$, $x_{t+2} = k$; then, if the Markov property holds, n_{ijk} follows a Binomial distribution with parameters n_{ij} and p_{jk}.

A classical chi-square test can check this distributional assumption, since

$$\sum_i \sum_j \sum_k \frac{\left(n_{ijk} - n_{ij} p_{jk}\right)^2}{n_{ij} p_{jk}} \sim \chi^2(d) \quad (5)$$

where d is the number of degrees of freedom. The number of degrees of freedom d of the chi-square distribution is given by d = r − q + s − 1, where s denotes the number of states i in the state space such that $n_i > 0$, q denotes the number of pairs (i, j) for which $n_{ij} > 0$, and r denotes the number of triplets (i, j, k) for which $n_{ijk} > 0$.

2.3 Transition Modeling

Cronos models the states at each time point (response variable) as a function of the metadata at this time point and the state at a previous time point (explanatory variables) by applying multinomial logistic regression via the multinom function of the nnet R package. For each time point, we create a matrix of explanatory variables using the cluster label on a given time point and the metadata as columns and the samples as rows.

To evaluate the predictions, Cronos divides the dataset into training and test sets using two different methods. First, we apply a leave one out (LOO) procedure, where all the dataset is used to train the model except one sample, which is used as the test set. The second method refers to stratified splits, which is performed via the createDataPartition function of the caret R package and splits the dataset into train and test sets with the same ratio of samples per label.

Cronos evaluates the accuracy of classification as the percentage of correct predictions that the model made:

$$A = \frac{correctPredictions}{N} \quad (6)$$

where N is the number of samples on the set and returns the mean accuracy over a prespecified number of iterations for both the training and the test sets, all the division methods, and all the time points used to create the models. Mean accuracy of a model is calculated as follows:

$$Acc = \frac{1}{T} \sum_{i=1}^{T} \frac{correctPredictions}{N} \quad (7)$$

where N is the number of samples on the set and T is the number of iterations. Partitions with the LOO method are iterated over all samples, whereas the stratified splits method assigns samples on the test set ensuring that the train and test sets have approximately the same percentage of samples of each target class as the complete set.

Cronos performs classification to predict the cluster on all time points but the first, with both partitioning methods for all the possible combinations of metadata provided, combined with cluster assignment, including models without metadata, both for the training and test sets. The classification performance of Cronos is compared to the random classifier, which labels all the possible outcomes of the predicted variable with the same frequency. Cronos' complete pipeline is shown in **Figure 2**.

2.4 Cluster Representation

Every cluster of microbial profiles is represented via its medoid. Cronos describes every medoid composition at all taxonomic levels above the genus to provide further insight into its community structure via binning (cumulative abundance of all OTUs/ASVs belonging to the same taxon). Furthermore, the profiles are illustrated as barplots. To enhance the visualizations, there is an option to agglomerate low abundance taxa into the category called "Others" using a selected by the user threshold (default 5%).

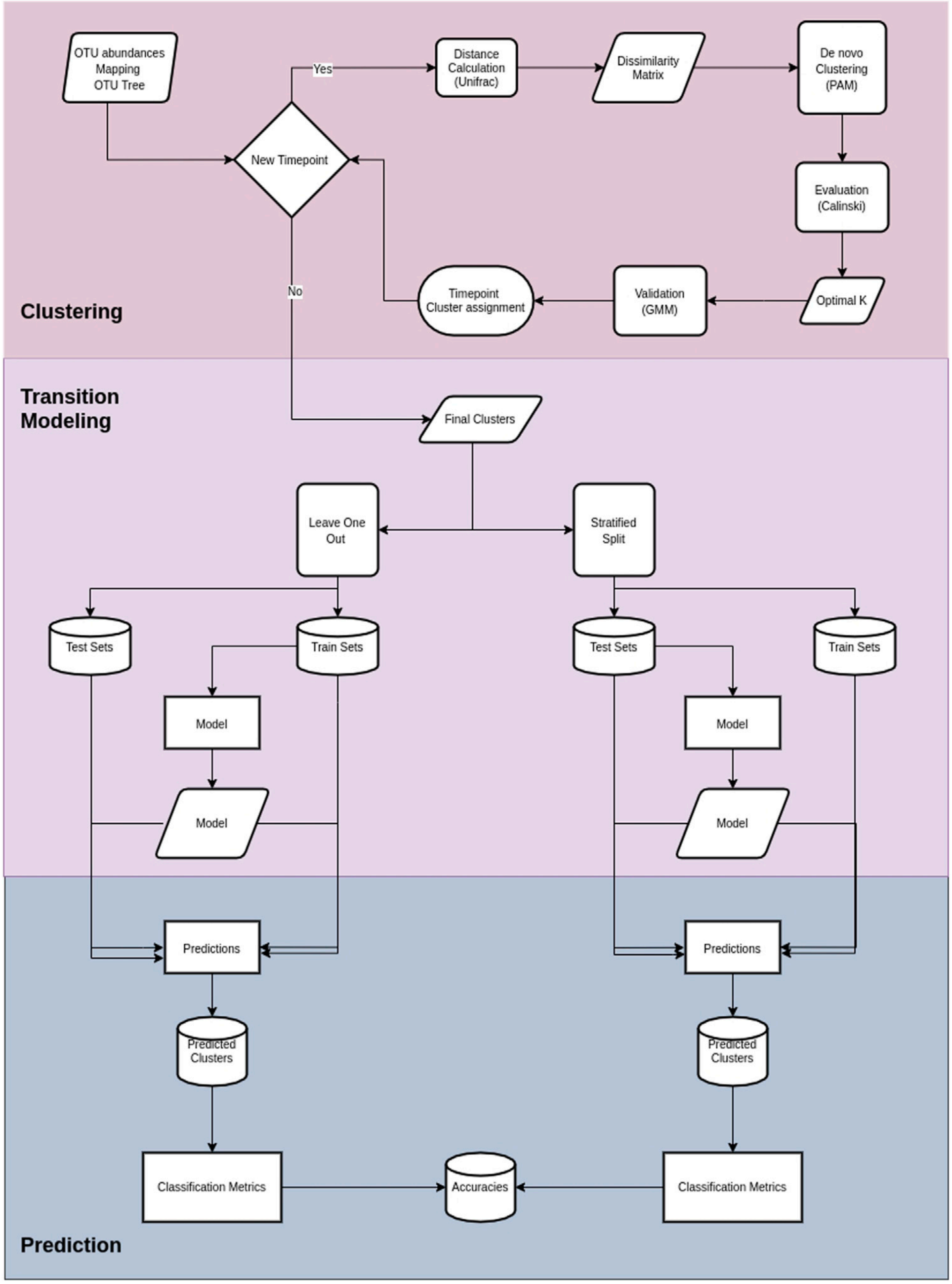


FIGURE 2 | Cronos' pipeline. The first section illustrates the procedures on Cronos from obtaining the data to forming complete clustering assignments.The middle section demonstrates the modeling of transitions from clusters on a time point to any later. The third section displays the prediction procedure and classification metrics.

2.5 Case Study

Cronos was tested on the fecal microbiome data from a study investigating the effects of formula milk and breastfeeding on infants' gut microbiome over the span of 2 years (Bazanella et al., 2017). The dataset consists of 106 infants from the Munich region with samples taken over 1, 3, 5, 7, 9, 12, and 24 months of age. Information on the mode of delivery (vaginal or Cesarean) was available and taken into account in our analysis. In addition to the infant data, we used as a reference for matured gut microbiome the sequence data from the stool samples from 216 healthy lean students of the Technical University of Munich. None of the students had been taking antibiotics in the last 3 months, had any known diseases, or were on long-term medication. The preprocessing of the raw data was performed with the IMNGS platform (Lagkouvardos et al., 2016) implementing the UNOISE version 3 (Edgar, 2016) and UPARSE (Edgar, 2013) pipelines, using the default parameters. The primary analysis outputs were used as inputs in Cronos. The raw data of the two studies are

TABLE 1 | Optimal number of clusters selected automatically in Cronos for all time points. The first row represents the time point in months of age, whereas the second shows the different number of similar microbiome profiles.

Time point (Months of age)	1	3	5	7	9	12	24	References
Optimal Number of Clusters	2	3	2	2	3	3	2	3

publically available at European Nucleotide Archive under accessions PRJEB21196 and PRJEB47555.

3 RESULTS

We applied Cronos to the data retrieved from the infant study of Bazanella et al. (2017) combined with the healthy students reference dataset. The samples were characterized in terms of OTU abundance via the IMNGS platform; the outputs were used as direct input for the Cronos tool.

3.1 *De novo* Profile Clustering

The Calinski–Harabasz indexes calculated for each clustering procedure are graphically demonstrated and stored automatically using Cronos (**Supplementary Figure S1**). Cronos' automated method for selection of the optimal number of the *de novo* clusters suggested that partitioning the data into two or three groups reflects the intrinsic organization of the microbial profiles of the infants at each time point and of the students used as an external reference (**Table 1**).

3.2 Maturation Process

Maturation, as a time-dependent process, is illustrated in Cronos via an MDS plot of all cluster medoids, to compare the relative distances between clusters within the dataset and any external reference time point given. Every microbiome profile cluster is represented by its medoid. The evolution trajectory of the microbiome over time is demonstrated by connecting the medoids as shown in **Figure 3**.

Microbiome profiles of 24 months of age children are relatively close to the adult external references, whereas early life clusters occur closer to each other, highlighting the maturation process. Three main areas of microbiome profile similarity are shown in the graph. The first, on the bottom left side, contains almost half of the early life clusters, dominated by breastfed infants. The top center one contains almost the other half of early life clusters and the bottom right one holds the external reference and 2-year-old clusters. The average distance of infant clusters on all time points compared to the external reference clusters of students decreases as the infants age (**Supplementary Figure S5**), emphasizing the maturation process, as older infants have microbial profiles relatively closer to the adult students.

3.3 Sample Transitions Through Time

Sample transitions between clusters over time are visualized in Cronos via Alluvial graphs (**Figure 4**). For the first months

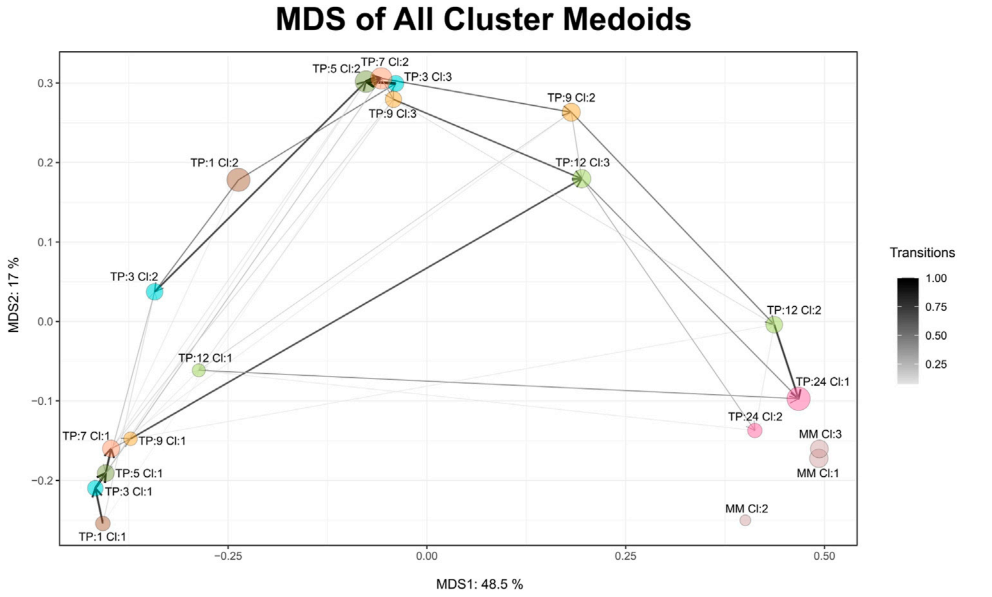


FIGURE 3 | MDS graph of cluster medoids of all clusters on all time points and the external reference as produced by Cronos. TP represents the time point and CL the cluster. MM abbreviation stands for mature microbiome and refers to the external reference samples. Circle size is relative to the percentage of the samples found to belong in the cluster, whereas the connecting arrows are directed from the earliest to the latest time point and their width is relative to the percentage of the samples following the transition.

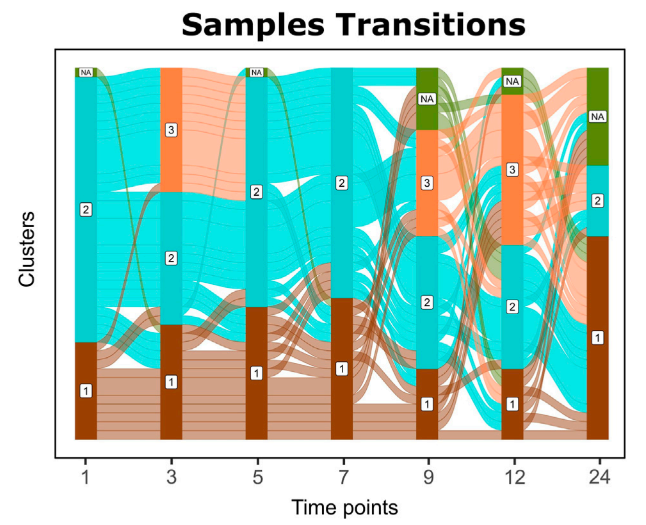


FIGURE 4 | Alluvial graph of sample transitions between clusters over time. NAs represent nonavailable data for the corresponding time point, whereas 1, 2, and 3 represent the cluster on the corresponding time point.

and until the seventh month of age, infants' profiles show common transition patterns switching largely in unison among the time point clusters. At later time points, the infants' microbiome endures many changes in terms of composition, illustrated by cluster alterations (samples entangled between clusters) on consecutive time points. As the infants age, their microbiome profiles tend to converge toward the adult reference. Longer periods between sampling and the introduction of a third cluster on 9 and 12-month-old children might explain the increase in sample transitions between clusters during these stages.

3.4 Cluster Representation

Every cluster is represented by its medoid. Cronos' automated pipeline describes and illustrates the microbial composition of all cluster medoids on all taxonomic levels above genus (**Supplementary Tables S1, S2, S3**). The representation of all clusters on a family level is shown in **Figure 3** (**Supplementary Figures S3, S4** on Order and Class levels).

The relative distances of cluster profiles can be shown even at a family level, highlighting the importance of a beta-diversity distance metric and the final number of cluster decisions. Clusters of 1-month-old infants are highly associated with the two types of diet. TP1-CL1 contains significantly more breastfed infants than expected (one-sided x^2 test $p = 0.00035$), whereas TP1-CL2 contains more than expected formula-fed infants (one-sided x^2 test $p = 0.03069$). TP1-CL1 is dominated by the *Bifidobacteriaceae* family, whereas TP1-CL2 has a more diverse profile, with lower *Bifidobacteriaceae* and higher *Streptococcaceae* and *Enterobacteriaceae* abundances (**Figure 5**). Clusters of 3, 5, and 7 months of age have similar compositions (**Figure 5**), reflected as close relative distances in the multidimensional scaling projection (MDS plot, **Figure 3**). The majority of 9- and 12-month-old infants' profiles start diverging. TP9-CL1 and TP12-CL1 represent late immature profiles, where the *Bifidobacteriaceae* family dominates. TP9-CL2 and TP12-CL2 show an increase in *Bacteroidaceae* family abundance, whereas TP9-CL3 and TP12-CL3 have a higher abundance of the *Lachnospiraceae* family (**Figure 5**). Microbial profiles of 2-year-old infants separate into two clusters, where the feeding groups co-occur. Thus, there is no association between the two types of diet and microbial profile clustering for any of the two clusters (one-sided x^2 test $p = 0.65$ and 0.45, respectively). TP24-CL1 and TP24-CL2 are

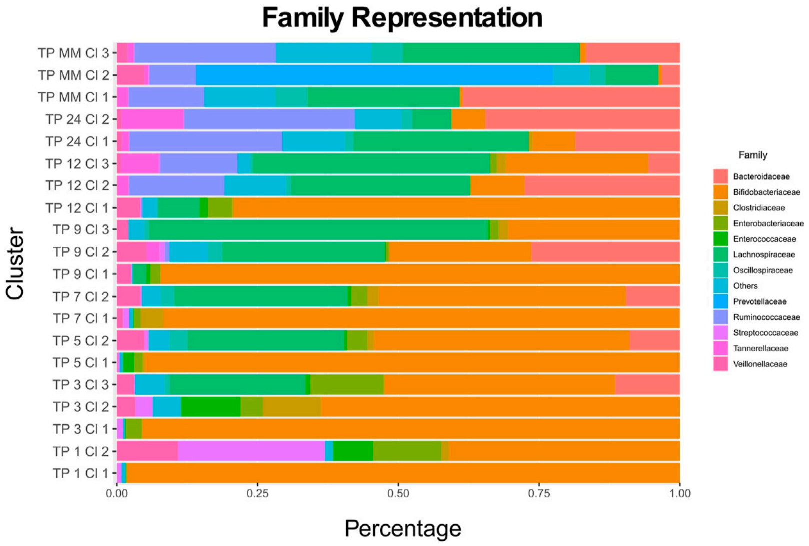


FIGURE 5 | Every cluster composition at a family level. TP stands for time point and Cl for the cluster. Families with an abundance lower than 5% across all medoids are cumulatively shown as Others.

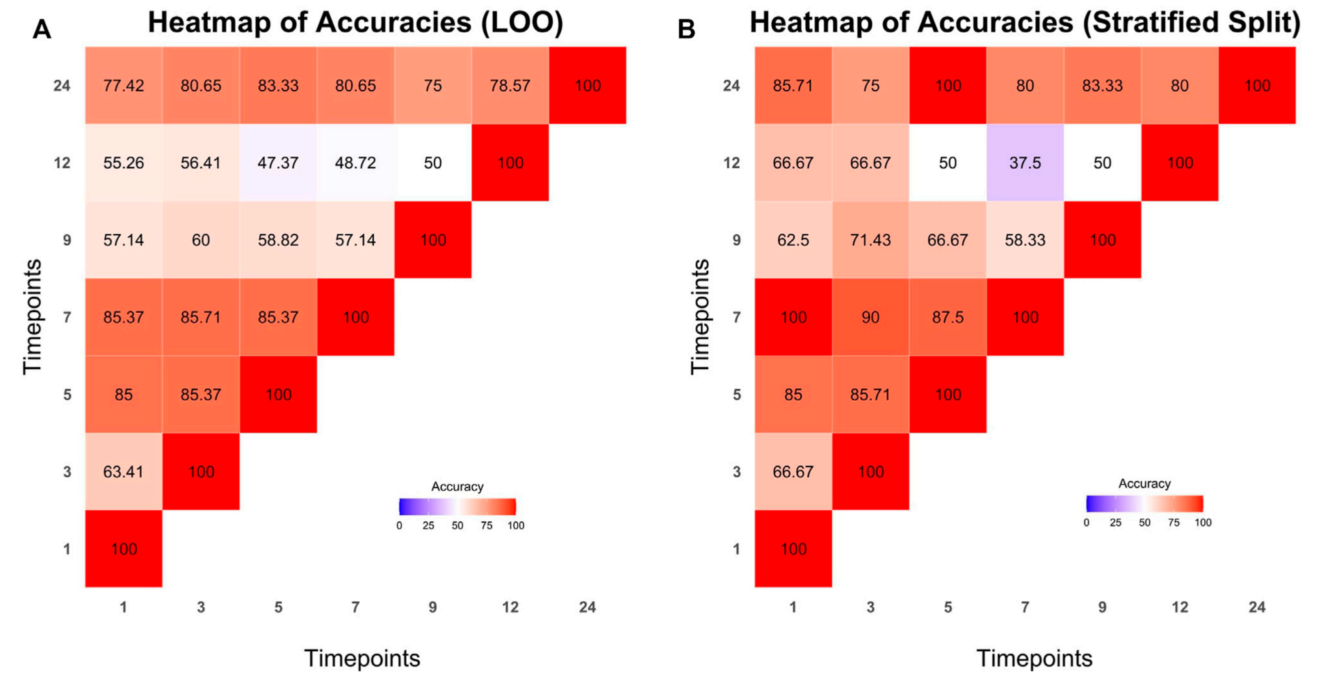


FIGURE 6 | Heatmaps presenting the prediction accuracies achieved from leave one out **(A)** and stratified splits methods **(B)** respectively. Time points from which predictions are made are shown on the *x* axis while predicting time points are represented on the *y* axis.

characterized by higher *Bacteroidaceae* and *Lachnospiraceae* abundances, respectively, whereas both contain a sizable proportion of *Ruminococcaceae* (20%). Clusters of 2-year-old infants are relatively closer to the reference profiles of mature individuals. The reference group is partitioned into three clusters that resemble the described enterotypes with MM-CL1 being the "*Bacteroides*" group, MM-CL2 the "*Prevotella*" and MM-CL3 the "*Ruminococcus*" group (Arumugam et al., 2011).

3.5 Transition Modeling

The dataset was split into train and test sets with the aforementioned methods (LOO and stratified splits). Microbiome profile transitions between clusters on different time points of all possible train sets were modeled by Cronos via multinomial logistic regression. Furthermore, using the model created by the training sets, Cronos predicted the clusters on all time points of the samples based on the provided matrix with metadata. Prediction performance was evaluated via the accuracy metric. The achieved accuracies are visualized in Cronos with multiple barplots according to the predicting and explanatory time point. Moreover, Cronos' automated pipeline creates heatmaps for both splitting methods (**Figure 6**).

All the predictions made by Cronos are compared to a trivial classifier, the random one, where the probability of all clusters is equal (i.e., 1/N where N is the number of clusters **Supplementary Tables S4, S5** show the comparison of the highest accuracies achieved from models with LOO and stratified splits methods to the trivial random classifiers into the test sets).

4 DISCUSSION

4.1 *De novo* Clustering and Cluster Validation

We apply a "Zoom out" methodology by assessing every sample as its whole microbial profile, rather than individual taxa. Cronos' automated pipeline incorporates the beta-diversity distance between samples by exploiting the advantages of the GUniFrac distance metric. Dirichlet multinomial mixtures (Holmes et al., 2012) widely used on microbiome data (Hosoda et al., 2020; Subedi et al., 2020) assume a prior distribution and are based on the abundances. Here, *de novo* clusters reflect the profile distance between samples adding another layer of information. For the clustering of the samples, we apply the partitioning around medoids algorithm, which allows us to represent every cluster by its medoid. This method has been successfully applied in studies spanning from the gut (Stokholm et al., 2018; Khine et al., 2019; Lee et al., 2020) to saliva (Acharya et al., 2017) microbiome.

De novo clustering is applied to all time points separately to specify the exact stages and future transitions of the microbial profiles. The maturation process through clustering has been well established (Stewart et al., 2018; de Muinck and Trosvik, 2018), whereas the divergence in specific time points remains unexplored. Here, by dividing the dataset into time points and applying clustering procedures to all, we provide a deeper understanding of microbial profile divergence.

A novel approach is incorporated to effectively divide the samples at a time point into clusters of a similar microbial profile, based on the GMM clustering algorithm (Pasarkar et al., 2021; Zhang et al., 2017). We compare clustering results for the optimal number of clusters to 1 as GMM components, in order to examine whether the data effectively separate.

4.2 Transitions Through Time and Modeling

Exploring the sample transitions between clusters at different time points enables the understanding of the effectors that shape a microbial profile's fate. Many machine learning techniques have been applied to microbiome data (Marcos-Zambrano et al., 2021). Cronos operates under the assumption that minor compositional differences among the members of a certain cluster of profiles are less important when the fate of the community as a whole is examined. When this assumption is not fulfilled and the presence or absence of taxa with little contribution to the overall cluster assignment determines the future of the community structure, the accuracy of the method might be low. The selection of cluster assignment rather than taxa abundances, and the introduction of metadata results in a small number of explanatory features. Due to the low number of features and interpretability losses that come with high complexity classification algorithms (Marcos-Zambrano et al., 2021), we select multinomial logistic regression, a method widely used on microbiome data (Kaszubinski et al., 2020; Lundgren et al., 2018; Xia et al., 2013) to model the transitions between clusters on different time points.

The importance of features on microbial profile fate is translated as predictability. Features or combinations of features that can better interpret cluster assignment on predicting time points are deemed to be the most important in the development of the microbiome profile in the time between examining and predicting time points. Cronos models for every possible transition and possible mixture of features to fully reflect the predictability of features on all combinations of timepoints and overall, aiming to detect the best time for interventions to steer a microbial profile's fate. Every model designed in Cronos is compared to the trivial random classifier that predicts all classes with equal probability.

4.3 Maturation

Our findings are in accordance with the well-documented microbiome patterns of early life. Breastfed infant profiles consist, mainly, of *Bifidobacteriaceae* family members, whereas formula-fed infants show higher diversity, colonized earlier by *Enterobacteriaceae*, *Bacteroidaceae*, and *Lachnospiraceae* members (Milani et al., 2017; Fallani et al., 2011; Koenig et al., 2011). Furthermore, our analysis, captures the decrease in *Bifidobacteriaceae* and the gradual increase of *Ruminococcaceae*, *Lachnospiraceae*, and *Bacteroidaceae* relative abundances, after the introduction of solid food, until the second year of life as established before (Laursen et al., 2016; Fallani et al., 2011). Cronos provides comparisons of taxonomic composition for the cluster medoids as a proxy of the corresponding cluster. The statistical comparisons of similar profiles fall outside of the scope of the tool. Therefore, using the outputs of Cronos, external tools like Rhea (Lagkouvardos et al., 2017) or QIIME (Caporaso et al., 2010) can easily perform these statistical comparisons of taxa among clusters, considering all their constituting members.

5 APPLICATIONS AND FUTURE WORK

Cronos is a bioinformatic tool that could also be used for other types of environments where bacterial communities dominate, such as soil or marine over the course of the year or several years, aiming to understand the microbiome progression or the suitable response to direct the microbial composition of the environment. Uses of Cronos extend from natural environments to man-made environments, such as open pond bioreactors. Possible uses might also include human gut microbiome over the progression of diseases, sampling over different stages of the disease, aiming to discover the proper antibiotic response or microbiome role in disease progression and phenotype.

For further understanding of infant gut microbiome profiles, more data are required, since the dataset used here as a case study was obtained from a limited geographical region and thus may not include all the possible states. Greater sample size could furthermore benefit the prediction of future states by training a model with more samples.

In future versions of Cronos, we want to include more classification techniques, such as random forest and support vector machines to acquire models that could enhance our transition description. In addition, we would like to introduce further classification performance metrics, such as precision, recall, and F1-score in order to represent model prediction performance extensively. Moreover, we would like to add further clustering performance metrics, such as the Akaike information criterion and silhouette coefficient to further describe cluster divergence.

DATA AVAILABILITY STATEMENT

The raw data of the studies are publicly available at ENA (European Nucleotide Archive https://www.ebi.ac.uk/ena/browser/) under accessions PRJEB21196 and PRJEB47555. The preprocessed data used for the demonstration run (OTUs table, OTUs Tree and mapping file are available at the tools github page: https://github.com/Lagkouvardos/Cronos/tree/main/Cronos_example.

AUTHOR CONTRIBUTIONS

IL conceived and designed the experiments. AL and EI performed the experiments, contributed the code, and analyzed the data. AL, EI, PP, and IL prepared figures and tables and wrote the study.

FUNDING

This research was co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Program "Human Resources Development, Education and Lifelong Learning" in the context of the project "Reinforcement of Postdoctoral Researchers 2nd Cycle" (MIS-5033021), implemented by the State Scholarships Foundation (IKY). IL also received funding from the German Research Foundation (SFB 1371, Project No. 395357507). The Technical University of Munich supported this publication within the Open Access Publishing Program.

ACKNOWLEDGMENTS

We would like to thank Antonios Kioukis for assisting in editing this manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.866902/full#supplementary-material

REFERENCES

Acharya, A., Chan, Y., Kheur, S., Kheur, M., Gopalakrishnan, D., Watt, R. M., et al. (2017). Salivary Microbiome of an Urban Indian Cohort and Patterns Linked to Subclinical Inflammation. *Oral Dis.* 23, 926–940. doi:10.1111/odi.12676

Arumugam, M., Raes, J., Pelletier, E., Le Paslier, D., Yamada, T., Mende, D. R., et al. (2011). Enterotypes of the Human Gut Microbiome. *Nature* 473, 174–180. doi:10.1038/nature09944

Bäckhed, F., Roswall, J., Peng, Y., Feng, Q., Jia, H., Kovatcheva-Datchary, P., et al. (2015). Dynamics and Stabilization of the Human Gut Microbiome during the First Year of Life. *Cell. Host Microbe* 17, 852. doi:10.1016/j.chom.2015.05.012

Bazanella, M., Maier, T. V., Clavel, T., Lagkouvardos, I., Lucio, M., Maldonado-Gòmez, M. X., et al. (2017). Randomized Controlled Trial on the Impact of Early-Life Intervention with Bifidobacteria on the Healthy Infant Fecal Microbiota and Metabolome. *Am. J. Clin. Nutr.* 106, 1274–1286. doi:10.3945/ajcn.117.157529

Calinski, T., and Harabasz, J. (1974). A Dendrite Method for Cluster Analysis. *Comm. Stats. - Theory & Methods* 3, 1–27. doi:10.1080/03610927408827101

Camacho-Morales, A., Caba, M., García-Juárez, M., Caba-Flores, M. D., Viveros-Contreras, R., and Martínez-Valenzuela, C. (2021). Breastfeeding Contributes to Physiological Immune Programming in the Newborn. *Front. Pediatr.* 9, 744104. doi:10.3389/fped.2021.744104

Caporaso, J. G., Kuczynski, J., Stombaugh, J., Bittinger, K., Bushman, F. D., Costello, E. K., et al. (2010). Qiime Allows Analysis of High-Throughput Community Sequencing Data. *Nat. Methods* 7, 335–336. doi:10.1038/nmeth.f.303

Chaffron, S., Rehrauer, H., Pernthaler, J., and von Mering, C. (2010). A Global Network of Coexisting Microbes from Environmental and Whole-Genome Sequence Data. *Genome Res.* 20, 947–959. doi:10.1101/gr.104521.109

Chen, J., Bittinger, K., Charlson, E. S., Hoffmann, C., Lewis, J., Wu, G. D., et al. (2012). Associating Microbiome Composition with Environmental Covariates Using Generalized UniFrac Distances. *Bioinformatics* 28, 2106–2113. doi:10.1093/bioinformatics/bts342

Costea, P. I., Hildebrand, F., Arumugam, M., Bäckhed, F., Blaser, M. J., Bushman, F. D., et al. (2018). Enterotypes in the Landscape of Gut Microbial Community Composition. *Nat. Microbiol.* 3, 8–16. doi:10.1038/s41564-017-0072-8

de Muinck, E. J., and Trosvik, P. (2018). Individuality and Convergence of the Infant Gut Microbiota during the First Year of Life. *Nat. Commun.* 9, 2233. doi:10.1038/s41467-018-04641-7

Edgar, R. C. (2013). UPARSE: Highly Accurate OTU Sequences from Microbial Amplicon Reads. *Nat. Methods* 10, 996–998. doi:10.1038/nmeth.2604

Edgar, R. C. (2016). UNOISE2: Improved Error-Correction for Illumina 16S and ITS Amplicon Sequencing. *bioRxiv*. doi:10.1101/081257

Estrela, S., Vila, J. C. C., Lu, N., Bajić, D., Rebolleda-Gómez, M., Chang, C. Y., et al. (2022). Functional Attractors in Microbial Community Assembly. *Cell. Syst.* 13, 29–e7. e7. doi:10.1016/j.cels.2021.09.011

Fallani, M., Amarri, S., Uusijarvi, A., Adam, R., Khanna, S., Aguilera, M., et al. (2011). Determinants of the Human Infant Intestinal Microbiota after the Introduction of First Complementary Foods in Infant Samples from Five European Centres. *Microbiol. Read.* 157, 1385–1392. doi:10.1099/mic.0.042143-0

Goldford, J. E., Lu, N., Bajić, D., Estrela, S., Tikhonov, M., Sanchez-Gorostiaga, A., et al. (2018). Emergent Simplicity in Microbial Community Assembly. *Science* 361, 469–474. doi:10.1126/science.aat1168

Holmes, I., Harris, K., and Quince, C. (2012). Dirichlet Multinomial Mixtures: Generative Models for Microbial Metagenomics. *PLoS One* 7, e30126. doi:10.1371/journal.pone.0030126

Hosoda, S., Nishijima, S., Fukunaga, T., Hattori, M., and Hamada, M. (2020). Revealing the Microbial Assemblage Structure in the Human Gut Microbiome Using Latent Dirichlet Allocation. *Microbiome* 8, 95. doi:10.1186/s40168-020-00864-3

Iizumi, T., Battaglia, T., Ruiz, V., and Perez Perez, G. I. (2017). Gut Microbiome and Antibiotics. *Arch. Med. Res.* 48, 727–734. doi:10.1016/j.arcmed.2017.11.004

Jakobsson, H. E., Abrahamsson, T. R., Jenmalm, M. C., Harris, K., Quince, C., Jernberg, C., et al. (2014). Decreased Gut Microbiota Diversity, Delayed Bacteroidetes Colonisation and Reduced Th1 Responses in Infants Delivered by Caesarean Section. *Gut* 63, 559–566. doi:10.1136/gutjnl-2012-303249

Jiang, T., Liu, B., Li, J., Dong, X., Lin, M., Zhang, M., et al. (2018). Association between Sn-2 Fatty Acid Profiles of Breast Milk and Development of the Infant Intestinal Microbiome. *Food Funct.* 9, 1028–1037. doi:10.1039/c7fo00088j

Kaszubinski, S. F., Pechal, J. L., Smiles, K., Schmidt, C. J., Jordan, H. R., Meek, M. H., et al. (2020). Dysbiosis in the Dead: Human Postmortem Microbiome Beta-Dispersion as an Indicator of Manner and Cause of Death. *Front. Microbiol.* 11, 555347. doi:10.3389/fmicb.2020.555347

Khine, W. W. T., Zhang, Y., Goie, G. J. Y., Wong, M. S., Liong, M., Lee, Y. Y., et al. (2019). Gut Microbiome of Pre-adolescent Children of Two Ethnicities Residing in Three Distant Cities. *Sci. Rep.* 9, 7831. doi:10.1038/s41598-019-44369-y

Ki, B. M., Ryu, H. W., and Cho, K. S. (2018). Extended Local Similarity Analysis (eLSA) Reveals Unique Associations between Bacterial Community Structure and Odor Emission during Pig Carcasses Decomposition. *J. Environ. Sci. Health A Tox Hazard Subst. Environ. Eng.* 53, 718–727. doi:10.1080/10934529.2018.1439856

Kim, H., Sitarik, A. R., Woodcroft, K., Johnson, C. C., and Zoratti, E. (2019). Birth Mode, Breastfeeding, Pet Exposure, and Antibiotic Use: Associations with the Gut Microbiome and Sensitization in Children. *Curr. Allergy Asthma Rep.* 19, 22. doi:10.1007/s11882-019-0851-9

Koenig, J. E., Spor, A., Scalfone, N., Fricker, A. D., Stombaugh, J., Knight, R., et al. (2011). Succession of Microbial Consortia in the Developing Infant Gut Microbiome. *Proc. Natl. Acad. Sci. U. S. A.* 108 (Suppl. 1), 4578–4585. doi:10.1073/pnas.1000081107

Korpela, K., Salonen, A., Saxen, H., Nikkonen, A., Peltola, V., Jaakkola, T., et al. (2020). Antibiotics in Early Life Associate with Specific Gut Microbiota Signatures in a Prospective Longitudinal Infant Cohort. *Pediatr. Res.* 88, 438–443. doi:10.1038/s41390-020-0761-5

Lagkouvardos, I., Fischer, S., Kumar, N., and Clavel, T. (2017). Rhea: a Transparent and Modular R Pipeline for Microbial Profiling Based on 16s Rrna Gene Amplicons. *PeerJ* 5, e2836. doi:10.7717/peerj.2836

Lagkouvardos, I., Joseph, D., Kapfhammer, M., Giritli, S., Horn, M., Haller, D., et al. (2016). IMNGS: A Comprehensive Open Resource of Processed 16S rRNA Microbial Profiles for Ecology and Diversity Studies. *Sci. Rep.* 6, 33721. doi:10.1038/srep33721

Laursen, M. F., Andersen, L. B., Michaelsen, K. F., Mølgaard, C., Trolle, E., Bahl, M. I., et al. (2016). Infant Gut Microbiota Development Is Driven by Transition to Family Foods Independent of Maternal Obesity. *mSphere* 1. doi:10.1128/mSphere.00069-15

Lee, S. H., Yoon, S. H., Jung, Y., Kim, N., Min, U., Chun, J., et al. (2020). Emotional Well-Being and Gut Microbiome Profiles by Enterotype. *Sci. Rep.* 10, 20736. doi:10.1038/s41598-020-77673-z

Lemas, D. J., Yee, S., Cacho, N., Miller, D., Cardel, M., Gurka, M., et al. (2016). Exploring the Contribution of Maternal Antibiotics and Breastfeeding to Development of the Infant Microbiome and Pediatric Obesity. *Semin. Fetal Neonatal Med.* 21, 406–409. doi:10.1016/j.siny.2016.04.013

Lozupone, C., and Knight, R. (2005). UniFrac: a New Phylogenetic Method for Comparing Microbial Communities. *Appl. Environ. Microbiol.* 71, 8228–8235. doi:10.1128/AEM.71.12.8228-8235.2005

Lundgren, S. N., Madan, J. C., Emond, J. A., Morrison, H. G., Christensen, B. C., Karagas, M. R., et al. (2018). Maternal Diet during Pregnancy Is Related with the Infant Stool Microbiome in a Delivery Mode-dependent Manner. *Microbiome* 6, 109. doi:10.1186/s40168-018-0490-8

Marcos-Zambrano, L. J., Karaduzovic-Hadziabdic, K., Loncar Turukalo, T., Przymus, P., Trajkovik, V., Aasmets, O., et al. (2021). Applications of Machine Learning in Human Microbiome Studies: A Review on Feature Selection, Biomarker Identification, Disease Prediction and Treatment. *Front. Microbiol.* 12, 634511. doi:10.3389/fmicb.2021.634511

Mesa, M. D., Loureiro, B., Iglesia, I., Fernandez Gonzalez, S., Llurba Olivé, E., García Algar, O., et al. (2020). The Evolving Microbiome from Pregnancy to Early Infancy: A Comprehensive Review. *Nutrients* 12. doi:10.3390/nu12010133

Milani, C., Duranti, S., Bottacini, F., Casey, E., Turroni, F., Mahony, J., et al. (2017). The First Microbial Colonizers of the Human Gut: Composition, Activities, and Health Implications of the Infant Gut Microbiota. *Microbiol. Mol. Biol. Rev.* 81. doi:10.1128/MMBR.00036-17

Miller, M. B., and Bassler, B. L. (2001). Quorum sensing in Bacteria. *Annu. Rev. Microbiol.* 55, 165–199. doi:10.1146/annurev.micro.55.1.165

Morris, B. E., Henneberger, R., Huber, H., and Moissl-Eichinger, C. (2013). Microbial Syntrophy: Interaction for the Common Good. *FEMS Microbiol. Rev.* 37, 384–406. doi:10.1111/1574-6976.12019

Mukherjee, S., Stamatis, D., Bertsch, J., Ovchinnikova, G., Sundaramurthi, J. C., Lee, J., et al. (2021). Genomes OnLine Database (GOLD) v.8: Overview and Updates. *Nucleic Acids Res.* 49, D723. doi:10.1093/nar/gkaa983

Pannaraj, P. S., Li, F., Cerini, C., Bender, J. M., Yang, S., Rollie, A., et al. (2017). Association between Breast Milk Bacterial Communities and Establishment and Development of the Infant Gut Microbiome. *JAMA Pediatr.* 171, 647–654. doi:10.1001/jamapediatrics.2017.0378

Pasarkar, A. P., Joseph, T. A., and Pe'er, I. (2021). Directional Gaussian Mixture Models of the Gut Microbiome Elucidate Microbial Spatial Structure. *mSystems* 6, e0081721. doi:10.1128/mSystems.00817-21

Schubert, E., and Rousseeuw, P. J. (2021). Fast and Eager K-Medoids Clustering: O(k) Runtime Improvement of the PAM, CLARA, and CLARANS Algorithms. *Inf. Syst.* 101, 101804. doi:10.1016/j.is.2021.101804

Sharon, I., Morowitz, M. J., Thomas, B. C., Costello, E. K., Relman, D. A., and Banfield, J. F. (2013). Time Series Community Genomics Analysis Reveals Rapid Shifts in Bacterial Species, Strains, and Phage during Infant Gut Colonization. *Genome Res.* 23, 111–120. doi:10.1101/gr.142315.112

Soderborg, T. K., Clark, S. E., Mulligan, C. E., Janssen, R. C., Babcock, L., Ir, D., et al. (2018). The Gut Microbiota in Infants of Obese Mothers Increases Inflammation and Susceptibility to NAFLD. *Nat. Commun.* 9, 4462. doi:10.1038/s41467-018-06929-0

Steele, J. A., Countway, P. D., Xia, L., Vigil, P. D., Beman, J. M., Kim, D. Y., et al. (2011). Marine Bacterial, Archaeal and Protistan Association Networks Reveal Ecological Linkages. *ISME J.* 5, 1414–1425. doi:10.1038/ismej.2011.24

Stewart, C. J., Ajami, N. J., O'Brien, J. L., Hutchinson, D. S., Smith, D. P., Wong, M. C., et al. (2018). Temporal Development of the Gut Microbiome in Early Childhood from the TEDDY Study. *Nature* 562, 583–588. doi:10.1038/s41586-018-0617-x

Stokholm, J., Blaser, M. J., Thorsen, J., Rasmussen, M. A., Waage, J., Vinding, R. K., et al. (2018). Maturation of the Gut Microbiome and Risk of Asthma in Childhood. *Nat. Commun.* 9, 141. doi:10.1038/s41467-017-02573-2

Stubbendieck, R. M., Vargas-Bautista, C., and Straight, P. D. (2016). Bacterial Communities: Interactions to Scale. *Front. Microbiol.* 7, 1234. doi:10.3389/fmicb.2016.01234

Subedi, S., Neish, D., Bak, S., and Feng, Z. (2020). Cluster Analysis of Microbiome Data by Using Mixtures of Dirichlet-Multinomial Regression Models. *J. R. Stat. Soc. C* 69, 1163–1187. doi:10.1111/rssc.12432

Sugino, K. Y., Ma, T., Paneth, N., and Comstock, S. S. (2021). Effect of Environmental Exposures on the Gut Microbiota from Early Infancy to Two Years of Age. *Microorganisms* 9, 2140. doi:10.3390/microorganisms9102140

Tan, C. H., Yeo, Y. P., Hafiz, M., Ng, N. K. J., Subramoni, S., Taj, S., et al. (2021). Functional Metagenomic Analysis of Quorum Sensing Signaling in a Nitrifying Community. *NPJ Biofilms Microbiomes* 7, 79. doi:10.1038/s41522-021-00250-3

Vergin, K. L., Beszteri, B., Monier, A., Thrash, J. C., Temperton, B., Treusch, A. H., et al. (2013). High-resolution SAR11 Ecotype Dynamics at the bermuda Atlantic Time-Series Study Site by Phylogenetic Placement of Pyrosequences. *ISME J.* 7, 1322–1332. doi:10.1038/ismej.2013.32

Xia, F., Chen, J., Fung, W. K., and Li, H. (2013). A Logistic Normal Multinomial Regression Model for Microbiome Compositional Data Analysis. *Biometrics* 69, 1053–1063. doi:10.1111/biom.12079

Xia, L. C., Steele, J. A., Cram, J. A., Cardon, Z. G., Simmons, S. L., Vallino, J. J., et al. (2011). Extended Local Similarity Analysis (eLSA) of Microbial Community and Other Time Series Data with Replicates. *BMC Syst. Biol.* 5 (Suppl. 2), S15. doi:10.1186/1752-0509-5-S2-S15

Zhang, F., Sun, F., and Luan, Y. (2019). Statistical Significance Approximation for Local Similarity Analysis of Dependent Time Series Data. *BMC Bioinforma.* 20, 53. doi:10.1186/s12859-019-2595-x

Zhang, Y., Hu, X., and Jiang, X. (2017). Multi-View Clustering of Microbiome Samples by Robust Similarity Network Fusion and Spectral Clustering. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 14, 264–271. doi:10.1109/TCBB.2015.2474387

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Litos, Intze, Pavlidis and Lagkouvardos. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Frontiers in Bioinformatics

Explores innovation in the analysis and interpretation of biological data

An innovative journal that provides a forum for new discoveries in bioinformatics. It focuses on how new tools and applications can bring insights to specific biological problems.

Discover the latest Research Topics

See more →

frontiers

Frontiers in
Bioinformatics

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact