frontiers
RESEARCH TOPICS

QUANTITATIVE ASSESSMENT AND VALIDATION OF NETWORK INFERENCE METHODS IN BIOINFORMATICS

Topic Editors
Benjamin Haibe-Kains and Frank Emmert-Streib

frontiers in
GENETICS

FRONTIERS COPYRIGHT STATEMENT

Cover image provided by Ibbl sarl, Lausanne CH

ISSN 1664-8714
ISBN 978-2-88919-478-0
DOI 10.3389/978-2-88919-478-0

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: researchtopics@frontiersin.org

QUANTITATIVE ASSESSMENT AND VALIDATION OF NETWORK INFERENCE METHODS IN BIOINFORMATICS

Topic Editors:
Benjamin Haibe-Kains, Princess Margaret Cancer Centre, Canada
Frank Emmert-Streib, Queen's University Belfast, United Kingdom

Scientists today have access to an unprecedented arsenal of high-tech tools that can be used to thoroughly characterize biological systems of interest. High-throughput "omics" technologies enable to generate enormous quantities of data at the DNA, RNA, epigenetic and proteomic levels. One of the major challenges of the post-genomic era is to extract functional information by integrating such heterogeneous high-throughput genomic data. This is not a trivial task as we are increasingly coming to understand that it is not individual genes, but rather biological pathways and networks that drive an organism's response to environmental factors and the development of its particular phenotype. In order to fully understand the way in which these networks interact (or fail to do so) in specific states (disease for instance), we must learn both, the structure of the underlying networks and the rules that govern their behaviour.

In recent years there has been an increasing interest in methods that aim to infer biological networks. These methods enable the opportunity for better understanding the interactions between genomic features and the overall structure and behavior of the underlying networks. So far, such network models have been mainly used to identify and validate new interactions between genes of interest. But ultimately, one could use these networks to predict large-scale effects of perturbations, such as treatment by multiple targeted drugs. However, currently, we are still at an early stage of comprehending methods and approaches providing a robust statistical framework to quantitatively assess the quality of network inference and its predictive potential.

The scope of this Research Topic in Bioinformatics and Computational Biology aims at addressing these issues by investigating the various, complementary approaches to quantify the quality of network models. These "validation" techniques could focus on assessing quality of specific interactions, global and local structures, and predictive ability of network models. These methods could rely exclusively on in silico evaluation procedures or they could be coupled with novel experimental designs to generate the biological data necessary to properly validate inferred networks.

Table of Contents

frontiers in
GENETICS

EDITORIAL
published: 16 July 2014
doi: 10.3389/fgene.2014.00221

Quantitative assessment and validation of network inference methods in bioinformatics

Benjamin Haibe-Kains[1,2]* and Frank Emmert-Streib[3]*

[1] *Bioinformatics and Computational Genomics, Princess Margaret Cancer Centre, University Health Network, Toronto, ON, Canada*
[2] *Medical Biophysics Department, University of Toronto, Toronto, ON, Canada*
[3] *Computational Biology and Machine Learning Laboratory, Center for Cancer Research and Cell Biology, Queen's University Belfast, Belfast, UK*
**Correspondence: bhaibeka@uhnresearch.ca; v@bio-complexity.com*

Edited by:
Richard D. Emes, University of Nottingham, UK

Reviewed by:
Matt Loose, University of Nottingham, UK

Keywords: computational biology, network inference, computational genomics, personalized medicine, bioinformatics, network validation

The last years following the completion of the human genome project (Quackenbush, 2011) have given raise to major breakthroughs in the development of novel biotechnologies, such as next-generation sequencing, that sparked the generation of high-throughput "omics" data. The robustness and the cost-efficiency of these technologies increasing over time enabled the conduction of large screening experiments containing hundreds and even thousands of samples. As a consequence of these "big" biological and biomedical high-throughput datasets, advanced statistical methodology can now be employed requiring such large sample sizes.

This is one reason explaining the recent interest in methods that aim to infer biological networks. These methods offer the opportunity for better understanding the interactions between genomic features and the overall structure and behavior of the underlying networks. In order to foster this research direction we edited a Research Topic entitled "Quantitative Assessment and Validation of Network Inference Methods in Bioinformatics." This research topic was perceived as relevant and timely by the scientific community and we consequently received 15 contributions from research groups all over the world (Boucher and Jenna, 2013; Chun et al., 2013; de Matos Simoes et al., 2013; Lopes and Bontempi, 2013; Qian and Dougherty, 2013; Schrynemackers et al., 2013; Scott-Boyer et al., 2013; Staiger et al., 2013; Tran et al., 2013; Ho et al., 2014; Horn et al., 2014; Montojo et al., 2014; Olsen et al., 2014; Peng and Schork, 2014; Santra, 2014).

The topics addressed by these contributions can be broadly grouped into the following categories:

- Data integration (Boucher and Jenna, 2013; Chun et al., 2013; Scott-Boyer et al., 2013; Ho et al., 2014; Horn et al., 2014; Olsen et al., 2014; Santra, 2014)
- Network validation (de Matos Simoes et al., 2013; Lopes and Bontempi, 2013; Qian and Dougherty, 2013; Schrynemackers et al., 2013; Montojo et al., 2014; Olsen et al., 2014)
- Network inference (Lopes and Bontempi, 2013; Schrynemackers et al., 2013)
- Time series data (Lopes and Bontempi, 2013)
- Network interpretation (Boucher and Jenna, 2013; Chun et al., 2013; de Matos Simoes et al., 2013; Montojo et al., 2014; Scott-Boyer et al., 2013; Tran et al., 2013)
- Diagnostic applications (Staiger et al., 2013; Peng and Schork, 2014)
- Network modeling (Tran et al., 2013)

First of all, it is important to note that there is still no commonly accepted term to denote 'networks' that are inferred from gene expression data, which the vast majority of the contributed papers used for their inference. Indeed, depending on the context, these networks are called gene regulatory networks (de Matos Simoes et al., 2013; Lopes and Bontempi, 2013; Qian and Dougherty, 2013; Santra, 2014), molecular interaction networks (Horn et al., 2014; Olsen et al., 2014), gene co-expression networks (Scott-Boyer et al., 2013) or biological networks (Schrynemackers et al., 2013). We believe that this plurality denotes the diversity of usages and interpretations of such networks, while it may also reflect the lack of agreement due to the interdisciplinary nature of network inference in Bioinformatics. For the future it would be beneficial to find a common terminology for such networks, because this would certainly enhance the communicability within the community. At the moment, the term 'gene regulatory networks' seems to be the most frequent denotation in use, however, a thorough discussion of this important topic seems indispensable.

The two topics that attracted most interest in the submitted contributions are network validation and data integration. The former is a good reminder that the assessment of inferred networks is not trivial due to two major reasons. First, we still have only partial knowledge about gene regulatory networks even in organisms like *Saccharomyces cerevisiae* (yeast) or *E. coli*, which are considerably simpler than Human. Second, networks are structured objects that means we cannot only assess errors on the global scale for the whole network, but also on intermediate levels down to single interactions and any combination thereof, e.g., motifs or modules (Emmert-Streib and Altay, 2010). In addition, for labeled data enabling the usage of supervised learning methods further issues need to be addressed, as indicated and discussed in the review paper by Schrynemackers et al. (2013).

The integration of different datasets, either of the same or of different types, is certainly a topic that will gain even more attention in the future when more and new high-throughput technologies become available and the access to such datasets is simplified by a policy change of funding agencies making it imperative for grant holders to provide free access to such data. It appears that Bayesian methods (Santra, 2014) provide a natural framework that is particularly suited for such an integration because of its flexibility and widespread acceptance as a fundamental statistical inference paradigm. However, other methods have also been proposed to tackle the challenge of heterogeneous data integration, such as the regression-based framework integrating priors extracted from the biomedical literature and other sources (Olsen et al., 2014). This provides opportunities for comparing novel methodological developments with well-established statistical approaches. We would like to emphasize that networks inferred from the integration of different datasets require a reassessment of their validation for similar reasons as for a supervised learning of gene regulatory networks (Schrynemackers et al., 2013).

For the future, we think that applications of inferred network, e.g., for diagnostic, predictive or therapeutic purposes in medicine will become very important for translational research because of their potential to provide a systems-approach, certainly required to understand complex disorders like cancer. However, until we reach this point more work is needed. For our Research Topic, two contributions have been submitted that are good examples for a better understanding of this problem. In Peng and Schork (2014) the authors found that network centrality measures, which are characterizing the importance of nodes within a gene network that has been constructed from the gene expression patterns, can be used to identify therapeutic targets. In contrast, in Staiger et al. (2013) the authors showed that current composite-feature classification methods considering a network structure, do not outperform simple single-genes classifiers in predicting outcome in breast cancer for prognostic purposes. It is interesting to note that the outcome of both studies allows opposing conclusions. Whereas the results in Peng and Schork (2014) can be seen as an encouragement for further studies employing network-based approaches, the results in Staiger et al. (2013) do not support this. However, by changing the perspective, the study by Staiger et al. (2013) suggests that we do not need to focus on single-gene studies because we can get similar results from network-based approaches. Now, the crucial question is which perspective should we chose? The choice of perspective actually depends on the use of the inferred networks, and therefore the goal of the study. On the one hand, if one is interested in building a predictive model, which does not need to be interpretable (often referred to as "black box" in the literature), then only performance of the inferred model matters; in this case scenario Staiger et al. (2013) showed that, for cancer prognosis, network-based approaches may not be relevant as they do not outperform simpler methods (singe genes). On the other hand, if one is more interested in the biological knowledge that could be extracted from statistical models, network-based approaches are extremely relevant as they are efficient ways to represent complex biological patterns while retaining good predictive ability.

Overall, we believe that, in a translational application, the underlying choice of perspective is of central importance. That means the utility of a network-based approach is expected to depend crucially on the biological question to which such a method should be applied to.

REFERENCES

Boucher, B., and Jenna, S. (2013). Genetic interaction networks: better understand to better predict. *Front. Genet.* 4:290. doi: 10.3389/fgene.2013.00290

Chun, H., Chen, M., Li, B., and Zhao, H. (2013). Joint conditional gaussian graphical models with multiple sources of genomic data. *Front. Genet.* 4:294. doi: 10.3389/fgene.2013.00294

de Matos Simoes, R., Dehmer, M., and Emmert-Streib, F. (2013). B-cell lymphoma gene regulatory networks: biological consistency among inference methods. *Front. Genet.* 4:281. doi: 10.3389/fgene.2013.00281

Emmert-Streib, F., and Altay, G. (2010). Local network-based measures to assess the inferability of different regulatory networks. *IET Syst. Biol.* 4, 277–288. doi: 10.1049/iet-syb.2010.0028

Ho, Y.-Y., Cope, L. M., and Parmigiani, G. (2014). Modular network construction using eQTL data: an analysis of computational costs and benefits. *Front. Genet.* 5:40. doi: 10.3389/fgene.2014.00040

Horn, F., Rittweger, M., Taubert, J., Lysenko, A., Rawlings, C., and Guthke, R. (2014). Interactive exploration of integrated biological datasets using context-sensitive workflows. *Front. Genet.* 5:21. doi: 10.3389/fgene.2014.00021

Lopes, M., and Bontempi, G. (2013). Experimental assessment of static and dynamic algorithms for gene regulation inference from time series expression data. *Front. Genet.* 4:303. doi: 10.3389/fgene.2013.00303

Montojo, J., Zuberi, K., Shao, Q., Bader, G., and Morris, Q. (2014). Network assessor: an automated method for quantitative assessment of a network's potential for gene function prediction. *Front. Genet.* 5:123. doi: 10.3389/fgene.2014.00123

Olsen, C., Bontempi, G., Emmert-Streib, F., Quackenbush, J., and Haibe-Kains, B. (2014). Relevance of different prior knowledge sources for inferring gene interaction networks. *Front. Genet.* 5:177. doi: 10.3389/fgene.2014.00177

Peng, Q., and Schork, N. (2014). Utility of network integrity methods in therapeutic target identification. *Front. Genet.* 5:12. doi: 10.3389/fgene.2014.00012

Qian, X., and Dougherty, E. (2013) Validation of gene regulatory network inference based on controllability. *Front. Genet.* 4:272. doi: 10.3389/fgene.2013.00272

Quackenbush, J. (2011). *The Human Genome: The Book of Essential Knowledge*. New York, NY: Imagine Publishing, Curiosity Guides.

Santra, T. (2014). A bayesian framework that integrates heterogeneous data for inferring gene regulatory networks. *Front. Bioeng. Biotechnol.* 2:13. doi: 10.3389/fbioe.2014.00013

Schrynemackers, M., Kueffner, R., and Geurts, P. (2013). On protocols and measures for the validation of supervised methods for the inference of biological networks. *Front. Genet.* 4:262. doi: 10.3389/fgene.2013.00262

Scott-Boyer, M.-P., Haibe-Kains, B., and Deschepper, C. F. (2013). Network statistics of genetically-driven gene co-expression modules in mouse crosses. *Front. Genet.* 4:291. doi: 10.3389/fgene.2013.00291

Staiger, C., Cadot, S., GyŽrffy, B., Wessels, L. F., and Klau, G. W. (2013). Current composite-feature classification methods do not outperform simple single-genes classifiers in breast cancer prognosis. *Front. Genet.* 4:289. doi: 10.3389/fgene.2013.00289

Tran, V., McCall, M. N., McMurray, H., and Almudevar, A. (2013). On the underlying assumptions of threshold boolean networks as a model for genetic regulatory network behavior. *Front. Genet.* 4:263. doi: 10.3389/fgene.2013.00263

Conflict of Interest Statement: 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.

Received: 17 June 2014; accepted: 26 June 2014; published online: 16 July 2014.
Citation: Haibe-Kains B and Emmert-Streib F (2014) Quantitative assessment and validation of network inference methods in bioinformatics. Front. Genet. ***5****:221. doi: 10.3389/fgene.2014.00221*
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

REVIEW ARTICLE
published: 03 December 2013
doi: 10.3389/fgene.2013.00262

On protocols and measures for the validation of supervised methods for the inference of biological networks

Marie Schrynemackers[1]*, Robert Küffner[2] and Pierre Geurts[1]

[1] Systems and Modeling, Department of Electrical Engineering and Computer Science and GIGA-R, University of Liège, Liège, Belgium
[2] Institute for Practical Informatics and Bioinformatics, Ludwig-Maximilians-University, Munich, Germany

Edited by:
Benjamin Haibe-Kains, Institut de Recherches Cliniques de Montréal, Canada

Reviewed by:
John Pinney, Imperial College London, UK
Mikhail P. Ponomarenko, Institute of Cytology and Genetics of Siberian Branch of Russian Academy of Sciences, Russia
Teresa Núñez de Villavicencio Díaz, Center for Genetic Engineering and Biotechnology, Cuba

***Correspondence:**
Marie Schrynemackers, Systems and Modeling, Department of Electrical Engineering and Computer Science and GIGA-R, Institut Montefiore, University of Liège, Sart-Tilman B28, 4000, Liège, Belgium
e-mail: marie.schrynemackers@ulg.ac.be

Networks provide a natural representation of molecular biology knowledge, in particular to model relationships between biological entities such as genes, proteins, drugs, or diseases. Because of the effort, the cost, or the lack of the experiments necessary for the elucidation of these networks, computational approaches for network inference have been frequently investigated in the literature. In this paper, we examine the assessment of supervised network inference. Supervised inference is based on machine learning techniques that infer the network from a training sample of known interacting and possibly non-interacting entities and additional measurement data. While these methods are very effective, their reliable validation *in silico* poses a challenge, since both prediction and validation need to be performed on the basis of the same partially known network. Cross-validation techniques need to be specifically adapted to classification problems on pairs of objects. We perform a critical review and assessment of protocols and measures proposed in the literature and derive specific guidelines how to best exploit and evaluate machine learning techniques for network inference. Through theoretical considerations and *in silico* experiments, we analyze in depth how important factors influence the outcome of performance estimation. These factors include the amount of information available for the interacting entities, the sparsity and topology of biological networks, and the lack of experimentally verified non-interacting pairs.

Keywords: biological network inference, supervised learning, cross-validation, evaluation protocols, ROC curves, precision-recall curves

1. INTRODUCTION

Networks naturally represent entities such as genes, proteins, drugs or diseases (as nodes) and their mutual relationships (as edges). As immense experimental efforts would be required to comprehensively characterize such networks, computational approaches for network inference have been frequently investigated in the literature. Both unsupervised and supervised approaches have been proposed for network inference. In order to predict interactions, unsupervised inference methods generally derive a score expressing the confidence for a pair of nodes to interact, based on analysis of some experimental data such as gene expression measurements. In contrast to unsupervised methods, supervised approaches additionally require a partial knowledge of the gold standard network. They then exploit some supervised learning algorithm to construct a model that can subsequently be applied to classify the remaining untested pairs. As supervised methods take advantage of known interactions, they can model node specific properties (e.g., in gene regulatory networks, the experimental conditions where a specific regulator becomes active) and thus perform typically much better than unsupervised ones. Supervised learning approaches have been applied to predict several biological networks: protein–protein interaction networks (Yip and Gerstein, 2008; Tastan et al., 2009; Park and Marcotte, 2011), metabolic networks (Yamanishi and Vert, 2005; Bleakley et al., 2007; Geurts et al., 2007), gene regulatory networks (Mordelet and Vert, 2008; Cerulo et al., 2010), epistatic gene networks (Ulitsky et al., 2009; Ryan et al., 2010), or networks of drug-protein interactions (Yamanishi et al., 2008; Bleakley and Yamanishi, 2009; Cheng et al., 2012; Takarabe et al., 2012; Yu et al., 2012).

Performance estimation of both unsupervised and supervised inference methods requires a gold standard of experimentally tested interactions, i.e., pairs of entities labeled as interacting or non-interacting. The validation of supervised methods, however, generally requires special care and the application of cross validation techniques to avoid any sources of bias. Indeed both training and validation need to be performed on the basis of the same partially labeled gold standard. The case of supervised network inference is even more complex as it works on pairs of objects so that the traditional cross validation techniques are not sufficient. In the paper, we propose a critical review of protocols and measures found in the literature for the validation of supervised network inference methods and derive specific guidelines on how to best exploit machine learning techniques for network inference.

The paper is structured as follows. In section 2, we define the problem of supervised network inference and review existing approaches to solve this problem. Section 3 discusses common metrics used to evaluate network predictions (that are common to unsupervised and supervised inference methods). Appropriate ways to perform cross-validation in this context are discussed in section 4. The impact of the lack of negative examples in common biological networks is analyzed in section 5. Finally, section 6 discusses the positive bias on performance induced by the heavy-tailed degree distribution often met in biological networks.

2. SUPERVISED NETWORK INFERENCE

In this section, we first define the problem of supervised network inference more formally and lay out the notations for the rest of the paper. We then briefly review existing approaches to solve this problem.

2.1. PROBLEM DEFINITION

For the sake of generality, let us assume that we have two finite sets of nodes, $U_r = \{n_r^1, \ldots, n_r^{N_{U_r}}\}$ and $U_c = \{n_c^1, \ldots, n_c^{N_{U_c}}\}$ of respective sizes N_{U_r} and N_{U_c}. A network connecting these two sets of nodes can then be defined by an adjacency matrix Y of size $N_{U_r} \times N_{U_c}$, such that $y_{ij} = 1$ if the nodes n_r^i and n_c^j are connected and $y_{ij} = 0$ if not. Actually, the subscripts r and c stand, respectively for *row* and *column*, referring to the rows and columns of the targeted adjacency matrix Y. Y thus defines a bipartite graph over the two sets U_r and U_c. Standard graphs defined on only one family of nodes, that we call *homogeneous graphs*, can nevertheless be obtained as special cases of this general framework by considering only one set of nodes (i.e., $U = U_r = U_c$). Undirected or directed graphs can then both be represented using a symmetric or an asymmetric adjacency matrix Y.

For example, in the case of protein–protein interaction networks, $U_c = U_r$ is the set of all proteins of a given organism and the adjacency matrix is symmetric. A drug-protein interaction network can be modeled as a bipartite graph where U_r and U_c are respectively the sets of proteins and drugs of interest, and element y_{ij} of Y is equal to 1 if protein n_r^i interacts with drug n_c^j, 0 otherwise. A regulatory network can be modeled either as a bipartite graph where U_c is the set of all genes of the organism of interest and U_r is the set of all candidate transcription factors (TFs) among them or equivalently by an homogeneous graph and an asymmetric adjacency matrix, where $U_c = U_r$ is the set of all genes and $y_{ij} = 1$ if gene n_i regulates gene n_j, 0 otherwise.

In addition, we assume that each node n (in both sets) is described by a feature vector, denoted $x(n)$, typically lying in $\mathbb{R}^p$. For example, features associated to proteins/genes could include their expression in some conditions as measured by microarrays, the presence of motifs in their promotor region, information about their structure, etc. A feature vector $x(n_r, n_c)$ can also be associated to each pair of nodes. For example, features directly associated to pairs of proteins could code for the association of the two proteins in another network, their binding in a ChIP-sequencing experiments, etc.

In this context, the problem of supervised network inference can be formulated as follows:

> Given a partial knowledge of the adjacency matrix Y of the target network in the form of a learning sample of triplets:
>
> $$LS_p = \left\{ \left(n_r^{i_k}, n_c^{j_k}, y_{i_k j_k} \right) | k = 1, \ldots, N_{LS} \right\},$$
>
> and given the feature representation of the nodes and/or pairs of nodes, find a function $f : U_r \times U_c \to \{0, 1\}$ that best approximates the unknown entries of the adjacency matrix from the feature representation (on nodes or on pairs) relative to these unknown entries.

This problem can be cast as a supervised classification problem, with the peculiarity, however, that pairs of nodes, and not single nodes, need to be classified. Next, we discuss existing methods to solve this problem.

2.2. NETWORK INFERENCE METHODS

Mainly two approaches have been investigated in the literature to transform the network inference problem into standard classification problem (Vert, 2010) (see **Figure 1**). The first, more straightforward, approach, called *pairwise* or *global*, considers each pair as a single object and then apply any existing classification method on these objects (e.g., Takarabe et al., 2012). This approach requires a feature vector defined on pairs. When features on individual nodes are provided, they thus need to be transformed into features on pairs (Tastan et al., 2009). Several approaches have been proposed in the literature to achieve this, ranging from a simple concatenation or addition of the feature vectors of the nodes in the pair (Chen and Liu, 2005; Yu et al., 2012) to more complex combination schemes (Yamanishi et al., 2008; Maetschke et al., 2013). Different classification methods have been exploited in the literature: nearest neighbor algorithm (He et al., 2010), support vector machines (Paladugu et al., 2008), logistic regression (Ulitsky et al., 2009), tree-based methods (Wong et al., 2004; Yu et al., 2012), etc. In particular, in the context of support vector machines, several kernels have been proposed to compare pairs of objects on the basis of individual

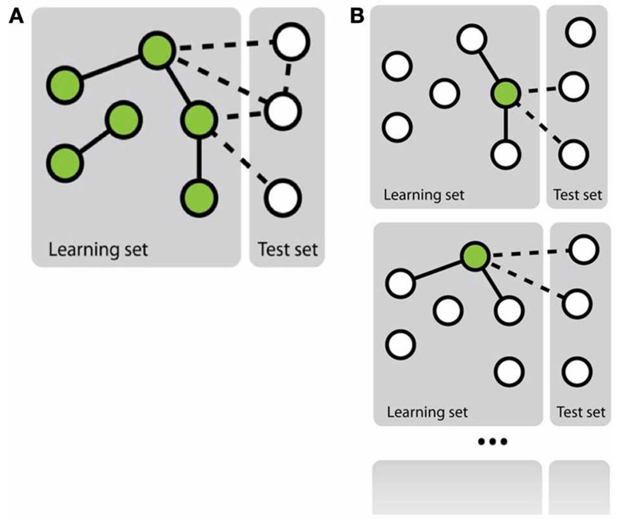


FIGURE 1 | Schematic representation of the two main approaches to solve the problem of network inference. (A) The global approach that solves a single supervised learning problem by considering each pair as an object for the learning. **(B)** The local approach that solves several supervised learning problems, each defined by a different node.

features defined on these objects that have been applied for supervised network inference (Vert et al., 2007; Hue and Vert, 2010; Brunner et al., 2012).

In the second approach, called *local* (Mordelet and Vert, 2008; Bleakley and Yamanishi, 2009; Vert, 2010; van Laarhoven et al., 2011; Mei et al., 2013), the network inference problem is divided into several smaller classification problems corresponding each to a node of interest and aiming at predicting, from the features, the nodes that are connected to this node in the network. More precisely, each of these classification problems is defined by a learning sample containing all nodes that are involved in a pair with the corresponding node of interest in LS_p. Interestingly, when trying to make a prediction for a given pair (n_r^i, n_c^j), one can aggregate the predictions of two classifiers: the one trained for n_r^i and the one trained for n_c^j. Note that it is only possible to train a classifier for a node that is involved in at least one positive and one negative interaction in LS_p. This prevents the use of the local approach to predict interactions for pairs where both nodes do not satisfy this property. Like for the global approach, in principle, any classification method can be used to train each of the classification models, but mainly support vector machines have been investigated in this context (Mordelet and Vert, 2008; Bleakley and Yamanishi, 2009).

From experiments in the literature, there does not seem to be a clear winner between the local and the global approach in terms of predictive accuracy. The global approach is typically more flexible as it can handle any kinds of features and can make prediction for pairs of unseen nodes, but it requires more computing times and resources, given that it aims to infer a network in one step.

Besides the global and local approaches that make use of existing classification methods, other more specific approaches have also been proposed for supervised network inference. For example, Kato et al. (2005) formulate the problem as a matrix completion problem (with input features) and solve it using an expectation-maximization-based approach. The problem has also been formulated as a distance metric learning problem (Vert and Yamanishi, 2005; Yamanishi, 2009): nodes of the graph are embedded into some euclidean space where they are close as soon as they are connected in the training graph and a mapping is then learned from the node feature space to this euclidean space. A related approach consists in defining a kernel between the nodes in the network that similarly encodes the connections between the nodes in the training graph and then exploit the kernel trick at the output of a regression method to learn an approximation of this kernel from the node features. This framework has been implemented using tree-based ensemble methods (Geurts et al., 2007) and ridge regression (Brouard et al., 2011) for example.

While our brief review focused on the inference of the network from node features, it is also possible to solve this problem by exploiting only the network itself. For example, Cheng et al. (2012) derive a similarly measure between nodes from the network topology and then use this similarity to infer new interactions. In a hybrid approach, some authors have also included features derived from the (training) network topology in the global approach to improve network inference (Ulitsky et al., 2009).

3. EVALUATION MEASURES

In this section, we review and discuss evaluation measures that have been used to quantify the quality of the predictions given by network inference methods. We focus here on statistical measures that compare a predicted network (or subnetwork) with the true one, as in the case of supervised network inference, some part of the true network is supposed to be available for training. In the general context of network inference, other performance measures have been proposed based either on functional annotations shared by genes/proteins or on topological properties of the inferred networks (see Emmert-Streib et al., 2012, for a survey).

The prediction given by a network inference method for a given pair of nodes can typically be of two kinds: a binary (0–1) value, coding for the presence or the absence of an interaction between the two nodes in the predicted network, or a real value, representing some confidence score associated to the pair: the higher the score, the higher the confidence or certainty of the model that there is an interaction between the nodes in the pair. Depending on the supervised network inference method used, this confidence score can have a probabilistic interpretation or not, but we will not assume it is the case. Of course, one can always transform a confidence score into a binary prediction using a decision threshold. The choice of an appropriate threshold is, however, not an easy problem in practice.

In this section, we assume that we have an adjacency matrix (of a complete or a partial graph) and an equivalent matrix of the binary or real scores predicted by a network inference method. In both cases, our goal is to quantify the quality of the predictions with respect to the true network represented by the adjacency matrix. Protocols to obtain these matrices will be discussed in section 4. We first discuss the case of binary predictions and then compare the receiver operating characteristic (ROC) curves and precision-recall (PR) curves that have been predominantly used to evaluate network inference methods that provide confidence scores. We end the section with a brief survey of other measures and a general discussion.

3.1. BINARY PREDICTIONS

Common criteria to evaluate binary predictions are the accuracy (the number of correctly predicted pairs divided by the total number of pairs) or equivalently the error rate (one minus the accuracy). However, network inference problems typically correspond to highly imbalanced classification problems as non-interacting pairs often far outnumber interacting ones. Accuracy is not appropriate in such situations because it greatly favors the majority class (high accuracy is given to a model predicting all pairs as non-interacting pairs). Alternative measures requires to differentiate between the possible types of errors, that are usually counted and compiled in a confusion matrix. In the case of binary classification, this matrix is a 2 × 2 matrix where the columns and rows represent, respectively the actual and the predicted classes and each cell contains the number of pairs corresponding to these classes. Denoting by positive an interaction and by negative a non-interaction, the confusion matrix is as follows:

	actual positive (P)	actual negative (N)
predicted positive (pred$P$)	true positive (TP)	false positive (FP)
predicted negative (pred$N$)	false negative (FN)	true negative (TN)

Several metrics can be then derived from this matrix to evaluate the performance of a model, among which:

- the *true positive rate* (TPR), also called the *sensitivity* or the *recall*, is equal to the number of true positives divided by the number of actual positives: $\frac{TP}{TP+FN}$ or $\frac{TP}{P}$,
- the *true negative rate* (TNR), also called the *specificity*, is equal to the number of true negatives divided by the number of actual negatives: $\frac{TN}{FP+TN}$ or $\frac{TN}{N}$,
- the *false positive rate* (FPR), corresponding to 1-*specificity*, is equal to the number of false positives divided by the number of actual negatives: $\frac{FP}{FP+TN}$ or $\frac{FP}{N}$,
- the *false negative rate* (FNR), also called the *miss*, is equal to the number of false negative divided by the number of actual negatives: $\frac{FN}{TP+FN}$ or $\frac{FN}{P}$,
- the *precision* is equal to the number of true positives divided by the number of predicted positives: $\frac{TP}{TP+FP}$.
- the *rate of positive predictions* (RPP) is equal to the number of predicted positive divided by the total number of examples: $\frac{TP+FP}{P+N}$ or $\frac{\text{pred}P}{P+N}$
- the *F-score* is equal to the harmonic mean of precision and recall:

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Except for the *F-score*, these measures should be combined to give a global picture of the performance of a method, e.g., sensitivity and specificity or precision and recall. In the case of confidence scores, all these performance measures can be computed for a given threshold on the confidence scores. Nevertheless, often, one would like to measure the performance of a method independently of the choice of a specific threshold. Several curves are used for that purpose that are exposed below.

3.2. ROC CURVES

ROC curves plot the TPR as a function of the FPR, when varying the confidence threshold (Fawcett, 2006). In concrete terms, the predictions are sorted from the most confident to the least confident, and the threshold is varied from the maximum to the minimum confidence score. Each value of the threshold corresponds to a different confusion matrix, and thus a different pair of values of the TPR and FPR, and corresponds to a point of the ROC curve. See **Figure 2A** for an example.

Sorted predictions	0.91	0.86	0.85	0.57	0.54	0.26	0.18	0.16	0.14	0.13
Rank	1	2	3	4	5	6	7	8	9	10
Actual values	**1**	**1**	0	**1**	0	0	**1**	0	0	0
FPR	0	0	1/6	1/6	1/3	1/2	1/2	2/3	5/6	1
TPR/Recall	1/4	1/2	1/2	3/4	3/4	3/4	1	1	1	1
Precision	1	1	2/3	3/4	3/5	1/2	4/7	1/2	4/9	2/5

Constants: $P = 4, N = 6$

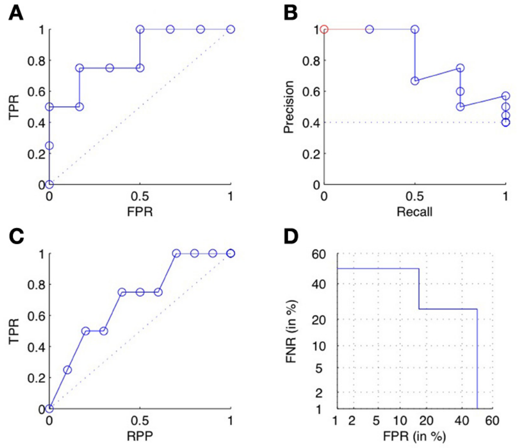


FIGURE 2 | ROC curve (A), precision-recall curve (B), lift chart (C), and DET curve (D) for the scores of the table above.

The two ends of the curve are always the two points (0, 0) and (1, 1), corresponding, respectively to pred$P = 0$ and pred$P = P + N$. A perfect classifier would give the highest values of prediction to the pairs that truly interact, and then would have a corresponding ROC curve passing through the point (0, 1). The curve relative to a random classifier corresponds to the diagonal connecting the two points (0, 0) and (1, 1) (the dotted line in **Figure 2A**).

For comparison purposes, it is often convenient to summarize a ROC curve with a single real number. The most common such measure is the area under the ROC curve (AUROC), which is equal to 1 for a perfect classifier and 0.5 for a random one. On the face of it, one typically assumes that the higher the AUROC, the better the predictions.

In many network prediction tasks, however, the number of interactions is much lower than the number of non-interactions. It is therefore important to achieve a low *FPR* as even moderate *FPR* can easily lead to much more *FP* predictions than *TP* predictions, and hence a very low precision. To better highlight the importance of small *FPR*, partial AUROC values are sometimes used instead of the full AUROC. For example, Tastan et al. (2009) propose statistics like $R50$, $R100$, $R200$, and $R300$ that measure the area under the ROC curve until reaching a *FP* equal to 50, 100, 200, and 300, respectively.

Another summary statistic of a ROC curve is the Youden index (Fluss et al., 2005), which is defined as the maximal value of TPR − FPR over all possible confidence thresholds. It corresponds to the maximal vertical distance between the ROC curve and the diagonal. The Youden index ranges between 0 (corresponding to a random classifier) and 1 (corresponding to a perfect classifier). This statistic was used for example in Hempel et al. (2011) to assess gene regulatory network inference methods.

3.3. PRECISION-RECALL CURVES

PR curves plot the precision as a function of the recall (equal to the TPR), when varying the confidence threshold. See **Figure 2B** for an example. A perfect classifier would give a PR curve passing through the point (1, 1), while a random classifier would have an average precision equal to $\frac{P}{P+N}$ (dotted line in **Figure 2B**). All PR curves end at the point $(1, \frac{P}{P+N})$ corresponding to predicting all pairs as positive. When all pairs are predicted as negative, recall is 0 but the precision is actually undefined. The coordinates of the first point of the PR curve will therefore be $(\frac{1}{P}, 1)$ if the most likely prediction is actually positive, and (0, 0) otherwise. To make all PR curve defined on the full [0, 1] interval, one sometimes adds a pseudo point to the curve at (0, 1) (**Figure 2B**).

The PR curve is also often summarized by the area under the curve (AUPR). The AUPR is sometimes called MAP, for Mean Average Precision (Manning et al., 2009; Tastan et al., 2009). Like for the AUROC, one typically assumes that the higher the AUPR, the better is the classifier, with the AUPR of a perfect classifier equal to 1 and the AUPR of a random classifier close to $\frac{P}{P+N}$. In practice, the AUPR can be computed from the curve completed with the additional pseudo-point or not. In the second case, one can rescale the area by dividing it by $1 - \frac{1}{P}$ so that its values is equal to 1 for a perfect classifier. Note that it is important to report exactly on which approach was used to compute the AUPR as it can make a significant difference when the number of positives is very small. For example, the AUPR of the PR curve of **Figure 2B** is equal to 0.81, 0.75, and 0.56, respectively with the pseudo-point, without the pseudo-point but with rescaling, and without the pseudo-point and without rescaling.

3.4. COMPARISON OF ROC AND PR CURVES

An important difference between ROC and PR curves is their different sensitivities to the ratio between positives and negatives (class imbalance) among the tested pairs: a ROC curve is independent of the precise value of this ratio, while a PR curve is not. To illustrate this fact, we triplicated every negative examples in the ranked list of predictions of **Figure 2** and plotted the new ROC and PR curves in **Figure 3**. As expected, we obtained exactly the same ROC curves, while the PR curves are different. This happens because, at fixed recall, a large change in *FP* will lead to no change in the *FPR* used in ROC curves (because to total number *N* of negatives will increase in the same proportion), but to a large change in the precision used in PR curves (Davis and Goadrich, 2006).

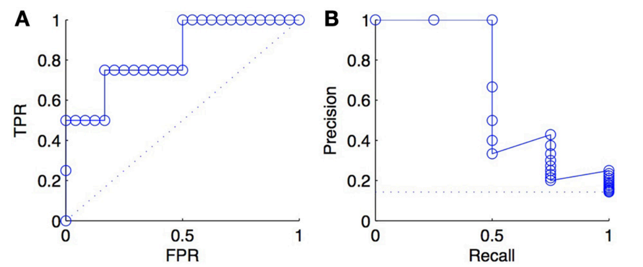


FIGURE 3 | ROC curve (A) and PR curve (B) for a list of scores where negative examples were tripled with respect to scores of Figure 2. The comparison with the curves in **Figure 2** shows that the ROC curve is unchanged and that the PR curve degrades, as a consequence of tripling the negatives.

This independence with respect to the particular content of the test sample in terms of positives and negatives is actually the main advantage of the ROC curve over the PR curve when it comes to compare different classification methods (Fawcett, 2006). ROC curves allow to compare classification methods whatever will be the ratio between positives and negatives expected when practically applying the model. Because of this independence, however, ROC curves do not really emphasize a particular intervals of values of this ratio and therefore favor methods that are good for a large range of such values. If one knows for example that the ratio between positives and negatives will be very low when applying the classification model, then one is typically only interested in the bottom-left part of the ROC curve. PR curves, on the other hand, provide a better picture of the performance of a method when the ratio between positives and negatives in the test data is close to the ratio one expects when practically applying the model.

The dependence of the PR curve on the ratio between positives and negatives can also be seen as a drawback. First, it means that PR curves (and their associated AUPR) obtained from different datasets can not really be compared when the ratio $\frac{P}{N}$ is very different. This is a limitation if one wants to compare the performance of a method across several networks for example. Second, because of this dependence, it is important that the ratio of positive and negative interactions in the subset of pairs used to validate the method is representative of the final application of the method. Otherwise, the PR curve will not provide a realistic evaluation of the method. Note, however, that it is possible to adapt a given PR curve to a ratio between positives and negatives different than the one adopted to generate it (Hue et al., 2010). Mathematical details are given in the supplementary information.

Another drawback of the PR curve is the potential unstability of the precision for small recall values. Indeed, for small values of pred*P*, the vertical changes of the curve from one confidence threshold to the next can be very huge, independently of the size of the dataset. This is more noticeable when the value of *P* is small because the horizontal changes are then also relatively large. This unstability makes the estimation of the true PR curve highly imprecise (Brodersen et al., 2010). It is, however, actually a direct consequence of the stronger focus put by the PR curve on the top of the ranking with respect to the ROC curve.

Despite these differences, it is interesting to note that a deep connection exists between the ROC and the PR spaces, in that a model dominates another model in the ROC space if and only if it dominates the same model in the PR space (Davis and Goadrich, 2006). In practice, however, it is often the case that a model does not dominate another model over the whole ROC and PR spaces and it might thus happen that a method's AUROC is greater than another method's AUROC, while the opposite is true concerning the AUPR.

3.5. OTHER MEASURES AND CURVES

ROC curves and PR curves are the most popular ways to estimate the performance of biological network inference methods, but some other measures and curves can also be found in the literature.

Lift charts (or cumulative lift charts), often used in marketing (Witten and Frank, 2005), plot the TPR, or recall, as a function of the RPP (rate of positive predictions), when varying the confidence threshold. See **Figure 2C** for an example. A perfect classifier would give a curve going through the points (0, 0), ($\frac{p}{p+n}$, 1) and (1, 1), while a random classifier would be equal to the diagonal connecting the two points (0, 0) and (1, 1).

For example, Geurts (2011) used a lift chart to evaluate the performance of supervised methods for the prediction of regulatory networks, and Yabuuchi et al. (2011) for the prediction of compound-protein interactions. Lift charts explicitly show the number of positive predictions (expressed as a percentage of all possible interactions) that one needs to accept to retrieve a given percentage of all truly positive interactions (recall). This is an important information when one is looking at the experimental validation of the predictions: a method that dominates another in terms of lift chart would require to experimentally test less interactions to achieve a given recall.

Note that when the number of positive examples is much smaller than the number of negative ones, as it often happens in biological networks, there is not much difference between the ROC curve and the lift chart.

Detection error tradeoff (DET) curves plot the two types of errors versus each other, i.e., FNR as a function of FPR (Martin et al., 1997). In addition, the two axes are log scaled. An example of DET curve is given in **Figure 2D**. Without the axis rescaling, a DET curve would be equivalent to a ROC curve (because FNR = 1 − TPR). The interest of the log scale is to expand the lower left part of the curve (which corresponds to the upper left part of the corresponding ROC curve), which as argued in Martin et al. (1997) makes the comparison between different methods easier. DET curves were used in Brunner et al. (2012) to evaluate classification methods working on pairs of objects.

Several authors (Li et al., 2009; Junaid et al., 2010; Lapins and Wikberg, 2010; Niijima et al., 2011) use a *correlation coefficient* for the evaluation of the performance of network inference methods. In this context, the latter is defined as

$$Q^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where the sum runs over all tested pairs, y_i and $\hat{y}_i$ are the true and predicted value corresponding to the ith pair and $\bar{y}$ is the average value of y_i. Q^2 values vary between 0 and 1, with $Q^2 = 1$ for a perfect classifier.

The *average normalized rank* is another way to compare the performance of different classifiers (Karni et al., 2009; Geurts, 2011). It computes the average rank of all actual positives in the ranking of all pairs according to their confidence score, and then divide it by the total number of pairs. Obviously smaller is the average rank and better is the model.

3.6. DISCUSSION

Biological network inference problems, as binary classification problems, are usually substantially imbalanced in favor of the negative class, as the proportion of interacting pairs among all possible pairs is very small. Given the discussion in section 3.4, this speaks in favor of the PR curve over the ROC curve. Let us nevertheless consider three typical scenarios related to the use of supervised network inference techniques and discuss the most appropriate use of these measures in each of these scenarios:

- *Development of new supervised network inference methods:* when trying to design a new supervised network inference method, one needs to assess its performance against existing methods, either on a specific target biological network if the method is specialized or on several networks if the method is generic. In this scenario, one has typically no specific application of the method in mind and the combination of both ROC and PR curves can be a good idea. While AUROC and AUPR summary values can be used for comparison purpose, it is always useful to actually report full ROC and PR curves to better characterize the areas of the ROC and PR where the new method dominates competitors.
- *Prioritizing interactions for experimental validation:* From a ranking of all the pairs from the most likely to interact to the less likely to interact, a biologist may want to validate experimentally the top-ranked pairs, i.e., the potentially new interacting pairs. More locally, he also may want to find the nodes (e.g., genes/proteins) the most likely to interact with a specific node of special interest for him. In this scenario, the biologist probably wants to find the best tradeoff between the number of true interactions he will find through the experimental validation and the cost associated to this validation. The former is measured by the recall and the latter is typically proportional to the *RPP*, which suggests the use of a lift chart. In addition, if the goal is also to minimize the rate of unsuccessful validation experiments (i.e., the precision), then also looking at the PR curve might be a good idea.
- *Global analysis of the predicted network:* We may want to use the top-ranked pairs to create a new network, or to complete an already known network, for visualization or a more global analysis of its main statistics. In these cases, we need to find the best possible tradeoff between precision (not to infer wrong things) and recall (to maximize the coverage of the true network). This tradeoff can be found from a PR curve. For example, one could derive from the PR curve the lowest confidence threshold corresponding to a precision greater than 50%.

4. EVALUATION PROTOCOLS

Given a learning set LS_p of pairs labeled as interacting or not, the goal of the application of supervised network inference methods is to get a prediction for all pairs not present in LS_p (or a subset of them depending on the application). In addition, one would like to compute an estimate of the quality of these predictions as measured with any of the metrics defined in the previous section. To obtain such estimation, one could rely only on the learning set LS_p as nothing is known about pairs outside this set by construction.

Standard supervised classification methods are typically validated using cross-validation (CV), i.e., leaving part of the examples in the learning sample aside as a test set, training a model from the remaining examples, and testing this model on the test set (and possibly repeat this procedure several times and average). Applying CV in the context of network inference, where we have to classify pairs, needs special care (Park and Marcotte, 2011). Indeed, the predictive performance of a method for a given pair highly depends on the availability in the training data of interactions involving any of the two nodes in the tested pair. It is typically much more difficult to predict pairs with nodes for which no example of interactions are provided in the training network.

As a consequence of this, pair predictions have to be partitioned into four sets, depending on whether the nodes in the pair to predict are represented or not in the learning sample of pairs LS_p. Denoting by LS_c (resp. LS_r) the nodes from U_c (resp. U_r) that are present in LS_p (i.e., which are involved in some pairs in LS_p) and by $TS_c = U_c \backslash LS_c$ (resp. $TS_r = U_r \backslash LS_r$) unseen nodes from U_c (resp. U_r), the pairs of nodes to predict (i.e., outside LS_p) can be divided into the following four families:

- $(LS_r \times LS_c) \backslash LS_p$: predictions of (unseen) pairs between two nodes which are represented in the learning sample.
- $LS_r \times TS_c$ or $TS_r \times LS_c$: predictions of pairs between one node represented in the learning sample and one unseen node, where the unseen node can be either from U_c or from U_r.
- $TS_r \times TS_c$: predictions of pairs between two unseen nodes.

These pairs are represented in the adjacency matrix in **Figure 4A**. In this representation, the rows and columns of the adjacency matrix have been ordered, without loss of generality, in order to make nodes from LS_r and LS_c appear first in the ranking and as a consequence, all four groups define rectangular and contiguous subregions of the adjacency matrix. Such ordering is always possible but the respective sizes of the four groups of pairs that this ordering defines is problem dependent. Thereafter, we simplify the notations by dropping the subscript r and c and denote the prediction sets as $LS \times LS$, $LS \times TS$, $TS \times LS$, and $TS \times TS$. In the case of an homogeneous undirected graph, only three sets can be defined as the two sets $LS \times TS$ and $TS \times LS$ are confounded.

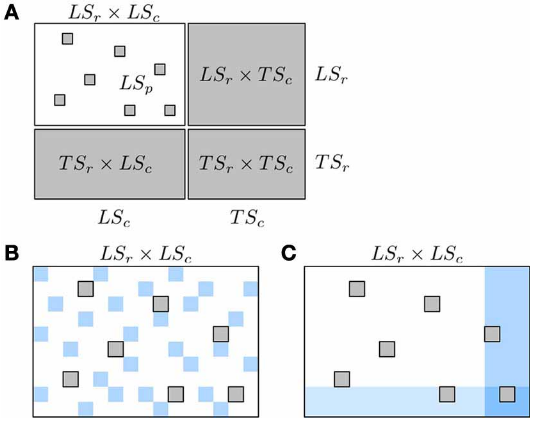


FIGURE 4 | Schematic representation of known and unknown pairs in the network adjacency matrix (A) and of the two kinds of CV, CV on pairs (B) and CV on nodes (C). In **(A)**: known pairs (that can be interacting or not) are in white and unknown pairs, to be predicted, are in gray. Rows and columns of the adjacency matrix have been rearranged to highlight the four families of unknown pairs described in the text: $LS_r \times LS_c$, $LS_r \times TS_c$, $TS_r \times LS_c$, and $TS_r \times TS_c$. In **(B)**,**(C)**: pairs from the learning fold are in white and pairs from the test fold are in blue. Pairs in gray represent unknown pairs that do not take part to the CV.

Typically, one expects different prediction performances for these different kinds of pairs and in particular, that $TS \times TS$ pairs will be the most difficult to predict since less information is available at training about the corresponding nodes. In consequence, we need ways to evaluate the quality of the predictions of these four groups separately. Below, we first present the two main CV procedures that have been proposed in the literature to evaluate supervised network inference methods and discuss which of these four kinds of predictions these procedures are evaluating (sections 4.1, 4.2). We then proceed with suggestions on how to practically assess network inference methods (section 4.3) and give an illustration on an artificial gene regulatory network (section 4.4).

4.1. CROSS-VALIDATION ON PAIRS

The most straightforward way to generate the learning and test sets needed for the CV, is to randomly select pairs from all the known pairs in LS_p (see **Figure 4B**). For example, in a specific step of a 10-fold CV, 90% of all the pairs from LS_p are chosen to be in the learning set, while the remaining 10% are then part of the test set. We call such CV *CV on pairs*. Many papers from the literature on supervised network inference only consider this sampling method (see e.g., Qi et al., 2006; Chang et al., 2010; Park and Marcotte, 2011; Yabuuchi et al., 2011).

With CV on pairs, each test set could in principle mix pairs from the four groups aforementioned. If LS_p is relatively dense, however, (i.e., there are only very few or no pairs in $LS_r \times LS_c \backslash LS_p$), the chance to have a node in a test set pair not present in any learning set pair will be very low. The test set will then be largely dominated by pairs from the $LS \times LS$ group. In this case, one can thus only consider the performance evaluated by CV on pairs as representative of the performance for the $LS \times LS$ pairs. When used to assess the global performance of a method, however, CV on pairs will in general give too optimistic estimates.

To obtain an estimate of the four kinds of predictions using CV on pairs, one could partition the pairs in the test fold into the four groups and then estimate the performance for each group separately. The CV scheme proposed in the next section provides, however, a more natural way to assess the three types of predictions involving the TS. CV on pairs should thus be reserved for the evaluation of $LS \times LS$ pairs. For that purpose, removing pairs in the test folds that do not belong to the $LS \times LS$ group might be useful to obtain a better estimate, especially when the size of LS_p is small with respect to the size of $LS_c \times LS_r$.

4.2. CROSS-VALIDATION ON NODES

Instead of sampling pairs, several authors have proposed to sample nodes. In the general case of a bipartite graph, the idea is to randomly split both sets LS_c and LS_r into two sets, respectively denoted LS'_c and TS'_c for LS_c and LS'_r and TS'_r for LS_r. The model is trained on the pairs in $(LS'_c \times LS'_r) \cap LS_p$ and then evaluated separately on three subsets (see **Figure 4C**):

- $(LS'_c \times TS'_r) \cap LS_p$ that gives an estimate of the $LS \times TS$ performance,

- $(TS'_c \times LS'_r) \cap LS_p$ that gives an estimate of the $TS \times LS$ performance,
- $(TS'_c \times TS'_r) \cap LS_p$ that gives an estimate of the $TS \times TS$ performance.

In addition, it might be interesting to evaluate the performance on the union of the three previous subsets of pairs to give an idea of the overall performance of the method. Better estimates could also be obtained by averaging results over k splits instead of one, where the different splits can be obtained either by repeated random resampling or by partitioning the two sets into k-folds and considering each fold in turn as a test set. In this latter case, partitioning LS_c and LS_r into k folds will lead to k^2 candidate (LS'_c, LS'_r) pairs for training and (TS'_c, TS'_r) pairs for evaluation but one could select only k of them arbitrarily to reduce the computational burden. The same approach can also be applied to homogeneous graphs to obtain estimate of the $LS \times TS$ and $TS \times TS$ performances.

CV on nodes has been applied, for example, for evaluating $LS \times TS$ and $TS \times TS$ performances for the prediction of a protein–protein interaction network and an enzyme network in Kato et al. (2005), Vert and Yamanishi (2005), and Geurts et al. (2007); or for evaluating $LS \times TS$, $TS \times LS$, and $TS \times TS$ performances for the prediction of drug-protein interactions in Yamanishi et al. (2008).

4.3. DISCUSSION

CV on pairs provides a natural way to estimate $LS \times LS$ predictions, while CV on nodes provide a natural way to estimate $LS \times TS$, $TS \times LS$, and $TS \times TS$ predictions. A global performance assessment of a method can therefore only be obtained by combining these two protocols. This was done only by a few authors (e.g., Yip and Gerstein, 2008; Bleakley and Yamanishi, 2009; Takarabe et al., 2012). The necessity to evaluate all four groups is, however, problem dependent. Again, when designing a new supervised network inference method, it is useful to report performances for all families separately, as a method can work well for one family and less good for another. If one is interested in the completion of a particular biological network, then the need for the evaluation will depend, on the one hand, on the content of the learning sample LS_p and, on the other hand, on which kinds of predictions the end user is interested in. Indeed, if all nodes are covered by at least one known interaction in LS_p, then there is no point in evaluating $LS \times TS$ or $TS \times TS$ predictions. If LS_p corresponds to a complete rectangular submatrix of the adjacency matrix (i.e., $LS_p = LS_c \times LS_r$), then there is no point in evaluating $LS \times LS$ predictions. Also, for some applications, the end-user might not be interested in the extension of the network over one of the two dimensions. For example, when inferring a regulatory network, one might only be interested in the prediction of new target genes for known TFs and not in the prediction of new TF (e.g., Mordelet and Vert, 2008).

In addition to the four groups previously defined, it is also possible to evaluate independently the predictions related to each individual node (to get for example an idea of the quality of the predictions of new target genes for a given TF). This can be achieved by dividing the test folds according to one of the nodes in the pairs and then to assess performance for each partition so obtained. In practice also, the quality of a prediction depends not only on the fact that the nodes in the pair belong or not to the learning sample, but also on the number of pairs in the learning sample that concern these nodes. We can indeed expect that, for a given node, the more interactions or non-interactions are known in the learning sample for this node, the better will be the predictions for the pairs that involve this node. Assessing each node separately can thus make sense to better evaluate this effect. We will illustrate this idea in section 4.4.2.

When using k-fold CV to estimate ROC or PR curves, one question we have not addressed so far is how to aggregate the results over the different folds. There are several ways to do that. If one is interested only in AUROC or AUPR values, then one could simply average AUROC or AUPR values over the k folds. If one wants to estimate the whole ROC or PR curves, there are two ways to obtain them: first, by averaging the k curves to obtain a single one, second by merging pairs from the k test folds with their confidence score and building a curve from all these pairs. In the first case, there are several alternative ways to average ROC (and PR) curves. One of them is to sample the x-axis in each curve and then average the k y-axis values corresponding to these points [this is called vertical averaging in Fawcett (2006)]. Merging all predictions together is easier to implement but it assumes that the confidence scores obtained from the k different models are comparable, which is not trivially true for all methods. Note that our own practical experience shows that there are only very small differences between these two methods of aggregation and we usually prefer to average the individual ROC curves so that they do not have to address the question of the compatibility of the confidence scores.

Finally, we have seen in section 3.4 that PR curves depend on the ratio between positives and negatives. This dependence should be taken into account when performing CV. If CV on pairs and CV on nodes use uniform random sampling, resp. of pairs and of nodes, to define the test folds, then they implicitly assume that the ratio between positives and negatives is the same in the test fold as in the learning sample of pairs. This seems a reasonable assumption in most situations but if one expects a different ratio among the predictions, then the procedure developed in section 3.4 can be used to correct the PR curve accordingly.

4.4. ILLUSTRATION

In this section, we will illustrate the use of CV with experiments on an artificial network. An artificial network was chosen so that it is possible to accurately estimate performance and therefore assess the different biases discussed in the paper. The chosen network is the artificial regulatory network simulated in the context of the DREAM5 network inference challenge (Marbach et al., 2012). This network is an artificial (bipartite) regulatory network, composed of 1565 genes, 178 TFs, and 4012 interactions, corresponding to 1.4% of all the pairs. The network has to be inferred from 804 artificial microarray expression values obtained in various conditions and mimicking typical real microarray compendia. To provide experiments on a homogeneous network as well, we transformed this network into a co-regulatory network composed of 1565 genes and in which there is an interaction between two genes if they are regulated by at least one common TF. The resulting network is composed of 4,191,120 interactions, corresponding to 17.1% of all pairs.

4.4.1. Performance over the four families of predictions

We performed a 10-fold CV on both the bipartite and homogeneous networks, with a local approach using Random Forests (Breiman, 2001). For the bipartite network, we sample first on pairs, and second on genes and on TFs. The resulting curves and areas under the curves are given in **Figures 5A,B**. Surprisingly, the prediction of interactions involving a TF present in the learning set, and a new gene (*LS* × *TS*) gives slightly better scores than the prediction of interactions involving a gene and a TF both present in the learning set (*LS* × *LS*). On the other hand, the prediction of pairs involving a new gene and a TF present in the learning set (*LS* × *TS*) or not (*TS* × *TS*) gives performances barely better than random. Finding new interactions for a known TF is thus much easier than finding interactions for a known gene.

For the homogeneous network, we sample first on the pairs and second on the genes. The resulting curves are shown in **Figures 5C,D**. Prediction of coregulation between two genes belonging to the learning set gives the best AUROC and AUPR. As expected prediction of coregulation between one known gene and one new gene gives less good performance, followed by prediction of coregulation between two new genes.

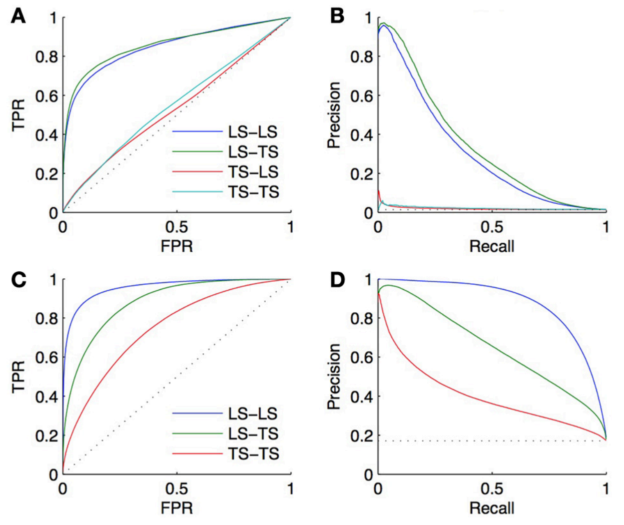


FIGURE 5 | ROC curves (A) and PR curves (B) for the four groups of predictions obtained by 10-fold CV on the DREAM5 artificial gene regulatory network. AUROC are, respectively, equal to 0.85, 0.86, 0.53, and 0.55 and AUPR are equal to 0.31, 0.34, 0.02, and 0.02. The performance of prediction of a pair involving a gene and a TF present in the learning set (*LS* × *LS*) is as good as the performance of prediction of a pair involving a gene absent and a TF present in the learning set (*LS* × *TS*). On the contrary, predicting an interaction involving a new TF is much more difficult (*TS* × *LS* and *TS* × *TS*). Bottom: ROC curves **(C)** and PR curves **(D)** obtained by 10-fold CV on the corresponding DREAM5 co-regulatory network. AUROC are, respectively, equal to 0.96, 0.88, and 0.75 and AUPR are equal to 0.88, 0.65, and 0.40. Predictions on pairs involving two genes from the learning set are the best, while predictions on pairs involving two genes from the test set are the worst.

These two examples clearly highlight the fact that all pairs are not as easy to discover as the others, and that it is thus important to distinguish them during the validation.

4.4.2. Per-node evaluation

As a second experiment, we computed the ROC and PR curves for each of the 178 TFs separately, from the result of the 10-fold CV on genes (bipartite graph). **Figure 6** shows the (average) AUROC and AUPR values for all TFs according to their degree. This plot shows that the quality of the predictions differs greatly from one TF to another and that the number of known pairs seems to affect this quality. For low values of degree (lower than about 20), the AUROC globally increases when the degree increases, but for higher values the AUROC does not seem to depend on it. On the other hand, AUPR values globally increase when the degree increases, for all values of TF.

4.4.3. A more realistic setting

The goal of CV is to estimate, from the training subnetwork, the performance one expects on the prediction of new interactions. We carried out a last experiment to evaluate the quality of the estimation obtained by CV in a realistic setting. In this setting, we assume that the known pairs are obtained by first randomly drawing 2/3 of the genes and 2/3 of the TFs and then randomly drawing 2/3 of all interacting and non-interacting pairs between these genes and TFs. The resulting training set thus contains about 30% of all possible pairs and the goal is to predict the remaining 70% pairs, which are divided into, respectively 15%, 22%, 22%, and 11% of *LS* × *LS*, *LS* × *TS*, *TS* × *LS*, and *TS* × *TS* pairs.

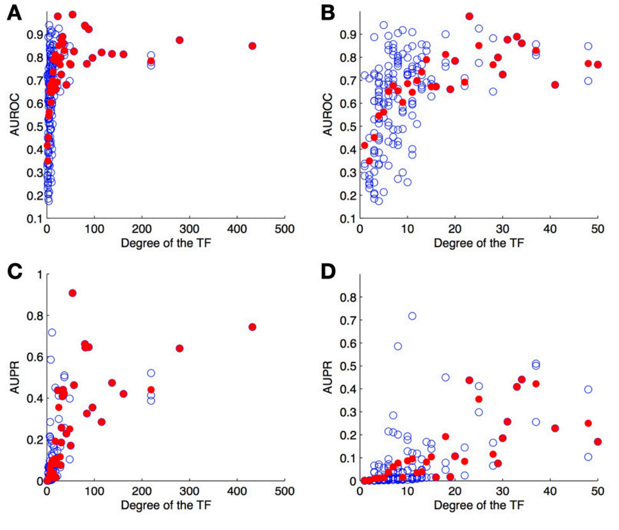


FIGURE 6 | AUROC (A,B) and AUPR (C,D) for each TF as a function of its degree (number of targets) on the DREAM5 network. Each value was obtained by 10-fold CV on genes. Each blue point corresponds to a particular TF and plots its average AUROC or AUPR value over the 10-folds. Each red point correspond to the average AUROC or AUPR values over all TFs of the corresponding degree. Globally, the higher the degree, the higher are the areas under the curve and so the better are the predictions.

Two validation experiments were performed. First, we evaluated the performance of the (global) Random Forests method by CV across pairs and across nodes on the 30% of known pairs (experiment A). Second, we trained local models based on Random Forests on the known pairs and we evaluated them on the 70% of pairs not used during training. Experiment A is therefore supposed to provide a CV estimate of the true performance as estimated by experiment B. The resulting ROC and PR curves obtained from these two experiments for the *LS* × *LS* and *LS* × *TS* families are shown in **Figure 7**. As expected, for both kinds of predictions, the curves obtained by the two experiments are very similar, with a very slight advantage to experiment B. This small difference comes from the fact that the number of pairs in the learning set of experiment B is 10% greater than the number of pairs in the learning sets of experiment A (because of 10-fold CV).

5. LACK OF NEGATIVE EXAMPLES

In biological networks, often truly non-interacting pairs are not available. Indeed it is often impossible for biologists to experimentally support the lack of an interaction between two nodes. For example you can prove that a specific drug acts on a set of proteins, and you may want to find other proteins being affected by this drug by using machine learning techniques, but you cannot prove that a particular set of proteins is not affected by the drug. This lack of negative examples leads to problems both when training and when evaluating a model. We discuss these two steps separately below and conclude with an illustration.

5.1. TRAINING A MODEL

Standard supervised machine learning methods require both positive and negative examples for training. The most common way to get around this limitation in the presence of only positive examples is to take as negative examples all, or a subset of, the unlabeled examples, i.e., in our context, considering all or some pairs that have not been measured as interacting as actually non-interacting. This approach has been adopted by most authors in the literature, e.g., in Geurts et al. (2007), Mordelet and Vert (2008), Yamanishi et al. (2008), Yip and Gerstein (2008), Bauer et al. (2011), van Laarhoven et al. (2011), and Takarabe et al. (2012), the authors use all unlabeled pairs as negatives and in Yip and Gerstein (2008), Chang et al. (2010), Hue et al. (2010), Yabuuchi et al. (2011), and Yu et al. (2012) they use only a subset of them. Although there is a risk that the presence of false negatives in the learning sample will affect the performance of the machine learning method, using only a subset of the unlabeled pairs as negative examples will, however, substantially reduce this risk in the context of biological networks. Indeed, the fraction of positive interactions is expected to be very small in common biological networks, which will lead to only a very small number of false negatives in the learning sample as soon as the size of the negative set is not too large with respect to the size of the positive set. For example, for the protein–protein interaction network of the yeast, it is estimated that 1 pair over 600 is actually interacting (Qi et al., 2006), which corresponds to ∼0.2% of all the possible pairs. A learning sample composed of 1000 positive and 1000 unlabeled pairs is therefore expected to contain in average only about two or three false negatives. In addition to the reduction of the number of false negatives, sampling the unlabeled pairs has also the advantage of decreasing the computational cost at the training stage and of improving the class imbalance in the training sample, which might affect the performance of classification methods (Pandey et al., 2010; Park and Marcotte, 2011).

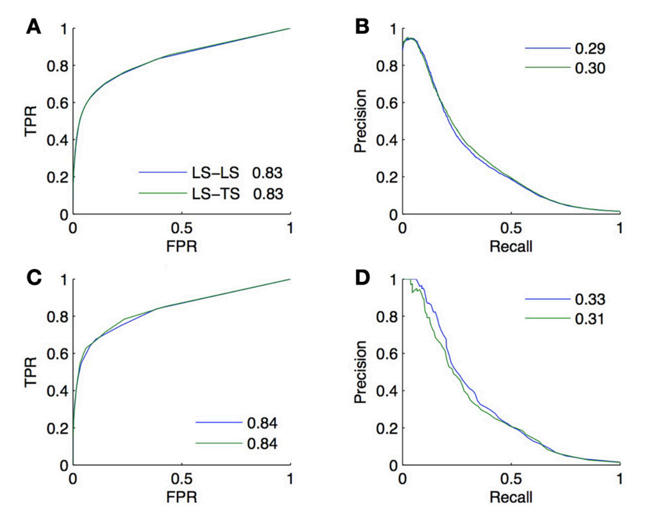


FIGURE 7 | Comparison of the CV estimates of the *LS* × *LS* and *LS* × *TS* scores, ROC curve in (A) and PR curve in (B), with true score values for the same two families of predictions, ROC curve in (C) and PR curve in (D). AUROC and AUPR values are found in the legends.

To even further reduce the risk of incorporating false negatives in the training data, one could also replace random sampling from the unlabeled pairs by a selection of a subset of more reliable negative examples using prior knowledge about the biological interactions of interest. This approach was considered for example in Ben-Hur and Noble (2006) for protein–protein interactions, in Ceccarelli and Cerulo (2009) for gene-TF interactions, and in Yousef et al. (2008) for microRNA-gene interactions.

Note that the presence of false negatives is not necessarily detrimental. Elkan and Noto (2008) showed that, under the assumption that the interactions in the learning sample are selected uniformly at random among all interactions, the presence of false negatives in the learning sample will only affect the confidence scores by a constant factor, which will thus leave ROC and PR curves for example unaffected. Although their assumption is quite strong, this nevertheless suggests that the presence of false negatives might have just a marginal effect on performance. As an illustration, we run the same experiment as in section 4.4 on the DREAM5 regulatory network only turning 10% of positives into negatives when training the model. The AUPR reduces from 0.31 to 0.29 and the AUROC from 0.85 to 0.84, showing that the presence of false negatives only very slightly affects the performance of Random Forests.

One drawback of considering unlabeled pairs as negative pairs for training the model is that the predictions provided by the model for these pairs will be biased toward low confidence scores. One way to obtain unbiased predictions for all unlabeled pairs is to use CV: construct a model using all known positive pairs and a random subset of the unlabeled pairs as negatives, use this model to obtain a prediction for all unlabeled pairs not used during the training stage, and repeat the procedure several times using different subsets of unlabeled pairs until all unlabeled pairs have obtained at least one prediction. Based on this general scheme, Mordelet and Vert (2013) proposed to train several models using small random subsamples of unlabeled pairs, leading to several predictions for each unlabeled pairs that are then aggregated.

Another approach to deal with the lack of negative examples is to forget about unlabeled examples and exploit machine learning methods, such as one-class support vector machines (Schölkopf et al., 2001), that can learn a model only from the positive examples. This approach was for example adopted in Yousef et al. (2008) to predict miRNA-gene interactions. Machine learning literature also provides several specific algorithms for dealing with positive and unlabeled examples, among which for example (Lee and Liu, 2003; Denis et al., 2005; Geurts, 2011), that could also be used in the context of supervised network inference. Geurts (2011) validated his method for the inference of regulatory networks, which showed improvement over standard two-classes methods.

5.2. EVALUATING A MODEL

The absence of true non-interacting pairs in the training data has also an impact on the validation of the model, as the different evaluation measures described in section 3 all rely on the availability of a set of known interacting and non-interacting pairs on which to perform the CV.

Like for training, the simplest way to deal with the lack of negatives for validating the model is to consider all unlabeled pairs within the test folds (generated in the context of CV on pairs or CV on nodes) as non-interacting pairs and then estimate ROC or PR curves under this assumption. The presence of false negatives in the gold standard will obviously affect the estimation of the performance. Let us assume that the ranking of the examples in a test fold is fixed and that a proportion x of positives are turned into negatives. Under this assumption, it can be shown that the *TPR* remains unchanged while *FPR* and *Prec* are modified as follows:

$$\mathrm{FPR}_{\mathrm{new}} = \frac{FP + TP{\cdot}x}{N + P{\cdot}x} > \mathrm{FPR} \qquad (1)$$

$$\mathrm{Prec}_{\mathrm{new}} = (1 - x)\mathrm{Prec} < \mathrm{Prec}, \qquad (2)$$

where the first inequality holds as soon as the ranking is better than random (see the supplementary information for the details). One can thus expect that the introduction of false negatives will systematically degrade both the ROC and the PR curves.

As an illustration, we carried out simulations on the DREAM5 regulatory network (see section 4.4). The model was trained with Random Forests with the local approach and we focus our experiment on the $LS \times LS$ pairs. The learning sample was kept unchanged but in each of the 10 CV folds (CV on pairs), we randomly turned a fraction x of positives into negatives, in order to simulate the introduction of false negatives. We tried several proportions $x \in \{0, 0.1, 0.2, \ldots, 0.9\}$ and got the curves shown in **Figures 8A,B**. As expected, the PR curves degrade when the ratio increases. More surprisingly, the ROC curves do not seem to be influenced by the ratio of false negatives. This can be explained by the fact that in Equation (1), $TP \cdot x$ becomes negligible compared to FP and $P \cdot x$ is negligible compared to N, even for small FPR values as soon as N is large with respect to P.

Actually, there are potentially two effects that play a role in the degradation of the PR curve in **Figure 8B**: the introduction of false negatives but also the alteration of class imbalance. Indeed, we have seen in section 3.4 that the PR curve was affected by this ratio. To try to assess both effects separately, we also generated the PR curves obtained from the initial curve by increasing the number of negatives in such a way that the ratio of P/N matches the ratio of P/N in the previous experiment for x ranging from 0 to 0.9. These curves are plotted in **Figure 8C**. They are also systematically degraded by the introduction of more negatives but the degradation is not as high as the degradation obtained by the addition of false negatives.

We can conclude from these experiments that PR curves are much more sensitive than ROC curves to false negatives in the true dataset. Interestingly, given Equation (2), if we can estimate the ratio x of false negatives, we can modify the PR curve simply by dividing the precision by $1 - x$, to obtain a more realistic PR curve. Note, however, that the correction in Equation (2) only applies under the assumption that false negatives will get scores distributed similarly as positives. This assumption is not unrealistic in practice as we indeed expect that false negatives will be predicted most often as positives (since they are in fact positives). However, it is also possible that for a given biological network, known interactions are the strongest ones (i.e., those with the

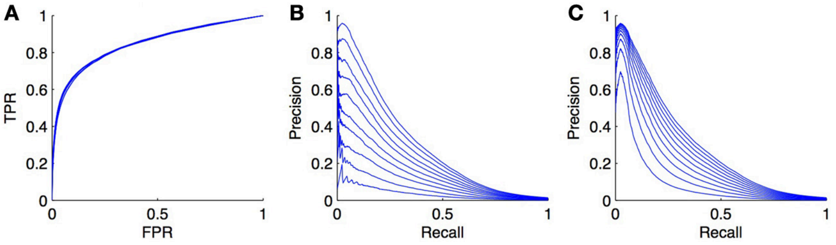


FIGURE 8 | Effect of false negatives on ROC and PR curves. We simulated false negatives in the DREAM5 regulatory network, during the testing stage. The ratio of false negatives does not influence the ROC curve **(A)**, but the PR curve **(B)** decreases while the ratio of positives turned into negatives increases. The ratio varies from 0 to 0.9. Curves **(C)** show the evolution of the PR curve when the ratio P/N is set similarly as in **(B)**. Although the PR curve degrades also in this case, the degradation is not as important as when false negatives are introduced.

strongest experimental support) and therefore false negatives will typically correspond to weaker interactions. Their scores, as predicted by network inference methods, can then be smaller than those of known positives. In this case, the degradation of the PR curve will most probably be somewhere in between curves in **Figures 8B,C**. Note that even though PR curves are affected by the introduction of false negatives, this is not really problematic when it comes to compare different inference methods on the same networks, as all methods will be affected in the same way by these false negatives. In this case, correcting the PR curve is not necessary.

Finally, we would like to note that the ratio between positives and negatives used to evaluate PR curves should be as close as possible to the expected ratio in the pairs to predict. Indeed, one could be tempted to estimate performance by CV on pairs on the positives and the selected negatives (randomly or from prior knowledge). The resulting PR curves will be, however, representative only for the given observed ratio between positives and negatives. If this ratio is different from the expected one, then one should apply the PR curve correction presented in section 3.4.

5.3. ILLUSTRATION

To illustrate the practical impact of the absence of negatives on validation, we reproduced the experiment of section 4.4.3 on the DREAM5 network, this time assuming that only positive (and unlabeled) pairs are available in the training data. More concretely, we again first randomly drew 2/3 of the genes and 2/3 of the TFs and then randomly drew 2/3 of the positive pairs existing among these genes and TFs. This set of positive pairs then defines our training network and the goal is to find new positive pairs among all the other ones (that are then considered as unlabeled). The positive pairs in the training set were chosen so that they match the positive pairs in the training set in the experiment of section 4.4.3.

Two validation experiments were performed. First, CV across pairs and nodes was carried out on all pairs between the selected genes (2/3) and TFs (2/3), considering all unlabeled pairs as negative (experiment A). Second, we randomly split the whole set of unlabeled pairs into two subsets. We trained a model on the positive pairs and each of these subsets taken in turn as the set of negative pairs and then used this model to obtain a prediction for the unlabeled pairs in the other subset. The resulting predictions were then evaluated against the true network (experiment B). Experiment A is thus supposed to provide a CV estimate of the true performance as computed by experiment B. The resulting ROC and PR curves obtained from these two experiments are shown in **Figure 9** for the $LS \times LS$ and $LS \times TS$ families.

ROC curves and AUROC scores obtained from experiments A and B are very close but noticeable differences appear in PR curves and AUPR scores. Indeed, experiment A gives higher AUPR than experiment B for $LS \times LS$ pairs, but gives lower AUPR for $LS \times TS$ pairs. In other words, CV overestimates the AUPR for $LS \times LS$ pairs and underestimates it for $LS \times TS$ pairs. As discussed above, these differences can be explained, on the one hand, by the presence of false negatives in the test data generated by the CV and, on the other hand, by the differences in the ratio between positives and negatives that exist in the two families of pairs between experiments A and B.

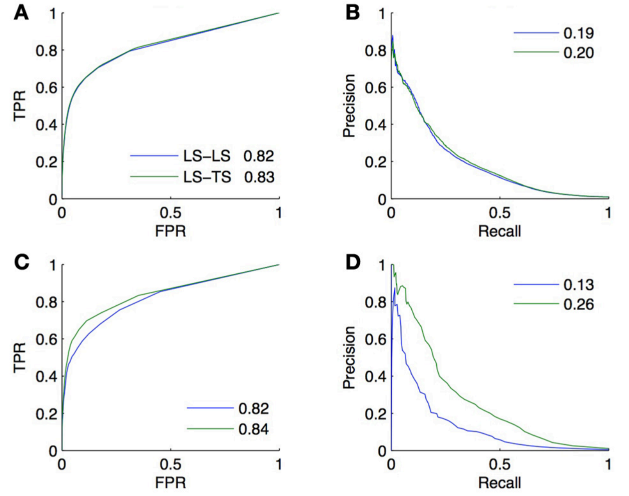


FIGURE 9 | Comparison of the CV estimates of the *LS* × *LS* and *LS* × *TS* scores, ROC curve in (A) and PR curve in (B), with true score values for the same two families of predictions, ROC curve in (C) and PR curve in (D), when only positive and unlabeled pairs are available. AUROC and AUPR values are found in the legends.

Assuming that both the ratio of false negatives in the training pairs and the ratio of positives and negatives among the unlabeled pairs are known or can be estimated, PR curves and AUPR scores obtained from experiment A can be corrected using results in sections 3.4, 5.2, so that they match the conditions of the application of the model in experiment B. Since these quantities are known for our artificial network, we performed these corrections, first adjusting the precision to account for the false negatives and then correcting the curve to account for the different ratio of positives versus negatives. The corrected AUPR are respectively 0.16 and 0.26 for $LS \times LS$ and $LS \times TS$, which are now closer to the value obtained from experiment B.

Note that another factor that could introduce a difference between CV scores and real scores is the composition of the training data in terms of positives and negatives, which might affect learning algorithms. In our experiment, however, the ratios of positives versus negatives in the training data are very close ($\sim 0.9\%$ for experiment A and $\sim 1.0\%$ for experiment B).

6. IMPACT OF HEAVY-TAILED NODE DEGREE DISTRIBUTION

Biological networks are typically non-random. In particular, many of them have a heavy-tailed distribution of node degrees: several nodes, called hubs, have degrees greatly higher than the average (Stumpf and Porter, 2012). In such networks, a new node, without consideration of its features, is more likely to interact with a hub than with a less connected node. As a consequence, it is possible in such network to obtain better than random interaction predictions without exploiting the node features, by simply connecting any new node with the more connected nodes in the training network.

Let us illustrate this on the DREAM5 *in silico* network. The topology of this network is based on known transcriptional regulatory networks of model organisms such as *S. cerevisiae* and *E. coli*. It clearly has a heavy-tailed node degree distribution (5% of the TFs collect about 50% of all interactions). **Figures 10A,B** shows the ROC and PR curves obtained using the same 10-CV folds as in section 4.4.1. The $LS \times LS$ pairs are now ranked according to the sum of the degrees of the nodes, computed in the training network, and the $LS \times TS$ and $TS \times LS$ pairs are now ranked according to the degree of the TF and of the gene, respectively. The AUROC and AUPR are, respectively, equal to 0.83 and 0.14 for $LS \times LS$, 0.83 and 0.17 for $LS \times TS$, and 0.54 and 0.02 for $TS \times LS$. We can conclude from these results that the degree of a TF is indeed greatly linked with the probability for it to interact with a known or a new gene. On the contrary, the degree of a gene does not influence its chance to interact with a new TF. Although better than random, it is important to note, however, that the degree-based ranking of $LS \times TS$ pairs does not allow to distinguish potential targets of a given TF since they all inherits the degree of the TF.

That it appears possible to complete a network based only on the degree of LS nodes shows that using a random classifier as a baseline for assessing the performance of supervised network inference methods is inappropriate. A network inference method that does not perform better than the simple degree-based ranking of the interactions is potentially unable to effectively extract useful information from the features. As a consequence, we believe that one should always report the performance of the degree-based ranking as a baseline for assessing the performance of a supervised network inference method. As an illustration, on the DREAM5 network, we obtained with the Random Forests method AUROC values of 0.85 and 0.86 and AUPR values of 0.31 and 0.34, respectively for $LS \times LS$ and $LS \times TS$ pairs (see section 4.4.1). The AUROC values of 0.85 and 0.86, although very good in absolute values, should be treated cautiously; they are indeed only slightly greater than the 0.83 AUROC of the degree-based ranking. In contrast, the doubling of the more robust AUPR value (from 0.14 and 0.17 for the degree-based random predictor to 0.31 and 0.34 for the trained model) indicates that the Random Forests are able to capture information from the feature vectors and indeed enable reliable predictions.

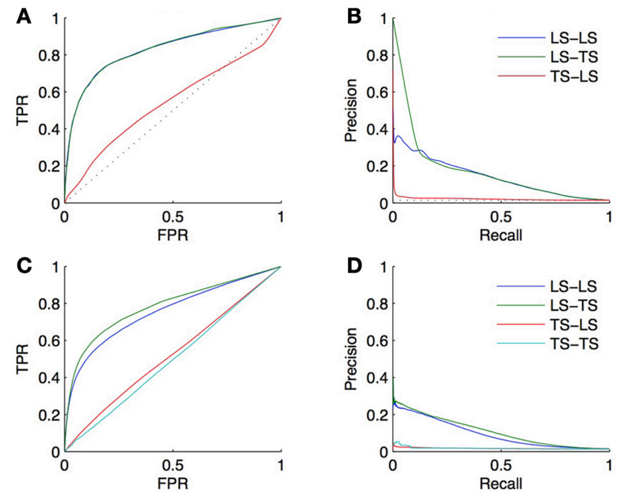


FIGURE 10 | The heavy-tailed degree distribution of many biological networks can lead to better than random predictions, only by exploiting the network topology and ignoring node or pair features. First row: ROC curves **(A)** and PR curves **(B)** obtained from predictions made on the DREAM5 dataset using the degree of the nodes in the learning set. Second row: ROC curves **(C)** and PR curves **(D)** obtained from predictions made on the DREAM5 dataset when randomly permuting the feature vectors relative to different nodes.

Even when the features are uninformative, supervised inference methods should be in principle able to "learn" and exploit this positive bias for interactions with nodes of high degree within the training data. Indeed, this is in this case the only way to get non-random predictions. To illustrate this assumption, we carried out an experiment on the DREAM5 network with the same protocol as in section 4.4.1 but making the features uniformative. To decorrelate the features from the network, the model is trained and tested by 10-fold CV on new data obtained by keeping the labels of the pairs unchanged but randomly permuting the feature vectors of the nodes. Resulting ROC and PR curves for $LS \times LS$ and $LS \times TS$ pairs are shown in **Figures 10C,D**. The AUROC and AUPR are, respectively, equal to 0.76 and 0.09 for $LS \times LS$ and 0.78 and 0.11 for $LS \times TS$. These results are slightly worse than the results obtained by the degree-based ranking but they are much better than random, although the features do not convey any information about the network by construction. Note that the AUROC and AUPR values averaged over each TF (as done in section 4.4.2) are, respectively, equal to 0.48 and 0.02 for $LS \times TS$ pairs. Like the degree-based ranking, the model trained on permuted features is unable to distinguish between possible targets of a given TF. This latter experiment further confirms that the degree-based ranking should be preferred to a random ranking as a baseline to assess the performance of supervised network inference methods.

7. DISCUSSION

In this paper, we discussed measures and protocols for the validation *in silico* of supervised methods for the inference of biological networks, i.e., methods that infer a biological network from a training sample of known interacting and non-interacting pairs and a set of features defined on the network nodes (or directly on pairs of nodes). Although this problem is very close to a standard supervised classification problem, it requires to address several important issues related to the need to classify pairs of entities in a candidate interaction and to the nature of biological networks. We carried out a rigorous examination of these issues that we supported by experiments on an artificial gene regulatory network. The main guidelines that can be drawn from this examination are as follows:

- Network inference methods have been assessed mainly using PR curves and ROC curves. The choice of an appropriate metric should be dictated mainly by the application but generally PR curves are more appropriate than ROC curves given the highly imbalanced nature of the underlying classification problem, related to the very sparse nature of most biological networks. While PR curves are sensitive to the ratio of positives versus negatives in the test data, we show that it is straightforward to adapt them to a new ratio. A further important characteristic of biological networks that should influence the choice of a performance metric is the heavy-tailed degree distribution. We show that this degree distribution severely affects the ROC curves, making it difficult to estimate the performance of inference methods by the AUROC, while PR curves are much less affected.

- When validating a model, it is necessary to divide the predictions into four groups, given that the two nodes might either be present or absent in the learning sample of interactions. Indeed, performance is typically very different from one group to another and improves when the number of training interactions involving the nodes in the pairs to be predicted increases. The quality of the predictions for pairs where both nodes have interactions in the training network can be assessed using CV over pairs in the training data. The quality of the predictions for the three other groups of pairs, where at least one node is not represented in the training data, is best assessed by using CV over nodes. Unless the inference problem at hand makes some subgroups of predictions irrelevant, we advocate the joint use of both kinds of CV to get a more detailed assessment of the performance of an inference method.
- We discussed the lack of experimental support for non-interacting pairs in most biological networks. We reviewed several ways to address this problem at training time and showed that the presence of false negatives does not really affect ROC curves but can result in an underestimation of the PR curve. Assuming that the proportion of false negatives in the test data is known and that false negatives are selected randomly among positives, we show that it is possible to correct the PR curve so that it better reflect true performances. The correction is, however, not necessary when one only wants to compare different methods.
- We showed empirically that a heavy-tailed node degree distribution seemingly enables a better than random inference only by exploiting the topology of the training network. As a consequence, random guesses should not be taken as valid baselines for supervised network inference methods, in order not to overestimate the performance. Every validation of a supervised inference method should always be supplemented by a reporting of the performance of the simple degree-based score (or a classifier grown from randomly permuted feature vectors).

Thereby, we provided the most comprehensive examination and discussion of issues in the evaluation of supervised inference techniques so far. Given that the examined supervised techniques exploiting prior information on the network are typically superior in performance to unsupervised approaches, a reliable assessment is particularly desirable. Following the guidelines we derived will enable a more rigorous assessment of supervised inference methods, will contribute to an improved comparability of the different approaches in this field and will thus furthermore aid researchers in improving the state of the art methods.

Still, there remain several open questions about supervised network inference methods and their validation. First, with a few exceptions, most papers in the domain focus on a given type of biological network. Yet, unlike unsupervised methods that need some prior knowledge to derive their confidence scores, supervised methods are most of the time generic in that they could be applied to any network without much adaptation. A thorough empirical comparison of these methods on several networks with different characteristics is missing to really understand the advantages and limitations of all these methods. While we argue, as others, that predictions within the different pair subgroups should be assessed separately, we have not discussed ways to take into account the resulting information to obtain better global network predictions. Indeed, most methods eventually provide a single ranking of all pairs to be predicted. How to take into account the performance differences between the different groups of pairs to reorganize this ranking into a better one, and whether this is actually possible, remains an interesting open question for future research. In this review, we focus on the statistical and *in silico* validation of network inference methods using CV techniques. Such validation helps assess the quality of the predictions and therefore decide on a confidence threshold that best suits application needs. However, even more important is the experimental validation of the predictions provided by network inference techniques. Experimental validation depends on the nature of the biological network at hand and therefore a discussion of these techniques is out of the scope of this review. Note nevertheless that experimental validation will be influenced also by the lack of experimental support for non-interacting pairs and that for some (more abstract) networks, experimental validation might be very difficult (e.g., disease-gene networks).

ACKNOWLEDGMENTS

FUNDING

This work was partially funded by the Interuniversity Attraction Poles Programme (IAP P7/19 DYSCO), initiated by the Belgian State, Science Policy Office and by the French Community of Belgium (ARC REFRACT). Marie Schrynemackers is recipient of a F.R.I.A. fellowship of F.R.S.-FNRS, Belgium.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://www.frontiersin.org/journal/10.3389/fgene.2013.00262/abstract

REFERENCES

Bauer, T., Eils, R., and Konig, R. (2011). Rip: the regulatory interaction predictor–a machine learning-based approach for predicting target genes of transcription factors. *Bioinformatics* 27, 2239–2247. doi: 10.1093/bioinformatics/btr366

Ben-Hur, A., and Noble, W. (2006). Choosing negative examples for the prediction of protein-protein interactions. *BMC Bioinformatics* 7(Suppl. 1):S2. doi: 10.1186/1471-2105-7-S1-S2

Bleakley, K., Biau, G., and Vert, J.-P. (2007). Supervised reconstruction of biological networks with local models. *Bioninformatics* 23, i57–i65. doi: 10.1093/bioinformatics/btm204

Bleakley, K., and Yamanishi, Y. (2009). Supervised prediction of drug-target interactions using bipartite local models. *Bioinformatics* 25, 2397–2403. doi: 10.1093/bioinformatics/btp433

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1017934522171

Brodersen, K. H., Ong, C. S., Stephan, K. E., and Buhmann, J. M. (2010). "The binormal assumption on precision-recall curves," in *International Conference on Pattern Recognition* (Istanbul), 4263–4266. doi: 10.1109/ICPR.2010.1036

Brouard, C., d'Alché-Buc, F., and Szafranski, M. (2011). "Semi-supervised penalized output kernel regression for link prediction," in *Proceedings of ICML* (Bellevue, Washington), 593–600.

Brunner, C., Fischer, A., Luig, K., and Thies, T. (2012). Pairwise support vector machines and their application to large scale problems. *J. Mach. Learn. Res.* 13, 2279–2292.

Ceccarelli, M., and Cerulo, L. (2009). "Selection of negative examples in learning gene regulatory networks," in *IEEE International Conference on Bioinformatics and Biomedicine Workshop, BIBMW 2009* (Washington, DC), 56–61. doi: 10.1109/BIBMW.2009.5332137

Cerulo, L., Elkan, C., and Ceccarelli, M. (2010). Learning gene regulatory networks from only positive and unlabeled data. *BMC Bioinformatics* 11:228. doi: 10.1186/1471-2105-11-228

Chang, D. T.-H., Syu, Y.-T., and Lin, P.-C. (2010). Predicting the protein-protein interactions using primary structures with predicted protein surface. *BMC Bioinformatics* 11(Suppl. 1):S3. doi: 10.1186/1471-2105-11-S1-S3

Chen, X.-W., and Liu, M. (2005). Prediction of protein-protein interactions using random decision forest framework. *Bioinformatics* 21, 4394–4400. doi: 10.1093/bioinformatics/bti721

Cheng, F., Chuang, L., Jiang, J., Lu, W., Li, W., Liu, G., et al. (2012). Prediction of drug-target interactions and drug repositioning via network-based inference. *PLos Compuat. Biol.* 8:e1002503. doi: 10.1371/journal.pcbi.1002503

Davis, J., and Goadrich, M. (2006). "The relationship between precision-recall and ROC curves," in *Proceedings of the 23rd International Conference on Machine Learning* (Pittsburgh, PA), 233–240. doi: 10.1145/1143844.1143874

Denis, F., Gilleron, R., and Letouzey, F. (2005). Learning from positive and unlabeled examples. *Theor. Comput. Sci.* 348, 70–83. doi: 10.1016/j.tcs.2005.09.007

Elkan, C., and Noto, K. (2008). "Learning classifiers from only positive and unlabeled data," in *KDD '08: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New Yor, NY), 213–220. doi: 10.1145/1401890.1401920

Emmert-Streib, F., Glazko, G. V., Altay, G., and de Matos Simoes, R. (2012). Statistical inference and reverse engineering of gene regulatory networks from observational expression data. *Front. Genet.* 3, 1–15. doi: 10.3389/fgene.2012.00008

Fawcett, T. (2006). An introduction to {ROC} analysis. *Pattern Recogn. Lett.* 27, 861–874. doi: 10.1016/j.patrec.2005.10.010

Fluss, R., Faraggi, D., and Reiser, B. (2005). Estimation of the Youden Index and its associated cutoff point. *Biom. J. (Biometrische Zeitschrift)* 47, 458–472. doi: 10.1002/bimj.200410135

Geurts, P. (2011). "Learning from positive and unlabeled examples by enforcing statistical significance," in *JMLR: Workshop and Conference Proceedings*. Vol. 15 (Lauderdale, FL), 305–314.

Geurts, P., Touleimat, N., Dutreix, M., and d'Alché Buc, F. (2007). Inferring biological networks with output kernel trees. *BMC Bioinformatics* 8(Suppl. 2):S4. doi: 10.1186/1471-2105-8-S2-S4

He, Z., Zhang, J., Shi, X.-H., Hu, L.-L., Kong, X., Cai, Y.-D., et al. (2010). Predicting drug-target interaction networks based on functional groups and biological features. *PLoS ONE* 5:e9603. doi: 10.1371/journal.pone.0009603

Hempel, S., Koseska, A., Nikoloski, Z., and Kurths, J. (2011). Unraveling gene regulatory networks from time-resolved gene expression data—a measures comparison study. *BMC Bioinformatics* 12:292. doi: 10.1186/1471-2105-12-292

Hue, M., Riffle, M., Vert, J.-P., and Noble, W. S. (2010). Large-scale prediction of protein-protein interactions from structures. *BMC Bioinformatics* 11:144. doi: 10.1186/1471-2105-11-144

Hue, M., and Vert, J.-P. (2010). "On learning with kernels for unordered pairs," in *Proceedings of the 27th International Conference on Machine Learning* (Haifa, Israel). 463–470.

Junaid, M., Lapins, M., Eklund, M., Spjuth, O., and Wikberg, J. E. S. (2010). Proteochemometric modeling of the susceptibility of mutated variants of the HIV-1 virus to reverse transcriptase inhibitors. *PLoS ONE* 5:e14353. doi: 10.1371/journal.pone.0014353

Karni, S., Soreq, H., and Sharan, R. (2009). A network-based method for predicting disease-causing genes. *J. Comput. Biol.* 16, 181–189. doi: 10.1089/cmb.2008.05TT

Kato, T., Tsuda, K., and Kiyoshi, A. (2005). Selective integration of multiple biological data for supervised network inference. *Bioinformatics* 21, 2488–2495. doi: 10.1093/bioinformatics/bti339

Lapins, M., and Wikberg, J. E. (2010). Kinome-wide interaction modelling using alignment-based and alignment-independent approaches for kinase description and linear and non-linear data analysis techniques research article. *BMC Bioinformatics* 11:339. doi: 10.1186/1471-2105-11-339

Lee, W., and Liu, B. (2003). Learning with positive and unlabeled examples using weighted logistic regression. *Proc. Int. Conf. Mach. Learn.* 20, 448.

Li, S., Xi, L., Wang, C., Li, J., Lei, B., Liu, H., et al. (2009). A novel method for protein-ligand binding affinity prediction and the related descriptors exploration. *J. Comput. Chem.* 30, 900–909. doi: 10.1002/jcc.21078

Maetschke, S. R., Madhamshettiwar, P. B., Davis, M. J., and Ragan, M. A. (2013). Supervised, semi-supervised and unsupervised inference of gene regulatory networks. *Brief. Bioinformatics* doi: 10.1093/bib/bbt034. [Epub ahead of print].

Manning, C., Raghavan, P., and Schütze, H. (2009). *An Introduction to Information Retrieval*. New York, NY: Cambridge University Press.

Marbach, D., Costello, J., Küffner, R., Vega, N., Prill, R., Camacho, D., et al. (2012). wisdom of crowds for robust network inference. *Nat. Meth.* 9, 794–804. doi: 10.1038/nmeth.2016

Martin, A., Doddington, G., Kamm, T., Ordowski, M., and Przybocki, M. (1997). "The DET curve in assessment of detection task performance," in *Fifth European Conference on Speech Communication and Technology, EUROSPEECH 1997* (Rhodes, Greece), 1899–1903.

Mei, J.-P., Kwoh, C.-K., Yang, P., Li, X.-L., and Zheng, J. (2013). DrugÐtarget interaction prediction by learning from local information and neighbors. *Bioinformatics* 29, 238–245. doi: 10.1093/bioinformatics/bts670

Mordelet, F., and Vert, J.-P. (2008). Sirene: supervised inference of regulatory networks. *Bioninformatics* 24, i76–i82. doi: 10.1093/bioinformatics/btn273

Mordelet, F., and Vert, J.-P. (2013). A bagging SVM to learn from positive and unlabeled examples. *Pattern Recog. Lett.* doi: 10.1016/j.patrec.2013.06.010. (in press).

Niijima, S., Yabuuchi, H., and Okuno, Y. (2011). Cross-target view to feature selection: identification of molecular interaction features in ligand-target space. *J. Chem. Inf. Model.* 51, 15–24. doi: 10.1021/ci1001394

Paladugu, S. R., Zhao, S., Ray, A., and Raval, A. (2008). Mining protein networks for synthetic genetic interactions. *BMC Bioinformatics* 9:426. doi: 10.1186/1471-2105-9-426

Pandey, G., Zhang, B., Chang, A. N., Myers, C. L., Zhu, J., Kumar, V., et al. (2010). An integrative multi-network and multi-classifier approach to predict genetic interactions. *PLoS Comput. Biol.* 6:e1000928. doi: 10.1371/journal.pcbi.1000928

Park, Y., and Marcotte, E. M. (2011). Revisiting the negative example sampling problem for predicting protein-protein interactions. *Bioinformatics* 27, 3024–3028. doi: 10.1093/bioinformatics/btr514

Qi, Y., Bar-Joseph, Z., and Klein-Seetharaman, J. (2006). Evaluation of different biological data and computational classification methods for use in protein interaction prediction. *Proteins* 63, 490–500. doi: 10.1002/prot.20865

Ryan, C., Greene, D., Cagney, G., and Cunningham, P. (2010). Missing value imputation for epistatic maps. *BMC Bioinformatics* 11:197. doi: 10.1186/1471-2105-11-197

Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A., and Williamson, R. (2001). Estimating the support of a high-dimensional distribution. *Neural Comput.* 13, 1443–1471. doi: 10.1162/089976601750264965

Stumpf, M. P. H., and Porter, M. A. (2012). Critical truths about power laws. *Science* 335, 665–666. doi: 10.1126/science.1216142

Takarabe, M., Kotera, M., Nishimura, Y., Goto, S., and Yamanishi, Y. (2012). Drug target prediction using adverse event report systems: a pharmacogenomic approach. *Bioinformatics* 28, i611–i618. doi: 10.1093/bioinformatics/bts413

Tastan, O., Qi, Y., Carbonell, J. G., and Klein-Seetharaman, J. (2009). Prediction of interactions between hiv-1 and human proteins by information integration. *Pac. Symp. Biocomput.* 14, 516–527.

Ulitsky, I., Krogan, N., and Shamir, R. (2009). Towards accurate imputation of quantitative genetic interactions. *Genome Biol.* 10, R140. doi: 10.1186/gb-2009-10-12-r140

van Laarhoven, T., Nabuurs, S. B., and Marchiori, E. (2011). Gaussian interaction profile kernels for predicting drug-target interaction. *Bioinformatics* 27, 3036–3043. doi: 10.1093/bioinformatics/btr500

Vert, J.-P. (2010). "Reconstruction of biological networks by supervised machine learning approaches," in *Elements of Computational Systems Biology*, eds

H. Lodhi and S. Muggleton (Oxford: John Wiley & Sons, Inc.), 165–188 (Chapter 7). doi: 10.1002/9780470556757.ch7

Vert, J.-P., Qiu, J., and Noble, W. S. (2007). A new pairwise kernel for biological network inference with support vector machines. *BMC Bioninformatics* 8(Suppl. 10):S8. doi: 10.1186/1471-2105-8-S10-S8

Vert, J.-P., and Yamanishi, Y. (2005). "Supervised graph inference," in *Advances in Neural Information and Processing System*, (Vancouver, BC), 1433–1440.

Witten, I. H., and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques.* 2nd Edn. Boston, MA: Morgan Kaufmann.

Wong, S. L., Zhang, L. V., Tong, A. H. Y., Li, Z., Goldberg, D. S., King, O. D., et al. (2004). Combining biological networks to predict genetic interactions. *PNAS* 101, 15682–15687. doi: 10.1073/pnas.0406614101

Yabuuchi, H., Niijima, S., Takematsu, H., Ida, T., Hirokawa, T., Hara, T., et al. (2011). Analysis of multiple compound-protein interactions reveals novel bioactive molecules. *Mol. Syst. Biol.* 7, 472. doi: 10.1038/msb.2011.5

Yamanishi, Y. (2009). Supervised bipartite graph inference. *Adv. Neural Inform. Process. Syst* 21, 1841–1848.

Yamanishi, Y., Araki, M., Gutteridge, A., Honda, W., and Kanehisa, M. (2008). Prediction of drug-target interaction networks from the integration of chemical and genomic spaces. *Bioinformatics* 24, i232–i240. doi: 10.1093/bioinformatics/btn162

Yamanishi, Y., and Vert, J.-P. (2005). Supervised enzyme network inference from the integration of genomic data and chemical information. *Bioinformatics* 21, i468–i477. doi: 10.1093/bioinformatics/bti1012

Yip, K. Y., and Gerstein, M. (2008). Training set expansion: an approach to improving the reconstruction of biological networks from limited and uneven reliable interactions. *Bioinformatics* 25, 243–250. doi: 10.1093/bioinformatics/btn602

Yousef, M., Jung, S., Showe, L. C., and Showe, M. K. (2008). Learning from positive examples when the negative class is undetermined-microRNA gene identification. *Algorithms Mol. Biol.* 3, 2. doi: 10.1186/1748-7188-3-2

Yu, H., Chen, J., Xu, X., Li, Y., Zhao, H., Fang, Y., et al. (2012). A systematic prediction of multiple drug-target interactions from chemical, genomic, and pharmacological data. *PLoS ONE* 7:e37608. doi: 10.1371/journal.pone.0037608

Conflict of Interest Statement: 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.

Received: 15 September 2013; paper pending published: 03 October 2013; accepted: 13 November 2013; published online: 03 December 2013.

*Citation: Schrynemackers M, Küffner R and Geurts P (2013) On protocols and measures for the validation of supervised methods for the inference of biological networks. Front. Genet. **4**:262. doi: 10.3389/fgene.2013.00262*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 11 December 2013
doi: 10.3389/fgene.2013.00263

On the underlying assumptions of threshold Boolean networks as a model for genetic regulatory network behavior

Van Tran[1], Matthew N. McCall[1]*, Helene R. McMurray[2] and Anthony Almudevar[1]

[1] Department of Biostatistics and Computational Biology, University of Rochester Medical Center, Rochester, NY, USA
[2] Department of Biomedical Genetics, University of Rochester Medical Center, Rochester, NY, USA

Edited by:
Benjamin Haibe-Kains, Institut de recherches cliniques de Montréal, Canada

Reviewed by:
George Georgakilas, Al. Fleming BSRC, Greece
Jèrôme Feret, INRIA (Institut National en Informatique et Automatique), France

***Correspondence:**
Matthew N. McCall, Department of Biostatistics and Computational Biology, University of Rochester Medical Center, 265 Crittenden Blvd, Rochester, NY 14642, USA
e-mail: mccallm@gmail.com

Boolean networks (BoN) are relatively simple and interpretable models of gene regulatory networks. Specifying these models with fewer parameters while retaining their ability to describe complex regulatory relationships is an ongoing methodological challenge. Additionally, extending these models to incorporate variable gene decay rates, asynchronous gene response, and synergistic regulation while maintaining their Markovian nature increases the applicability of these models to genetic regulatory networks (GRN). We explore a previously-proposed class of BoNs characterized by linear threshold functions, which we refer to as *threshold Boolean networks* (TBN). Compared to traditional BoNs with unconstrained transition functions, these models require far fewer parameters and offer a more direct interpretation. However, the functional form of a TBN does result in a reduction in the regulatory relationships which can be modeled. We show that TBNs can be readily extended to permit self-degradation, with explicitly modeled degradation rates. We note that the introduction of variable degradation compromises the Markovian property fundamental to BoN models but show that a simple state augmentation procedure restores their Markovian nature. Next, we study the effect of assumptions regarding self-degradation on the set of possible steady states. Our findings are captured in two theorems relating self-degradation and regulatory feedback to the steady state behavior of a TBN. Finally, we explore assumptions of synchronous gene response and asynergistic regulation and show that TBNs can be easily extended to relax these assumptions. Applying our methods to the budding yeast cell-cycle network revealed that although the network is complex, its steady state is simplified by the presence of self-degradation and lack of purely positive regulatory cycles.

Keywords: Boolean network, genetic regulatory network, attractor, steady state, state augmentation, asynchronous update, feedback loop, yeast cell-cycle

1. INTRODUCTION

Dynamic models are used frequently to study the evolution of a genetic regulatory network (GRN) over time [see De Jong (2002) for a review]. Often accompanying these models is a graph representing the relationships among the genetic components (e.g., proteins, DNA, RNA). The components are represented by nodes and the regulatory relationships by edges. The dynamic models range from highly quantitative frameworks such as systems of differential equations [see Heinrich and Schuster (1996) for an introduction] to more qualitative models such as Boolean networks (BoN) (Kauffman, 1969). Although systems of differential equations are explicit and detailed in their description of network trajectories, they require specialized knowledge of kinetic parameters, time constants, and the mechanism underlying the process. In comparison, BoN are easier to construct and interpret. In a BoN, gene expression is discretized into one of two states, e.g., on/off, up/down, or active/inactive. Regulation is modeled by logic functions (e.g., AND, OR, NOT) that code the influence of the effector genes. Genetic regulation is either positive, resulting in increased gene expression, or negative, resulting in decreased gene expression. While discretizing gene expression is certainly a simplification, similar approaches have resulted in increased reproducibility and robustness when estimating both absolute and differential gene expression (Parmigiani et al., 2002; Scharpf et al., 2003; Zilliox and Irizarry, 2007; McCall et al., 2011), and Boolean network models have been used to successfully model gene regulatory networks (Albert and Othmer, 2003; Espinosa-Soto et al., 2004; Li et al., 2004; Davidich and Bornholdt, 2008). For certain small networks, systems of differential equations and BoN are qualitatively similar in their state transitions and long term behavior (Glass and Kauffman, 1972, 1973). These two types of models can differ in their results when applied to networks with many nodes and complex gene interactions.

Ultimately a desirable model is one that retains the relative ease of modeling and interpretation of a BoN and the quantitative precision of differential equations. A model that possesses these qualities is the BoN proposed by Li et al. (2004) to study the

Network Inference

- Q: Which kinds of biological networks have been inferred in the paper?
- A: We studied genetic regulatory networks (GRN), specifically the budding yeast cell-cycle network, using a threshold Boolean network (TBN) model specified by linear functions and a threshold.
- Q: How was the quality/utility of the inferred networks assessed? How were these networks validated?
- A: We studied how the TBN model behaves under different assumptions of gene self-degradation and different parameter specifications. We Markovianized self-degradation and showed that the resulting model is more tractable. We proposed and proved two theorems relating gene self-degradation to a TBN's attractor set and used these results to assess the behavior of the budding yeast cell cycle. Our results were then compared to those of a widely cited GRN model.
- Q: A few sentences explaining the main positive/negative results described in the paper.
- A: We showed how the TBN model accommodates aspects of GRNs such as variable Markovian self-degradation, asynchronous gene update, and synergistic relationships, making the model more representative of real biological networks. Additionally, we found that the complexity of a GRN can be summarized by the presence of self-degradation and cycles comprised of only positive regulations. The primary limitation of TBNs is that they cannot easily model all possible regulatory relationships. Nevertheless, the mathematical tractability and qualitative characteristics of a TBN make it a desirable model for understanding GRNs.

budding yeast cell-cycle. Cited by more than 600 articles, their BoN employs a simple, elegant linear function with a threshold that utilizes far fewer parameters than a BoN specified by truth tables. Because of the influential results of Li et al.'s threshold Boolean network (TBN) model, a thorough analysis of the model's mathematical properties and fidelity to true network behavior are important. A key aspect of their model is the treatment of genetic degradation. Degradation primarily occurs in three ways: (a) negative regulation by other genes in the network, (b) negative regulation by other (unmeasured) genes not in the network, and (c) intrinsic protein degradation. The latter two are indistinguishable in a GRN and are commonly referred to as *self-degradation*.

Our evaluation of the TBN consists of: (1) characterizing the regulatory relationships that the TBN can and cannot express, (2) showing how self-degradation has a substantial impact on a GRN's steady state behavior, (3) Markovianizing self-degradation, (4) proving that steady states of a GRN are sensitive to gene interaction strengths, (5) commenting on the role of self-degradation and interaction strength in asynchronous gene update, and (6) augmenting the TBN to allow for synergistic and antagonistic relationships. The extensions improve a TBN's representation of a GRN and the theoretical results break down its complexity. In Section 2, we formally introduce BoN, their dynamic properties and Li et al.'s cell-cycle TBN. In Section 3, we evaluate the TBN and present our theorems relating self-degradation to steady state behavior. A summary and discussion of our findings follows in Section 4.

2. MATERIALS AND METHODS

2.1. A REVIEW OF BOOLEAN NETWORKS AND DYNAMIC PROPERTIES

A Boolean Network (BoN) is defined as a directed graph $\mathcal{G}(\mathcal{X}, \mathcal{E})$ with Boolean transition functions. The graph $\mathcal{G}$ is composed of a set of nodes $\mathcal{X} = \{1, \ldots, N\}$ and a set of edges $\mathcal{E}$, in which a directed edge represents a causal relationship between two nodes. Each node i can have either state $x_i = 0$ or $x_i = 1$. Whenever there is an edge $i \to j \in \mathcal{E}$, j is called the *child* of i and i is called the *parent* of j in $\mathcal{G}$. Associated with each node is a Boolean function $f_i : \mathcal{B}^N \mapsto \mathcal{B}$ where $\mathcal{B} = \{0, 1\}$. This function specifies how the state of node i changes over time. Denote the state of node i at time t as $x_i(t)$. Node i updates its state by the Markovian process, $x_i(t+1) = f_i(x_1(t), \ldots, x_k(t))$ where $1, \ldots, k$ are its parents. In other words, the current state of a node is determined by a function of its parents' previous states. Although f_i is defined to take N inputs, the relevant arguments are the parents' states since all other nodes do not directly affect i. In GRNs, an f_i specifies the regulatory relationship between gene i and the rest of the network. The entire network updates synchronously by the process, $\mathbf{x}(t+1) = A(\mathbf{x}(t))$, where $\mathbf{x} = (x_1, \ldots, x_N)$ is a state vector and $A : \mathcal{B}^N \mapsto \mathcal{B}^N$ is the model's operator. To be exact, A is a vector whose components are the functions, f_i. A network path is a sequence,

$$\mathbf{x}(0) \to \mathbf{x}(1) \to \mathbf{x}(2) \to \ldots$$

The long term behavior or steady state of a BoN can be characterized by its attractors. An *attractor* is a set of network states that occur infinitely often in the sequence $A^t(\mathbf{x}(0))$ with t $\geq$ 1. If the set contains only one element, then the attractor is referred to as a fixed point, otherwise the attractor is periodic. Formally, a *fixed point* is defined as $\mathbf{x} = A(\mathbf{x})$. An important feature of an attractor is its *basin of attraction*, which is the set of state vectors from which the network reaches the attractor. The size of the basin of attraction represents the attractor's pull on the network states. Growing evidence suggests that an attractor represents a particular cell fate (Kauffman, 1969; Huang et al., 2005).

2.2. THE CELL-CYCLE THRESHOLD BOOLEAN NETWORK

The cell-cycle of the budding yeast *Saccharomyces cerevisiae* is a phenomenon that continues to fascinate and generate knowledge even after years of research. Li et al. (2004) developed a dynamic BoN to model the cycle and "demonstrated that the cell-cycle network is extremely stable and robust for its function" (p.4781).

Their BoN uses a linear transition function with a threshold, henceforth referred to as a TBN, in the following manner:

$$x_i(t+1) = \begin{cases} 1, \ \sum_j a_{ij}x_j(t) > 0 \\ 0, \ \sum_j a_{ij}x_j(t) < 0 \\ x_i(t), \ \sum_j a_{ij}x_j(t) = 0 \end{cases} \quad (1)$$

where $x_j(t)$ is the expression of the regulator protein j at the current time t, $x_i(t+1)$ is the expression of the regulated protein i at the next time $t+1$, and interaction coefficient a_{ij} codes the strength and type of regulation that protein j exerts on protein i. Positive regulation is specified by positive values of a_{ij} and negative regulation by negative values of a_{ij}. Any regulation is a product of the parent's state $x_j(t)$ and the type and strength of the regulation a_{ij}. The next state of a protein depends only on its parents' current states. Specifically, the next state $x_i(t+1)$ of protein i is 'on' if the sum of its parents' regulatory effects surpasses 0, "off" if the sum is below 0, and when the sum is 0, the state remains the same. *Self-degradation* is a process not incorporated in Equation (1), but defined separately as: if $\sum_j a_{ij}x_j(t) = 0$ from $t = t_s$ to $t = t_s + t_d - 1$ then $x_i(t_s + t_d) = 0$, where t_d is referred to as the protein's *lifetime*. A higher value of t_d translates to a slower rate of decay. In the cell cycle TBN constructed in Li et al. (2004), only proteins not negatively regulated by others possess the self-degradation property (we note, however, that Swi5 appears to be an exception, as indicated in Figure 1 of Li et al. (2004)). Proteins that do not self-degrade maintain their current state according to line 3 of Equation (1). For ease of reference, we refer to these proteins as having the *persistence* property.

Proteins in the cell-cycle network belong to one of four classes: (a) cyclins (Cln1,-2,-3, Clb1,-2,-5,-6), (b) inhibitors/competitors of cyclins (Sic1, Cdh1, Cdc20, Cdc14), (c) transcription factors (SBF, MBF, Mcm1/SFF, Swi5), and (d) checkpoints. We focus on a simplified network having only the cell size checkpoint. The cell-cycle starts at phase G1 where the cell size becomes large enough and Cln3 reaches a high enough concentration, i.e., its Boolean state is equal to 1. When these two conditions are met, the cell commits to division. Next, the cell moves into S phase in which DNA is synthesized. After S phase is the gap phase G2, and in the final phase M, chromosomes separate and the yeast cell divides into two cells. This phenomenon repeats when the right conditions encourage cell growth and division.

Accompanying the TBN model in Equation (1) is a graph depicting the relationships among the proteins in the cell-cycle network. We reproduced the cell-cycle network in **Figure 1**. The graph is identical to Li et al.'s except for green self loops that we added to proteins that are assumed to persist. Functionally, **Figure 1** is equivalent to theirs. An edge between two nodes represent one of four regulatory relationships, negative regulation, positive regulation, self-degradation and persistence. These relationships are represented with a *red edge*, *green edge*, *yellow loop*, and *green self loop* respectively (note that all genes possess either a green self loop or a yellow loop). Li et al. assigned all positive regulations (green edges) the same interaction coefficient $a_{ij} = a_g$, and all negative regulations (red edges) $a_{ij} = a_r$. Although a_{ij} is allowed to take on any real value, Li et al.'s main results are based on $a_g = -a_r = 1$. They claimed that "the results are insensitive to the values of the weights a_g and a_r ...and to the protein lifetime t_d, as long as $-a_r \geq a_g$ and $t_d > 0$" (p. 4785).

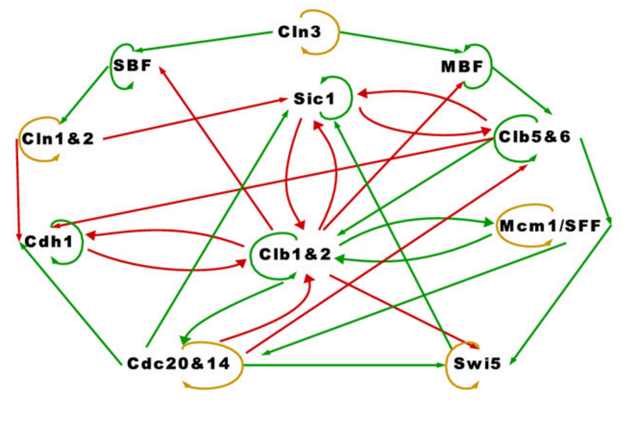


FIGURE 1 | The simplified yeast cell-cycle network.

The cell-cycle network in **Figure 1** appears to be very complex. The network contains 11 proteins, some proteins have as many as five regulators, and there are many feedback loops. With the exception of Swi5, a protein that is not negatively regulated by others in the network self-degrades (yellow loop), otherwise it persists (green self loop). We will show how the attractor set changes when Swi5 is set to persist instead of degrade, which illustrates the network's sensitivity to the assumptions of self-degradation. An important feature of this network is that the positive regulations (green edges) are almost acyclic except for the cycle between Clb1&2 and Mcm1/SFF, key players in the M phase or mitosis. We will discuss in more detail how this cycle plays a crucial role in the simplicity of the network's long term behavior.

Compared to a BoN specified by truth tables, the TBN in Equation (1) captures genetic relationships with far fewer parameters, which is especially convenient when the model space is relatively large. As an illustration, suppose a network has N nodes and each node i has k_i parents. Defining a BoN with truth tables requires $\sum_i^N 2^{k_i}$ parameters, 2^{k_i} parameters per node, while specifying the TBN in Equation (1) requires only $\sum_i^N k_i$ parameters, k_i of a_{ij} per node. The TBN is a hybrid between a BoN and a system of differential equations that retains the interpretability of the former and the mathematical tractability of the latter.

In the next section, we analyze the TBN model and propose extensions related to self-degradation, asynchronous gene update and synergistic relationships. We also state theoretical results that translate self-degradation and network cycles to network steady state behavior.

3. RESULTS

3.1. THRESHOLD BOOLEAN NETWORK MODEL

The primary limitation of the model described by Equation (1) is that only the regulatory relationship OR can be expressed. For example, given proteins, i, j, and k, expressing i if $j \cup k$ can be achieved by setting $a_{ij} = a_{ik} = 1$. However, expressing i if $j \cap k$ is impossible with any combinations of a_{ij} and a_{ik}. To encode an AND relationship and other types of regulations, the threshold needs to be greater than zero. An example of a TBN with a non-zero threshold was implemented by Davidich and Bornholdt (2008) to model the fission yeast cell-cycle. We present a more general form of the model in Equation (1) by including a threshold parameter $\alpha_i \geq 0$:

$$x_i(t+1) = \begin{cases} 1, \ \sum_j a_{ij}x_j(t) > \alpha_i \\ 0, \ \sum_j a_{ij}x_j(t) < \alpha_i \\ x_i(t), \ \sum_j a_{ij}x_j(t) = \alpha_i. \end{cases} \tag{2}$$

Clearly, Equation 1 is a special case of Equation 2 in which $\alpha_i = 0 \ \forall i$. By varying thresholds and interaction coefficients, it is possible to encode many regulatory relationships. Given proteins, i, j, and k, encoding the relationship i if $j \cap k$ would simply require setting $a_{ij} = a_{ik} = 0.5$ and $\alpha_i = 0.99$. Even more complicated relationships can be expressed using the TBN model. For example, i if $(j \cup k) \cap l$ could be achieved by setting $a_{ij} = a_{ik} = 0.1$, $a_{il} = 0.95$, and $\alpha_i = 1$.

However, not all relationships can be expressed. One such relationship is i if $(j \cap k) \cup (l \cap m)$. The following example illustrates this issue:

Example. In order to encode the relationship i if $(j \cap k) \cup (l \cap m)$, the coefficients a_{ij}, a_{ik}, a_{il}, a_{im} and the threshold α_i would have to satisfy the following inequalities:

$$\begin{aligned} a_{ij} + a_{ik} &> \alpha_i \\ a_{il} + a_{im} &> \alpha_i \\ a_{ij} + a_{il} &\leq \alpha_i \\ a_{ij} + a_{im} &\leq \alpha_i \\ a_{ik} + a_{il} &\leq \alpha_i \\ a_{ik} + a_{im} &\leq \alpha_i. \end{aligned}$$

Summing the first 2 inequalities produces $a_{ij} + a_{ik} + a_{il} + a_{im} > 2\alpha_i$. Summing the last four inequalities produces $2a_{ij} + 2a_{ik} + 2a_{il} + 2a_{im} \leq 4\alpha_i$. The contradiction shows that it is not possible to encode the above relationship using any TBN of the form in Equation (2). Although inclusion of the threshold parameter α_i permits a far wider range of regulatory relationships, some limitations remain.

3.2. SELF-DEGRADATION

3.2.1. Steady state characteristics

Setting negative regulations (red edges) at the same rate $a_{ij} = a_r = -1$, positive regulations (green edges) at the same rate $a_{ij} = a_g = 1$ and protein lifetime $t_d = 1$, the main result of the cell-cycle TBN, reported in Li et al. (2004), is the set of attractors in **Table 1A**. The largest basin of attraction shown is 1764. Of $2^{11} = 2048$ possible network states, 1764 states flow toward the fixed point (0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0), in which inhibitor proteins Cdh1 and Sic1 stay active indefinitely even when the rest of the network is off. Although the cell-cycle network is very complex, the attractor set has only seven attractors, which are all fixed points.

Thomas (1981) explored the effects of different *regulatory circuits* or feedback loops on the composition of the attractor set. Regulatory circuits are classified as positive or negative depending on whether the number of negative regulations (red edges) in the circuit is odd or even. Thomas proposed that positive circuits are necessary to generate multiple attractors and negative circuits are necessary to generate fixed points and periodic attractors. These ideas were later formalized in theorems by Remy et al. (2008); Richard (2010), and various conditions for a unique fixed point attractor set have been developed by Robert (1980); Shih and Dong (2005); Richard (2013). The theorems and results in this manuscript build upon these works by examining the effect of self-degradation and regulatory circuits on a network's long term behavior.

Theorem 1. *Let $\mathcal{G} = (\mathcal{X}, \mathcal{E})$ be a TBN of the form in Equation* (2) *with N nodes, $\mathcal{X} = \{1, \ldots, N\}$ and edges $\mathcal{E}$. Suppose each threshold parameter satisfies $\alpha_i \geq 0$ for each i. If every node has a self-degradation loop and network cycles must have at least 1 negative regulation (red edge), then the network's attractor is a unique fixed point, the null state.*

The proof requires the following definition. Let $\mathcal{G} = (\mathcal{X}, \mathcal{E})$ be a graph. An ordering of nodes $1, \ldots, N$ is a *topological ordering* relative to $\mathcal{G}$ if, whenever we have $i \to j \in \mathcal{E}$, then $i < j$. A parent node has a lower order than a child node. Most importantly, a graph is directed acyclic or *DAG* if and only if it has a topological ordering.

Proof. Denote the set of nodes having either an incoming or outgoing positive regulation (green edge) as $\mathcal{X}_n = \{1, \ldots, n\} \subset \mathcal{X}$. Given that cycles with all positive regulation (green edges) do not exist, choose a topological ordering (with respect to green edges only) for $\mathcal{X}_n$, say $\mathcal{T}$, and add directed null edges, which have no real regulatory effect, to all pairs of nodes in $\mathcal{X}_n$ not having an edge such that $\mathcal{T}$ is not violated. Then $\mathcal{X}_n$ has the unique topological ordering $\mathcal{T} = 1, \ldots, n$. The expression of a node in $\mathcal{G} = (\mathcal{X}, \mathcal{E})$ at time t is a function of nodes with smaller topological order and other nodes in $\mathcal{X}$ at the previous time $t-1$, i.e.,

$$\begin{aligned} x_i(t) = f_i(\{x_1(t-1), \ldots, x_{i-1}(t-1)\}, \{x_i(t-1), \ldots, \\ x_n(t-1), \ldots, x_N(t-1)\}) \end{aligned}$$

where f_i is the transition function for node i of the form in Equation (2) in which the parameter a_{ij} can take any magnitude so long as positive regulation is defined by a positive sign and negative regulation by a negative sign.

The proof proceeds from the observation that, under the stated hypothesis, if for t_d consecutive time points all nodes with topological ordering smaller than i have value 0, at the time point t immediately following we must also have $x_i(t) = 0$.

By mathematical induction, we will show that $(x_1(k), \ldots, x_n(k)) = (0, \ldots, 0)$ for some time k and remains at $\vec{0}$ after time k. At some time $t < k$,

$$\begin{aligned} x_1(t) &= f_1(\{\emptyset\}, \{x_1(t-1), \ldots, x_N(t-1)\}) \\ &= 0, \end{aligned}$$

Table 1 | The attractor set for the cell-cycle threshold Boolean network under different interaction coefficients.

Basin size	Cln3	MBF	Clb5&6	Mcm1/SFF	Swi5	Cdc20&14	Cdh1	Cln1&2	SBF	Sic1	Clb1&2
(A) $a_g = 1$											
1764	0	0	0	0	0	0	1	0	0	1	0
151	0	0	0	0	0	0	0	1	1	0	0
109	0	1	0	0	0	0	1	0	0	1	0
9	0	0	0	0	0	0	0	0	0	1	0
7	0	1	0	0	0	0	0	0	0	1	0
7	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	0
(B) $a_g = 2$											
1978	0	0	0	0	0	0	1	0	0	1	0
57	0	0	0	0	0	0	0	1	1	0	0
7	0	0	0	0	0	0	0	0	0	1	0
5	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	0
(C) $a_g = 3$											
1936	0	0	0	1	1	1	1	0	0	1	1
59	0	0	0	0	0	0	1	0	0	1	0
40	0	0	0	0	0	0	0	1	1	0	0
7	0	0	0	0	0	0	0	0	0	1	0
5	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	0

Protein lifetime is set at $t_d = 1$. All negative regulations are assigned a common coefficient $a_{ij} = a_r = -1$. All positive regulations are assigned $a_{ij} = a_g$. (A) Shows the attractor set associated with $a_g = 1$. (B) Shows the attractor set associated with $a_g = 2$. (C) Shows the attractor set associated with $a_g = 3$. For each panel, the rows are the attractors, which are all fixed points, and columns 2 through 12 indicate whether a protein is on (1) or off (0) in the attractor. Column 1 lists the basin size of each attractor.

and remains at 0 indefinitely through negative regulation or self-degradation. At some $t' > t$,

$$\begin{aligned} x_2(t') &= f_2^{(t'-t)}(\{x_1(t)\}, \{x_2(t), \ldots, x_n(t), \ldots, x_N(t)\}) \\ &= f_2^{(t'-t)}(\{0\}, \{x_2(t), \ldots, x_n(t), \ldots, x_N(t)\}) \\ &= 0 \end{aligned}$$

where the composite function $f_2^{(t'-t)}$ is the $(t'-t)$th iteration of the transition function f_2, and $(t'-t) \leq t_d$, for any t_d. Node 2 remains at 0 indefinitely through negative regulation or self-degradation. Assume that for some l nodes, all with order less than n, satisfies at time $t'' > t'$,

$$x_1(t'') = \ldots = x_l(t'') = 0,$$

and remains at 0 indefinitely through negative regulation or self-degradation.

Then at time $k > t''$,

$$\begin{aligned} x_n(k) &= f_n^{(k-t'')}(\{x_1(t''), \ldots, x_{n-1}(t'')\}, \{x_n(t''), \ldots, x_N(t'')\}) \\ &= f_n^{(k-t'')}(\{0, \ldots, 0\}, \{x_n(t''), \ldots, x_N(t'')\}) \\ &= 0. \end{aligned}$$

where $f_n^{(k-t'')}$ is the $(k-t'')$th iteration of the transition function f_n, and $(k-t'') \leq t_d$, for any t_d. Node n remains at 0 indefinitely. For all nodes not in $\mathcal{X}_n$, they remain at state 0 through negative regulation or self-degradation. Therefore, $(x_i(k), \ldots, x_N(k)) = \vec{0}$ and remains a fixed point after time k. □

In short, the proof shows that when upstream positive regulations are shut down by self-degradation, the network turns off in a cascading fashion due to the topological order and self-degradation. The theorem applies to an entire class of networks whose member graphs may have any number of genes, any number of cycles with at least one negative regulation (red edge), differing interaction coefficients a_{ij} and differing protein lifetimes t_d. The theorem is invariant to a_{ij} and t_d because these parameters only work to speed up or slow down the rate at which the network reaches the null attractor. An example of a network belonging to this class is displayed in **Figure 2A**.

Consider a more general network class that is still acyclic in the positive regulations (green edges) but has the additional feature of persistence (green self loops). An example of such a network is shown in **Figure 2B**.

We noted above that the degradation model defined here implies an assignment to each gene of either a yellow loop or a

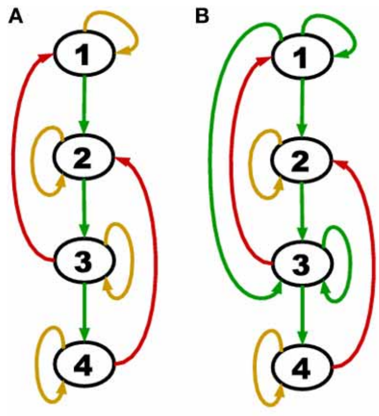


FIGURE 2 | (A) A network with all genes self degrading (yellow loop on each node) and acyclic positive regulations (green edges). **(B)** A network with persistence (green self loop) in addition to self-degradation and acyclic positive regulations.

green self loop. Theorem 1 concerns the special case in which all genes are assigned yellow loops. A green self loop is formally a cycle (which does not contain a red edge), and so the hypothesis of Theorem 1 does not hold if any persistent nodes are present.

However, suppose we are given a TBN which does satisfy the hypothesis of Theorem 1, but we then alter the model by designating a set of nodes as persistent, otherwise leaving the model unchanged. We wish to determine how this affects the complexity of the resulting attractor structure. It must have some effect. To take a trivial case, suppose we have n unconnected persistent nodes. Each may be analyzed as an independent TBN, each of which can sustain a fixed point of value 0 or 1. The total number of unique fixed points for the entire network is therefore 2^n. Of course, the complexity of the attractor structure in this case is due entirely to the lack of any exogenous degradation pathways, and not to any connectivity structure of the network (which does not exist in our example).

We next show that this type of reasoning can be extended to TBNs which have the type of acyclicity defined by Theorem 1, but which also have persistent nodes. It is possible to describe mathematically weaker properties of acyclicity within cyclic networks in a way which bounds the complexity of attractor structure. For example, Skodawessely and Klemm (2011) found the maximum number of fixed points in such a network to be $2^{|V|}$ where $V \subseteq N$ is a set of nodes whose removal leaves the network acyclic.

Here, we extend our notion of acyclicity in the following way. We say j is an *ancestor* of i if there is a directed path from j to i. Define the two sets of nodes:

$$\begin{aligned} S_G &= \{\text{ all persistent nodes }\} \\ S_A &= \{\text{ all nonpersistent nodes not possessing} \\ &\qquad \text{a persistent node as an ancestor}\}. \end{aligned} \tag{3}$$

Theorem 2. *Suppose we are given a TBN in which the subnetwork defined by the nodes S_A of (3) satisfies the hypothesis of Theorem 1, or for which $S_A = \emptyset$.*

Next, define the following sequence of subsets of nodes:

$$\begin{aligned} E_1 &= S_G \cup S_A, \\ E_j &= \{\text{ all nodes not in } \cup_{i<j} E_i \text{ with all parents in } \cup_{i<j} E_i \}, \quad j > 1, \end{aligned}$$

and suppose for some J all nodes are included in $\cup_{i \le J} E_i$. Then any two fixed points with identical values for the persistent nodes must be equal, and therefore the maximum number of fixed points is 2^g, where g is the number of persistent nodes.

Proof. Suppose we are given any fixed point. The nodes in S_A (if any) form a TBN satisfying the hypothesis of Theorem 1, so any fixed point must be 0 on these nodes. This implies that the fixed point values of the nodes in E_2 are determined entirely by those of S_G. The argument may be repeated for $E_3, E_4, \ldots$, until the fixed point values of all nodes are determined. □

Theorem 2 complements the result of Skodawessely and Klemm (2011). The conclusion implies a similar upper bound of 2^g for the number of distinct fixed points, where g is the number of persistent nodes. However, while the class of BoNs considered by Theorem 2 is more restricted, removal of the persistent nodes does not necessarily leave the network acyclic, so that the result of Skodawessely and Klemm (2011) does not imply Theorem 2.

The hypothesis of Theorem 2 is satisfied by both TBNs of **Figure 2**. In particular, for **(B)** we have $S_G = \{1, 3\}$, $S_A = \emptyset$, $E_2 = \{4\}$, $E_3 = \{2\}$. However, if a negative regulation from node 2 to node 4 was added, the hypothesis would no longer hold (we would have $E_j = \emptyset$ for all $j \geq 2$) and a counter-example could be constructed.

Next, consider, the cell-cycle network of **Figure 1**. This TBN satisfies the hypothesis of Theorem 2 by setting

$$\begin{aligned} S_G &= \{MBF, Clb5\&6, Cdh1, SBF, Sic1, Clb1\&2\} \\ S_A &= \{Cln3\} \\ E_2 &= \{Mcm1/SFF, Cln1\&2\} \\ E_3 &= \{Cdc20\&14\} \\ E_4 &= \{Swi5\}. \end{aligned}$$

It is interesting to note that the hypothesis of Theorem 2 is satisfied despite the existence of a cycle of green edges between Mcm1/SFF and Clb1&2 (due the the fact that one of these nodes is persistent).

We can see from the application of Theorem 2 to the cell-cycle network that the relationship between the attractor structure and the configuration of persistent nodes is similar to the previous example of the completely unconnected TBN, in the sense that all fixed points are fully determined by their values on the persistent nodes, so that the complexity of the attractor structure must be understood to be driven by a selective lack of exogenous degradation pathways.

3.2.2. Self-degradation assumptions

The assignment of self-degradation (yellow loops) to certain proteins in a network is not a trivial task and cannot be completed *ad-hoc* because self-degradation influences the network's long term behavior. The simplicity of the attractor set associated with the cell-cycle network in **Table 1A** is attributable to the presence of self-degradation and a lack of active network cycles composed entirely of positive regulations (green edges). We exemplify this claim with protein Swi5, the transcription factor for inhibitor protein Sic1. According to Li et al.'s rule of assigning self degradation only to proteins without negative regulators (incoming red edges), Swi5 should not self-degrade since it has the inhibitor Clb1&2. However, their representation of the network allowed Swi5 to have both attributes. Suppose we don't allow Swi5 to self-degrade since it has an inhibitor. How would this change affect the network's steady state behavior? We computed the attractor set for the cell-cycle TBN (Equation (1)) disallowing Swi5 to have the self-degradation property in **Table 2**. Compared to the attractor set with Swi5 self degrading (yellow loop) in **Table 1A**, the attractor set in **Table 2** is bigger with 14 fixed points, half of which has Swi5 on. The attractor set in **Table 1A** is a subset of that in **Table 2**, meaning that the new attractors are due to Swi5 not degrading to 0. The biggest attractor in this new set is $(0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0)$ which differs from the biggest attractor in **Table 1A** only by the presence of Swi5. This exercise has shown that slightly altering the degradation assumption dramatically affected the size and complexity of the cell-cycle's long term behavior.

As noted above, the only cycle constructed with all positive regulations in **Figure 1** is between Clb1&2 and Mcm1/SFF, and this cycle is not sustained (both proteins are at state 0) in the network's long term behavior. To leave the cycle on indefinitely, that is, to keep Clb1&2 and Mcm1/SFF at state 1 perpetually, the sum of the interaction coefficients a_g associated with the positive regulations (green edges) must exceed the sum of a_r associated with the negative regulations (red edges) acting on Clb1&2. Since $-a_r = a_g = 1$, the cycle between Clb1&2 and Mcm1/SFF may get turned on, but does not endure. If this cycle is deleted, the network satisfies the hypothesis of Theorem 1. Because the cycle between Clb1&2 and Mcm1/SFF does not stay on, the network therefore yields a null attractor when all proteins are forced to self-degrade. Thus, following Theorem 2, the variety of fixed points in **Table 1A** is attributable to the 6 proteins with persistence (green self loop) and the cardinality of the attractor set satisfies the upper bound of 2^6. Note that the fixed points in **Table 1A** differ at the proteins with persistence (green self loop), as predicted by Theorem 2. In Section 3.3, we present a network in which the cycle remains active in the steady state.

3.2.3. Markovian self-degradation

Since self-degradation is not built into the Markovian transition functions of the TBN model in Equation (1), specifying incremental degradation is a cumbersome separate process that requires tracking each gene with the self-degradation property and counting the t_d time steps prior to a state change. More importantly, by not explicitly modeling degradation, the model in Equation (1) does not have the typical Boolean network behavior. In particular, a state can be repeated without the network having reached an attractor. For example, suppose we have a two member network in which the only regulations are: protein 1 positively regulates (green edge) protein 2, protein 1 self degrades (yellow loop), and protein 2 persists (green self loop). The interaction coefficient is $a_{21} = 1$. Further, suppose that a protein's lifetime is $t_d = 2$. Using the TBN of Equation (1), a network path is $(1, 1) \rightarrow (1, 1) \rightarrow (0, 1)$. Markovianizing degradation via the following model eliminates this problem by augmenting the state space to express the degradation counter.

$$x_i(t+1) = \begin{cases} 1, & \sum_j a_{ij}\mathrm{I}(x_j(t) > 0) > \alpha_i \\ 0, & \sum_j a_{ij}\mathrm{I}(x_j(t) > 0) < \alpha_i \\ \max(x_i(t) - \epsilon_i, 0), & \sum_j a_{ij}\mathrm{I}(x_j(t) > 0) = \alpha_i \end{cases} \qquad (4)$$

Table 2 | The attractor set for the cell-cycle threshold Boolean network which does not contain Swi5's self-degradation property.

Basin size	Cln3	MBF	Clb5&6	Mcm1/SFF	Swi5	Cdc20&14	Cdh1	Cln1&2	SBF	Sic1	Clb1&2
1383	0	0	0	0	1	0	1	0	0	1	0
380	0	1	0	0	1	0	0	1	1	1	0
139	0	0	0	0	1	0	0	1	1	1	0
108	0	1	0	0	1	0	1	0	0	1	0
10	0	0	0	0	1	0	0	0	0	1	0
8	0	0	0	0	0	0	0	1	1	0	0
6	0	1	0	0	1	0	0	0	0	1	0
5	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	1	0	0	1	1	0	0
1	0	0	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	1	0	0	1	0
1	0	0	0	0	0	0	1	0	0	1	0

The results are based on setting the interaction coefficients $a_g = -a_r = 1$ and protein lifetime $t_d = 1$.

Here $\mathrm{I}(x_j(t) > 0)$ is an expression indicator for protein j; $\epsilon_i \in [0, 1]$ is the degradation rate for protein i; all other parameters are as previously defined in Equations (1) and (2). Whether a protein degrades is determined by the degradation parameter ϵ_i. A protein degrades quickly with a large value of ϵ_i and persists at $\epsilon_i = 0$. The TBN model in Equation (1) with the protein lifetime parameter $t_d = 1$ is equivalent to setting $\epsilon = 1$ for proteins with self-degradation (yellow loop) and $\epsilon = 0$ for proteins with persistence (self green loop). Note that $\epsilon = 1/t_d$. Compared to the TBN model in Equation (1) for which self-degradation must modeled in a side process, Equation (4) explicitly models self-degradation as part of the TBN.

The third line in Equation (4) is meant solely as a device for Markovianizing degradation and persistence. Thus, $x_i(t + 1) \in [0, 1]$, but the regulatory relationships remain Boolean via the indicator $\mathrm{I}(x_j(t) > 0)$. The state space has simply been augmented to allow self-degradation. A further modification that would bring a TBN model closer to a system of differential equations would be to eliminate $\mathrm{I}(x_j(t) > 0)$ and allow node j to take state $x_j \in [0, 1]$ in Equation (4).

So far self-degradation has been treated as a triggered event, i.e., decays occurs after the net influence on the protein is equal to the threshold. The model can be extended to have decay in the presence of a net regulatory effect (Hanel et al., 2012) by letting a protein be its own parent. The sums in Equation (4) would then include node i and line 3 could be omitted with $< \alpha_i$ replaced by $\leq \alpha_i$. These extensions of Equation (4) need to be further studied to understand their properties and appropriateness for modeling a genetic regulatory network.

3.3. SENSITIVITY TO INTERACTION COEFFICIENT

To test the robustness of the cell-cycle TBN to different values of the interaction coefficient a_{ij}, we changed the coefficient of the positive regulations (green edges) to $a_g \in \{2, 3\}$. The attractor sets associated with $a_r = -1$ and $a_g = 2$ and with $a_r = -1$ and $a_g = 3$ are in **Tables 1B,C**. The attractor set for the model with $a_r = -1$ and $a_g = 2$ is a subset, with different basin sizes, of the attractor set for the model with $a_r = -1$ and $a_g = 1$ (**Table 1A**). When $a_r = -1$ and $a_g = 3$, the network cycle between Clb1,2 and Mcm1/SFF is turned on indefinitely in the biggest attractor $(0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1)$ which has a basin size of 1936 states. This is a consequence of positive regulations overcoming negative regulations acting on Clb1,2. With negative interactions fixed at $a_r = -1$, the attractor sets for networks with $a_g > 3$ are either identical or very similar to the set corresponding to $a_g = 3$ (**Table 1C**). For those attractor sets not identical with **Table 1C**, the main difference is the appearance of a two state attractor $\{(0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0), (0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1)\}$. This periodic attractor is very similar to the biggest fixed point in **Table 1C** because all the same proteins get turned on. The unequal attractor sets corresponding to different parameters indicate that the TBN model is not robust to variable interaction coefficients; the cell-cycle network exhibit different behaviors depending on the model specifications. Furthermore, certain parameter values sustain the network cycle between Clb1&2 and Mcm1/SFF and express cellular activities not previously seen.

Next we explored how increasing the degradation delay t_d changed the cell-cycle network's behavior. When we set $-a_r = a_g = 1$ and $t_d > 1$ in the cell-cycle TBN (Equation (1)) the same 7 attractors in **Table 1A** appear. Simulation results show that varying a_r and a_g with t_d yielded attractor sets that are sensitive only to the interaction coefficient.

3.4. ASYNCHRONOUS GENE RESPONSE

The assumption that all genes in a network update simultaneously, *synchronous response*, may be too simplistic. For example, synchronous BoN models may yield attractors driven by the synchrony assumption (Ingerson and Buvel, 1984; Klemm and Bornholdt, 2005). While synchronous response is well-defined, asynchronous response has been defined and modeled in a variety of ways. One model of asynchrony works via an operator external to the BoN that randomly selects a subset of genes to update at each iteration while keeping the unselected genes constant (Ingerson and Buvel, 1984; Greil and Drossel, 2005; Skodawessely and Klemm, 2011). Another model of asynchrony is achieved by allowing different regulatory relationships to have different reaction rates (Thomas and d'Ari, 1990; Silvescu and Honavar, 2001; Shmulevich and Zhang, 2002). Unlike stochastic asynchrony, asynchrony due to varying reaction rates can be incorporated into a deterministic BoN. One type of deterministic asynchronous response can be modeled by allowing genes and proteins to have different self-degradation rates and different interaction coefficients a_{ij}. A protein with a larger lifetime t_d in Equation (1) will take a longer time to reach state 0. Allowing different proteins to have different lifetimes imply different response times. A positive regulator with a higher interaction strength, $|a_{ij}|$, can dominate a negative regulator with a smaller interaction strength and turn on the affected gene. Suppose in a four member network, the relationships $\{2 \rightarrow 1, 3 \rightarrow 1, 4 \rightarrow 1\}$ have the following attributes: $a_{12} = -1, a_{13} = 1, a_{14} = 3$. Compared to gene 3, gene 4 can neutralize the effect of the inhibitor gene 2 and turn on gene 1. In the absence of gene 4, gene 3 would not be able to turn on gene 1 if the inhibitor gene 2 is also on. In this perspective, the magnitude of the interaction, $|a_{ij}|$, can be thought of as a rate. Assigning different interaction coefficients to proteins in a network may be a way to model asynchronous gene update. As we've discussed in Section 3.3, different choices of the coefficient may produce different attractor sets. More work is required to identify which attractors are insensitive to variable a_{ij} and their importance to the cell-cycle.

3.5. SYNERGY AND ANTAGONISM

Thus far the TBN in Equation (1) assumes the regulatory effects are additive. However, some genes act together such that their combined effect is more or less than the sum of the individual effects. *Synergistic* regulation occurs when the joint effect of multiple parents is more than the sum of the individual effects. In contrast, *antagonistic* regulation results in a joint effect that is less than the sum of the individual effects. Such relationships have been studied in cancer cells in which genes exhibit a synergistic response to the combined effort of oncogenic mutations (McMurray et al., 2008). Since synergistic and antagonistic regulations can be critical to the function of a GRN, the interactions should be properly modeled. The TBN model in Equation (1)) can be extended to model these types of regulation by including the statistical interaction terms, $\sum_{j,k} a_{i(jk)} x_j(t) x_k(t)$, where the interaction coefficient $a_{i(jk)}$ between parents j and k and child i are defined analogously to a_{ij}. Synergy is represented by a positive $a_{i(jk)}$ and antagonism by a negative $a_{i(jk)}$. Interactions of order greater than two are similarly constructed.

4. DISCUSSION

A TBN specified by linear functions and a threshold instead of truth tables is more quantitative at describing genetic regulatory network (GRN) dynamics. We illustrate how this framework can accommodate aspects of GRNs such as variable Markovian self-degradation, asynchronous gene update, and synergistic relationships. Furthermore, we found that the complexity of a GRN can be summarized by the presence of self-degradation and cycles comprised of only positive regulations. Although the model is more analytical compared to networks specified by truth tables, it still retains the qualitative interpretation of a BoN.

Inspection of the TBN model in Equation (1) to model the budding yeast cell-cycle showed that the attractor set relied on the assumptions of self-degradation and choice of interaction coefficient a_{ij}. Changing these two aspects of the model changed the steady state behavior of the cell-cycle. Our extension of the TBN model using a threshold parameter as in Equation (2) permits greater flexibility in describing regulatory relationships. Another modification we suggested was Markovianizing degradation to facilitate incremental or delayed degradation. We also proposed varying the protein lifetime t_d and interaction coefficient among proteins to simulate asynchronous gene update and adding statistical interaction terms to account for synergistic effects.

Our theorems claimed that the composition of a TBN's attractor set depends on the presence and abundance of self-degradation (yellow loops), persistence (green self loops), and network cycles. Theorem 1 states that the null attractor is the only attractor for a network acyclic in the positive regulations (green edges) and in which all nodes self degrade. This result holds under varying interaction strength and degradation rates. Although the theorem was proved for TBNs, it applies to other Boolean network models that are not of the form in Equation (1) because the proof relies only on topological ordering in the positive regulations and self-degradation on all genes. Theorem 2 states that under a weaker definition of acyclicity, the complexity of the attractor structure is entirely determined by the configuration of persistent genes.

Future work includes characterizing the attractor set, e.g., determine an upper bound on its cardinality, for (a) the class of TBNs containing network cycles of positive regulations (green edges), and (b) the class of TBNs containing both persistence and network cycles of positive regulations in the presence of self-degradation and asynchronicity.

AUTHOR CONTRIBUTIONS

Van Tran performed the majority of the analyses and primarily wrote the manuscript; Matthew N. McCall and Anthony Almudevar performed some analyses, wrote portions of the manuscript, and helped conceive the project; Helene R. McMurray provided biological expertise and helped conceive the project. All authors edited and approved the manuscript.

ACKNOWLEDGMENTS

This work was funded in part by the National Institutes of Health (CA009363, ES007271, HG006853) and an Edelman-Gardner Foundation Award.

REFERENCES

Albert, R., and Othmer, H. G. (2003). The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in *Drosophila melanogaster*. *J. Theor. Biol.* 223, 1–18. doi: 10.1016/S0022-5193(03)00035-3

Davidich, M., and Bornholdt, S. (2008). Boolean network model predicts cell-cycle sequence of fission yeast. *PLoS ONE* 3:e1672. doi: 10.1371/journal.pone.0001672

De Jong, H. (2002). Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.* 9, 67–103. doi: 10.1089/10665270252833208

Espinosa-Soto, C., Padilla-Longoria, P., and Alvarez-Buylla, E. R. (2004). A gene regulatory network model for cell-fate determination during *Arabidopsis thaliana* flower development that is robust and recovers experimental gene expression profiles. *Plant Cell Online* 16, 2923–2939. doi: 10.1105/tpc.104.021725

Glass, L., and Kauffman, S. A. (1972). Co-operative components, spatial localization and oscillatory cellular dynamics. *J. Theor. Biol.* 34, 219–237. doi: 10.1016/0022-5193(72)90157-9

Glass, L., and Kauffman, S. A. (1973). The logical analysis of continuous, non-linear biochemical control networks. *J. Theor. Biol.* 39, 103–129. doi: 10.1016/0022-5193(73)90208-7

Greil, F., and Drossel, B. (2005). The dynamics of critical kauffman networks under asynchronous stochastic update. *Phys. Rev. Lett.* 95:048701. doi: 10.1103/PhysRevLett.95.048701

Hanel, R., Pöchacker, M., Schölling, M., and Thurner, S. (2012). A self-organized model for cell-differentiation based on variations of molecular decay rates. *PLoS ONE* 7:e36679. doi: 10.1371/journal.pone.0036679

Heinrich, R., and Schuster, S. (1996). *The Regulation of Cellular Systems*, vol. 416. New York, NY: Chapman & Hall. doi: 10.1007/978-1-4613-1161-4

Huang, S., Eichler, G., Bar-Yam, Y., and Ingber, D. E. (2005). Cell fates as high-dimensional attractor states of a complex gene regulatory network. *Phys. Rev. Lett.* 94, 128701. doi: 10.1103/PhysRevLett.94.128701

Ingerson, T. E., and Buvel, R. L. (1984). Structure in asynchronous cellular automata. *Phys. D Nonlin. Phenom.* 10, 59–68. doi: 10.1016/0167-2789(84)90249-5

Kauffman, S. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.* 22, 437 – 467. doi: 10.1016/0022-5193(69)90015-0

Klemm, K., and Bornholdt, S. (2005). Stable and unstable attractors in boolean networks. *Phys. Rev. E* 72, 055101. doi: 10.1103/PhysRevE.72.055101

Li, F., Long, T., Lu, Y., Ouyang, Q., and Tang, C. (2004). The yeast cell-cycle network is robustly designed. *Proc. Natl. Acad. Sci. U.S.A.* 101, 4781–4786. doi: 10.1073/pnas.0305937101

McCall, M. N., Uppal, K., Jaffee, H. A., Zilliox, M. J., and Irizarry, R. A. (2011). The gene expression barcode: leveraging public data repositories to begin cataloging the human and murine transcriptomes. *Nucl. Acids Res.* 39(Suppl. 1), D1011–D1015. doi: 10.1093/nar/gkq1259

McMurray, H. R., Sampson, E. R., Compitello, G., Kinsey, C., Newman, L., Smith, B. et al. (2008). Synergistic response to oncogenic mutations defines gene class critical to cancer phenotype. *Nature* 453, 1112–1116. doi: 10.1038/nature06973

Parmigiani, G., Garrett, E. S., Anbazhagan, R., and Gabrielson, E. (2002). A statistical framework for expression-based molecular classification in cancer. *J. R. Stat. Soc. B* 64, 717–736. doi: 10.1111/1467-9868.00358

Remy, É., Ruet, P., and Thieffry, D. (2008). Graphic requirements for multistability and attractive cycles in a boolean dynamical framework. *Adv. Appl. Math.* 41, 335–350. doi: 10.1016/j.aam.2007.11.003

Richard, A. (2010). Negative circuits and sustained oscillations in asynchronous automata networks. *Adv. Appl. Math.* 44, 378–392. doi: 10.1016/j.aam.2009.11.011

Richard, A. (2013). Fixed point theorems for boolean networks expressed in terms of forbidden subnetworks. *arXiv preprint arXiv*:1302.6346.

Robert, F. (1980). Iterations sur des ensembles finis et automates cellulaires contractants. *Linear Algebra Appl.* 29, 393–412. doi: 10.1016/0024-3795(80) 90251-7

Scharpf, R., Garrett, E. S., Hu, J., and Parmigiani, G. (2003). Statistical modeling and visualization of molecular profiles in cancer. *Biotechniques* 34, S22–S29.

Shih, M.-H., and Dong, J.-L. (2005). A combinatorial analogue of the jacobian problem in automata networks. *Adv. Appl. Math.* 34, 30–46. doi: 10.1016/j.aam.2004.06.002

Shmulevich, I., and Zhang, W. (2002). Binary analysis and optimization-based normalization of gene expression data. *Bioinformatics* 18, 555–565. doi: 10.1093/bioinformatics/18.4.555

Silvescu, A., and Honavar, V. (2001). Temporal boolean network models of genetic networks and their inference from gene expression time series. *Comp. Syst.* 13, 61–78.

Skodawessely, T., and Klemm, K. (2011). Finding attractors in asynchronous boolean dynamics. *Adv. Comp. Syst.* 14, 439–449. doi: 10.1142/S0219525911003098

Thomas, R. (1981). "On the relation between the logical structure of systems and their ability to generate multiple steady states or sustained oscillations," in *Numerical Methods in the Study of Critical Phenomena* (Springer), 180–193.

Thomas, R., and D'Ari, R. (1990). *Biological Feedback*. Boca Raton, FL: CRC press.

Zilliox, M. J., and Irizarry, R. A. (2007). A gene expression bar code for microarray data. *Nat. Methods* 4, 911–913. doi: 10.1038/nmeth1102

Conflict of Interest Statement: 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.

Received: 15 September 2013; paper pending published: 17 October 2013; accepted: 15 November 2013; published online: 11 December 2013.

Citation: Tran V, McCall MN, McMurray HR and Almudevar A (2013) On the underlying assumptions of threshold Boolean networks as a model for genetic regulatory network behavior. Front. Genet. ***4****:263. doi: 10.3389/fgene.2013.00263*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 12 December 2013
doi: 10.3389/fgene.2013.00272

Validation of gene regulatory network inference based on controllability

Xiaoning Qian[1]* *and Edward R. Dougherty*[1,2]*

[1] *Department of Electrical and Computer Engineering, Texas A & M University, TX, USA*
[2] *Computational Biology Division, Translational Genomics Research Institute, Phoenix, AZ, USA*

Edited by:
Frank Emmert-Streib, Queen's University Belfast, UK

Reviewed by:
Boris A. Reva, Memorial Sloan-Kettering Cancer Center, USA
Yang Dai, University of Illinois at Chicago, USA

****Correspondence:***
Xiaoning Qian and Edward R. Dougherty, Department of Electrical and Computer Engineering, Texas A& M University, College Station, MS 3128, TAMU, TX 77843, USA
e-mail: xqian@ece.tamu.edu; edward@ece.tamu.edu

There are two distinct issues regarding network validation: (1) Does an inferred network provide good predictions relative to experimental data? (2) Does a network inference algorithm applied within a certain network model framework yield networks that are accurate relative to some criterion of goodness? The first issue concerns scientific validation and the second concerns algorithm validation. In this paper we consider inferential validation relative to controllability; that is, if an inference procedure is applied to data generated from a gene regulatory network and an intervention procedure is designed on the inferred network, how well does it perform on the true network? The reasoning behind such a criterion is that, if our purpose is to use gene regulatory networks to design therapeutic intervention strategies, then we are not concerned with network fidelity, *per se*, but only with our ability to design effective interventions based on the inferred network. We will consider the problem from the perspectives of stationary control, which involves designing a control policy to be applied over time based on the current state of the network, with the decision procedure itself being time independent. The objective of a control policy is to optimally reduce the total steady-state probability mass of the undesirable states (phenotypes), which is equivalent to optimally increasing the total steady-state mass of the desirable states. Based on this criterion we compare several proposed network inference procedures. We will see that inference procedure ψ may perform poorer than inference procedure ξ relative to inferring the full network structure but perform better than ξ relative to controllability. Hence, when one is aiming at a specific application, it may be wise to use an objective-based measure of inference validity.

Keywords: network inference, genetic regulatory network, control, validation, probabilistic Boolean network

1. INTRODUCTION

Network validity can be approached from two perspectives: *scientific* and *inferential*. Scientific validity is an epistemological issue concerning the ability of a network model to yield observations concordant with those predicted by the model (Dougherty and Bittner, 2011). It involves relations between model characteristics and experimental observations such that mathematical predictions based on the model are manifested in the phenomena via these relations. Inferential validity concerns the ability of an inference procedure to operate on data generated from the model and yield an inferred model close to the original network relative to some distance function. Inferential validity is purely a mathematical issue concerning the inference algorithm. The two issues, scientific and inferential validity, are not unrelated because in practice an inferential procedure is used to construct a model from real data and the scientific validity is therefore dependent upon the performance of the inferential procedure. In this paper we are interested in inferential validity [see Dougherty (2011) for a discussion of the two types of validity].

The validity of inference procedures for gene regulatory networks is discussed in Dougherty (2007), where validation is relative to some network characteristic and quantified by some distance between the characteristic for the original network and the characteristic for the inferred network, such as a norm between the steady-state distributions of the original and inferred networks. Generally speaking (we shall be more rigorous shortly), (1) a characteristic is derived for the network; (2) a data sample is generated from the network; (3) an inference procedure operates on the sample to produce an inferred network; (4) the corresponding characteristic is derived for the inferred network; (5) the corresponding characteristics for the original and inferred networks are compared; and (6) the validity of the inference procedure is determined by some distance between the characteristics.

The preceding validation protocol focuses solely on the network itself, not any objective to which the network is to be used, although clearly successful use of the inferred network will depend to some extent on the closeness of the inferred and original networks. Our aim here is to characterize the notion of *objective inferential validity*, where inferential validity is measured relative to the objective for which the network will be used. In particular, we are concerned with controllability. Specifically, if the objective is to derive a control procedure from the inferred network, then it is of utmost importance that the control procedure works well on the original network (from which the sample data have been generated). In other words, to what extent is

Network Inference

- Q: What types of biological networks have been inferred in the paper?
- A: We focus on the inferential validity of genetic regulatory network inference. We evaluate and compare different inference algorithms in the framework of probabilistic Boolean networks (PBNs) by both synthetic random PBNs and a melanoma metastatic network inferred from gene expression data.
- Q: How was the quality/utility of the inferred networks assessed?
- A: We propose and discuss different inferential validity criteria for inferring genetic regulatory networks, including (1) Hamming distance to measure the network topology closeness; (2) steady-state mass difference for network dynamic behavior similarity; and (3) expected difference of desirable steady-state mass shift by applying derived optimal control when the operational objective is intervention. We would like to emphasize that objective inferential validity criteria based on operational objectives such as intervention are viable choices when we typically do not have the ground truth of real-world gene regulatory networks.
- Q: How were these networks validated?
- A: Both synthetic random networks and the melanoma metastatic network are considered as benchmark networks. From these network models, we simulate the network dynamics with perturbations, inferred networks by different algorithms are evaluated by ground truth network models based on the aforementioned three inferential validity criteria.

controllability preserved by the inference procedure? It may be that the original and inferred networks are a quire discordant; however, if their lack of agreement has little impact on derivation of the control procedure, then this lack of agreement is of little consequence.

Two basic intervention approaches have been considered for gene regulatory networks in the framework of probabilistic Boolean networks (PBNs) (Dougherty and Datta, 2005; Datta and Dougherty, 2007; Shmulevich and Dougherty, 2007), structural intervention and external control. Both take advantage of the fact that the probabilistic characteristics of a PBN are characterized by an associated Markov chain. *Structural intervention* involves a one-time change of the network structure (wiring) to beneficially alter the long-run behavior (steady state) of the network (Shmulevich et al., 2002b; Xiao and Dougherty, 2007; Qian and Dougherty, 2008). Given a class of potential structural changes, the problem is to find the optimal structural intervention resulting in a desired alteration of the steady-state distribution. *Stationary control* is generally based on flipping (or not flipping) the value of a control gene(s) over time in an effort to favorably move the steady-state mass. Efforts have mainly focused on infinite-horizon stationary external control. The first proposed approach utilizes dynamic programming to find an optimal policy relative to a cost function, in which case the steady-state distribution is altered as a by-product of this optimization (Pal et al., 2006). A second approach is to utilize a greedy (no optimality) algorithm to find a policy that directly aims at altering the steady-state distribution Qian et al. (2009). Here we will use a more recently proposed approach for gene regulatory networks that uses linear programming to find a policy that is optimal relative to minimizing undesirable steady-state mass (Yousefi and Dougherty, 2013). This latter approach avoids the introduction of a subjectively defined cost function as in Pal et al. (2006) and avoids the sub-optimality of greedy algorithms (Qian et al., 2009). Instead, the amount of shift in the steady-state distribution gives an intrinsic network measure, as it also does in the case of structural intervention. The situation is analogous to classification, where the Bayes error is intrinsic to the feature-label distribution, as opposed to errors resulting from suboptimal classifiers that have been derived from data via some *ad hoc* classification rule. In this paper we restrict our attention to stationary control because it is very possible that the optimal structural controller for an inferred network is based on an inferred function that may not exist in the original network. In such a case it would not be feasible to apply the identified intervention for the inferred network back to the original network.

Figure 1 illustrates the main idea of objective inferential validity for quantifying the performance of different network inference procedures with respect to controllability. Assuming that we are interested in an impaired biological system that has a higher risk of entering into aberrant phenotypes, from the collected measurements, our goal is to design effective stationary control policies to reduce the risk of entering into these undesirable or bad states. One way to characterize network states is based on the prior knowledge of biomarkers. As a hypothetic example, x_1 in **Figure 1** is considered as the marker gene, whose value being 1 (up-regulated) are not desirable as it may represent metastasizing phenotypes in cancerous systems, for example. Based on what we can observe, from microarray profiling or other high-throughput techniques, we may infer the underlying network model that governs the state dynamics. Many previous inferential validity measures are solely interested in the network itself. However, in this scenario, inference procedures should be evaluated in regard to our final objective of effectively reducing the undesirable risk by evaluating the control performance of intervention strategies derived using the network model inferred from partially observed data. In fact, in real-world scenario, we typically do not have the ground truth of the underlying system. Objective inferential validity may be the only reasonable framework for network inference validation.

2. SYSTEMS AND METHODS

2.1. PROBABILISTIC BOOLEAN NETWORKS

Probabilistic Boolean networks (Shmulevich et al., 2002a) extend the classical Boolean networks (Kauffman, 1969, 1993) by introducing uncertainty in the rule structure [see Shmulevich and

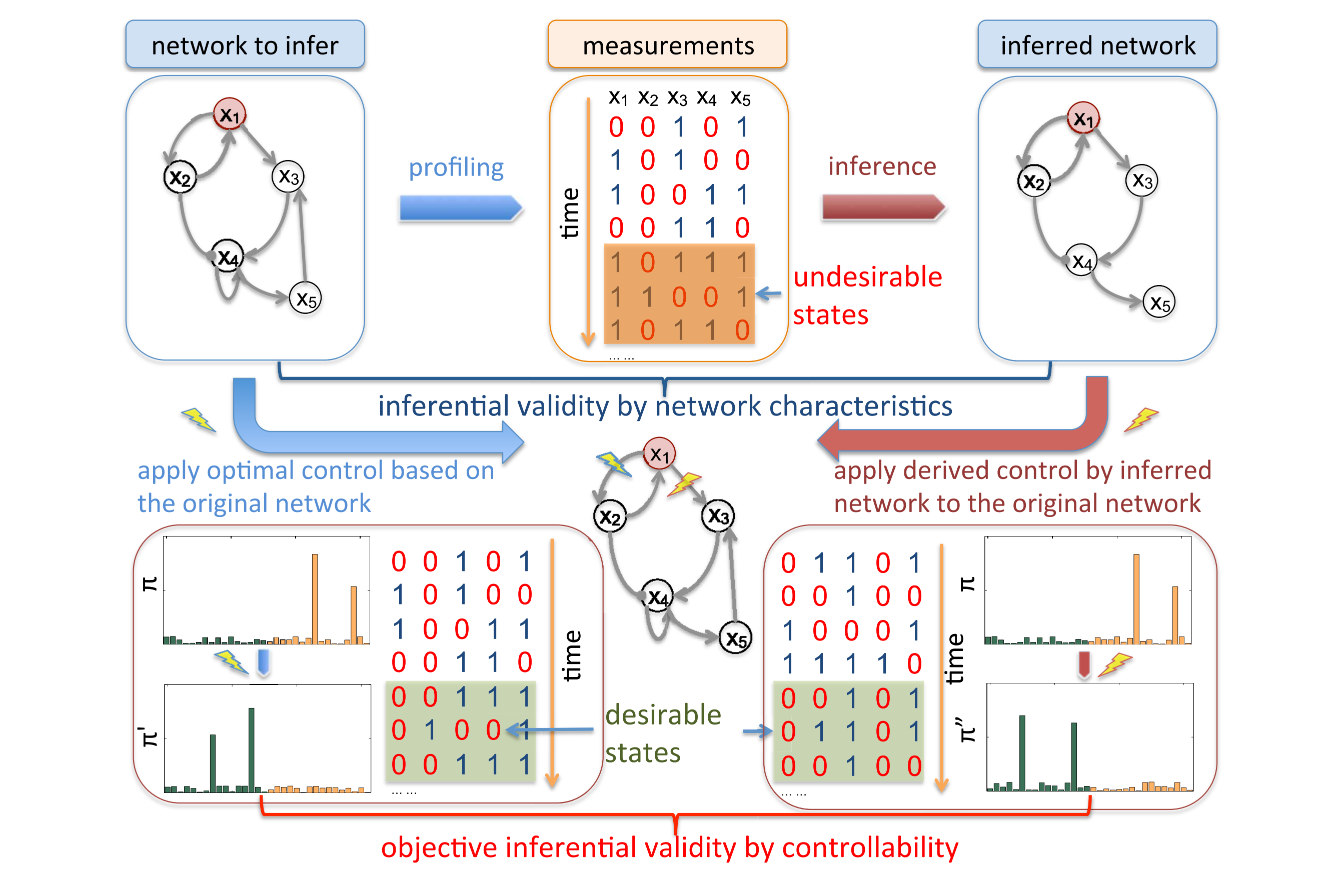


FIGURE 1 | Schematic illustration of inferential validity. There are different criteria to evaluate inferred networks from available temporal measurements. For example, we can directly measure the difference of inferred regulatory relationships among genes by the commonly adopted Hamming distance between the original network adjacency matrix and the inferred adjacency matrix. We are interested in objective-based inferential validity based on controllability. For example, assuming that x_1 is a genetic marker marked in red, the network is considered in "undesirable" states when it is up-regulated ($x_1 = 1$). Hence, from the translational perspective, the ultimate goal of studying this network system is to develop effective therapeutic strategies based on collected data from the system. Hence when evaluating network inference algorithms, instead of comparing other network characteristics, it may be more appropriate to directly investigate how the derived intervention strategies based on inferred networks perform on the original networks by reducing the long-run probability of entering into undesirable states, which leads to our controllability-based inferential validity. As shown in the figure, assume that we derive the optimal control based on the original network to block the regulation from x_1 to x_2 while the derived control from the inferred network is to block the regulation from x_1 to x_3. Note that both of the derived control policies have to be validated on the true network. One criterion to evaluate the inferred network as our "objective-based inferential validity" is to check how the steady-state distribution π'' by blocking $x_1 \rightarrow x_3$ on the original network compares to the optimally controlled steady-state distribution π' after blocking $x_1 \rightarrow x_2$ with respect to the reduction of undesirable steady-state mass in the original steady-state distribution π before intervention. This difference reflects the cost of using the derived control from the inferred network instead of the optimal control designed from the true network.

Dougherty (2010) for a comprehensive review]. This uncertainty is motivated by randomness in the inference procedure, inherent biological randomness, and model stochasticity owing to latent variables outside the model that are involved in regulation.

A binary Boolean network $G(V, F)$ is defined by a set $V = \{x_1, x_2 \ldots, x_n\}$ of binary variables, $x_i \in \{0, 1\}$, $i = 1, \ldots, n$, and a list of Boolean functions $F = \left(f_1, f_2, \ldots, f_n\right)$. The value of x_i at time $t + 1$ is completely determined by a subset $\{x_{i1}, x_{i2}, \cdots, x_{ik_i}\} \subset V$ at time t via a Boolean function $f_i : \{0, 1\}^{k_i} \mapsto \{0, 1\}$. Transitions are homogeneous in time and we have the update $x_i(t + 1) = f_i(x_{i1}(t), x_{i2}(t), \cdots, x_{ik_i}(t))$. Each x_i represents the state (expression) of gene i, where $x_i = 1$ and $x_i = 0$ represent gene i being expressed and not expressed, respectively. It is commonplace to refer to x_i as the ith gene. The list F of Boolean functions represents the rules of regulatory interactions between genes. All genes are assumed to update synchronously in accordance with the functions assigned to them and this process is then repeated. At any time t, the state of the network is defined by a state vector $\mathbf{x}(t) = (x_1(t), x_2(t), \ldots, x_n(t))$, called a *gene activity profile* (GAP). Given an initial state, a BN will eventually reach a set of states, called an *attractor cycle*, through which it will cycle endlessly. Each initial state corresponds to a unique attractor cycle and the set of states leading to a specific attractor cycle is known as the *basin of attraction* (BOA) of the attractor cycle.

A *Boolean network with perturbation* (BNp) is defined by allowing each gene to possess the possibility of randomly flipping its value with a positive probability p. Implicitly, we assume that there is an i.i.d. random perturbation vector $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_n)$, where $\gamma_i \in \{0, 1\}$, the ith gene flips if and only if $\gamma_i = 1$, and $p = P(\gamma_i = 1)$ for $i = 1, 2, \ldots, n$. If $\mathbf{x}(t)$ is the

GAP at time t, then the next state $\mathbf{x}(t+1)$ is either $\mathbf{f}(\mathbf{x}(t))$ with probability $(1-p)^n$ or $\mathbf{x}(t) \oplus \boldsymbol{\gamma}$ with probability $1-(1-p)^n$, where $\mathbf{f}$ is the multi-output function from the truth table and $\oplus$ is component-wise addition modulo 2. Larger values of p result in the regulatory rules being overridden by random alterations to the regulatory signaling, which one might call "noise."

A binary *probabilistic Boolean network* (PBN) is composed of a family $\{B_1, B_2, \ldots, B_m\}$ of BNps together with probabilities governing the selection of a BNp at each time. The m constituent BNps are characterized by m network functions, $\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m\}$. At any time point there is a positive probability q of switching from the current governing constituent BNp (context) to another, with the selection probabilities for transitioning to $B_1, B_2, \ldots, B_m$ given by $c_1, c_2, \ldots, c_m$, respectively. Note that the probability of switching to any constituent network B_ℓ, $1 \leq \ell \leq m$ is independent of the current network; indeed, when a switch is called for, the current network may "switch" to itself. By definition, a PBN inherits the attractor cycles of its constituent BNps. There are two modeling interpretations regarding q. If $q < 1$, the PBN is said to be *context-sensitive* (Brun et al., 2005); if $q = 1$, as in the original formulation of PBNs (Shmulevich et al., 2002a), then the PBN is said to be *instantaneously random*. The modeling interpretation is that there are latent variables outside the network model controlling the context of the network and larger values of q correspond to greater effects of latent variables. Although we have defined PBNs as having binary gene values, there is nothing inherent in this restriction and the general definition assumes that each gene can take a finite number of values, say in the set $\{0, 1, \ldots, d\}$.

Transition rules of any PBN can be modeled by a homogeneous Markov chain, whose states of the transition probability matrix (TPM) P are the GAPs of the PBN [see Faryabi et al. (2009) for the particulars on how the Markov chain is derived for different classes of PBNs]. Perturbation makes the corresponding Markov chain of a PBN irreducible and ergodic. Hence, the network possesses a steady-state distribution $\pi^T = \pi^T P$, describing its long-run behavior. For small q and p, most of the steady-state mass lies in the attractors of the PBN (Brun et al., 2005), which by definition are the attractors of the constituent BNs. Let $\mathcal{S} = \{(\mathbf{x}, y) : \mathbf{x} \in \mathcal{B}, y \in \{1, 2, \ldots, m\}\}$ be the state space of the PBN, where $\mathcal{B}$ denotes the space of all GAPs or network states for any constituent BN with n genes and y is the index to which constituent BN currently governs the dynamics. We note that when we have BNps with only one constituent BN, y is redundant. Let $\{\mathbf{Z}_k \in \mathcal{S}, k = 0, 1, \ldots\}$ be the stochastic process of the state of the PBN that has both the information about the current constituent BN and GAP of the underlying network. Originating from state $\mathbf{i} \in \mathcal{S}$, the successor state $\mathbf{j} \in \mathcal{S}$ is selected randomly according to the TPM P, with its $\mathbf{ij}$th element defined by $p_{\mathbf{ij}} \triangleq P(\mathbf{Z}_{k+1} = \mathbf{j} \mid \mathbf{Z}_k = \mathbf{i})$ for all $k = 0, 1, \ldots$.

2.2. MAXIMAL STEADY-STATE ALTERATION

We now briefly outline the setting in which an infinite-horizon policy can be found that achieves maximal steady-state alteration, meaning that it optimizes the shift of steady-state mass from undesirable to desirable states. Let $\mathcal{D}$ and $\mathcal{U}$ denote the sets of desirable and undesirable states, respectively. One way to define $\mathcal{D}$ and $\mathcal{U}$ is based on the values of given genetic markers as illustrated in **Figure 1**. For instance, undesirable states may be those in which gene WNT5A is up-regulated because such states are associated with increased risk of metastasis in melanoma, whereas the desirable states would be those in which WNT5A is down-regulated (see Section 4.3). We assume that the PBN admits an external control input A from a set of actions, $\mathcal{A}$, specifying the type of intervention on a set of control genes. For instance, $A = 0$ may indicate no-intervention and $A = 1$ may indicate that the expression level of a single gene, g^c, $c \in \{1, 2, \ldots, n\}$, is flipped. In this intervention scenario, the control action $A = 1$ at state $(\mathbf{x}, y)$ replaces the row corresponding to the state $(\mathbf{x}, y)$ in the original TPM of the underlying Markov chain by the row corresponding to the state $(\tilde{\mathbf{x}}, y)$, where the binary representation of $\tilde{\mathbf{x}}$ is the same as $\mathbf{x}$ except in bit v^c, where it is flipped.

Denote by $\{\mathbf{z}_k, k = 0, 1, \ldots\}$ and $\{a_k, k = 0, 1, \ldots\}$ the sequences of observed states and actions. A *policy* is a prescription for taking actions at each time point k. Actions may be taken in accordance with a random mechanism, possibly a function of the entire history of the system up to time k. For time k, let $h_k = (\mathbf{z}_0, a_0, \mathbf{z}_1, a_1, \ldots, \mathbf{z}_k, a_k)$ denote the observed history. A policy $\upsilon = (\upsilon_0, \upsilon_1, \ldots)$ is a sequence prescribed by the decision maker that steers the dynamics of the underlying system. If the history h_{k-1} is observed up to time k, then the decision maker chooses an action $a \in \mathcal{A}(\mathbf{z}_k)$ with probability $\upsilon_k(a \mid h_{k-1}, \mathbf{z}_k)$.

The goal is to find an intervention policy to maximally shift the long-run probability mass of undesirable states to desirable ones. Let $\mathcal{A} = \mathcal{A}(\mathbf{j}) = \{0, 1\}$ for all $\mathbf{j} \in \mathcal{S}$. The amount of shift in the aggregated probability of undesirable states for a PBN controlled under υ is defined as

$$\Delta\pi_{\mathcal{U}}(\upsilon) = \sum_{\mathbf{j} \in \mathcal{U}} \pi_{\mathbf{j}} - \sum_{\mathbf{j} \in \mathcal{U}} \pi_{\mathbf{j}}(\upsilon), \tag{1}$$

where π and $\pi(\upsilon)$ are the steady-state vectors for the Markov chains governed by the original and controlled PBNs, respectively. The goal is to maximize $\Delta\pi_{\mathcal{U}}(\upsilon)$. An optimal policy that is both stationary (time-invariant) and deterministic can be obtained by solving a linear programming problem, which we refer to as the Maximal Steady-State Alteration (MSSA) algorithm (Yousefi and Dougherty, 2013). The optimal policy depends on the choice of undesirable states and the control input. In our case, these will be determined by the values of certain genes, which can be considered as *a priori* known biomarkers for example. Since we are interested in quantifying the performance of inference procedures on the network, these marker genes will be selected randomly for random networks without loss of generality.

2.3. INFERENTIAL VALIDATION

Network comparison is based on a distance function, μ, which need only be a semi-metric because we do not want to require that $\mu(\mathcal{M}, \mathcal{H}) = 0$ implies $\mathcal{M} = \mathcal{H}$, the point being that we compare networks via characteristics and two distinct networks might possess the same characteristic yet be quite different. For instance, consider the steady-state distribution. If $\pi = (\pi_1, \pi_2, \ldots, \pi_m)$ and $\omega = (\omega_1, \omega_2, \ldots, \omega_m)$ are the steady-state distributions for networks $\mathcal{H}$ and $\mathcal{M}$, respectively, then a network distance is defined by $\mu_{ss}(\mathcal{M}, \mathcal{H}) = \|\pi - \omega\|$, where $\|\bullet\|$ is some vector norm. As a second example, suppose one is interested in network topology. Define the adjacency matrix in the following manner: given an n-gene network, for $i, j = 1, 2, \ldots, n$, the (i, j) entry in the matrix is 1 if there is a directed edge from the ith to the jth gene; otherwise, the (i, j) entry is 0. If $\mathbf{A} = (a_{ij})$ and $\mathbf{B} = (b_{ij})$ are the adjacency matrices for networks $\mathcal{H}$ and $\mathcal{M}$, respectively, where $\mathcal{H}$ and $\mathcal{M}$ possess the same gene set, then the Hamming distance between the networks is defined by $\mu_{\text{ham}}(\mathcal{M}, \mathcal{H}) = \sum_{i,j=1}^{n} |a_{ij} - b_{ij}|$. Both μ_{ss} and μ_{ham} are semi-metrics.

Focusing on full network inference (and following Dougherty, 2007), the goodness of an inference procedure ψ relative to distance μ is measured by $\mu(\psi(S), \mathcal{H})$, where $\mathcal{H}$ is the original network and sample S is a realization of the random process, Σ, governing data generation from $\mathcal{H}$. Hence, $\mu(\psi(\Sigma), \mathcal{H})$ is a random variable and the performance of ψ is characterized by the distribution of $\mu(\psi(\Sigma), \mathcal{H})$, which depends on the distribution of Σ. We adopt the expectation of the distribution of $\mu(\psi(\Sigma), \mathcal{H})$ as the measure for inferential validity, $E_\Sigma\,[\mu(\psi(\Sigma), \mathcal{H})]$ taken with respect to Σ.

Rather than considering a single network, we can consider a distribution, H, of random networks, where the occurrences of realizations $\mathcal{H}$ of H are governed by a probability distribution. Averaging over the class of random networks, our interest focuses on $E_{\text{H}}\,[E_\Sigma\,[\mu(\psi(\Sigma), \mathcal{H})]]$. Inference procedure ψ_1 is better than the inference procedure ψ_2 relative to the distance μ, the random network H, and the sampling procedure Σ if $E_{\text{H}}\,[E_\Sigma\,[\mu(\psi_1(\Sigma), \mathcal{H})]] < E_{\text{H}}\,[E_\Sigma\,[\mu(\psi_2(\Sigma), \mathcal{H})]]$. In practice, the expectation must be estimated by an average $\frac{1}{m}\sum_{j=1}^{m} \mu(\psi(S_j), \mathcal{H}_j)$, where $S_1, S_2, \ldots, S_m$ are sample point sets generated according to Σ from networks $\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_m$ randomly chosen from H.

The preceding analysis applies unchanged when measuring validity relative to controllability; indeed, it is just a matter of defining the distance function. Let $\mathcal{H}$ denote the original network, S be a sample generated from $\mathcal{H}$, $\upsilon_{\mathcal{H}}$ and $\upsilon_{\psi(S)}$ be the maximal steady-state alteration policies for $\mathcal{H}$ and $\psi(S)$, respectively, and $\pi^{\mathcal{H}}$ and $\pi^{\psi(S)}$ be the steady-state vectors for $\mathcal{H}$ controlled by $\upsilon_{\mathcal{H}}$ and $\upsilon_{\psi(S)}$, respectively. Then the inferential-validity distance relative to controllability is defined by

$$\mu_{\text{ctrl}}(\psi(S), \mathcal{H}) = \sum_{\mathbf{i} \in \mathcal{U}} \pi_{\mathbf{i}}^{\psi(S)} - \sum_{\mathbf{i} \in \mathcal{U}} \pi_{\mathbf{i}}^{\mathcal{H}}, \tag{2}$$

where $\mathcal{U}$ is the class of undesirable states. Applying this distance to a distribution $\mathbf{H}$, of random networks yields the expectation in which we are interested, namely,

$$E_{\text{H}}\,[E_\Sigma\,[\mu_{\text{ctrl}}(\psi(\Sigma), \mathcal{H})]] = E_{\text{H}}\left[E_\Sigma\left[\sum_{\mathbf{i} \in \mathcal{U}} \pi_{\mathbf{i}}^{\psi(S)} - \sum_{\mathbf{i} \in \mathcal{U}} \pi_{\mathbf{i}}^{\mathcal{H}}\right]\right]. \tag{3}$$

For analyzing PBNs, we are confronted by computational issues in regard to transition probability matrices of their underlying Markov chains. In the case of controlling binary discrete-time networks, one is looking at a matrix of dimension $N \times N$, where N is the number of states. For a PBN, $N = m \times 2^n$, where m is the number of contexts and n is the number of genes. Generally speaking, networks beyond 15 genes become computationally intractable with regard to deriving control policies. Larger networks require first the application of a reduction algorithm to reduce the size of the state space (Qian and Dougherty, 2009b; Ivanov et al., 2010; Qian et al., 2010). These inevitably lose information. If one is going to study inference for networks larger than 15 genes, then the analysis must include the reduction algorithm as part of the design. This can certainly be done but it would not essentially change the kind of inference analysis in which we are involved. The price would be that, whereas by using the MSSA algorithm the entire matter is intrinsic, there being no subjective cost functions, prior use of a reduction algorithm would destroy the intrinsic nature of the analysis.

2.4. NETWORK INFERENCE ALGORITMS

Learning regulatory relationships among genes is a major challenge in computational biology. Numerous methods based on different mathematical models have been developed; however, performance evaluation remains problematic (Marbach et al., 2010). In this paper, we focus on network inference algorithms for PBNs from one or several time series of observed gene expression states $\mathbf{x}(t)$. We have implemented a few commonly adopted inference algorithms for PBNs with modifications to allow for more than one time series: REVEAL (REVerse Engineering ALgorithm) and its extension (Liang et al., 1998; Akutsu et al., 1999; Murphy and Mian, 1999; Martin et al., 2007), MDL (Minimal Description Length) (Tabus and Astola, 2001; Zhao et al., 2006; Dougherty et al., 2008), and Best-Fit (Lähdesmäki et al., 2003; Marshall et al., 2007; Lähdesmäki and Shmulevich, 2012).

These inference algorithms aim for identifying regulatory relationships among genes as well as finding corresponding Boolean functions for them so that the observed state transitions in time series data are most "consistently" explained by the inferred functions. For example, REVEAL (Liang et al., 1998) identifies predictors for each gene by estimating the mutual information between the temporal profile of each gene and all the combination profiles of potential genes as regulators, starting from one regulator per gene. In order to find a unique solution, in the worst case, the algorithm requires an exponential number of state transitions in the observed time course data, with respect to the number of genes n in the network. However, as most of biological networks are sparse (Arnone and Davidson, 1997; Thieffry et al., 1998), REVEAL works effectively in practice and (Akutsu et al., 1999) also have proven that only $O(\log n)$ state transitions are required when the maximum number of predictors, $K = \max_{i=1}^{n} k_i$, for all the genes in the network is small. However, the original REVEAL algorithm and the exhaustive algorithm in Akutsu et al. (1999) focus on inferring BNs instead of PBNs and require finding the "consistent" Boolean functions for each gene. They assume that the observed time course data themselves are completely consistent based on underlying Boolean functions without errors.

With random perturbations introduced in PBNs, instead of finding consistent Boolean functions, the inference algorithm Best-Fit (Lähdesmäki et al., 2003; Marshall et al., 2007; Lähdesmäki and Shmulevich, 2012) searches for the best-fit function for each gene by exhaustively searching for all the combination of potential regulator sets. Similarly, with small K, the algorithm is feasible with a given number of state transitions and is efficient with the time complexity $O\left(m \log m \mathrm{poly}(n)\right)$ with m state transitions, in which $\mathrm{poly}(n)$ is time to compute the minimum error for one given state transition Lähdesmäki et al. (2003). For our implementations (Murphy and Mian, 1999; Lähdesmäki et al., 2003; Lähdesmäki and Shmulevich, 2012) based on both REVEAL and Best-Fit algorithms, we have modified the algorithms to get both regulator sets and corresponding best-fit functions. Finally, with a limited number of observed state transitions and potential random perturbations, the inferred regulatory functions may still be *partially defined Boolean functions* (Lähdesmäki et al., 2003). To obtain a unique solution, we can further impose other biologically motivating constraints. For example, in Pal et al. (2005), BNs are inferred simply based on the attractor structure of network dynamics, which can be extended to impose dynamic constraints to search for suitable solutions.

In this work, we adopt the MDL-based network inference algorithm (Tabus and Astola, 2001; Zhao et al., 2006; Dougherty et al., 2008) to penalize the model complexity of inferred networks. We have modified the algorithm proposed in Zhao et al. (2006) to identify the best regulator set with the minimum combination of network coding length, capturing the model complexity, and data coding length, which is similar to REVEAL based on mutual information. The MDL network coding length in Zhao et al. (2006) has similar asymptotic performance to the Bayesian Information Criterion (BIC) model complexity, which we also have implemented in our set of inference algorithms. Finally, both MDL (Zhao et al., 2006) and BIC (Murphy and Mian, 1999) adopt *ad hoc* measures of model description length that necessitate tuning parameters as weighting coefficients to balance the model and data coding lengths (Tabus and Astola, 2001; Dougherty et al., 2008) and inference performances or validity measures may change with different tuning parameters. To overcome this difficulty, we also adopt a universal MDL (uMDL) network inference algorithm (Dougherty et al., 2008) in which the model and data coding length together is a theoretical measure derived from a universal normalized maximum likelihood model and no tuning parameters are needed (Tabus and Astola, 2001).

3. IMPLEMENTATION

We will compare network inference algorithms for their inferential validity based on both synthetic networks as well as a well-studied metastatic melanoma network (Bittner et al., 2000; Kim et al., 2002; Weeraratna et al., 2002; Qian and Dougherty, 2008; Yousefi and Dougherty, 2013).

To evaluate the inference algorithms based on simulated time series of network states, we first generate random PBNs with properties that resemble those of biological networks so that we have the ground truth networks for validation. For appropriate evaluation, we have imposed a few assumptions: First, as genetic regulatory networks are commonly believed to have sparse connectivity topology, we have restricted the Boolean functions in random PBNs to have at most five predictors: $K = \max_{i=1}^{n} k_i \leq 5$. This assumption also enables all the inference algorithms to run smoothly on these random PBNs as the computational complexity of these algorithms, especially those based on exhaustive enumerations, reduces significantly as shown in Akutsu et al. (1999); Lähdesmäki et al. (2003). Second, as the network state space is exponential with respect to the number of genes or the network size, the number of state transitions observed will usually not be large enough to uniquely determine the network structure and thereafter the regulatory functions. For the inference algorithms adopted in this paper, all of which are based on solving the consistency problem (Liang et al., 1998; Akutsu et al., 1999; Lähdesmäki et al., 2003; Zhao et al., 2006; Martin et al., 2007), we take the most sparse network as the final solution within the feasible networks that give the same minimum prediction errors in REVEAL and Best-Fit or the same objective function values in the inference algorithms with BIC and MDL regularization. The motivation is that biological networks are usually stable and robust to random perturbations and larger k_i leads to increased sensitivity of the steady-state distribution to random gene perturbations Shmulevich and Dougherty (2007), Qian and Dougherty (2009a, 2010).

With either simulated or real ground truth networks, we can generate time series of gene expression profiles with different numbers of state transitions based on their underlying Markov chains so that we can investigate the inference performances with different available sample sizes. We have implemented REVEAL, MDL, BIC, uMDL, and Best-Fit to infer networks with these simulated time series. Our implementations of these different algorithms are based on the PBN Toolbox (http://code.google.com/p/pbn-matlab-toolbox/), the Bayes Net Toolbox (https://code.google.com/p/bnt/), as well as the source code provided by the authors of Dougherty et al. (2008). The detailed descriptions of these different algorithms can be found in the corresponding papers (Liang et al., 1998; Murphy and Mian, 1999; Lähdesmäki et al., 2003; Zhao et al., 2006; Dougherty et al., 2008; Lähdesmäki and Shmulevich, 2012).

We compute three distance functions $\mu(\psi(S), \mathcal{H})$ to evaluate an inference algorithm ψ: (1) the Hamming distance μ_{ham}; (2) the L_1 norm μ_{ss} between the steady-state distributions of $\psi(S)$ and $\mathcal{H}$; and (3) the controllability distance μ_{ctrl} defined in (2). For inferential validity based on controllability, we find the optimal stationary control policies for the original and inferred networks based on the MSSA algorithm (Yousefi and Dougherty, 2013).

4. RESULTS AND DISCUSSION

4.1. SIMULATED BNps WITH 7 GENES

We first evaluate different inference algorithms on synthetically generated random networks. We generate 1000 random BNps with $n = 7$ genes, maximum input degree $K = 3$, and perturbation probability $p = 0.01$. For each node, we uniformly assign 1 to K regulators. Hence the average connectivity in this set of random networks is 2. After determining the regulatory relationships among nodes, the regulatory functions for each node are determined by randomly filling in the corresponding truth tables with Bernoulli random numbers with the bias following a Beta distribution with mean 0.5 and standard deviation 0.01. For each random BNp, we simulate time series of different numbers of state transitions based on its underlying Markov chain. The number of "observed" state transitions M ranges from 10 to 60 to reflect the difficulty level of network inference. For control, we choose the first node as the marker gene and define the undesirable states as these network states with the first node down-regulated. In the binary representation of network states, $\mathcal{U} = \{\mathbf{x}|x_1 = 0\}$. As the networks are randomly generated, without loss of generality, we allow intervention on the last node as the control gene, which we can either knock up or down to derive control policies. In our simulated random BNps, we have the original average undesirable steady state mass $\pi_{\mathcal{U}}^{\text{org}} = 0.5071$ with standard deviation 0.3575, with $\pi_{\mathcal{U}}^{\text{org}} \approx 0.5$ because we set the bias to 0.5. When we apply the MSSA algorithm to derive the optimal stationary control policies for these random BNps, the average controlled undesirable steady state mass is $\pi_{\mathcal{U}} = 0.3703$ with the standard deviation 0.3749.

Based on these simulated time series, we have implemented REVEAL, BIC, MDL, uMDL, and Best-Fit inference algorithms and modified accordingly to reconstruct BNps, including regulatory relationships and regulatory functions represented as general truth tables. For BIC and MDL, we set the regularization coefficients to values previously reported to have good performance in Zhao et al. (2006), $\lambda = 0.5$ for BIC and $\lambda = 0.3$ for MDL.

Table 1 provides the network inferential validity measurements: normalized Hamming distance μ_{ham} (Hamming distance over the total number of edges in true networks), the steady-state distance μ_{ss}, and the controllability distance μ_{ctrl} for different network inference algorithms given different numbers of state transitions. As discussed in (Zhao et al., 2006), BIC and MDL perform similarly. Regarding the accurate recovery of regulatory relationships, it is interesting to see that Best-Fit appears to achieve the best performance with respect to μ_{ham} while REVEAL does not perform very well. One explanation could be that REVEAL introduces many false positives, hopefully to best fit the data by using the functions with more regulators. This is in fact what we observe from our experiments. All the other inference algorithms choose the functions with the smallest number of regulators either by complexity regularization in BIC, MDL, and uMDL; or choosing the "parsimonious" functions with the minimum prediction errors in Best-Fit. For uMDL, we note that μ_{ham} improves quickly with the increasing sample size compared to other complexity regularization algorithms BIC and MDL. Based on our experiments, uMDL consistently generates very low false positive edges (close to zero), even with a very limited number of samples, which is the main advantage of the uMDL algorithms. This has also been shown in the original paper (Dougherty et al., 2008). For μ_{ss}, both REVEAL and Best-Fit perform consistently better than BIC, MDL, and uMDL, since both REVEAL and Best-Fit aim to find the network models that best fit the observed state transitions. With regularization on model complexity by BIC, MDL, and uMDL, the steady-state distances are greater. As mentioned earlier, REVEAL and Best-Fit, especially REVEAL, reconstruct networks with more edges to explain the observed data, which leads to smaller μ_{ss}.

When we investigate the inferential validity with respect to controllability, μ_{ctrl}, we see interesting changes of tendency between the five algorithms. Especially with very few state transitions, $M = 10$, BIC, MDL, and uMDL algorithms perform better than REVEAL and Best-Fit, which indicates that the regularization on model complexity with a limited number of observations helps reconstruct network models that yield better controllers. With more observations, REVEAL and Best-Fit gradually perform better than BIC, MDL, and uMDL due to introduced bias by model complexity regularization.

Figure 2 plots μ_{ham}, μ_{ss}, and the average undesirable steady-state mass using the control policy designed on the inferred network via the MSSA algorithm. For comparison purposes, the latter average is compared to the average original undesirable mass and the average undesirable mass following application of the MMSA control policy designed on the original network. As M increases from 10 to 60, all algorithms improve. In fact, with more than 50 observed state transitions for these generated random BNps, the derived stationary control policies achieve almost the same performance compared to the optimal control policies with complete knowledge of the network models. The average performances from inferred networks are in fact within 5% for all five inference algorithms when $M = 60$.

We further evaluate inference algorithms on a similar set of 1000 random BNps with $n = 7$ genes with the same settings but change the maximum input degree $K = 5$, which increases the average connectivity to 3. For this set of random BNps, we have the average undesirable original steady state mass $\pi_{\mathcal{U}}^{\text{org}} = 0.4841$ with standard deviation 0.3171 . When we apply the MSSA algorithm to derive the optimal stationary control policies for these random BNps, the average controlled undesirable steady state mass is $\pi_{\mathcal{U}} = 0.2529$ with the standard deviation 0.3144. The average shift of undesirable masses is higher compared to the previous set of random networks, which is expected as the network sensitivity monotonically increases with the average network connectivity (Kauffman, 1993; Shmulevich and Dougherty, 2007; Qian and Dougherty, 2009a). With higher sensitivity, networks can be more effectively controlled. We again compare the inferential validity as in the previous experiment. **Figure 3** shows

Table 1 | The comparison of network inference algorithms (REVEAL, BIC, MDL, uMDL, and Best-Fit) with *M* different number of observed state transitions.

Validity	μ_{ham}			μ_{ss}			μ_{ctrl}		
M	**10**	**30**	**50**	**10**	**30**	**50**	**10**	**30**	**50**
REVEAL	0.7774	0.6111	0.6511	0.6743	0.4657	0.4216	0.1067	0.0275	0.0049
BIC	0.6966	0.4196	0.3304	0.8679	0.7089	0.5492	0.0739	0.0300	0.0126
MDL	0.7204	0.4260	0.3294	0.9414	0.7225	0.5435	0.0775	0.0311	0.0121
uMDL	0.8000	0.3728	0.2471	1.1957	0.6973	0.4935	0.1058	0.0352	0.0093
Best-Fit	0.7311	0.3919	0.2913	0.6378	0.4244	0.4098	0.1027	0.0250	0.0045

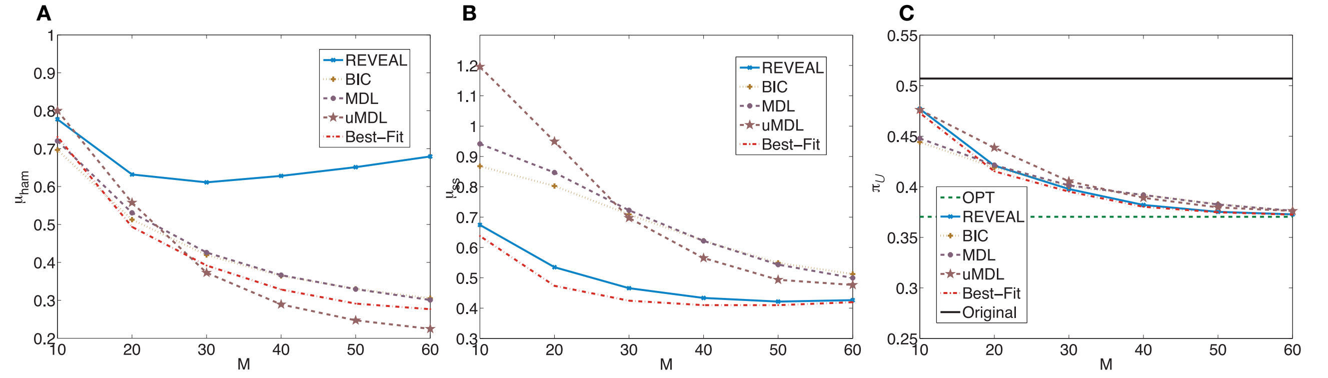


FIGURE 2 | Performance comparison of five network inference algorithms by different validity indices based on simulated BNps with 7 genes and $K = 3$. (A) Average normalized Hamming distance μ_{ham}; **(B)** μ_{ss}; **(C)** average undesirable steady-state mass $\pi_{\mathcal{U}}$ after applying derived stationary control policies based on inferred networks to the original ground truth BNps, compared to the average undesirable mass obtained by the optimal control policy (OPT) based on the complete knowledge of original BNps and the average undesirable mass before intervention (Original).

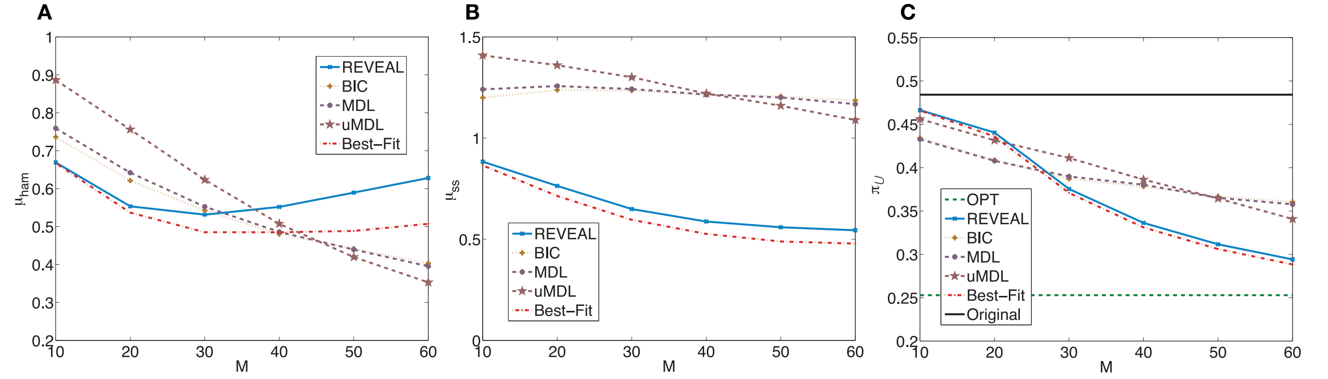


FIGURE 3 | Performance comparison of five network inference algorithms by different average validity indices based on BNps with 7 genes and $K = 5$. (A) Average normalized Hamming distance μ_{ham}; **(B)** μ_{ss}; **(C)** average undesirable steady-state mass $\pi_{\mathcal{U}}$ after applying derived stationary control policies based on inferred networks to the original ground truth BNps, compared to the average undesirable mass obtained by the optimal control policy (OPT) based on the complete knowledge of original BNps and the average undesirable mass before intervention (Original).

plots analogous to **Figure 2**. Especially, we note that in this set of experiments, we can achieve close-to-optimal intervention with fairly small sample size as illustrated in **Figure 3C**. It is clear that the performance of different inference algorithms depends on the characteristics of the networks, especially the network sensitivity. More specifically, all three indices become worse for all the inference algorithms, illustrating that with increasing network sensitivity, the inference problem becomes more difficult. It is also clear that the performance improves at slower rates with the increasing sample size when we have higher network sensitivity. Another important difference is that for this set of random networks, both REVEAL and Best-Fit have higher μ_{ham} when the number of samples increase above 40. The reason may be due to the tendency of random perturbations forcing both algorithms to bias toward more complex Boolean functions with more input variables as regulators.

4.2. SIMULATED BNps WITH 9 GENES

For simulations with 9 genes, owing to run time, we generate 200 BNps with $n = 9$ genes and perturbation probability $p = 0.01$. We again make uniformly random assignments of 1 to K regulators, with $K = 3$ so that the average connectivity is 2. The bias for the corresponding truth tables follows the same Beta distribution with mean 0.5 and stand deviation 0.01. The number of "observed" state transitions M range from 10 to 60. The derivation of control policies is still based on the definition of the undesirable states $\mathcal{U} = \{\mathbf{x}|x_1 = 0\}$ and the last node is the control gene. In the simulated random BNps, the average undesirable steady state mass is $\pi_{\mathcal{U}}^{org} = 0.4886$ with the standard deviation 0.3764. When we apply the MMSA algorithm to derive the optimal stationary control policies for these random BNps, the average controlled undesirable steady state mass is $\pi_{\mathcal{U}} = 0.3668$ with the standard deviation 0.3863. **Figure 4** shows plots analogous to **Figure 2** with the trends similar as those observed in the previous experiments with corresponding random BNps with 7 genes and $K = 3$.

In the second set of simulated random BNps with 9 genes, the settings are the same except that $K = 5$. In these random networks, the average undesirable steady state mass is $\pi_{\mathcal{U}}^{org} = 0.4895$ with standard deviation 0.3269. When we apply the

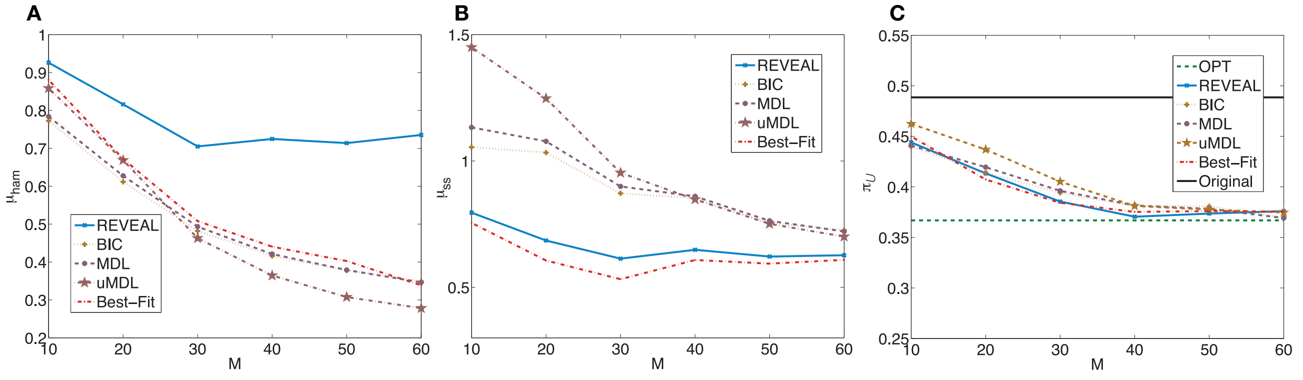


FIGURE 4 | Performance comparison of five network inference algorithms by different average validity indices based on BNps with 9 genes and $K = 3$. (A) Average normalized Hamming distance μ_{ham}; **(B)** μ_{ss}; **(C)** average undesirable steady-state mass $\pi_{\mathcal{U}}$ after applying derived stationary control policies based on inferred networks to the original ground truth BNps, compared to the average undesirable mass obtained by the optimal control policy (OPT) based on the complete knowledge of original BNps and the average undesirable mass before intervention (Original).

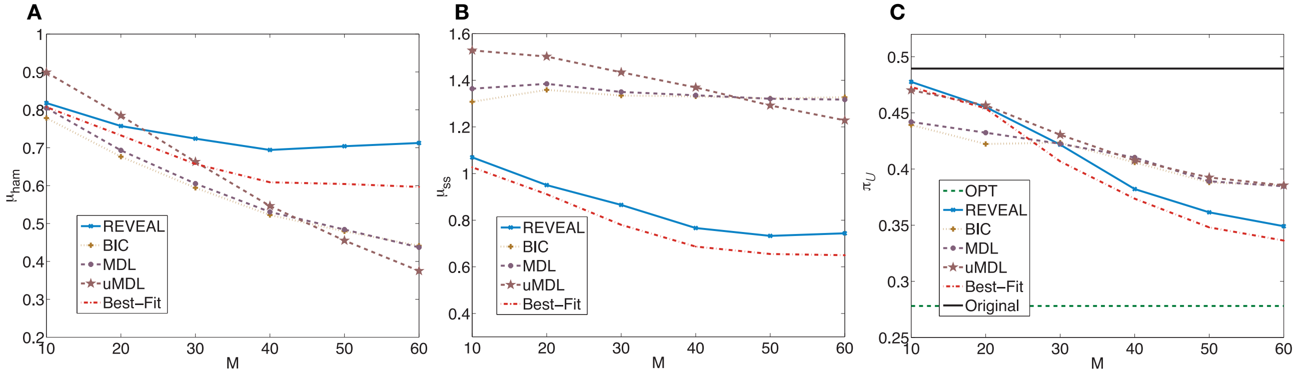


FIGURE 5 | Performance comparison of five network inference algorithms by different average validity indices based on BNps with 9 genes and $K = 5$. (A) Average normalized Hamming distance μ_{ham}; **(B)** μ_{ss}; **(C)** average undesirable steady-state mass $\pi_{\mathcal{U}}$ after applying derived stationary control policies based on inferred networks to the original ground truth BNps, compared to the average undesirable mass obtained by the optimal control policy (OPT) based on the complete knowledge of original BNps and the average undesirable mass before intervention (Original).

MSSA algorithm to derive the optimal stationary control policies for these random BNps, the average controlled undesirable steady state mass is $\pi_{\mathcal{U}} = 0.2781$ with standard deviation 0.3268. **Figure 5** is analogous to **Figure 3**.

In summary, when we evaluate different inference procedures with respect to different inferential validity criteria, different inference procedures show different trends with their increasing sample size. Their performance overall depends on network characteristics as well as available samples. Finally, when effective intervention is our final operational objectivel, it is promising that we can achieve effective intervention based on inferred networks, even with fairly small sample size as illustrated in **Figures 2C, 3C, 4C, 5C**.

4.3. A METASTATIC MELANOMA NETWORK

Finally, we evaluate different inference algorithms based on a metastatic melanoma network used in previous studies on network intervention (Qian and Dougherty, 2008; Qian et al., 2009; Yousefi and Dougherty, 2013). The network has 10 genes listed in the order from the most to the least significant bit: WNT5A, PIR, S100P, RET1, MMP3, PLCG1, MART1, HADHB, SNCA, and STC2. The order does not affect our analysis. We note here that this network was derived from gene expression data (Kim et al., 2002) collected in studies of metastatic melanoma (Bittner et al.,

Table 2 | Regulatory functions in the metastatic melanoma network [Modified from Table 1 in Yousefi and Dougherty (2013)].

Node	Gene	Boolean function
x_1	WNT5A	$(x_3 \wedge x_5 \wedge \neg x_6) \vee (\neg x_5 \wedge x_6)$
x_2	PIR	$(\neg x_1 \wedge \neg x_3 \wedge x_5) \vee (x_1 \wedge \neg x_3 \wedge \neg x_5)$
x_3	S100P	x_7
x_4	RET1	$(\neg x_1 \wedge x_2 \wedge x_4) \vee (\neg x_2 \wedge x_4)$
x_5	MMP3	$(x_4 \wedge x_9) \vee (\neg x_9)$
x_6	PLCG1	$(\neg x_4 \wedge \neg x_7) \vee (x_4 \wedge x_7 \wedge x_{10})$
x_7	MART1	x_7
x_8	HADHB	$(x_1 \wedge x_5) \vee (\neg x_5 \wedge \neg x_9) \vee (x_1 \wedge \neg x_5 \wedge x_9)$
x_9	SNCA	$(\neg x_1 \wedge \neg x_7 \wedge \neg x_{10}) \vee (x_4 \wedge \neg x_7 \wedge x_{10}) \vee x_7$
x_{10}	STC2	$\neg x_3$

2000; Weeraratna et al., 2002). **Table 2** and **Figure 6** together illustrate the regulatory relationships among these selected 10 genes from 587 genes profiled in Bittner et al. (2000), Weeraratna et al. (2002), which were derived based on gene expression data rather than curated regulatory relationships among genes in literature. We believe that the model is appropriate for the purpose of illustrating the effectiveness of objective inferential validity on quantifying the performance of inference procedures in this work. Based on these information, we construct a BNp with the perturbation probability $p = 0.01$. As in the previous studies, the control objective is based on the fact that up-regulation of WNT5A is associated with increased metastasis. Thus, $\mathcal{U} = \{\mathbf{x}|x_1 = 1\}$. For this network, the undesirable steady-state mass is $\pi_{\mathcal{U}} = 0.2073$ in the original network, which can be reduced as illustrated in **Table 3** with different genes as potential targets using the MSSA algorithm on the original network. Based on this model, we simulate 20, 60, and 80 state transitions and infer the network based on these time series data using all five algorithms. As the primary objective here is to reduce the undesirable steady-state mass with WNT5A up-regulated, we focus on its shift derived by the MSSA algorithm based on the inferred networks using different inference algorithms.

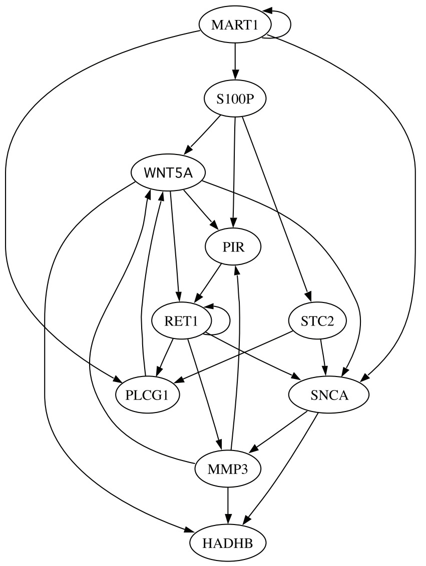


FIGURE 6 | Multivariate relationships among genes in the metastatic melanoma network.

Table 3 compares this network inferential validity μ_{ctrl} for different algorithms. According to the table, even with small sample size, we may obtain effective intervention strategies in most cases from all five inference algorithms. For example, with $M = 60$ samples, RET1, MMP3, PLCG1, and MART1 can be successfully identified as effective intervention targets based on inferred networks using different inference algorithms. These potential targets have been similarly identified in previous publications (Qian and Dougherty, 2008, 2009a; Qian et al., 2009; Yousefi and

Table 3 | The shifted undesirable steady-state mass in the metastatic melanoma network by the MSSA algorithm for different control genes derived on inferred networks from five network inference algorithms (REVEAL, BIC, MDL, uMDL, and Best-Fit) with $M = [20, 60, 80]$ different number of observed state transitions, compared to the optimal shift by applying the MSSA algorithm to the original network.

Control	WNT5A	PIR	S100P	RET1	MMP3	PLCG1	MART1	HADHB	SNCA	STC2
					OPT (UC)					
	0.0847	0.1340	0.1767	0.1766	0.1965	0.1965	0.1799	0.0000	0.0259	0.1680
M					**REVEAL**					
20	−0.0421	0.1319	0.1702	0.1761	0.1739	−0.1375	0.1777	0.0000	0.0227	0.1027
60	0.0054	0.1316	0.1754	0.0634	0.1961	0.1940	0.1622	0.0000	−0.0448	0.1660
80	0.0728	0.1339	0.1737	0.1727	0.1965	0.1965	0.1795	0.0000	0.0235	0.1678
M					**BIC**					
20	−0.0421	0.0789	0.1696	−0.4032	0.1802	−0.1246	0.0026	0.0000	−0.2800	0.0132
60	−0.0421	0.0789	0.1264	0.1765	0.1965	0.1965	0.1799	0.0000	0.0259	0.0023
80	0.0738	0.1340	0.1767	0.1766	0.1965	0.1965	0.1799	0.0000	0.0259	0.1680
M					**MDL**					
20	−0.0421	0.0628	0.1696	−0.2655	0.1802	−0.2695	0.0026	0.0000	−0.3586	−0.0574
60	−0.0421	0.0789	0.1264	0.1764	0.1965	0.1965	0.1799	0.0000	0.0259	0.0023
80	0.0738	0.1340	0.1767	0.1766	0.1965	0.1965	0.1799	0.0000	0.0259	0.1680
M					**uMDL**					
20	−0.0421	0.0829	−0.2255	−0.2645	−0.2300	−0.1810	−0.2952	0.0000	−0.2295	0.1065
60	−0.0421	0.1309	0.0363	0.1766	0.1965	0.1965	0.1799	0.0000	0.0259	0.0189
80	0.0844	0.1340	0.1767	0.1766	0.1965	0.1965	0.1799	0.0000	0.0259	0.1680
M					**Best-Fit**					
20	0.0588	0.1330	0.1724	0.1762	0.1793	−0.0662	0.1796	0.0000	0.0256	0.0115
60	0.0816	0.1340	0.1767	0.1764	0.1965	0.1965	0.1798	0.0000	0.0258	0.1677
80	0.0728	0.1339	0.1737	0.1766	0.1965	0.1965	0.1799	0.0000	0.0259	0.1680

Dougherty, 2013), which demonstrates the feasibility of deriving effective therapeutic strategies even with partially observed data from the original system. All the algorithms achieve almost optimal performance for all possible control genes when $M = 80$. In fact, Best-Fit appears to obtain the best performance when $M = 60$ compared to all the other algorithms as it better captures network dynamics manifested as steady-state distributions. Hence, Best-Fit appears to be the best-performing inference algorithm when we consider the operational objective to be beneficial alteration of network dynamics. We also note that with small samples ($M = 20$), it is relatively difficult to derive effective control based on the inferred network by uMDL; however, when we have enough samples ($M = 80$), we can derive the most effective control for all the target genes based on uMDL. This is again due to its advantage of obtaining consistently close to zero false positive regulators, which leads to the best performance when we have enough samples. This is consistent with the previous results we have seen using simulated networks.

5. CONCLUDING REMARKS

We have considered inferential validity from three perspectives: (1) Hamming distance, which relates to accurate network topology; (2) steady-state distribution, which corresponds to accurate phenotyping because attractors dominate the steady-state mass and attractors correspond to phenotypes; and (3) controllability. From a translational perspective, controllability is an important criterion because a key interest in translational genomics is to derive intervention strategies from gene network models. We have observed from the experiments that controllability provides quite a different view of validation than either Hamming distance or steady-state mass, with performance comparison depending strongly on the number of observations. The upside is that one can achieve decent control when there is still considerable distance between the original and inferred networks relative to Hamming distance and steady-state mass. This depends on network size, connectivity, sample size, and the inference procedure. The general point is that it may be wise to use objective-based measures of validity for practical applications. While the individual components and connections in a system may overall be fairly inaccurate, it may be that those that matter for the objective are determined fairly accurately so that the inaccuracy of the others is of little consequence. The situation is analogous to uncertainty in model classes. While entropy provides an overall measure of model uncertainty, it may be better to use a measure of uncertainty that accounts for the cost of the uncertainty relative to a particular objective because uncertainty that does not negatively impact attainment of the objective is of no practical consequence (Yoon et al., 2013).

REFERENCES

Akutsu, T., Miyano, S., and Kuhara, S. (1999). Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. *Pac. Symp. Biocomp.* 4, 17–28.

Arnone, M. I., and Davidson, E. H. (1997). The hardwiring of development: organization and function of genomic regulatory systems. *Development* 124, 1851–1864.

Bertsekas, D. P. (2001). *Dynamic Programming and Optimal Control*, Vol. 1, 2. Nashua, NH: Athena Scientific.

Bittner, M., Meltzer, P., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., et al. (2000). Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature* 406, 536–540. doi: 10.1038/35020115

Brun, M., Dougherty, E. R., Shmulevich, I. (2005). Steady-state probabilities for attractors in probabilistic Boolean networks. *EURASIP J. Signal Process.* 85, 1993–2013. doi: 10.1016/j.sigpro.2005.02.016

Datta, A., and Dougherty, E. R. (2007). *Introduction to Genomic Signal Processing with Control*. New York, NY: CRC Press.

Dougherty, E. R., and Bittner, M. (2011). *Epistemology of the Cell: A Systems Perspective on Biological Knowledge*. New York, NY: IEEE Press Series on Biomedical Engineering, John Wiley. doi: 10.1002/9781118104866

Dougherty, E. R., and Datta, A. (2005). Genomic signal processing: diagnosis and therapy. *IEEE Signal Process. Mag.* 22, 107–112. doi: 10.1109/MSP.2005.1407722

Dougherty, J., Tabus, I., and Astola, J. (2008). Inference of gene regulatory networks based on a universal Minimum Description Length. *EURASIP J. Bioinform. Syst. Biol.* 2008:482090. doi: 10.1155/2008/482090

Dougherty, E. R. (2007). Validation of inference procedures for gene regulatory networks. *Curr. Genomics* 8, 351–359. doi: 10.2174/138920207783406505

Dougherty, E. R. (2011). Validation of gene regulatory networks: scientific and inferential. *Brief. Bioinform.* 12, 245–252. doi: 10.1093/bib/bbq078

Faryabi, B., Vahedi, G., Chamberland, J.-F., Datta, A., and Dougherty, E. R. (2009). Intervention in context-sensitive probabilistic Boolean networks revisited. *EURASIP J. Bioinform. Syst. Biol.* 2009:360864. doi: 10.1155/2009/360864

Hashimoto, R. F., Kim, S., Shmulevich, I., Zhang, W., Bittner, M. L., and Dougherty, E. R. (2004). Growing genetic regulatory networks from seed genes. *Bioinformatics* 20, 1241–1247. doi: 10.1093/bioinformatics/bth074

Ivanov, I., Simeonov, P., Ghaffari, N., Qian, X., and Dougherty, E. R. (2010). Selection policy induced reduction mappings for boolean networks. *IEEE Trans. Signal Process.* 58, 4871–4882. doi: 10.1109/TSP.2010.2050314

Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Theor. Biol.* 22, 437–467. doi: 10.1016/0022-5193(69)90015-0

Kauffman, S. A. (1993). *The Origins of Order*. New York, NY: Oxford University Press.

Kim, S., Li, H., Dougherty, E. R., Cao, N. W., Chen, Y. D., Bittner, M., et al. (2002). Can Markov chain models mimic biological regulation? *J. Biol. Syst.* 10, 337–357. doi: 10.1142/S0218339002000676

Lähdesmäki, H., Shmulevich, I., Yli-Harja, O. (2003). On Learning gene regulatory networks under the Boolean network model. *Mach. Learn.* 52, 147–167. doi: 10.1023/A:1023905711304

Lähdesmäki, H. and Shmulevich, I. (2012). “Inference of Genetic Regulatory Networks via Best-Fit Extensions.” in *Computational And Statistical Approaches To Genomics*, eds W. Zhang and I. Shmulevich (Boston, MA: Kluwer Academic Publishers), 259–278.

Liang, S., Fuhrman, S., and Somogyi, R. (1998). REVEAL: a general reverse engineering algorithm for inference of genetic network architectures. *Pac. Symp. Biocomput.* 3, 18–29.

Marbach, D., Prill, R. J., Schaffter, C., Mattiussi, C., Floreano, D., and Stolovitzky G. (2010). Revealing strengths and weaknesses of methods for gene network inference. *Proc. Natl. Acad. Sci. U.S.A.* 107, 6286–6291. doi: 10.1073/pnas.0913357107

Marshall, S., Yu, L., Xiao, Y., and Dougherty, E. R. (2007). Inference of probabilistic Boolean networks from a single observed temporal sequence. *EURASIP J. Bioinform. Syst. Biol.* 2007:32454. doi: 10.1155/2007/32454

Martin, S., Zhang, Z., Martino, A., and Faulon, J.-L. (2007). Boolean dynamics of genetic regulatory networks inferred from microarray time series data. *Bioinformatics* 23, 866–874. doi: 10.1093/bioinformatics/btm021

Murphy, K., and Mian, S. (1999). “Modelling gene expression data using dynamic Bayesian networks,” in *Technical Report, University of California,* (Berkeley, CA).

Pal, R., Ivanov, I., Datta, A., Bittner, M. L., and Dougherty, E. R. (2005). Generating Boolean networks with a prescribed attractor structure. *Bioinformatics* 21, 4021–4025. doi: 10.1093/bioinformatics/bti664

Pal, R., Datta, A., and Dougherty, E. R. (2006). Optimal infinite horizon control for probabilistic Boolean networks. *IEEE Trans. Signal Process.* 54, 2375–2387. doi: 10.1109/TSP.2006.873740

Qian, X., and Dougherty, E. R. (2008). Effect of function perturbation on the steady-state distribution of genetic regulatory networks: optimal structural intervention. *IEEE Trans. Signal Process.* 56, 4966–4975. doi: 10.1109/TSP.2008.928089

Qian, X., and Dougherty, E. R. (2009a). On the long-run sensitivity of probabilistic Boolean networks. *J. Theor. Biol.* 257, 560–577. doi: 10.1016/j.jtbi.2008.12.023

Qian, X., and Dougherty, E. R. (2009b). "Control-compatible state reduction for Boolean networks," in *IEEE International Workshop on Genomic Signal Processing and Statistics* (Minneapolis, MN).

Qian, X., and Dougherty, E. R. (2010). "Comparative study on sensitivities of Boolean networks," in *IEEE International Workshop on Genomic Signal Processing and Statistics (GENSIPS)*, Cold Spring Harbor Laboratory, NY: IEEE.

Qian, X., Ivanov, I., Ghaffari, N., and Dougherty, E. R. (2009). Intervention in gene regulatory networks via greedy control policies based on long-run behavior. *BMC Syst. Biol.* 3:16. doi: 10.1186/1752-0509-3-61

Qian, X., Ghaffari, N., Ivanov, I., and Dougherty, E. R. (2010). State reduction for network intervention in probabilistic Boolean networks. *Bioinformatics* 26, 3098–3104. doi: 10.1093/bioinformatics/btq575

Shmulevich, I., and Dougherty, E. R. (2007). *Genomic Signal Processing* Princeton, NJ: Princeton University Press. doi: 10.1137/1.9780898717631

Shmulevich, I., and Dougherty E. R. (2010). *Probabilistic Boolean Networks: The Modeling and Control of Gene Regulatory Networks*. New York, NY: SIAM Press.

Shmulevich, I., Dougherty, E. R., Kim, S., and Zhang, W. (2002a). Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics* 18, 261–274. doi: 10.1093/bioinformatics/18.2.261

Shmulevich, I., Dougherty, E. R., and Zhang, W. (2002b). Control of stationary behaviour in probabilistic Boolean networks by means of structural intervention. *Biol. Syst.* 10, 431–446. doi: 10.1142/S0218339002000706

Tabus, I., and Astola, J. (2001). On the use of MDL principle in gene expression prediction. *J. Appl. Signal Process.* 4, 297–303. doi: 10.1155/S1110865701000270

Thieffry, D., Huerta, A. M., Pèrez-Rueda, E., and Collado-Vides, J. (1998). From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in *Escherichia coli*. *BioEssays* 20, 433–440.

Weeraratna, A. T., Jiang, Y., Hostetter, G., Rosenblatt, K., Duray, P., Bittner, M., et al. (2002). Wnt5a signalling directly affects cell motility and invasion of metastatic melanoma. *Cancer Cell* 1, 279–288. doi: 10.1016/S1535-6108(02)00045-4

Xiao, Y., and Dougherty, E. R. (2007). The impact of function perturbations in Boolean networks. *Bioinformatics* 23, 1265–1273. doi: 10.1093/bioinformatics/btm093

Yoon, B.-J., Qian, X., Dougherty, E. R. (2013). Quantifying the objective cost of uncertainty in complex dynamical systems. *IEEE Trans. Signal Process.* 61, 2256–2266. doi: 10.1109/TSP.2013.2251336

Yousefi, N. R., and Dougherty, E. R. (2013). Intervention in gene regulatory networks with maximal phenotype alteration. *Bioinformatics* 29, 1758–1767. doi: 10.1093/bioinformatics/btt242

Zhao, W., Serpedin, E., and Dougherty, E. R. (2006). Inferring gene regulatory networks from time series data using the minimum description length principle. *Bioinformatics* 22, 2129–2135. doi: 10.1093/bioinformatics/btl364

Zhou, X., Wang, X., Pal, R., Ivanov, I., Bittner, M. L., and Dougherty, E. R. (2004). A Bayesian connectivity-based approach to constructing probabilistic gene regulatory networks. *Bioinformatics* 20, 2918–2927. doi: 10.1093/bioinformatics/bth318

Conflict of Interest Statement: 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.

Received: 26 June 2013; accepted: 18 November 2013; published online: 12 December 2013.

Citation: Qian X and Dougherty ER (2013) Validation of gene regulatory network inference based on controllability. Front. Genet. ***4****:272. doi: 10.3389/fgene.2013.00272*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

APPENDIX

We plot the normalized false positive rates (the ratio of the number of false positive regulators over the total number of edges) in **Figure A1**, in which we can see that the performance of different algorithms are consistent as we discussed previously.

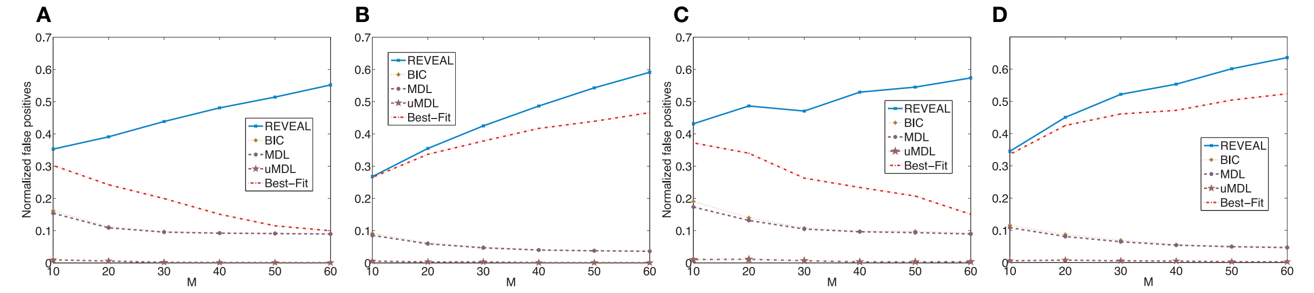


FIGURE A1 | Comparison of five network inference algorithms by normalized false positive rates. (A) BNps with 7 genes and $K = 3$; **(B)** BNps with 7 genes and $K = 5$; **(C)** BNps with 9 genes and $K = 3$; **(D)** BNps with 9 genes and $K = 5$.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 16 December 2013
doi: 10.3389/fgene.2013.00281

B-cell lymphoma gene regulatory networks: biological consistency among inference methods

Ricardo de Matos Simoes*[1]*, Matthias Dehmer*[2] *and Frank Emmert-Streib*[1]**

[1] Computational Biology and Machine Learning Laboratory, Faculty of Medicine, Health and Life Sciences, Center for Cancer Research and Cell Biology, School of Medicine, Dentistry and Biomedical Sciences, Queen's University Belfast, Belfast, UK
[2] Institute for Bioinformatics and Translational Research, UMIT, Hall in Tirol, Austria

Edited by:
Benjamin Haibe-Kains, Institut de Recherches Cliniques de Montréal, Canada

Reviewed by:
Victor P. Andreev, University of Miami, USA
Jaume Bacardit, University of Nottingham, UK

****Correspondence:***
Frank Emmert-Streib, Computational Biology and Machine Learning Laboratory, Faculty of Medicine, Health and Life Sciences, Center for Cancer Research and Cell Biology, School of Medicine, Dentistry and Biomedical Sciences, Queen's University Belfast, 97 Lisburn Road, Belfast, BT9 7BL, UK
e-mail: v@bio-complexity.com

Despite the development of numerous gene regulatory network (GRN) inference methods in the last years, their application, usage and the biological significance of the resulting GRN remains unclear for our general understanding of large-scale gene expression data in routine practice. In our study, we conduct a structural and a functional analysis of B-cell lymphoma GRNs that were inferred using 3 mutual information-based GRN inference methods: C3Net, BC3Net and Aracne. From a comparative analysis on the global level, we find that the inferred B-cell lymphoma GRNs show major differences. However, on the edge-level and the functional-level—that are more important for our biological understanding—the B-cell lymphoma GRNs were highly similar among each other. Also, the ranks of the degree centrality values and major hub genes in the inferred networks are highly conserved as well. Interestingly, the major hub genes of all GRNs are associated with the G-protein-coupled receptor pathway, cell-cell signaling and cell cycle. This implies that hub genes of the GRNs can be highly consistently inferred with C3Net, BC3Net, and Aracne, representing prominent targets for signaling pathways. Finally, we describe the functional and structural relationship between C3Net, BC3Net and Aracne gene regulatory networks. Our study shows that these GRNs that are inferred from large-scale gene expression data are promising for the identification of novel candidate interactions and pathways that play a key role in the underlying mechanisms driving cancer hallmarks. Overall, our comparative analysis reveals that these GRNs inferred with considerably different inference methods contain large amounts of consistent, method independent, biological information.

Keywords: gene regulatory network, C3Net, BC3Net, Aracne, GPEA, statistical inference

1. INTRODUCTION

To date, a vast amount of gene regulatory network (GRN) inference methods are being developed with the future goal to establish qualitative and quantitative procedures for a structural, biological and experimental validation of the inferred networks (Friedman, 2004; Wille et al., 2004; Werhli et al., 2006; Margolin and Califano, 2007; Yip et al., 2010; Zhang et al., 2011; Emmert-Streib et al., 2012). One of the most conservative approaches for GRN inference was introduced with the C3Net (Altay and Emmert-Streib, 2010, 2011) method that inferres at most one interaction (edge) for each gene with the strongest mutual dependency. An extension of C3Net was introduced by the bagging (Breiman, 1996; Zhang and Singer, 2010) approach BC3Net (de Matos Simoes and Emmert-Streib, 2012) that allows to aggregate ensembles of C3Net networks that are inferred from bootstrap (Efron and Tibshirani, 1994; Davison and Hinkley, 1997) datasets. The main advantage of a C3Net and BC3Net over many other methods is the intuitive interpretation of the inferred interactions that correspond to gene-pairs with the strongest significant mutual dependency, present in the data. Notably, a C3Net GRN has the property to infer very sparse, modular networks with a preference for interactions in the periphery of the network corresponding to genes with a less complex mutual dependency structure.

In de Matos Simoes and Emmert-Streib (2011), C3Net was used to infer GRNs from simulated gene expression data using a known underlying network structure. This study demonstrated that interactions (edges) of genes with a low number of direct neighbors (low degree) are more likely to be inferred correctly compared to interactions of genes with a large number of direct neighbors. From this observation one can presume that the interaction periphery of the unknown gene network is more prominently represented in an inferred GRN due to the lower complexities of the gene expression dependencies between the genes. However, the underlying gene network is unknown when a GRN is inferred from real biological gene expression data. Thus, the periphery and the center of the gene network is restricted to known experimental interactions that provide only a static and incomplete representation of the gene network. Furthermore, in de Matos Simoes et al. (2012) it was shown that the giant connected component (GCC) of the GRN using C3Net is highly enriched with membrane associated proteins. This observation suggested that the periphery of a gene network represents, to some extend, also the physical periphery of the biological cell that

Network Inference

- Q: What types of biological networks have been inferred in the paper?
- A: We use gene expression data from B-cell lymphoma and infer GRNs.
- Q: How was the quality/utility of the inferred networks assessed?
- A: We compare the inferred GRNs with a protein-protein interaction network and a transcriptional regulatory network. Furthermore, we compare 3 GRNs among each other to identify their similarity. This analysis is conducted by using the Gene Ontology database and a variety of additional databases.
- Q: How were these networks validated?
- A: All networks are analyzed computationally and statistical hypotheses testing is employed to test various hypotheses about the network structure and the biological function of the investigated GRNs.

is centered around signaling receptors that represent the major hubs of the GRN.

When comparing different methods with each other for inferring GRN it is important to conduct this comparison on similar grounds. For this reason, we are comparing in this paper only methods with each other that employ statistical hypothesis testing (Lehman, 2005; Young and Smith, 2005) and utilizing mutual information to estimate the interactions within regulatory networks. In this way we are avoiding a potential bias that could result from comparisons between networks with a different meaning.

The performance of GRN inference methods have been often compared using simulated data from biological or simulated network structures (Van den Bulcke et al., 2006; Schaffter et al., 2011; Emmert-Streib, 2013). One major problem with a simulation-based analysis is that the assumed mechanisms to simulate gene expression are only partially understood biologically, leaving a certain uncertainty about the resulting properties of the expression data. On the other hand, when real data are used, the underlying network structure remains unknown or highly incomplete. Furthermore, differences between inferred GRN using different methods may be negligible due to small sample sizes of real data sets and the presence of noise in these gene expression data.

Of great importance is the question “what” and “how consistent” is the information that can be extracted from a given large-scale gene expression data set to generate novel data-driven hypotheses. Unfortunately, to date, frequently, targets for wetlab studies are chosen based on the popularity of key genes rather than on the information within data sets. However, a non-data driven hypothesis ignores the limitations of an underlying data set to resolve known and unknown gene relationships. Furthermore, the efforts that have been performed for the validation of GRNs where mostly focusing on individual interactions, such as transcription factor target gene interactions (e.g., for MYC). To our knowledge, the most prominent genes appearing in a GRN, e.g., the actual hub genes, have not been considered for experimental validation.

In our study, we infer a C3Net, BC3Net and Aracne B-cell lymphoma GRN from a large-scale gene expression data set (Basso et al., 2005). We provide a structural and a functional comparison between the sparse, modular network structure inferred by C3Net and the more densely connected BC3Net and Aracne GRNs. Furthermore, we discuss the role of the hub genes and known cancer genes, such as MYC, we find in the inferred GRNs.

The paper is organized as follows. In the next section, we discuss the data we use for our analysis, the network inference methods and statistical measures we use for our analysis. In the results section, we present a comparative analysis and discuss differences between the 3 inferred GRNs and 2 reference networks (a PPN and a TRN). This article finishes with a discussion.

2. MATERIALS AND METHODS

2.1. GENE EXPRESSION DATA

For our study, we use the gene expression data with the GEO (Barrett et al., 2011) accession GSE2350 from Basso et al. (2005). The data set includes transformed and untransformed B-cell lymphoma samples. For our analysis, we consider only samples for which raw gene expression data in form of Affymetrix CEL files are available. From the total of 387 samples of the GSE2350 dataset, 344 samples were available in a CEL file format from the hgu95a and hgu95av2 chip platform. The data were preprocessed as described in detail in de Matos Simoes and Emmert-Streib (2011). Probeset identifiers were mapped to entrez gene symbols when available using the *org.Hs.eg.db* R-package (Carlson, 2013). Multiple probesets that mapped to the same gene were summarized using their median value. The final gene expression data set comprises 9684 genes and 344 samples. We subsequently applied a copula transformation to the processed gene expression data, as described in Margolin et al. (2006).

2.2. INFERENCE OF GENE REGULATORY NETWORKS

For the inference of the B cell lymphoma GRN, we use 3 mutual information-based GRN inference methods: C3Net, BC3Net and Aracne (Margolin et al., 2006; Altay and Emmert-Streib, 2010; de Matos Simoes and Emmert-Streib, 2012). Mutual information (MI) for all gene pairs is computed using a Pearson estimator (Meyer et al., 2007; Olsen et al., 2009),

$$I(X, Y) = -\frac{1}{2}\log\left(1 - \rho^2\right), \quad (1)$$

where ρ is the Pearson correlation coefficient.

2.2.1. *Null-distribution of mutual information values*

In order to determine the statistical significance of the mutual information values between genes we test for each pair of genes the following null hypothesis.

H_0^I : The mutual information between gene i and j is zero.

Because we are using a nonparametric test we need to obtain the corresponding null distribution for H_0^I from a randomization of the data. Principally, there are several ways to perform such a randomization. Here we permute the sample and gene labels for all genes of the entire expression matrix at once. In de Matos Simoes and Emmert-Streib (2011) we investigated three different randomization schemes and found that the randomization procedure applied here [in de Matos Simoes and Emmert-Streib (2011) called RM3] leads to similar results as other procedures that are computationally more demanding.

2.2.2. C3Net

The C3Net (Conservative Causal Core) algorithm consists of three main steps (Altay and Emmert-Streib, 2010, 2011). In the first step, mutual information values among all gene pairs are estimated. For this, we use a Pearson estimator for mutual information values, as given in Equation 1. In the second step, we select for each gene only the largest mutual information interaction (see **Figure 1**, indicated by the red elements in matrix I). This interaction corresponds also to the most significant gene among the neighbor edges. Third, we apply a non-parametric significance test for the mutual information values of the largest elements. The null distribution for this test is obtained from a randomization of the sample labels in the gene expression matrix. We use a significance level of $\alpha = 0.05$ in combination with a Bonferroni multiple testing correction (Dudoit and van der Laan, 2007).

Since C3Net employs mutual information values as test statistics among genes, there is no directional information that can be inferred thereof. Hence, the resulting network G_{C3Net} is undirected and unweighted (corresponding to a symmetric, binary adjacency matrix A; as indicated by the orange and yellow elements in **Figure 1**). For a detailed explanation of C3Net and its technical details, the reader is referred to Altay and Emmert-Streib (2010, 2011).

2.2.3. BC3Net

The BC3Net (de Matos Simoes and Emmert-Streib, 2012) algorithm is a bagging (Breiman, 1996) version of C3Net (Altay and Emmert-Streib, 2010, 2011). Briefly, BC3Net consists of 2 major steps. In the first step, a bootstrap ensemble of B data sets is generated. For each data set in the ensemble a GRN is inferred using C3Net; see **Figure 1**. In step two, the resulting ensemble of networks is combined into a weighted network, where the weights in this network G_{weighted} describe the ensemble consensus rate for an edge in the bootstrap ensemble. Then, we apply a binomial test to all edges in the weighted network and retain only edges that are statistically significance for a significance level of $\alpha = 0.05$ that pass a Bonferroni multiple testing correction (see **Figure 1B**—aggregation). This results into the final network G_{BC3Net}. For a statistically detailed description, the reader is referred to de Matos Simoes and Emmert-Streib (2012).

2.2.4. Aracne

The Aracne (algorithm for the reconstruction of accurate cellular networks) algorithm (Basso et al., 2005; Margolin et al., 2006) consists of two main steps. In step one, it estimates the mutual information values between all gene pairs and identifies their statistical significance. In **Figure 1**, these elements are represented as green elements in the matrix I'. In step two, all gene-triples (ijk), i.e., three genes with significant mutual information values, are used in combination with the *data processing inequality* (DPI) (Cover and Thomas, 1991) for thinning the resulting network. Specifically, for each triplet (ijk), the edge corresponding to the lowest mutual information value $I_1 = I_{i'j'}$, with $(i'j') = \text{argmin}\{I_{ij}, I_{jk}, I_{ik}\}$, is eliminated from the mutual information matrix I (in **Figure 1** indicated by the white circles) and the adjacency matrix A, if it is smaller than the second smallest mutual information value I_2, adjusted by a factor $(1 - \epsilon)$, i.e.,

$$A_{i'j'} = A_{j'i'} = \begin{cases} 0 & I_{i'j'} \leq I_2\,(1-\epsilon) \\ 1 & \text{otherwise.} \end{cases} \tag{2}$$

Here $0 \leq \epsilon \leq 1$. The introduction of this step has been motivated by the so called *data processing inequality* (DPI) (Cover and Thomas, 1991). The DPI is a relation between mutual information values, which means loosely that a post-processing of data cannot increase its information content. Specifically, one can show (Cover and Thomas, 1991) that the DPI for the following relation between the three random variables,

$$X \rightarrow Y \rightarrow Z, \tag{3}$$

implies that $I(X, Z) \leq I(X, Y)$. Due to the fact that the criteria in Equation 2 is for $\epsilon > 0$ less stringent than the DPI (Equation 3), ϵ is called *tolerance parameter*.

In order to ensure an unique solution that is independent of the order of the selected gene-triples, the procedure starts by listing all possible gene-triplets that can be found from the significant mutual information values after step one. Then, all of these gene-triplets are tested sequentially. Hence, the results of these tests have no influence on subsequent tests and the formation of gene-triplets.

For our practical application of Aracne, we use the standalone java executable Aracne2 (Basso et al., 2005; Margolin et al., 2006) available from (http://wiki.c2b2.columbia.edu/califanolab/index.php/Software/Aracne) to infer a GRN. For Aracne, we use the recommended parameter settings for this data set, listed in the following: For the mutual information estimator a kernel width of $w = 0.12918$ is defined with $b = 6$ bins. The significance threshold for MI was $t = 0.064394$ with a p-value threshold of $p = 1.0e - 7$. Aracne considers the removal of indirect interactions between a triplet of genes by applying the *data processing inequality* (DPI) with a tolerance parameter that is set to $\epsilon = 0.15$.

2.3. EXPERIMENTAL INTERACTIONS: REFERENCE NETWORKS

We use a meta collection of protein-protein interactions provided by iRefIndex (Razick et al., 2008). iRefIndex gathers protein interactions from BIND, BioGrid, DIP, HPRD, IntAct, MINT, MPact, MPPI and OPHID. Uniprot and refseq Ids were converted to entrez gene symbols using the *org.Hs.eg.db* R package (Carlson, 2013). If an identifier could not be mapped directly to entrez identifiers, the HUGO gene symbol was used. The remaining identifiers that could not be directly mapped to entrez gene symbols were not used. The resulting undirected protein network we

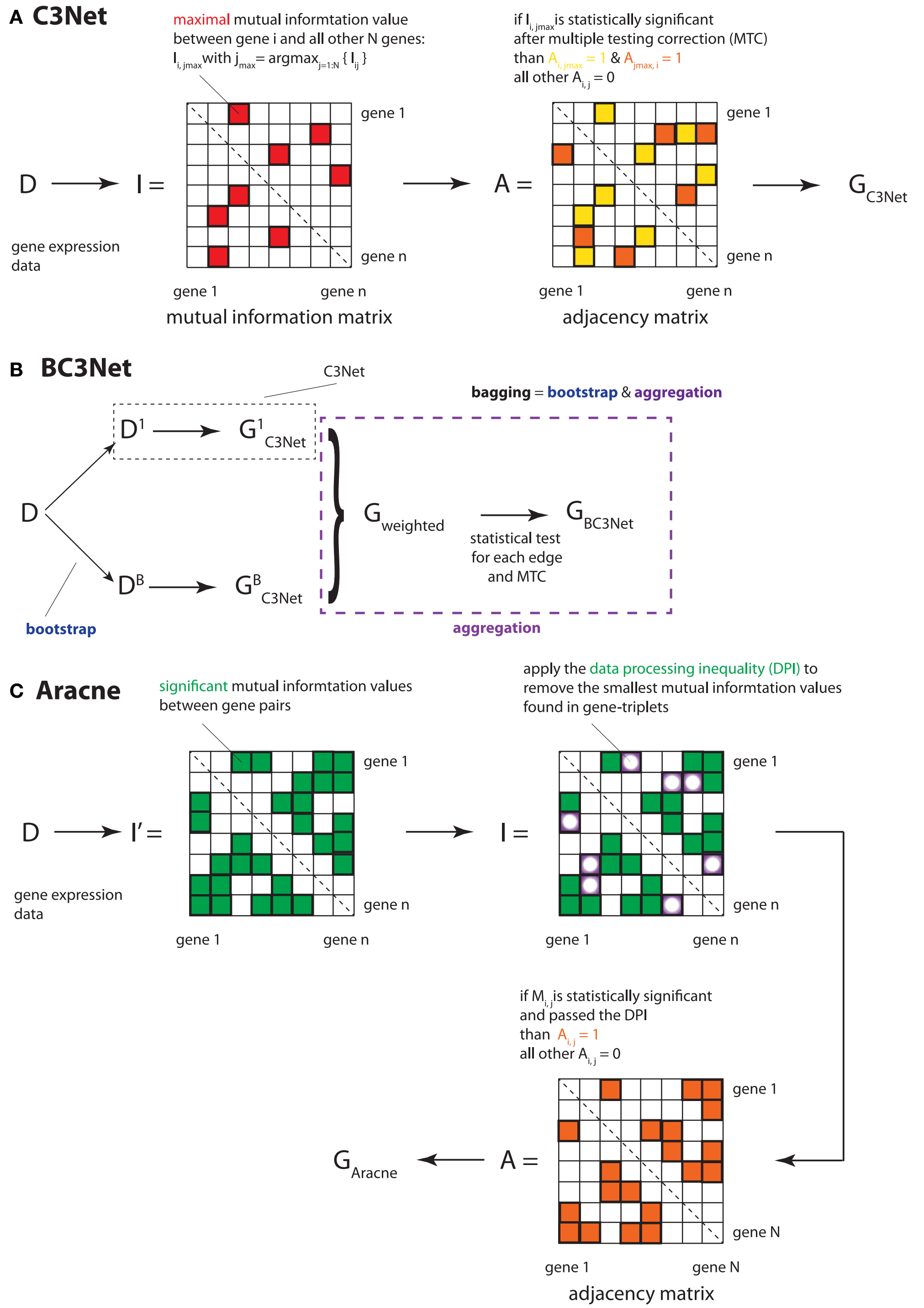

A C3Net
maximal mutual informtation value between gene i and all other N genes:
$I_{i,jmax}$ with j_{max} = $\text{argmax}_{j=1:N}$ { I_{ij} }
if $I_{i,jmax}$ is statistically significant after multiple testing correction (MTC) than $A_{i,jmax}$ = 1 & $A_{jmax,i}$ = 1 all other $A_{i,j}$ = 0
D
gene expression data
I =
gene 1
gene n
gene 1
gene n
mutual information matrix
A =
gene 1
gene n
gene 1
gene n
adjacency matrix
G_{C3Net}
B BC3Net
C3Net
bagging = bootstrap & aggregation
D^1
G^1_{C3Net}
D
D^B
G^B_{C3Net}
bootstrap
$G_{weighted}$
statistical test for each edge and MTC
G_{BC3Net}
aggregation
C Aracne
significant mutual informtation values between gene pairs
apply the data processing inequality (DPI) to remove the smallest mutual informtation values found in gene-triplets
D
gene expression data
I′ =
gene 1
gene n
gene 1
gene n
I =
gene 1
gene n
gene 1
gene n
if $M_{i,j}$ is statistically significant and passed the DPI than $A_{i,j}$ = 1 all other $A_{i,j}$ = 0
G_{Aracne}
A =
gene 1
gene N
gene 1
gene N
adjacency matrix

FIGURE 1 | Overview of the 3 applied inference methods and their key methodological analysis steps. (A) C3Net, **(B)** BC3Net and **(C)** Aracne.

use for our analysis includes a total of 185, 433 protein-protein interactions for 15, 233 proteins.

Furthermore, we use a transcriptional regulatory network (TRN) provided by the HTRidb database comprising a collection of experimentally validated transcription factor target gene interactions (Bovolenta et al., 2012). The database comprises a total of 51, 871 interactions for 284 transcription factors, regulating 18, 302 genes.

In the results section, we use these two experimental networks as *reference networks* to compare them with the inferred GRNs.

2.4. NETWORK CENTRALITY MEASURES

In the following, we describe 4 network-based measures we use for our analysis, namely, (A) degree centrality, (B) edge density, (C) transitivity and (D) assortativity. For a more detailed description see (Newman, 2010; Emmert-Streib and Dehmer, 2011).

The (A) degree centrality is defined as the total number of direct neighbors of a vertex v_i (gene). Formally, the degree centrality of v_i in an undirected network is given by Newman (2010),

$$C_1(v_i) = \sum_{j=1}^{n} A_{ij}, \tag{4}$$

where the adjacency matrix of the network is given by A and n is the total number of genes. That means $C_1(v_i)$ of node v_i is just the number of connections that node v_i has to other nodes in the network. Frequently, this is briefly called the *degree* of a node.

The (B) edge density of a network is the number of edges divided by the maximal number of possible edges. For an undirected network, this number of possible edges is given by $n(n-1)/2$, whereas n is the total number of genes. Hence, the edge density is a global measure for the connectivity of a network, whereas small values indicate sparsely connected networks and high values indicate densely connected networks.

The (C) transitivity centrality value of a vertex v_i, also called the local clustering coefficient, measures the proportion of edges of the direct neighbors of v_i in a clique of k vertices. The local clustering coefficient is given by Watts and Strogatz (1998),

$$C_3(v_i) = \frac{2|\{e_{ij}\}|}{k(k-1)}, \tag{5}$$

where $|\{e_{ij}\}|$ is the number of edges for vertex v_i to all its direct neighbors v_j, and $\frac{k(k-1)}{2}$ corresponds to the total number of edges in a clique of k vertices. Formally, the transitivity of a vertex is that probability that two neighbors of this vertex are connected with each other. Informally, this can be translated in "friends of mine are friends too," if a "friend" is defined as "connected with."

Finally, the (D) assortativity measure is the Pearson correlation coefficient of the degree centrality between the connected vertices in a network (Newman, 2002),

$$r = \frac{M^{-1}\sum_i j_i k_i - \left[M^{-1}\sum_i \frac{1}{2}(j_i + k_i)\right]^2}{M^{-1}\sum_i \frac{1}{2}\left(j_i^2 + k_i^2\right) - \left[M^{-1}\sum_i \frac{1}{2}\left(j_i + k_i\right)\right]^2}. \tag{6}$$

Here j_i and k_i correspond to the degrees of the vertices at the end of edge i, and M is the total number of edges in the network. We would like to remark that Equation 6 is symmetric in j_i and k_i. Informally, the assortativity is a global measure that gives positive values when—in average—vertices connect to other vertices that have a *similar* degree (e.g., high to high and low to low), and it has negative values—in average—when vertices connect to other vertices that have a *dissimilar* degree.

2.5. DEGREE CENTRALITY PATHWAY ANALYSIS

We define the test statistic δ, as the average degree centrality in the GRN, for a set of k genes defined by a Gene Ontology term. For an undirected network, δ is given by,

$$\delta_{obs} = \frac{1}{k}\sum_{i=1}^{k}\left(\sum_{j=1}^{n} A_{ij}\right), \tag{7}$$

where A is again the adjacency matrix of the network.

For each gene set (resulting from a GO term), the null distribution of δ is obtained from randomizations of the gene labels in the GRN. The p-value is estimated from the fraction of randomizations with a larger value than the test statistic, δ_{obs}, for a given term in the GRN, i.e.,

$$p = P\left(\delta \geq \delta_{obs}\right). \tag{8}$$

For each GO term, $R = 10,000$ randomizations are performed. We set the p-value to $p = 0.0001 = 1/R$ in cases when none of the randomizations exceed the test statistic for a given term. We perform a multiple hypothesis correction using the FDR by Benjamini and Hochberg (1995).

2.6. DRUGBANK

For the major hub genes in a GRN, we tabulated the associated drugs from the drugbank database (Knox et al., 2011). We use the drugbank version from july 2013. The drug to target protein links were extracted from *all_target_ids_all.csv* and the drugnames from *drug_links.csv*. We map uniprot identifiers to entrez gene symbols using org.Hs.eg.db R-package (Carlson, 2013).

3. RESULTS

3.1. GLOBAL PROPERTIES OF GENE REGULATORY NETWORKS

For the B-cell lymphoma gene expression data set in Basso et al. (2005), we infer 3 GRN using C3Net, BC3Net and Aracne. For the 3 inferred networks, we estimated the edge-density, maximal node degree, size of the giant component (GCC), assortativity and transitivity (**Table 1A**). Here, the GCC is the largest subnetwork and its size corresponds to the number of genes in this subnetwork.

As one can see from **Table 1A**, for these global measures the networks differ considerably for all measures. Specifically, the C3Net GRN has the lowest edge density (1.9×10^{-5}) and it is composed of 463 separated network components (subnetworks). In contrast, the Aracne GRN has the highest edge density (6.8×10^{-3}) followed by the BC3Net GRN (1.2×10^{-3}). The assortativity coefficient shows a weak negative correlation for C3Net indicating a tendency that, e.g., genes with a high degree have a tendency to be connected with genes with a low degree. For BC3Net and Aracne this cannot be observed.

From a pairwise comparison of the 3 GRNs in **Table 1B**, we find that the C3Net GRN is a subnetwork of BC3Net and Aracne, with almost all edges (99%) represented in both networks (see **Table 1B**). Also the BC3Net and the Aracne GRN show a large overlap with over 91.11% (52, 777/57, 905) of common edges that are present in BC3Net. In contrast, only 16.46% (52, 777/320, 668) of the common edges are present in the Aracne GRN.

Table 1 | (A) Global network properties of the B-cell lymphoma C3Net, BC3Net and Aracne GRN. (B) Edge-overlap between the 3 GRN.

	C3Net	BC3Net	Aracne
(A)			
Number of genes	9684	9684	9684
Number of edges	9221	57, 905	320, 668
Edge-density	1.9×10^{-5}	1.2×10^{-3}	6.8×10^{-3}
Max degree	46	169	2198
Number of components	463	8	1
Size of GCC	884	9668	9684
Assortativity	−0.144	−0.0195	0.0543
Transitivity	0.089	0.000	0.230
(B)			
C3Net	9221 (100%)	9215 (99.93%)	9167 (99.41%)
BC3Net	9215 (15.91%)	57, 905 (100%)	52, 777 (91.11%)
Aracne	9167 (2.86%)	52, 777 (16.46%)	320, 668 (100%)

(A, B) For this table, we compare the number of edges in both networks divided by the total number of edges of the network in the row.

Table 2 | (A) Functional enrichment using a GPEA for the C3Net, BC3Net, and Aracne GRN. Shown are the numbers of significant terms/number of total terms, and the percentage of significant terms. (B) Overlap percentage (%) of significant terms in the GPEA between the C3Net, BC3Net and Aracne gene regulatory networks.

	C3Net%	BC3Net%	Aracne%	
(A)				
BP	124/1673 (7.4)	166/2604 (6.3)	386/3565 (10.8)	
CC	30/241 (12.4)	49/357 (13.7)	110/477 (23.1)	
MF	8/308 (2.6)	25/535 (4.7)	38/774 (4.9)	
Reactome	92/270 (34.1)	129/387 (33.3)	186/492 (37.8)	
(B)				
C3Net vs. BC3Net	92.74	96.67	87.50	96.74
C3Net vs. Aracne	92.74	100.00	87.50	96.74
BC3Net vs. Aracne	97.59	97.96	64.00	99.22

On a general note, we would like to add that the differing number of edges in the 3 inferred GRN is related to the different inference methods applied (see Methods section). Whereas C3Net aims only to infer the interactions within a GRN that are strongest, as emphasized by its name (Conservative Causal Core = C3), BC3Net is a bagged (Breiman, 1996) version of C3Net that is capable of exploiting also less strong signals by estimating their variability from an ensemble approach. Finally, Aracne employs an entirely different inference strategy than C3Net or BC3Net. Whereas C3Net aims only to infer the strongest interactions and BC3Net aims to *add* additional interactions by bagging C3Net, Aracne uses the *data processing inequality* to thinning all significant mutual information values. Hence, C3Net is the most conservative approach, Aracne is the most anti-conservative approach and BC3Net is situated in-between them.

The results in **Table 1** indicate clearly that the 3 GRNs are considerably different among each other, if compared with global measures.

3.2. FUNCTIONAL ANALYSIS OF B-CELL LYMPHOMA NETWORKS

Next, we investigate the functional similarity of the 3 GRNs. In order to identify the most prominently represented biological processes in the 3 B-cell lymphoma GRN, we perform a *gene pair enrichment analysis* (GPEA). The GPEA analysis tests the null hypothesis whether the number of interactions in a GRN connecting genes from the same GO term is similar to the number of interactions connecting genes from different GO terms. This is tested by a hypergeometric test.

We perform the GPEA using gene sets, defined by the Gene Ontology database, for the categories biological process (BP), molecular function (MF) and cellular component (CC). In addition, we use terms defined in the reactome database. Furthermore, we compare the results obtained for the C3Net, BC3Net and Aracne gene regulatory networks among each other.

In **Table 2A**, we show an overview of the number of significant terms identified using the GPEA. For example for GO BP, we find 124 significant terms for C3Net, 166 significant for BC3Net and 386 significant terms for Aracne. The number of significant terms is similar between C3Net and BC3Net. For the Aracne GRN, the number of significant terms is almost twice as large. The number of significant terms for the reactome is similar for all three networks comprising a total of 30% of the terms. For MF the number of significant terms is the lowest for the three networks comprising only 5% of the terms. **Table 2B** shows the overlap of significant terms for BP, MF, CC and reactome between C3Net, BC3Net, and Aracne. For all pairwise comparisons, we observe an overlap of >90% of significant terms between pairs of GRNs, except for MF.

Another interesting observation we make is that the rank-order of significant GO terms is highly correlated between C3Net and BC3Net ($r = 0.88$, $p \leq 2.2 \times 10{-}16$), but also the other two pairs of GRNs. For instance, **Figure 2** shows a pairwise comparison of the rank-order of the GPEA analysis for BP terms between BC3Net and Aracne, whereas the topmost 25 pairs are highlighted in blue. That means for each network, we rank-ordered the analyzed GO terms according to their resulting *p*-values and we used these ranks as x-coordinates (Aracne) and y-coordinates (BC3Net) in **Figure 2**. On a technical note, we want to remark that we used logarithmically transformed (log-transformed) values to obtain a better visualization of the shown GO terms. However, because a logarithm is a monotonous function, the original rank-order of the GO terms remains unchanged by this transformation.

Biologically, from the top 25 BP GO terms in **Figure 2** we observe a variety of significant biological processes for protein translation, targeting and protein complex disassembly, viral transcription and cell cycle. Interestingly, in contrast to the results from the global analysis of GRNs, the functional analysis indicates that all 3 GRNs are biologically quite similar to each other.

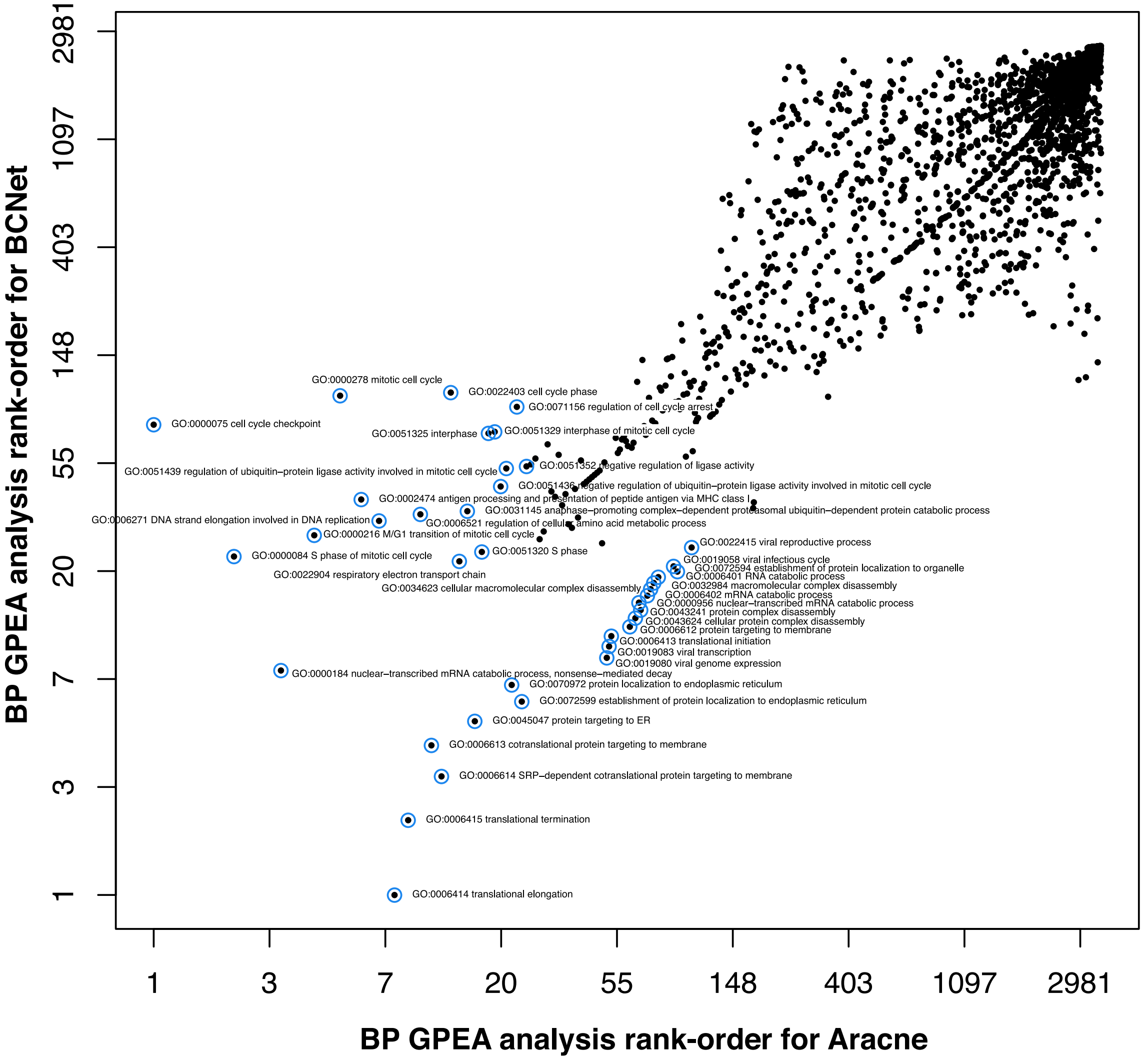


FIGURE 2 | Comparison of the rank-order of significant biological process (BP) GO terms from the GPEA analysis for BC3Net (*y*-axis) and Aracne (*x*-axis). The axis are log-transformed for a better visualization. The blue circles correspond to the top GO terms for Aracne and BC3Net.

3.3. EXPERIMENTAL INTERACTIONS: COMPARISON WITH REFERENCE NETWORKS

In this section, we compare the 3 inferred GRNs with experimental networks (serving as reference networks). Specifically, we use a protein-protein interaction network (PPN) and a transcriptional regulatory network (TRN) for this comparison. The transcriptional regulatory network is obtained from the HTRIdb database of experimental validated transcription factor target gene interactions (Bovolenta et al., 2012). The database comprises a total of 284 transcription factors and 18, 302 target genes comprising a total of 51, 871 interactions. The PPN is from iRefIndex containing a total of 185, 433 protein-protein interactions among 15, 233 proteins; see the Methods section for more details.

An overview of the pairwise comparisons between the B-cell lymphoma GRNs and the TRN is shown in **Table 3A** and the comparison with the PPN is shown in **Table 3B**. The percentage of shared interactions for all 3 GRNs is very low, and ranges around 0.1%. However, only for C3Net the number of shared interactions with the TRN is significant. For the comparison between the PPN and the inferred GRNs the number of shared interactions is significant for all three GRNs and the percentage of shared interactions is in the range between 1% to 2%. Again for C3Net we observe the highest overlap of edges between the GRN and the TRN.

3.3.1. Correlation between the degree centrality of the GRNs and the reference networks

In this section, we study the correlation between the degree centrality value of genes that we find in the GRNs and the experimental reference networks, i.e., the TRN and the PPN, using the Pearson correlation coefficient. Specifically, we start with the topmost 25 genes in these networks and then increase the number of the genes sequentially in step sizes of 25 genes, until all genes are included. This corresponds to an averaging window with one fixed side and one sliding side that increases in steps of 25 genes to lower ranked genes. The results of this analysis are shown in **Figure 3**. In this figure, the gray area indicates correlation values that would *not* be statistically significant for a significance level of $\alpha = 0.05$. In other words, all correlation values that are outside the gray area, are statistically significant. We obtained the values of the significance boundaries from using an assymtotic relation between a t-statistic, t, and a Pearson correlation coefficient, r, given by Sheskin (2004)

$$r = \frac{t}{\sqrt{df + t^2}}. \tag{9}$$

Here $df = n - 2$ is the degree of freedom of the data for a profile vector of length n. It is important to note that also a t-statistic is a function of the degree of freedom, i.e., $t(df)$. From selecting a significance level α one obtains for each profile vector of a certain length, n, the corresponding t-statistics, which gives via Equation 9 the corresponding values for the Pearson correlation coefficients. For our analysis we assumed a two-sided hypothesis explaining the symmetric values of the correlation around zero. As one can see from **Figure 3**, due to increasing sizes of the profile vectors for which correlations are assessed, these decision boundaries are not constant but are becoming narrower around zero when more genes are used in the analysis.

For all three GRNs and the PPN, we observe a tendency for the high-degree genes to show a statistically significant negative correlation to the degrees observed in the PPN (~ -0.2, **Figure 3A**). For larger window sizes, the correlation slowly decreases for C3Net and BC3Net, but much faster for Aracne. Interestingly, C3Net assumes positive statistically significant correlation values ($\sim$0.05) for very large window sizes.The observations for the comparison between the three GRNs and the TRN are similar, however, less strong (see **Figure 3B**). In this case, all three GRN inference methods C3Net, BC3Net and Aracne retain their negative statistically significant correlation values, even for very large window sizes.

Table 3 | Network comparison between the 3 B-cell lymphoma GRNs and the (A) TRN and (B) PPN.

(A) Transcriptional regulatory network (TRN)					
	Shared genes (%)	**Edges GRN**	**Edges TRN**	**Shared edges%**	***p*-value**
C3Net	7915	6361	22,668	8 (0.125)	0.045
BC3Net	7915	39, 507	22, 668	33 (0.084)	0.176
Aracne	7915	210, 036	22, 668	134 (0.064)	0.923
(B) Protein protein network (PPN)					
	Shared genes (%)	**Edges GRN**	**Edges PPN**	**Shared edges%**	***p*-value**
C3Net	7944	6429	100,074	145 (2.226)	0
BC3Net	7944	40,049	100,074	563 (1.406)	0
Aracne	7944	213,841	100,074	2110 (0.987)	0

Here, shared genes and shared edges correspond to the genes and edges that can be found in a GRN and the (A) TRN and (B) PPN.

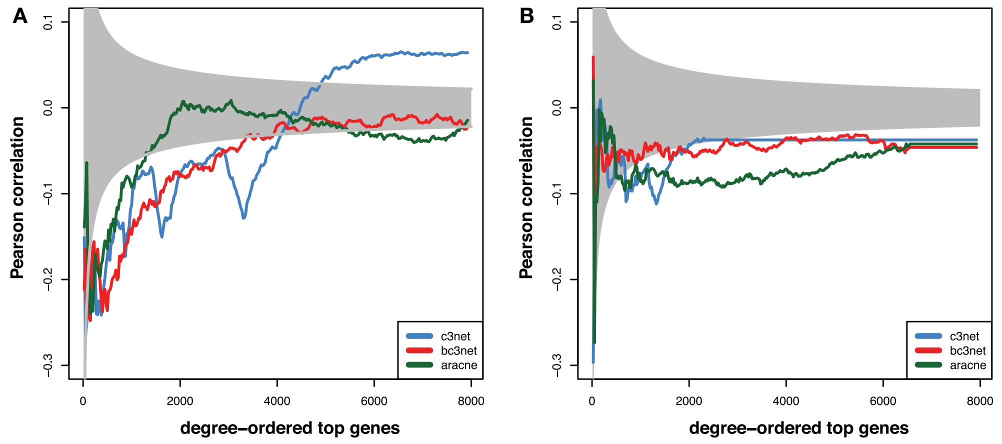


FIGURE 3 | Shown are Pearson correlation coefficients that are obtained for genes that are rank-ordered according the size of their degree centrality values in the GRNs compared to: (A) PPN and (B) TRN. The ordering is from high to low degree centrality values and the size of the underlying profile vectors increases with the number of the gene rank. The gray area indicates correlation values that are not statistically significant for $\alpha = 0.05$.

3.4. HUB GENES

In this section, we study the major hub genes in the B-cell lymphoma GRNs inferred from C3Net, BC3Net and Aracne. Furthermore, we conduct a functional analysis to elucidate the role of the involved biological processes of the major hub genes.

We start by performing a global comparison of the degree centrality values for all genes between the C3Net, BC3Net and Aracne GRN. The largest global rank-order Spearman correlation coefficient for all genes is observed between C3Net and BC3Net ($r = 0.72$, $p < 2.2 \times 10^{-16}$) and for BC3Net and Aracne ($r = 0.74$, $p < 2.2 \times 10^{-16}$). The lowest correlation is observed between C3Net and Aracne ($r = 0.54$, $p < 2.2 \times 10^{-16}$). In **Figure 4**, we show the pairwise comparison between the rank-order of the degree centrality of all genes for BC3Net and Aracne. Interestingly, we observe no substantial difference between the degree rank of genes with the highest node degree. This holds for all pairwise comparisons between the three GRNs. That means all 3 GRNs contain essentially the same hub genes.

In general, the connection between hub genes for different physiological contexts, as represented by a protein network (PPI) or a GRN, is not well studied. In a PPI network, lethal proteins have been observed to have the tendency to form hubs (Jeong et al., 2001), whereas non-lethal disease associated proteins, which are putative drug targets, are more likely to reside at the periphery of a PPI network (Goh et al., 2007). Interestingly, in contrast to a PPI network, the GRN hub genes of the B-cell lymphoma GRNs have the tendency to be associated with signaling receptors, such as from the G-protein coupled receptor pathway that comprises promising drug targets in cancer (Lappano and Maggiolini, 2011).

We would like to note that the hub genes of the B-cell lymphoma GRNs, see **Figure 4**, are not restricted to signaling receptors and can also include a variety of transcription factors such as, e.g., ZIC2 and ELAVL2 (HuB). Although, the literature does not show studies investigating these genes specifically for B-cell lymphoma, several studies point to their importance for the development of tumors. For instance, ZIC2 was observed with a higher expression in malignant ovarian tumors (Marchini et al., 2012) and overexpression analysis showed oncogenic properties of ZIC2 to drive tumor growth in ovarian cancer (Marchini et al., 2012). Also, proteins of the ELAV gene family (Hu genes) such as ELAVL2 are tumor antigens that are investigated for early stage lung cancer detection (D'Alessandro et al., 2010). Hu genes are usually expressed in neuron cells and were found to have an

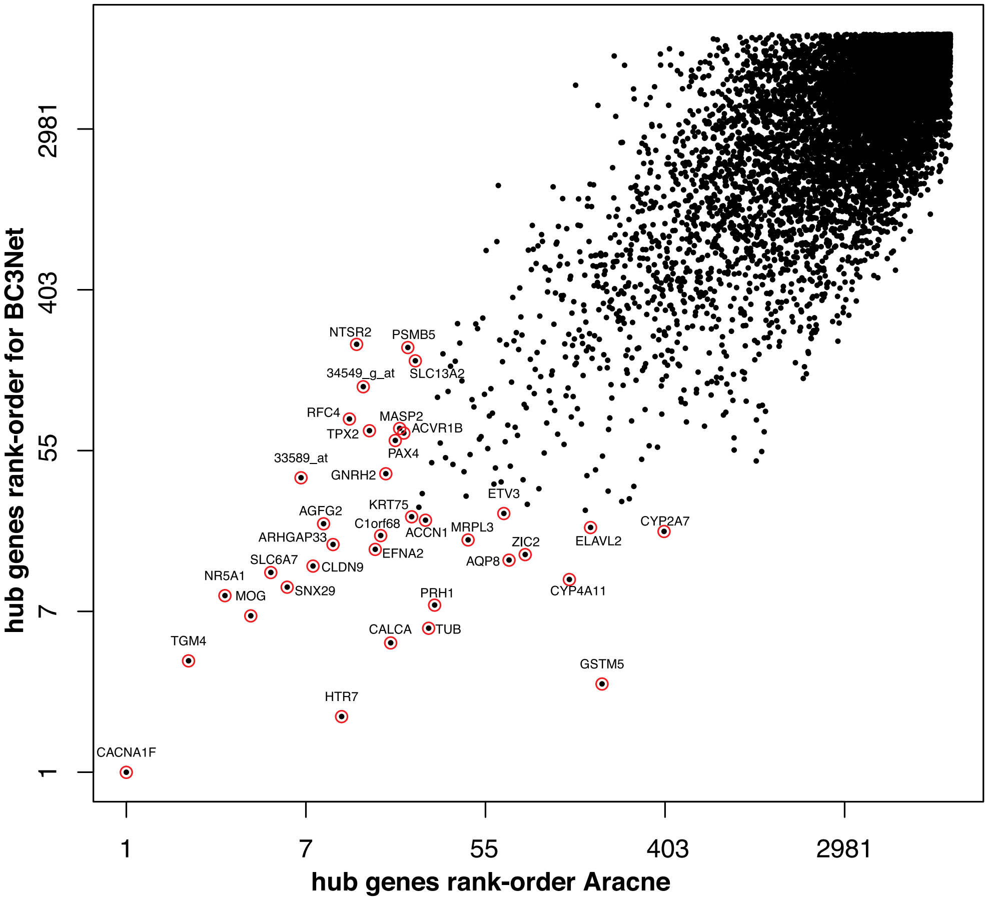


FIGURE 4 | Comparison of the rank-order of hub-genes for BC3Net (y-axis) and Aracne (x-axis). The axis are log-transformed for a better visualization. The red circles correspond to the top hub genes for Aracne and BC3Net.

Table 4 | Results for the degree centrality pathway analysis test for the BC3Net GRN.

GO	Term	δ_{obs}	δ (avg)	Size	*p*-value	FDR
GO:0007188	Adenylate cyclase-modulating G-protein coupled receptor signaling pathway	17.5	11.95	86	0.0001	0.02877
GO:0007267	Cell-cell signaling	13.73	11.96	776	0.0001	0.02877
GO:0007600	Sensory perception	14.69	11.95	263	0.0001	0.02877
GO:0009581	Detection of external stimulus	21.06	11.95	52	0.0001	0.02877
GO:0009582	Detection of abiotic stimulus	21.77	11.96	47	0.0001	0.02877
GO:0009583	Detection of light stimulus	25.27	11.99	26	0.0001	0.02877
GO:0050877	Neurological system process	13.45	11.96	772	0.0001	0.02877
GO:0051320	S phase	16	11.95	119	0.0001	0.02877
GO:0007187	G-protein coupled receptor signaling pathway, coupled to cyclic nucleotide second messenger	16.6	11.96	111	0.0001	0.02877
GO:0007601	Visual perception	16.24	11.97	132	0.0001	0.02877
GO:0008217	Regulation of blood pressure	15.92	11.97	110	0.0001	0.02877
GO:0050906	Detection of stimulus involved in sensory perception	23.46	11.98	26	0.0001	0.02877
GO:0050953	Sensory perception of light stimulus	16.24	11.96	132	0.0001	0.02877
GO:0003073	Regulation of systemic arterial blood pressure	18.6	11.96	50	0.0002	0.04675
GO:0051606	Detection of stimulus	17.45	11.95	84	0.0002	0.04675
GO:0000216	M/G1 transition of mitotic cell cycle	17.85	11.95	65	0.0002	0.04675
GO:0007189	Adenylate cyclase-activating G-protein coupled receptor signaling pathway	19.51	11.96	41	0.0003	0.06233
GO:0000084	S phase of mitotic cell cycle	15.96	11.96	113	0.0003	0.06233
GO:0045649	Regulation of macrophage differentiation	32	11.92	10	0.0004	0.07874

ectopic expression in neurodendocrine tumors (Gultekin et al., 2000). However, the association between tumor progression and Hu gene expression remains unclear on the molecular level.

For the functional analysis of the GRN hub genes, we applied a non-parametric test to identify biological processes that are related to genes with a large degree centrality value in the GRN. We perform a permutation-based test that defines the average degree centrality from the GRN as test statistic for the gene set of a given GO term; see Equation 7. As a result, **Table 4** shows the most significant biological process terms with the highest average degree centrality (δ_{obs}) in the GRN (with FDR$\leq$0.1). We observe a large variety of signaling related processes such as adenylate cyclase-modulating G-protein coupled receptor signaling pathway, cell-cell signaling, sensory perception, cell cycle processes (S phase), blood pressure regulation and macrophage differentiation.

We further studied whether major hub genes of a GRN are drugable by known drugs that are related to the treatment of B-cell lymphoma. For the 30 genes with the largest degree centrality in the B-cell lymphoma BC3Net we extracted associated drugs from the drugbank database (Knox et al., 2011). A variety of drugs were associated with 8 genes comprising calcium-channel blockers (calcium channel subunit CACNA1F), dopamine antagonists (serotonin receptor HTR7), metabolic compounds such as glutathione, NADH, L-proline and pituitary hormone analogues, see **Table 5** for an overview.

3.4.1. MYC

The study of Basso et al. (2005) provided a validation for some interactions of the transcription factor MYC. However, when considering the degree centrality values of MYC in the inferred networks, MYC has a low rank-order. Interestingly, this is consistent for all three inferred GRNs and holds also for the ranking of other network-based measures. In **Table 6**, the rank of MYC is shown for C3Net, BC3Net and Aracne (in decreasing order of the absolute degree value) for the degree centrality, betweenness and local transitivity. For example, MYC ranks for the degree centrality of the GRN for C3Net 3110 (9684), BC3Net 9322 (9684) and Aracne 2317 (9684). Here, the number in brackets corresponds to the total number of genes. In the C3Net GRN, we find that MYC has only one single direct neighbor. In the BC3Net GRN, MYC has 4 direct neighbors, namely, POLD2 (52 neighbors), NME1 (26 neighbors), SRM (30 neighbors), NINL (13 neighbors) (**Figure 5**). For the Aracne GRN, MYC has 68 direct neighbors. The direct neighbors of C3Net and BC3Net are also present in the Aracne GRN.

4. DISCUSSION

In this paper, we conducted a structural and a functional analysis of B-cell lymphoma GRNs that were inferred using 3 mutual information-based inference methods, namely, C3Net (Altay and Emmert-Streib, 2010), BC3Net (de Matos Simoes and Emmert-Streib, 2012) and Aracne (Basso et al., 2005). On the global-level, our analysis revealed that the inferred B-cell lymphoma GRNs have major differences in their edge density, maximal degree, transitivity and assortativity. However, on the edge-level, the 3 GRNs were highly similar among each other, whereas the C3Net GRN and the BC3Net GRN represent almost a subnetwork of the Aracne GRN. The global differences in the edge densities can be mainly explained by the different inference strategies employed

Table 5 | Drug targets for major hub genes in the BC3Net B-cell lymphoma gene regulatory network, see Figure 4.

Target gene	Drugbank	Drugname
CACNA1F	DB00393	Nimodipine
	DB00568	Cinnarizine
	DB00661	Verapamil
	DB01388	Mibefradil
	DB04855	Dronedarone
	DB04920	Clevidipine
HTR7	DB00216	Eletriptan
	DB00246	Ziprasidone
	DB00247	Methysergide
	DB00248	Cabergoline
	DB00334	Olanzapine
	DB00363	Clozapine
	DB00751	Epinastine
	DB01200	Bromocriptine
	DB01224	Quetiapine
	DB01238	Aripiprazole
	DB04946	Iloperidone
	DB06216	Asenapine
	DB06288	Amisulpride
	DB08815	Lurasidone
GSTM5	DB00143	Glutathione
TUB	DB02028	DB02028
NR5A1	DB04683	DB04683
	DB04752	Phosphatidyl ethanol
CYP4A11	DB00157	NADH
SLC6A7	DB00172	L-Proline
AVPR1B	DB00035	Desmopressin
	DB02638	Terlipressin

Table 6 | MYC rank (in decreasing order) for degree centrality, betweeness and transitivity for the C3Net, BC3Net, and Aracne GRN.

GRN	Degree rank	Betweeness rank	Transitivity rank
C3Net	3110 (1)	6955 (0)	1936 (0)
BC3Net	9322 (4)	2184 (4142.263)	3968 (0)
Aracne	2317 (68)	3397 (7778)	712 (0.23)

The absolute values of the centrality measure are shown in parenthesis. In total, the maximal rank order is 9684 corresponding to the total number of genes.

by the three methods (see Methods section) resulting always in GRNs with the following ordering in the number of edges; $\#\text{edges}_{\text{C3Net}} < \#\text{edges}_{\text{BC3Net}} < \#\text{edges}_{\text{Aracne}}$.

A C3Net GRN represents the *core* network structure of a GRN that considers only the strongest signal from a data set (Altay and Emmert-Streib, 2010). Although C3Net limits the analysis to a very sparse GRN structure, it provides a less complex and more clearly arranged structural organization of large GRN networks (de Matos Simoes et al., 2012). Furthermore, because only the strongest gene neighbors are considered for each gene, using C3Net or BC3Net, the number of putative indirect associations is highly reduced.

In our study, we compared also the biological functions that are significantly represented in GRNs. We observed high similarities between the GRN of C3Net, BC3Net and Aracne, where significant biological processes, cellular components and Reactome terms were overlapping with >90%. The tendency for Aracne to observe a larger number of significant terms for the GPEA analysis can be explained due to the larger edge-density that is beneficial for a GPEA analysis. Among the significant terms common in all 3 GRNs, we find biological processes for protein translation, targeting and protein complex disassembly, viral transcription and cell cycle.

Next, we compared the 3 inferred GRNs with 2 experimental networks (we called reference networks). Specifically, we compared the 3 GRNs with a protein-protein interaction network (PPN) and a transcriptional regulatory network (TRN). From this comparison, we determined the quantitative edge-overlap between the 2 reference networks and the 3 B-cell lymphoma GRNs. For the TRN, we observed $\sim 0.1\%$ of shared interactions with the GRNs. However, for the PPN, we observed a higher relative percentage of $1 - 2\%$ of shared interactions. A reason for this low, but significant (see *p*-values in **Table 3**), overlap is three-fold. First, the used reference networks are not *condition specific* for B-cell lymphoma. For instance, many interactions in the PPN are obtained from *yeast-two-hybrid* (Y2H) experiments providing only information about the potential binding of proteins outside a particular cellular context (Maslov and Sneppen, 2002). Similarly, the experimentally verified interactions in the TRN provided by the HTRidb database are identified from a wide range of different normal (not pathological) physiological conditions. Second, a GRN provides only an average representation of the interactions across the spatial and temporal separation of the cellular processes that are reflected by the observed gene expression dependencies. Third, due to the different data types used to assemble a PPN (e.g., Y2H), TRN (e.g., ChIP-chip) and a GRN (gene expression) they are all different from each other. The relation between these networks has been studied systematically for the model organisms *S. cerevisiae* and *E.coli* in de Matos Simoes et al. (2013).

We compared the degree centrality of the GRNs to the PPN and TRN. For the PPN and the TRN, we observed a statistically significant negative correlation for the genes with the largest degree centrality, independent of the GRN inference method. That means, the major hub genes of a GRN have a tendency to relate to proteins with a low(er) degree in the PPN or TRN. This analysis suggest that proteins with few direct neighbor interactions have a stronger relationship in gene expression data for the corresponding genes that are connected in a GRN, which may more likely represent the periphery of the gene network. However, one major limitation of defining the degree centrality from a PPN network is that protein interactions are not well defined and gathered from multiple experimental methods for different interaction types that are not distinguished and largely incomplete.

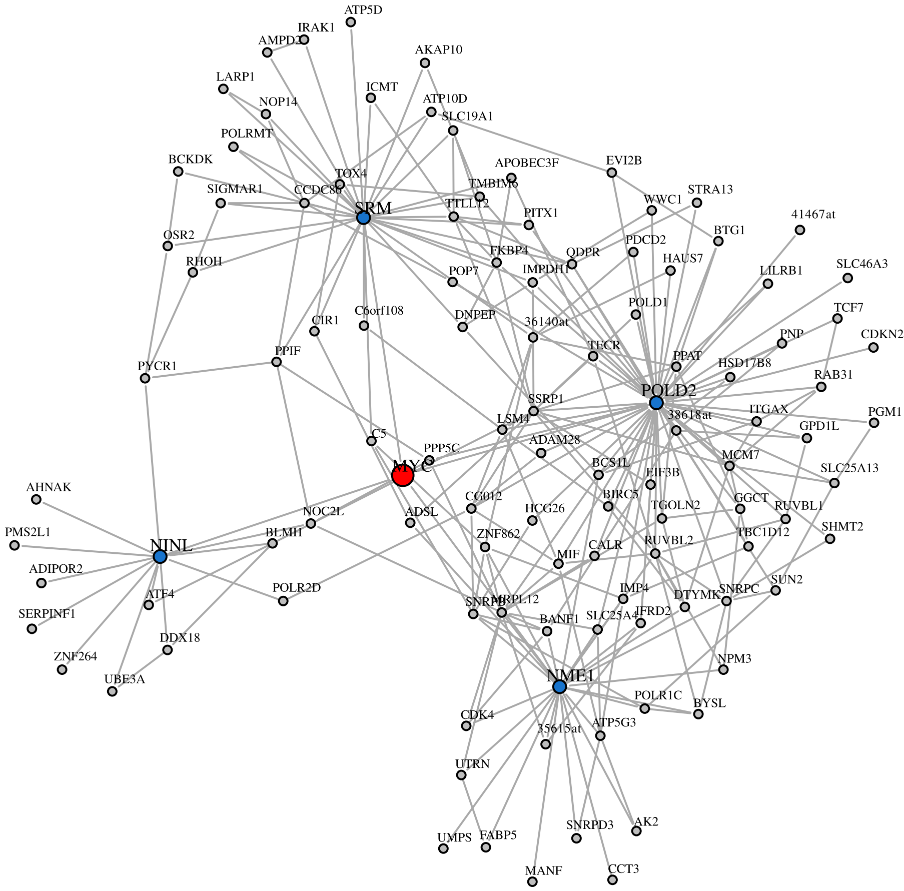


FIGURE 5 | BC3Net subnetwork including MYC (red) and its 2nd level nearest neighbors. The first degree MYC neighbors are shown in blue and the 2nd degree MYC neighbors are shown in gray.

In a PPI network lethal proteins have been observed to have a tendency to form hubs (Jeong et al., 2001), whereas non-lethal disease associated proteins, which are putative drug targets, are more likely to reside at the periphery of a PPI network (Goh et al., 2007). Our functional analysis to identify pathways with a significantly larger average degree centrality revealed pathways involved in the G-protein coupled receptor signaling pathway, sensory perception, cell-cell signaling and cell cycle. G-protein coupled receptors are prominent drug targets for a large catalogue of conditions such as cardiovascular related and neuropsychiatric disorders (Esposito et al., 2002; Albizu et al., 2010) and promising drug targets in cancer (Lappano and Maggiolini, 2011).

For example, the major hub gene CACNA1F, see **Table 5**, can be inhibited by a variety of channel blockers like nifedipine, amlodipine, verapamil, and diltiazem (Striessnig et al., 2010). Due to the importance of ion channels in signaling the calcium channel blockers are also being investigated for the treatment of B-cell lymphoma (Shamash et al., 1998). For CACNA1F 6 calcium channel blocking drugs were identified from drugbank. The combination of verapamil and antineoplastic agents is suggested to induce chemosensitivity in chemoresistant cells (Simpson, 1985). Furthermore, mibefradil was shown to slow tumor growth in glioblastoma cell lines (Keir et al., 2013). The serotonin receptor HTR7 is a G-protein coupled receptor. The drugs associated to HTR7 are dopamine antagonists used for neuropsychiatric disorders. GSTM5 is associated to Glutathione that is highly abundant and important for protecting the cell against free radicals, but also promote chemoresistance (Balendiran et al., 2004). A number of studies investigated the depletion of Glutathione following chemotherapy for increasing chemosensitization of cancer cells (Balendiran et al., 2004). Lastly, AVPR1B is associated to desmopressin which may impair metastasis of cancer cells (Gomez et al., 2006).

The hub genes of the B-cell lymphoma GRN are not restricted to signaling receptors and can also include transcription factors such as ZIC2 or RNA-binding proteins such as ELAVL2 (HuB). Although, the literature does not show studies investigating these genes specifically for B-cell lymphoma several studies point to their importance in tumorgenic processes. ZIC2 was observed with higher expression in malignant ovarian tumors (Marchini et al., 2012). Overexpression analysis showed oncogenic properties of ZIC2 to drive tumor growth in ovarian cancer (Marchini et al., 2012). Proteins of the ELAV gene family (Hu genes) such as ELAVL2 are tumor antigens that are investigated for early stage lung cancer detection (D'Alessandro et al., 2010). Hu genes are usually expressed in neuron cells and were found to have an ectopic expression in neurodendocrine tumors (Gultekin et al., 2000). However, the association between tumor progression and Hu gene expression remains unclear on the molecular level.

This discussion shows a potential application of the resulting GRNs. That means, major inferred hub genes could be used for the experimental validation of drugs to effect important biological pathways of B-cell lymphoma. In this way, data-driven hypothesis about drug targets could be derived from the inferred GRN (Ildirim et al., 2007; Hopkins, 2008; Ghosh and Basu, 2012). Additionally, in a similar way, hallmark pathways could be studied, because since the seminal work by Hanahan and Weinberg (2000, 2011) it is generally accepted that the molecular causes of cancer need to be approaches on this level, rather than on the level of individual genes.

Overall, our analysis sheds light on the biological similarity of GRNs inferred with C3Net, BC3Net and Aracne, and indicates that these network inference methods contain consistent biological information. This is a very important result, because it demonstrates the biological robustness of the information that can be reliably derived from such different GRNs, despite existing differences among various other aspects of such networks.

4.1. DATA SHARING

We provide the gene expression data, the inferred GRNs and the reference experimental networks from our analysis in the R-package BClymphomaGRN, available from CRAN.

ACKNOWLEDGMENTS

We would like to thank Shailesh Tripathi for fruitful discussions.

FUNDING

Matthias Dehmer thanks the Austrian Science Funds for supporting this work (project P22029-N13). Matthias Dehmer also gratefully acknowledges funding from the Standortagentur Tirol (formerly Tiroler Zukunftsstiftung).

REFERENCES

Albizu, L., Moreno, J., Gonzalez-Maeso, J., and Sealfon, S. (2010). Heteromerization of g protein-coupled receptors: relevance to neurological disorders and neurotherapeutics. *CNS Neurol. Disord. Drug Targets* 9, 636–650. doi: 10.2174/187152710793361586

Altay, G., and Emmert-Streib, F. (2010). Inferring the conservative causal core of gene regulatory networks. *BMC Syst. Biol.* 4:132. doi: 10.1186/1752-0509-4-132

Altay, G., and Emmert-Streib, F. (2011). Structural Influence of gene networks on their inference: analysis of C3NET. *Biol. Dir.* 6:31. doi: 10.1186/1745-6150-6-31

Balendiran, G., Dabur, R., and Fraser, D. (2004). The role of glutathione in cancer. *Cell Biochem. Funct.* 22, 343–352. doi: 10.1002/cbf.1149

Barrett, T., Troup, D., Wilhite, S., Ledoux, P., Evangelista, C., Kim, I., et al. (2011). NCBI GEO: archive for functional genomics data sets–10 years on. *Nucleic Acids Res.* 39, D1005–D1010. doi: 10.1093/nar/gkq1184

Basso, K., Margolin, A., Stolovitzky, G., Klein, U., Dalla-Favera, R., and Califano, A. (2005). Reverse engineering of regulatory networks in human B cells. *Nat. Genet.* 37, 382–390. doi: 10.1038/ng1532

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. B (Methodol.)* 57, 125–133.

Bovolenta, L., Acencio, M., and Lemke, N. (2012). HTRIdb: an open-access database for experimentally verified human transcriptional regulation interactions. *BMC Genomics* 13:405. doi: 10.1186/1471-2164-13-405

Breiman, L. (1996). Bagging Predictors. *Mach. Learn.* 24, 123–140. doi: 10.1007/BF00058655

Carlson, M. (2013). *org.Hs.eg.db: Genome wide annotation for Human*. R package version 2.9.0.

Cover, T., and Thomas, J. (1991). *Information Theory*. New York, NY: John Wiley & Sons, Inc.

D'Alessandro, V., Muscarella, L., la Torre, A., Bisceglia, M., Parrella, P., Scaramuzzi, G., et al. (2010). Molecular analysis of the HuD gene in neuroendocrine lung cancers. *Lung. Cancer* 67, 69–75. doi: 10.1016/j.lungcan.2009.03.022

Davison, A., and Hinkley, D. (1997). *Bootstrap Methods and Their Application*. New York, NY: Cambridge University Press. doi: 10.1017/CBO9780511802843

de Matos Simoes, R., Dehmer, M., and Emmert-Streib, F. (2013). Interfacing cellular networks of *S. cerevisiae* and *E. coli:* Connecting dynamic and genetic information. *BMC Genomics* 14:324. doi: 10.1186/1471-2164-14-324

de Matos Simoes, R., and Emmert-Streib, F. (2011). Influence of statistical estimators of mutual information and data heterogeneity on the inference of gene regulatory networks. *PLoS ONE* 6:e29279. doi: 10.1371/journal.pone.0029279

de Matos Simoes, R., and Emmert-Streib, F. (2012). Bagging statistical network inference from large-scale gene expression data. *PLoS ONE* 7:e33624. doi: 10.1371/journal.pone.0033624

de Matos Simoes, R., Tripathi, S., and Emmert-Streib, F. (2012). Organizational structure and the periphery of the gene regulatory network in B-cell lymphoma. *BMC Syst. Biol.* 6:38. doi: 10.1186/1752-0509-6-38

Dudoit, S., and van der Laan, M. (2007). *Multiple Testing Procedures with Applications to Genomics*. New York; London: Springer.

Efron, B., and Tibshirani, R. (1994). *An Introduction to the Bootstrap*. New York, NY:Chapman and Hall/CRC.

Emmert-Streib, F. (2013). Influence of the experimental design of gene expression studies on the inference of gene regulatory networks: environmental factors. *PeerJ* 1:e10. doi: 10.7717/peerj.10

Emmert-Streib, F., and Dehmer, M. (2011). Networks for Systems Biology: conceptual Connection of Data and Function. *IET Syst. Biol.* 5:185. doi: 10.1049/iet-syb.2010.0025

Emmert-Streib, F., Glazko, G., Altay, G., and de Matos Simoes, R. (2012). Statistical inference and reverse engineering of gene regulatory networks from observational expression data. *Front. Genet.* 3:8. doi: 10.3389/fgene.2012.00008

Esposito, G., Rapacciuolo, A., Prasad, S. V. N., and Rockman, H. A. (2002). Cardiac hypertrophy: role of g protein-coupled receptors. *J. Card. Fail.* 8(6 Part B), S409–S414. doi: 10.1054/jcaf.2002.129283

Friedman, N. (2004). Inferring cellular networks using probabilistic graphical models. *Science* 303, 799–805. doi: 10.1126/science.1094068

Ghosh, S., and Basu, A. (2012). Network medicine in drug design: implications for neuroinflammation. *Drug Discov. Today* 17, 600–607. doi: 10.1016/j.drudis.2012.01.018

Goh, K., Cusick, M., Valle, D., Childs, B., Vidal, M., and Barabasi, A. (2007). The human disease network. *Proc. Natl. Acad. Sci. U.S.A.* 104, 8685–8690. doi: 10.1073/pnas.0701361104

Gomez, D., Ripoll, G., Giron, S., and Alonso, D. (2006). Desmopressin and other synthetic vasopressin analogues in cancer treatment. *Bull. Cancer* 93, E7–E12.

Gultekin, S., Rosai, J., Demopoulos, A., Graus, Y., Posner, J., Dalmau, J., et al. (2000). Hu immunolabeling as a marker of neural and neuroendocrine differentiation in normal and neoplastic human tissues: assessment using a recombinant anti-Hu Fab Fragment. *Int. J. Surg. Pathol.* 8, 109–117. doi: 10.1177/106689690000800205

Hanahan, D., and Weinberg, R. A. (2000). The hallmarks of cancer. *Cell* 100, 57–70. doi: 10.1016/S0092-8674(00)81683-9

Hanahan, D., and Weinberg, R. A. (2011). Hallmarks of cancer: the next generation. *Cell* 144, 646–674. doi: 10.1016/j.cell.2011.02.013

Hopkins, A. L. (2008). Network pharmacology: the next paradigm in drug discovery. *Nat. Chem. Biol.* 4, 682–690. doi: 10.1038/nchembio.118

Ildirim, M., Goh, K.-I., Cuisick, M., Barabasi, A.-L., and Vidal, M. (2007). Drug-target network. *Nat. Biotechnol.* 25, 1119–1126. doi: 10.1038/nbt1338

Jeong, H., Mason, S., Barabasi, A., and Oltvai, Z. (2001). Lethality and centrality in protein networks. *Nature* 411, 41–42. doi: 10.1038/35075138

Keir, S., Friedman, H., Reardon, D., Bigner, D., and Gray, L. (2013). Mibefradil, a novel therapy for glioblastoma multiforme: cell cycle synchronization and interlaced therapy in a murine model. *J. Neurooncol.* 111, 97–102. doi: 10.1007/s11060-012-0995-0

Knox, C., Law, V., Jewison, T., Liu, P., Ly, S., Frolkis, A., et al. (2011). DrugBank 3.0: a comprehensive resource for 'omics' research on drugs. *Nucleic Acids Res.* 39, D1035–D1041. doi: 10.1093/nar/gkq1126

Lappano, R., and Maggiolini, M. (2011). G protein-coupled receptors: novel targets for drug discovery in cancer. *Nat. Rev. Drug. Discov.* 10, 47–60. doi: 10.1038/nrd3320

Lehman, E. (2005). *Testing Statistical Hypotheses*. New York, NY: Springer.

Marchini, S., Poynor, E., Barakat, R., Clivio, L., Cinquini, M., Fruscio, R., et al. (2012). The zinc finger gene ZIC2 has features of an oncogene and its overexpression correlates strongly with the clinical course of epithelial ovarian cancer. *Clin. Cancer Res.* 18, 4313–4324. doi: 10.1158/1078-0432. CCR-12-0037

Margolin, A., and Califano, A. (2007). Theory and limitations of genetic network inference from microarray data. *Ann. N.Y. Acad. Sci.* 1115, 51–72. doi: 10.1196/annals.1407.019

Margolin, A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Dalla Favera, R., et al. (2006). ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* 7 (Suppl. 1):S7. doi: 10.1186/1471-2105-7-S1-S7

Maslov, S., and Sneppen, K. (2002). Specificity and stability in topology of protein networks. *Science* 296, 910–913. doi: 10.1126/science.1065103

Meyer, P., Kontos, K., and Bontempi, G. (2007). Information-theoretic inference of large transcriptional regulatory networks. *EURASIP J. Bioinform. Syst. Biol.* 2007:79879. doi: 10.1155/2007/79879

Newman, M. (2002). Assortative mixing in networks. *Phys. Rev. Lett.* 89:208701. doi: 10.1103/PhysRevLett.89.208701

Newman, M. (2010). *Networks: An Introduction*. Oxford: Oxford University Press.

Olsen, C., Meyer, P., and Bontempi, G. (2009). On the impact of entropy estimator in transcriptional regulatory network inference. *EURASIP J. Bioinform. Syst. Biol.* 2009:308959. doi: 10.1155/2009/308959

Razick, S., Magklaras, G., and Donaldson, I. (2008). iRefIndex: a consolidated protein interaction database with provenance. *BMC Bioinformatics* 9:405. doi: 10.1186/1471-2105-9-405

Schaffter, T., Marbach, D., and Floreano, D. (2011). GeneNetWeaver: *in silico* benchmark generation and performance profiling of network inference methods. *Bioinformatics* 27, 2263–2270. doi: 10.1093/bioinformatics/btr373

Shamash, J., Salam, A., Davies, D., Williams, A., Joel, S., and Lister, T. (1998). *In vitro* testing of calcium channel blockers and cytotoxic chemotherapy in B-cell low-grade non-Hodgkin's lymphoma. *Br. J. Cancer.* 77, 1598–1603. doi: 10.1038/bjc.1998.262

Sheskin, D. J. (2004). *Handbook of Parametric and Nonparametric Statistical Procedures*. 3rd Edn. Boca Raton, FL: RC Press.

Simpson, W. (1985). The calcium channel blocker verapamil and cancer chemotherapy. *Cell Calcium* 6, 449–467. doi: 10.1016/0143-4160(85)90021-1

Striessnig, J., Bolz, H., and Koschak, A. (2010). Channelopathies in Cav1.1, Cav1.3, and Cav1.4 voltage-gated L-type Ca2+ channels. *Pflugers Arch.* 460, 361–374. doi: 10.1007/s00424-010-0800-x

Van den Bulcke, T., Van Leemput, K., Naudts, B., van Remortel, P., Ma, H., Verschoren, A., et al. (2006). SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC Bioinformatics* 7:43. doi: 10.1186/1471-2105-7-43

Watts, D., and Strogatz, S. (1998). Collective dynamics of 'small-world' networks. *Nature* 393. 440–442. doi: 10.1038/30918

Werhli, A., Grzegorczyk, M., and Husmeier, D. (2006). Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical gaussian models and bayesian networks. *Bioinformatics* 22, 2523–2531. doi: 10.1093/bioinformatics/btl391

Wille, A., Zimmermann, P., Vranova, E., Furholz, A., Laule, O., Bleuler, S., et al. (2004). Sparse graphical gaussian modeling of the isoprenoid gene network in arabidopsis thaliana. *Genome Biol.* 5:R92. doi: 10.1186/gb-2004-5-11-r92

Yip, K. Y., Alexander, R. P., Yan, K.-K., and Gerstein, M. (2010). Improved reconstruction of *in silico* gene regulatory networks by integrating knockout and perturbation data. *PLoS ONE* 5:e8121. doi: 10.1371/journal.pone.0008121

Young, G. A., and Smith, R. L. (2005). *Essentials of Statistical Inference*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511755392

Zhang, H., and Singer, B. H. (2010). *Recursive Partitioning and Applications*. New York, NY: Springer. doi: 10.1007/978-1-4419-6824-1

Zhang, X., Zhao, X.-M., He, K., Lu, L., Cao, Y., Liu, J., et al. (2011). Inferring gene regulatory networks from gene expression data by PC-algorithm based on conditional mutual information. *Bioinformatics* 28, 98–104. doi: 10.1093/bioinformatics/btr626

Conflict of Interest Statement: 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.

Received: 30 September 2013; accepted: 25 November 2013; published online: 16 December 2013.

Citation: de Matos Simoes R, Dehmer M and Emmert-Streib F (2013) B-cell lymphoma gene regulatory networks: biological consistency among inference methods. Front. Genet. ***4****:281. doi: 10.3389/fgene.2013.00281*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 17 December 2013
doi: 10.3389/fgene.2013.00294

Joint conditional Gaussian graphical models with multiple sources of genomic data

Hyonho Chun[1]*, Min Chen[2], Bing Li[3] and Hongyu Zhao[4]

[1] Department of Statistics, Purdue University, West Lafayette, IN, USA
[2] Department of Mathematical Sciences, University of Texas at Dallas, Dallas, TX, USA
[3] Department of Statistics, The Pennsylvania State University, University Park, PA, USA
[4] Department of Biostatistics, Yale School of Public Health, New Haven, CT, USA

Edited by:
Frank Emmert-Streib, Queen's University Belfast, UK

Reviewed by:
Subha Madhavan, Georgetown University, USA
Zoran Nikoloski, Max-Planck Institute of Molecular Plant Physiology, Germany
Igor B. Rogozin, National Institutes of Health, USA

***Correspondence:**
Hyonho Chun, Department of Statistics, Purdue University, West Lafayette, 150 N. University Street, IN 47906, USA
e-mail: chunh@purdue.edu

It is challenging to identify meaningful gene networks because biological interactions are often condition-specific and confounded with external factors. It is necessary to integrate multiple sources of genomic data to facilitate network inference. For example, one can jointly model expression datasets measured from multiple tissues with molecular marker data in so-called genetical genomic studies. In this paper, we propose a joint conditional Gaussian graphical model (JCGGM) that aims for modeling biological processes based on multiple sources of data. This approach is able to integrate multiple sources of information by adopting conditional models combined with joint sparsity regularization. We apply our approach to a real dataset measuring gene expression in four tissues (kidney, liver, heart, and fat) from recombinant inbred rats. Our approach reveals that the liver tissue has the highest level of tissue-specific gene regulations among genes involved in *insulin responsive facilitative sugar transporter mediated glucose transport pathway*, followed by heart and fat tissues, and this finding can only be attained from our JCGGM approach.

Keywords: Gaussian graphical models, gene networks, GGMs, conditional GGMs, joint sparsity

1. INTRODUCTION

Inference of gene networks plays an important role in revealing the interactions among genes that may lead to a better understanding of molecular mechanisms in organisms. Biologists routinely use high-throughput technologies (e.g., microarrays) to measure gene expression data at the genome scale to study various biological and biomedical problems. Statisticians are often charged to explore interactions among genes through statistical analysis of these large data sets. It is natural to use multivariate approaches to analyze these high-throughput datasets, because multivariate methods may reveal various interactions among genes that cannot be captured from individual gene based approaches.

In this paper we focus on a graphical model approach that aims at finding relationships among a group of genes, where a graph is used for encoding relationships among multiple variables. When a graph is used for a gene network, nodes represent genes and edges represent relationships between the connected genes. The edges can be defined with various relationships among genes. For example, pairwise correlations are used to define edges in a "relevance network." Similarly, we can define edges through conditional dependence, that is, any two genes connected with an edge in such graphical models are conditionally dependent of each other when the effects from all other genes are explained away. Therefore, when the expression profiles of two genes are correlated because they are both regulated by some other genes, the graphical model does not put an edge between these two genes because they are conditionally independent given the expressions of the common regulatory genes. In this way, the graphical model produces a more parsimonious graph than a relevance network.

Gene network inference is a complex problem, because the relationships of genes are often affected by external variables (e.g., genomic variations), and gene regulatory relationships may be altered under different conditions such as tissue types. This means that a single network inferred from gene expression measurements alone may not be adequate to describe the relationships among genes. Further, it is often desirable to jointly model gene networks under various conditions rather than considering them separately, because large parts of the networks are likely to share common topologies corresponding to similar underlying biological processes across conditions (e.g., the house keeping functions and the clock), and thus joint modeling may increase the power of detecting common gene interactions. Therefore, one may want to infer multiple condition-specific networks in a single model framework, while the network models may also need to incorporate all available external variables as well. Such inference is possible through the analysis of datasets in genetical genomic studies from same genetic origin (Jansen and Nap, 2001) where gene expressions from multiple tissues, as well as marker genotypes, are measured from the same set of individuals. These data allow us to perform an integrative analysis via joint conditional Gaussian graphical models (JCGGM) to infer relationships among genes. The JCGGM approach is an extension of the conditional Gaussian graphical model (CGGM) in order to increase power of the methods via joint modeling. The joint modeling is particularly important in the conditional models with a limited sample size, since the model's complexity increases very quickly

Network Inference

- Q: What types of biological networks have been inferred in the paper?
- A: We use gene expression data and marker data from recombinant inbred rats and infer gene regulation network by using genes consisting of the insulin responsive facilitative sugar transporter mediated glucose transport pathway.
- Q: How was the quality/utility of the inferred networks assessed?
- A: Our JCGGM found that the liver network has the highest tissue specificity, and this is in line with the role of SLC2A4 protein, which forms glucose concentration gradient of muscle and fat cells, as well as the specialized glycogen breakdown of glycogen phosphorylase that only occurs in liver tissue (Watson et al., 2004; Campbell et al., 2006).
- Q: How were these networks validated?
- A: We have performed simulation study to test performance of the proposed JCGGM approach and our approach performs the best over all simulation scenarios. We have also provided the scientific literature to support the validity of the inferred networks.

and the separate models have no power unless appropriately combined.

In Section 2, we first introduce CGGMs and joint regularization approaches, and then propose the JCGGM that uses both the CGGM and a joint regularization approach. In Section 3, we show the performance of our approach via a simulation study and then apply it to a genetical genomics study, where gene expressions from four different tissues are measured together with genotype data from recombinant inbred rats. We show that the JCGGM approach is able to find tissue-specific gene networks. The discussion follow in Section 4.

2. MATERIALS AND METHODS

2.1. MATERIAL

For a real data analysis, we used a dataset of Petretto et al. (2006) in which gene expression levels in four tissues (liver, kidney, heart and fat) were measured from a panel of 29 rat recombinant inbred (RI) strains. This strain was derived from a cross between the spontaneously hypertensive rat (SHR) and the brown norway (BN) strains (Hubner et al., 2005). We downloaded the dataset normalized by the robust multi-array average (RMA) algorithm from www.genenetwork.org (Accession numbers: GN70, GN79, GN221 and GN222). From the same website, we also downloaded a genetic marker dataset that consists of 556 markers.

2.2. METHODS

In this section, we briefly introduce recent approaches for CGGMs as well as those for joint estimation of multiple Gaussian graphical models. We then propose a new method to combine these approaches in order for inferring networks from multiple sources of biological data for finding multiple CGGMs. Finally, we explain the simulation process for generating datasets that are used for comparing the performance of our proposed method.

2.2.1. A brief summary on CGGM and joint estimation of multiple GGMs

A GGM describes the conditional independences of multiple random variables, $Y_1, \ldots, Y_p$ with a graph $G = (V, E)$, where $V = \{1, \ldots, p\}$ is a set of nodes and E is a set of edges, in which an edge between nodes represents that they are conditionally dependent. According to the Hammersely and Clifford theorem, a graphical model can be inferred from a factorization of the joint density of a multivariate random vector $Y = (Y_1, \ldots, Y_p)^T$. When Y is assumed to follow a multivariate Gaussian distribution $N_p(0, \Sigma)$, where Σ is a $p \times p$ covariance matrix, a factorization can be easily found from zero elements of the inverse covariance matrix (also known as the precision matrix), $\Sigma^{-1} = \Omega$. Hence, conditional independence can be directly inferred from zero entries of a precision matrix, when a multivariate Gaussian assumption is made. This model is called a GGM (Lauritzen, 1996). Finding a sparse precision matrix with various regularizations such as lasso and adaptive lasso (Tibshirani, 1996; Zou, 2006) has been studied by many researchers including Li and Gui (2006); Yuan and Lin (2007); Friedman et al. (2008).

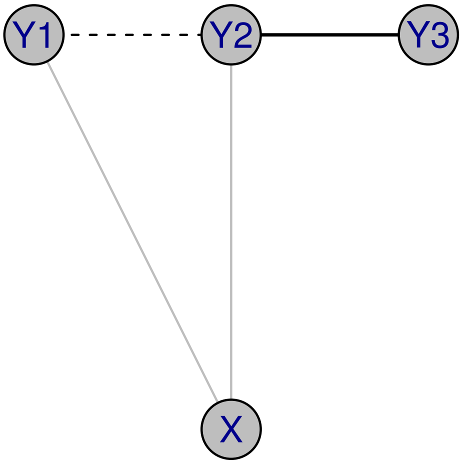


FIGURE 1 | Illustration of conditional GGM: *X* represents a single molecular marker, and Y_1, Y_2, Y_3 represent the expressions of three genes. When the marker effect is ignored, there are two edges in a graphical model: 1 ↔ 2 and 2 ↔ 3. After considering the marker effect, there is a single edge, represented with a solid line, in a conditional graphical model.

More recently, it has been noted that one can further elaborate a GGM by using extra sources of information. For example, as in **Figure 1**, let us assume that X represents a single molecular marker, and Y_1, Y_2, Y_3 represent the expressions of three genes. When the marker effect is ignored, there are two edges in the unconditional graphical model: 1 ↔ 2 and 2 ↔ 3. After considering the marker effect, there is only one edge, represented by the solid line, in the conditional graphical model. For this purpose, a conditional Gaussian graphical model (CGGM) is introduced by several researchers including Yin and Li (2011); Li et al. (2012); Cai et al. (2013).

In addition to the conditional modeling, there is recently an increasing needs for inferring multiple networks that vary across conditions. For example, gene expression levels are measured in multiple tissues so as to study the tissue specificity of the gene regulations. Since the sample size is often limited, we would achieve a more accurate network inference when an appropriate joint modeling is used than when a separate estimation is made for each network because such joint analysis allows borrowing information across conditions. The joint modeling problem has been studied by several researchers including Guo et al. (2011); Danaher et al. (2013); Chun et al. (unpublished). These approaches do not accommodate the conditional models, and we will consider a joint approach in the context of estimating the conditional models.

2.2.2. *Joint estimation of multiple conditional Gaussian graphical models*

In this section, we propose an approach to estimate the multiple CGGMs jointly. This approach is aimed to infer tissue-specific gene networks from a genetical genomic dataset that consists of a marker dataset and a collection of gene expression datasets from several tissues.

We assume that at the t-th condition, a p-dimensional gene expression measurement $Y^{(t)}$ is from $N_p(f^{(t)}(X), (\Omega^{(t)})^{-1})$, $t = 1, \ldots, T$, where $f^{(t)}(\cdot)$. is an arbitrary function, and X is a q-dimensional vector $(X_1, \ldots, X_q)^T$, describing an extra dataset such as a genetic marker dataset. We remark that $f^{(t)}(\cdot)$ varies along with the condition t, and thus our model is able to reflect the dynamic nature of genetic controls (Gerrits et al., 2009). A conditional model describes conditional independence between any two variables, Y_i and Y_j given the remaining variables $Y_{-\{i,j\}}$ and the extra information $f^{(t)}(X)$. Here, $Y_{-\{i,j\}}$ represents a $p-2$ dimensional subvector of Y excluding the i th and j th components. The interest is in estimating $\{\Omega^{(t)}\}_{t=1}^{T}$ jointly, while accounting for the effects from X. We will take a two-stage approach: (1) finding consistent conditional covariance matrix $\hat{\Sigma}^{(t)}$, $t = 1, \ldots, T$ and (2) finding sparse estimates of $\{\Omega^{(t)}\}_{t=1}^{T}$ by using a joint sparsity penalty.

The first step is finding $\hat{\Sigma}^{(t)}$ with a conditional covariance matrix estimator after carefully selecting a subset of X that are related to Y. Such $\Sigma^{(t)}$ can be estimated by using a conditional variance matrix of $\Sigma_{YY|X}$, based on a conditional variance operator between RKHSs of X and Y under some general model assumptions (Li et al., 2012). Assuming the $X^{(t),i}$ and $Y^{(t),i}$, $i = 1, \ldots, n$, are independently and identically distributed random vectors as with $X^{(t)}$ and $Y^{(t)}$, respectively, we can estimate the conditional variance matrix by using a kernel $\mathbf{K}_X$ as follows: $\frac{1}{n}\left(\mathbf{Y^{(t)}}^T\mathbf{QY^{(t)}} - \mathbf{Y^{(t)}}^T\mathbf{Q}(\mathbf{QK}_X\mathbf{Q})(\mathbf{QK}_X\mathbf{Q})^{\dagger}\mathbf{QY^{(t)}}\right)$, where $\mathbf{Y^{(t)}} = \left(Y^{(t),1}, \ldots, Y^{(t),n}\right)^T$, $\mathbf{Q} = I_n - \frac{1}{n}J_n$, I_n is an $n \times n$ identity matrix, J_n is an $n \times n$ matrix whose elements are all 1, and $A^{\dagger}$ means a generalized inverse of a matrix A. When a linear kernel is used, the conditional variance matrix becomes $S_{Y^{(t)}Y^{(t)}} - S_{Y^{(t)}X}S_{XX}^{-1}S_{XY^{(t)}}$, where $S_{XX} = \frac{1}{n}\sum_{i=1}^{n} X^iX^{i^T}$, $S_{XY^{(t)}} = \frac{1}{n}\sum_{i=1}^{n} X^iY^{(t),i^T}$ and $S_{Y^{(t)}Y^{(t)}} = \frac{1}{n}\sum_{i=1}^{n} Y^{(t),i}Y^{(t),i^T}$. Thus, one can obtain the estimate of the conditional variance as in Yin and Li (2011); Cai et al. (2013) by using linear kernels. When X represents marker genotypes of a backcross from a genetical genomics study, the linear model assumption is reasonable because the genotypes have two levels of genotype values (e.g., back cross population). With other kernels such as a polynomial and a radial basis function kernel, one can model an arbitrary form of f flexibly.

Second, we will use a penalized profiled likelihood that jointly estimate $\left\{\Omega^{(t)}\right\}_{t=1}^{T}$ with a joint sparsity penalization as follows:

$$\mathrm{PPL}\left(\left\{\Omega^{(t)}\right\}_{t=1}^{T}\right) = \sum_{t=1}^{T} n_t \left(-\log\det\left(\Omega^{(t)}\right) + \mathrm{tr}\left(\hat{\Sigma}^{(t)}\Omega^{(t)}\right)\right) + P\left(\left\{\Omega^{(t)}\right\}_{t=1}^{T}\right), \quad (1)$$

where $\hat{\Sigma}^{(t)}$ is the conditional covariance matrix estimate, and $P(\cdot)$ is a penalty function. In addition, $\mathrm{tr}(A)$ and $\det(A)$ denote trace and determinant of matrix A, respectively. The joint sparsity function $P(\cdot)$ can be chosen from the following different penalty functions:

- $\lambda_1 \sum_{j \neq j'} \sqrt{\sum_{t=1}^{T} \left|\omega_{j,j'}^{(t)}\right|}$ (Guo et al., 2011)
- $\lambda_1 \sum_{t=1}^{T} \sum_{j,j'} \left|\omega_{j,j'}^{(t)}\right| + \lambda_2 \sum_{j,j'} \sqrt{\sum_{t=1}^{T} {\omega_{j,j'}^{(t)}}^2}$ (Danaher et al., 2013)
- $\lambda_1 \sum_{j \neq j'} g\left(\sum_{t=1}^{T} \left|\omega_{j,j'}^{(t)}\right|\right)$ (Chun et al., unpublished),

where $\omega_{j,j'}^{(t)}$ is the (j, j')th element of $\Omega^{(t)}$, λ_1 and λ_2 are positive tuning parameters, and g is a nonconvex function such as $g(x) = x^{\beta}$, where $0 < \beta < 1$, or a truncated log function or a truncated inverse polynomial function.

The approach of Chun et al. (unpublished) is a generalization of Guo et al. (2011), where it allows the control in balance between common and condition-specific structures by the choice of the penalty function $P(\cdot)$. Through a simulation study, Chun et al. (unpublished) showed that the truncated log penalty performs well, when the majority of networks are shared across conditions. Interestingly, the approach of Danaher et al. (2013) uses two tuning parameters, which can make the algorithm computationally challenging. Also, in their approach, the common structure is defined as $\sqrt{\sum_{t=1}^{T} {\omega_{j,j'}^{(t)}}^2}$, whereas it is defined as $\sum_{t=1}^{T} \left|\omega_{j,j'}^{(t)}\right|$ in the other approaches. With the latter choice, the condition-specific regularization can be automatically achieved by the use of a nonconvex penalty function. Additionally, they proved that the estimator from the nonconvex penalty has a sparsistency (variable selection consistency) for edges that appear in any of the conditions. We thus use the truncated log penalty of Chun et al. (unpublished) for the joint estimation of multiple GGMs. That is, our penalty function is given by

$$P\left(\{\Omega\}_{t=1}^{T}\right) = \sum_{j \neq j'} \left\{ \left(\log \left(\sum_{t=1}^{T} \left| \omega_{j,j'}^{(t)} \right| \right) - \log \epsilon + 1 \right) I_A + \frac{\left| \sum_{t=1}^{T} \omega_{j,j'}^{(t)} \right|}{\epsilon} I_{A^c} \right\},$$

where $A = \left(\sum_{t=1}^{T} \left| \omega_{j,j'}^{(t)} \right| > \epsilon \right)$, $A^c = \left(\sum_{t=1}^{T} \left| \omega_{j,j'}^{(t)} \right| \leq \epsilon \right)$ and ϵ is a small positive constant (we used $\epsilon = 1e^{-3}$ in the current manuscript). We remark that the choice of a different penalty function corresponds to enforcing different level of joint sparsity in network inference. Hence we may obtain improved results from the different penalty function depending on the underlying truth. However, due to the limited sample size in biological datasets, it is often very difficult to find the optimal penalty function.

The objective function 1 can be optimized by using a local linear approximation as in Guo et al. (2011). We remark that the solution from the current optimization algorithm may not produce a global solution, and hence the choice of the good initial estimate is very important. However, our simulation study suggests that the current algorithm yields a good estimate in terms of performance of the approach. Specifically, at the $(k+1)$th iteration, the PL is decomposed into T individual optimization problems as follows:

$$\left(\Omega^{(t)}\right)^{(k+1)} = \text{argmin}_{\Omega^{(t)}} n_t \left(tr\left(S^{(t)} \Omega^{(t)} \right) - \log \left\{ \det \left(\Omega^{(t)} \right) \right\} \right) + \lambda \sum_{j \neq j'} \zeta_{j,j'}^{(k)} \left| \omega_{j,j'}^{(t)} \right|,$$

where $\zeta_{j,j'}^{(k)} = P' \left(\sum_{t=1}^{T} \left| \left(\omega_{j,j'}^{(t)} \right)^{(k)} \right| \right) = \max \left(\sum_{t=1}^{T} \left| \left(\omega_{j,j'}^{(t)} \right)^{(k)} \right|, \epsilon \right)^{-1}$ and $\left(\omega_{j,j'}^{(t)} \right)^{(k)}$ is the solution of the previous k-th step. Then, the formulation becomes a single precision matrix estimation problem with a weighted lasso penalty, which can be solved by the **glasso** algorithm (Friedman et al., 2008).

JCGGM algorithm

1. Compute $\hat{\Sigma}$ by using a kernel. When a linear kernel is used, $\hat{\Sigma}^t = S_{Y^tY^t} - S_{Y^tX} S_{XX}^{-1} S_{XY^t}$.
2. Initialize $\hat{\Omega}^t = \left(\hat{\Sigma}^t + \delta I_p \right)^{-1}$ for all $1 \leq t \leq T$, where I_p is the identity matrix and the constant δ is chosen so that $\hat{\Sigma}^t + \delta I_p$ is invertible. We added $1e^{-3}$ to the diagonals when the ratio of largest and smallest eigen values is larger than $1e^3$.
3. Update $\hat{\Omega}^t$ for all $1 \leq t \leq T$ by solving
$$\min_{\Omega^t} tr\left(\hat{\Sigma}^t \Omega^t \right) - \log \left\{ det \left(\Omega^t \right) \right\} + \lambda \sum_{j \neq j'} \frac{\left| \omega_{j,j'}^{t} \right|}{\left(\sum_{t=1}^{T} \left| \hat{\omega}_{j,j'}^{t} \right| \right)},$$
using a **glasso**, where $\hat{\omega}_{j,j'}^{t}$ is the estimate from the previous step.
4. Repeat step 2 until convergence is achieved.

For selecting the tuning parameter λ, one can use the following BIC criterion:

$$\text{BIC}(\lambda) = \sum_{t=1}^{T} \left\{ -\log \det \left(\hat{\Omega}^{(t)}(\lambda) \right) + tr \left(\hat{\Sigma}^{(t)} \hat{\Omega}^{(t)}(\lambda) \right) + \log(n_t)\, df_t / n_t \right\},$$

where $\{\hat{\Omega}^{(t)}(\lambda)\}_{t=1}^{T}$ are the estimates from solving the penalized negative log likelihood with a tuning parameter λ where df_t is $\text{card}\{(j, j') : j \leq j', \hat{\omega}_{j,j'}^{(t)} \neq 0\}$ with *card* representing the cardinality of a finite set.

2.3. METHODS FOR SIMULATION STUDY

For simulation study, we generate datasets by taking the number of conditions $T = 3$, the number of gene expression variables $p = 30$ and the number of markers $q = 10$. We set the sample sizes $n_t = 30$ and 100 to assess the small and large sample performances of the estimators. We first simulate X that mimics a marker dataset by using *sim.map* and *sim.cross* functions from **R/qtl** package. We consider a single chromosome with length 1000 cM and place 10 equally spaced markers. We use the backcross design, since it is the design used in our real data analysis in the next section.

The scale-free network structures, which are the most commonly observed structure in biology, are generated using the Barabasi–Albert algorithm (Barabasi and Albert, 1999). We start from six edges, and add one edge at each step. We first generate common edges from each of the network structures. For each condition, randomly selected 0.1 M edges are added as condition-specific edges, where M is the total number of edges in the common structure. Based on the network structures, we simulate the precision matrices by setting values for the off-diagonals that correspond to edges with random numbers from $Unif([-1, -0.5] \cup [0.5, 1])$, and by setting the diagonal elements with $\sum_{j \neq i} |\omega_{i,j}|$. The process is repeated until Ω^t becomes a positive definite matrix.

For simulating Y^t, we first consider a scenario where there is no external variable that causes dependence among genes. This is an extreme scenario where our proposed conditional approach does not have any advantage over the unconditional model. We simulate Y^t with the model $Y^t = XB^t + E^t$. The elements of B^t are zeros except for (1,1), (2,4) and (3,8)th positions. These nonzero coefficients are (−0.09, 0.789, −0.667), (1.361, 1.508, −2.608) and (0.687, 0.316, 2.020) for three conditions. The ith row of E^t is simulated from $N_p(0, \Omega^{t^{-1}})$.

We then consider a scenario where there exist hotspots that cause marginal associations among genes. This is the case where our proposed conditional approach is expected to perform better than the unconditional approach. Now, $Y_1^t, \ldots, Y_{18}^t$ are linked to X_1; $Y_{19}^t, \ldots, Y_{25}^t$ are to X_4^t; and $Y_{26}^t, \ldots, Y_{30}^t$ are to X_8. The nonzero coefficients are simulated by perturbing the coefficients used in Case 1. $B_{(i,1)}^1 = -0.09 + N(0, 0.1^2)$, for $i = 1, \ldots, 18$; $B_{(i,1)}^1 = 0.789 + N(0, 0.1^2)$, for $i = 19, \ldots, 25$; $B_{(i,1)}^1 = -0.667 + N(0, 0.1^2)$, for $i = 26, \ldots, 30$; $B_{(i,1)}^2 = 1.361 + N(0, 0.1^2)$, for $i = 1, \ldots, 18$; $B_{(i,1)}^2 = 1.508 + N(0, 0.1^2)$, for $i = 19, \ldots, 25$; $B_{(i,1)}^2 = -2.608 + N(0, 0.1^2)$, for

$i = 26, \ldots, 30; B^{3}_{(i,1)} = 0.687 + N(0, 0.1^2)$, for $i = 1, \ldots, 18$; and $B^{1}_{(i,1)} = 0.316 + N(0, 0.1^2)$, for $i = 19, \ldots, 25$; $B^{1}_{(i,1)} = 2.020 + N(0, 0.1^2)$, for $i = 26, \ldots, 30$. The ith row of E^t is simulated from $N_p(0, \Omega^{t^{-1}})$.

3. RESULTS

3.1. RESULTS FROM SIMULATION STUDY

We compare the performances of unconditional/conditional GGMs and joint conditional GGMs. We use the following five criteria for the comparison:

1. False positive rate at $\hat{\lambda}_{\text{BIC}}$:

$$FP(\hat{\lambda}_{\text{BIC}}) = \frac{1}{T}\sum_{t=1}^{T} \frac{\text{card}\{(i, j) : i > j, \omega^{t}_{i,j} = 0 \text{ and } \hat{\omega}^{t}_{i,j} \neq 0\}}{\text{card}\{(i, j) : i > j \text{ and } \omega_{i,j} = 0\}}.$$

2. False negative rate at $\hat{\lambda}_{\text{BIC}}$:

$$FN(\hat{\lambda}_{\text{BIC}}) = \frac{1}{T}\sum_{t=1}^{T} \frac{\text{card}\{(i, j) : i > j, \omega^{t}_{i,j} \neq 0 \text{ and } \hat{\omega}^{t}_{i,j} = 0\}}{\text{card}\{(i, j) : i > j \text{ and } \omega_{i,j} \neq 0\}}.$$

3. False positive rate for common zeros at $\hat{\lambda}_{\text{BIC}}$:

$$FPC(\hat{\lambda}_{\text{BIC}}) = \frac{\text{card}\left\{(i, j) : i > j;\ \omega^{t}_{i,j} = 0 \text{ for all } t = 1, \ldots, T;\ \text{and } \hat{\omega}^{t}_{i,j} \neq 0 \text{ for any } t, 1 \leq t \leq T\right\}}{\text{card}\{(i, j) : i > j;\ \text{and } \omega^{t}_{i,j} = 0 \text{ for all } t = 1, \ldots, T\}}.$$

4. False negative rate for common zeros at $\hat{\lambda}_{\text{BIC}}$:

$$FNC(\hat{\lambda}_{\text{BIC}}) = \frac{\text{card}\left\{(i, j) : i > j;\ \omega^{t}_{i,j} \neq 0 \text{ for any } t, 1 \leq t \leq T;\ \text{and } \hat{\omega}^{t}_{i,j} = 0 \text{ for all } t = 1, \ldots, T\right\}}{\text{card}\{(i, j) : i > j;\ \text{and } \omega_{i,j} \neq 0 \text{ for any } t, 1 \leq t \leq T\}}.$$

5. Relative Frobenius loss (RFL):

$$\text{RFL} = \frac{1}{T}\sum_{t=1}^{T} ||\Omega^{t} - \hat{\Omega}^{t}||^{2}_{F}/||\Omega^{t}||^{2}_{F}.$$

The results are given in **Tables 1**, **2**. First, one can see that the joint approach improves the performance greatly for the small sample cases. This effect is more pronounced for the conditional models. This may be explained by the fact that conditional models require the estimation of more parameters than unconditional ones. Second, for large sample sizes, JCGGM performs the best in both simulation scenarios. This also confirms that even if we include extra variables in a conditional model, it will perform well as long as the sample size is large enough. The current results depend on the BIC criterion, and one may have different results when different tuning parameter selection approach is used. We thus present ROC curves in **Figure 2**. These ROC curves are the average ROC curves of 200 replicates. The figure confirms that JCGGM performs the best in all simulation scenarios.

3.2. REAL DATA ANALYSIS

In this section, we demonstrate how to use the JCGGM approach in a real biological study. In this analysis, we focused on genes that consist of a particular pathway. Pathway information was obtained from rgd.mcw.edu, and we investigated *the insulin responsive facilitative sugar transporter mediated glucose transport pathway*. We were able to identify 34 genes in our dataset that belong to the pathway. We then used joint GGMs and joint CGGMs approach for finding a gene regulation networks. For the CGGM approach, we have selected a marker set based on **scanone** function of **R/qtl** package. For each of 34 genes, we selected markers that were significantly linked to the gene expression at the genome wide significance level of 0.05. We used permutation with 1000 replicates for computing the genome wide significance. We then took the union of those selected markers as covariates for our RKHS conditional covariance estimator with a linear kernel. We remark that the set of selected markers were tissue-specific.

Table 1 | Results for Case 1.

	FP	FN	FPC	FNC	RFL
			***n* = 30**		
GGMs	0.081 (0.002)	0.755 (0.004)	0.222 (0.004)	0.518 (0.008)	0.703 (0.002)
CGGMs	0.946 (0.001)	0.063 (0.002)	0.999 (0.000)	0.000 (0.000)	5087.146 (135.93)
JGGM	0.053 (0.002)	0.560 (0.004)	0.067 (0.002)	0.524 (0.005)	0.564 (0.002)
JCGGM	0.114 (0.013)	0.459 (0.007)	0.134 (0.014)	0.434 (0.008)	2.517 (0.624)
			***n* = 100**		
GGMs	0.051 (0.001)	0.475 (0.003)	0.144 (0.003)	0.262 (0.004)	0.577 (0.001)
CGGMs	0.054 (0.001)	0.335 (0.003)	0.152 (0.003)	0.164 (0.004)	0.348 (0.002)
JGGM	0.027 (0.002)	0.383 (0.002)	0.030 (0.001)	0.346 (0.003)	0.504 (0.001)
JCGGM	0.020 (0.001)	0.329 (0.002)	0.021 (0.001)	0.298 (0.003)	0.263 (0.001)

The performances of GGMs, CGGMs, JGGMs, and JCGGMs are compared with the comparison criteria explained in subsection 3.1. When the sample size is small, the separate CGGMs select many false positives, which can be alleviated with JCGGMs. Under the scenario which is favored to JGGM, the JCGGM performs as well as the JGGM in both small and large sample cases.

Table 2 | Results for Case 2.

	FP	FN	FPC	FNC	RFL
			***n* = 30**		
GGMs	0.143 (0.003)	0.685 (0.005)	0.367 (0.006)	0.359 (0.007)	0.692 (0.002)
CGGMs	0.945 (0.001)	0.066 (0.002)	1.000 (0.000)	0.000 (0.000)	5343.2 (142.343)
JGGM	0.011 (0.005)	0.907 (0.006)	0.014 (0.005)	0.890 (0.006)	71.99 (71.27)
JCGGM	0.112 (0.013)	0.467 (0.008)	0.133 (0.013)	0.444 (0.008)	2.992 (0.84)
			***n* = 100**		
GGMs	0.161 (0.002)	0.226 (0.002)	0.365 (0.004)	0.061 (0.002)	0.471 (0.002)
CGGMs	0.080 (0.001)	0.228 (0.002)	0.189 (0.003)	0.060 (0.002)	0.328 (0.002)
JGGM	0.103 (0.001)	0.164 (0.002)	0.135 (0.002)	0.132 (0.003)	0.392 (0.001)
JCGGM	0.023 (0.001)	0.162 (0.003)	0.024 (0.002)	0.127 (0.003)	0.234 (0.001)

The performances of GGMs, CGGMs, JGGMs, and JCGGMs are compared with the comparison criteria explained in subsection 3.1. When the sample size is small, the separate CGGMs select many false positives, which can be alleviated with JCGGMs. Under the scenario which is favored to JCGGMs, the JCGGM performs the best in both small and large sample cases.

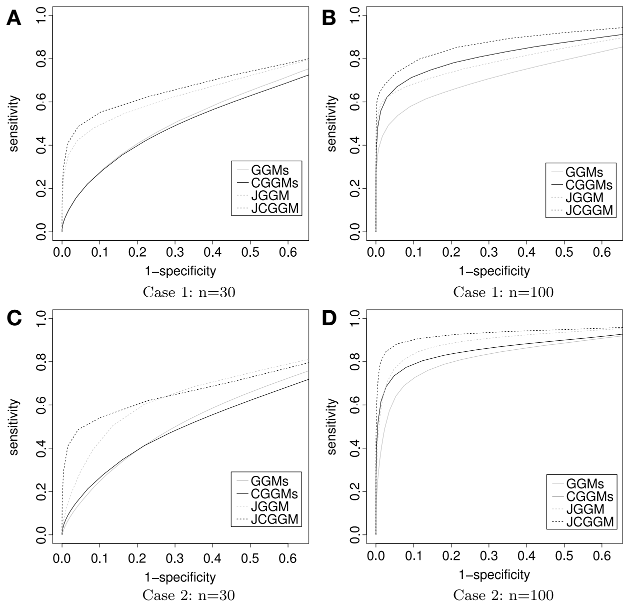


FIGURE 2 | ROC curves: the average ROC curves are presented. Throughout all scenarios, the JCGGM performs the best. **(A)** With no external variable and a small sample size, JGGM, and JCGGM perform well. **(B)** With no external variable and a large sample size, JCGGM performs the best, followed by CGGM and JGGM. These two performs similarly. **(C)** With external variables and a small sample size, only JCGGM performs well. **(D)** With external variables and a large sample size, JCGGM performs the best, followed by JGGM and CGGM.

Table 3 | Results from JGGM and JCGGM.

		Kidney	Liver	Heart	Fat
JGGMs	Number of edges	93	120	115	117
	% specific edges	1.1	5.8	6	4.2
JCGGMs	Number of edges	74	99	94	93
	% specific edges	0	9.1	3.2	2.1

The JGGM and JCGGM are applied to the expression measurements of genes involved in insulin responsive facilitative sugar transporter mediated glucose transport pathway. The JGGM implies that liver, heart, and fat tissues have the similar level of tissue-specificity, whereas the JCGGM implies that the liver tissue has the highest level of tissue specificity. The result from JCGGM is more convincing due to the fact that the specialized enzyme activity of glycogen phosphorylase only occurs in liver tissue.

The results are given in **Table 3**. First, in both JGGM and JCGGM, the liver networks have the largest numbers of edges. The heart and fat networks have similar numbers of edges to the liver network based on JGGM, but they have fewer edges based on JCGGM. This suggests that the pathway is the most activated in a liver tissue, and some tissue-specific controls in heart and fat might be from marker effects. We then computed the percentage of edges that present only in the corresponding tissue. Based on the JGGM, liver and heart networks have a high level of tissue-specific edges. But, the JCGGM found that the liver network has the highest tissue specificity. Interestingly, our finding is in line with the role of *SLC2A4* protein, which forms glucose concentration gradient of muscle and fat cells, as well as the specialized glycogen breakdown of glycogen phosphorylase that only occurs in liver tissue (Watson et al., 2004; Campbell et al., 2006). We also present the estimated graphs in **Figure 3**.

As demonstrated in the analysis, the CGGMs can distinguish intrinsic and extrinsic regulations and gives a better overview in tissue-specificity in intrinsic regulations. To our knowledge, the tissue-specificity in gene regulations has been studied in marker-expression relationships only, and the tissue specificity in intrinsic interactions has never been studied. The JCGGMs approach can be useful for studying tissue-specificity in gene interactions.

4. DISCUSSION

Genes interact with each other in various ways. Some genes interact directly, whereas some genes interact because they are both regulated by the same set of genes or other covariates. CGGM allows us to infer only direct interactions among genes by using the definition of a graphical model and using extra information as predictors. The joint sparsity regularization can be achieved by using various penalty functions. By combining these two

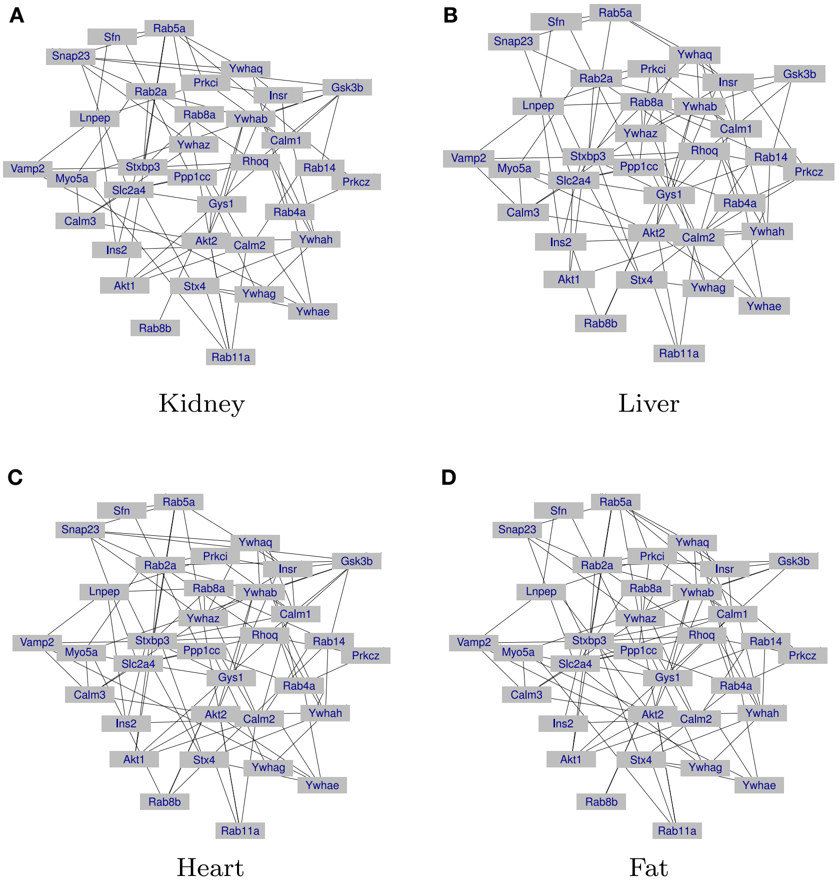


FIGURE 3 | Networks inferred from JCGGM: the liver network has the largest number of edges and the highest level of tissue-specificity. **(A)** The inferred gene regulation network of the kidney tissue is presented. **(B)** The inferred gene regulation network of the liver tissue is presented. **(C)** The inferred gene regulation network of the heart tissue is presented. **(D)** The inferred gene regulation network of the fat tissue is presented.

approaches, we have explained how to find multiple CGGMs jointly and applied the approach to a real biological dataset. The analysis showed that JCGGM is able to reveal tissue-specific interactions that cannot be explained by marker effects. In addition to the previous findings on tissue specificity in gene-marker regulations, studying the extra level of tissue-specificity in gene-gene interactions brings additional understanding of the complexity in gene interactions.

In the conditional model, it is important to include all relevant extra information in the model. However, it is not necessary to include only relevant predictors, which means that one can find a better network when one incorporates available extra variables into the model as long as the sample size is large compared to the number of included variables. The RKHS approach does not involve a variable selection step of X because it assumes that a proper set of covariates are available. However, when the number of covariate is is large, while the sample size is small, we need to consider a variable selection step for choosing only a relevant subset of covariates. Otherwise, the RKHS conditional covariance estimator would not be consistent. The only requirement for the conditional covariance matrix estimator is that the estimator is consistent and has a finite variance [Equation 24 of Li et al. (2012)], and thus any method that can produce such an estimator can work well for finding a CGGM. For example, one can use the approaches of Yin and Li (2011) or Cai et al. (2013) as long as it yields a reasonable set of covariates. In genetical genomics study, one can use a traditional quantitative trait loci (QTL) mapping method to select relevant markers, and the eQTL mapping method was used in our manuscript.

ACKNOWLEDGMENTS

The authors appreciate the constructive comments from the three reviewers. The work of Hyonho Chun was supported in part by NSF grant DMS 1107025, and the work of Min Chen was partly supported by NIH grant 1K25AR063761. The work of Bing Li was supported in part by NSF grant DMS 1106815. The work of Hongyu Zhao was supported in part by NIH grants GM59507 and CA154295 and NSF grant DMS 1106738.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://www.frontiersin.org/journal/10.3389/fgene.2013.00294/abstract

REFERENCES

Barabasi, A. L., and Albert, R. (1999). Emergence of scaling in random networks. *Science* 286, 509–512. doi: 10.1126/science.286.5439.509

Cai, T., Li, H., Liu, W., and Xie, J. (2013). Covariate-adjusted precision matrix estimation with an application in genetical genomics. *Biometrika* 100, 139–156. doi: 10.1093/biomet/ass058

Campbell, N. A., Williamson, B., and Heyden, R. J. (2006). *Biology: Exploring Life*. Boston, MA: Pearson Prentice Hall.

Danaher, P., Wang, P., and Witten, D. M. (2013). The joint graphical lasso for inverse covariance estimation across multiple classes. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* doi: 10.1111/rssb.12033

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9, 432–441. doi: 10.1093/biostatistics/kxm045

Gerrits, A., Li, Y., Tesson, B. M., Bystrykh, L. V., Weersing, E., Ausema, A., et al. (2009). Expression quantitative trait loci are highly sensitive to cellular differentiation state. *PLoS Genet.* 5:e1000692. doi: 10.1371/journal.pgen.1000692

Guo, J., Levina, E., Michailidis, G., and Zhu, J. (2011). Joint estimation of multiple graphical models. *Biometrika* 98, 1–15. doi: 10.1093/biomet/asq060

Hubner, N., Wallace, C. A., Zimdahl, H., Petretto, E., Schulz, H., Maciver, F., et al. (2005). Integrated transcriptional profiling and linkage analysis for identification of genes underlying disease. *Nat. Genet.* 37, 243–253. doi: 10.1038/ng1522

Jansen, R., and Nap, J. (2001). Genetical genomics: the added value from segregation. *Trends Genet.* 17, 388–391. doi: 10.1016/S0168-9525(01)02310-1

Lauritzen, S. L. (1996). *Graphical Models*. Oxford: Clarendon Press.

Li, B., Chun, H., and Zhao, H. (2012). Sparse estimation of conditonal graphical models with application to gene networks. *J. Am. Stat. Assoc.* 107, 152–167. doi: 10.1080/01621459.2011.644498

Li, H., and Gui, J. (2006). Gradient directed regularization for sparse gaussian concentration graphs, with applications to inference of genetic networks. *Biostatistics* 7, 302–317. doi: 10.1093/biostatistics/kxj008

Petretto, E., Mangion, J., Dickens, N. J., Cook, S. A., Kumaran, M. K., Lu, H., et al. (2006). Heritability and tissue specificity of expression quantitative trait loci. *PLoS Genet.* 2:e172. doi: 10.1371/journal.pgen.0020172

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. B* 58, 267–288.

Watson, R., Kanzaki, M., and Pessin, J. (2004). Regulated membrane trafficking of the insulin-responsive glucose transporter 4 in adipocytes. *Endocr. Rev.* 25, 177–204. doi: 10.1210/er.2003-0011

Yin, J., and Li, H. (2011). A sparse conditional gaussian graphical model for analysis of genetical genomics data. *Ann. Appl. Stat.* 5, 2630–2650. doi: 10.1214/11-AOAS494

Yuan, M., and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika* 94, 19–35. doi: 10.1093/biomet/asm018

Zou, H. (2006). The adaptive lasso and its oracle properties. *J. Am. Stat. Assoc.* 101, 1418–1429. doi: 10.1198/016214506000000735

Conflict of Interest Statement: 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.

Received: 18 September 2013; accepted: 01 December 2013; published online: 17 December 2013.

Citation: Chun H, Chen M, Li B and Zhao H (2013) Joint conditional Gaussian graphical models with multiple sources of genomic data. Front. Genet. ***4****:294. doi: 10.3389/fgene.2013.00294*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

REVIEW ARTICLE
published: 17 December 2013
doi: 10.3389/fgene.2013.00290

Genetic interaction networks: better understand to better predict

Benjamin Boucher and Sarah Jenna*

Laboratory of Integrative Genomics and Cell Signalling, Pharmaqam, Biomed, Department of Chemistry, Université du Québec à Montréal, Montréal, QC, Canada

Edited by:
Benjamin Haibe-Kains, Institut de Recherches Cliniques de Montréal, Canada

Reviewed by:
Ao Li, University of Science and Technology of China, China
Ivan Merelli, Institute for Biomedical Technologies, Italy

****Correspondence:***
Sarah Jenna, Laboratory of Integrative Genomics and Cell Signalling, Pharmaqam, Biomed, Department of Chemistry, Université du Québec à Montréal, CB4010, Case postale 8888, Succursale Centre-ville, Montréal, QC H3C 3P8, Canada
e-mail: jenna.sarah@uqam.ca

A genetic interaction (GI) between two genes generally indicates that the phenotype of a double mutant differs from what is expected from each individual mutant. In the last decade, genome scale studies of quantitative GIs were completed using mainly synthetic genetic array technology and RNA interference in yeast and *Caenorhabditis elegans*. These studies raised questions regarding the functional interpretation of GIs, the relationship of genetic and molecular interaction networks, the usefulness of GI networks to infer gene function and co-functionality, the evolutionary conservation of GI, etc. While GIs have been used for decades to dissect signaling pathways in genetic models, their functional interpretations are still not trivial. The existence of a GI between two genes does not necessarily imply that these two genes code for interacting proteins or that the two genes are even expressed in the same cell. In fact, a GI only implies that the two genes share a functional relationship. These two genes may be involved in the same biological process or pathway; or they may also be involved in compensatory pathways with unrelated apparent function. Considering the powerful opportunity to better understand gene function, genetic relationship, robustness and evolution, provided by a genome-wide mapping of GIs, several *in silico* approaches have been employed to predict GIs in unicellular and multicellular organisms. Most of these methods used weighted data integration. In this article, we will review the later knowledge acquired on GI networks in metazoans by looking more closely into their relationship with pathways, biological processes and molecular complexes but also into their modularity and organization. We will also review the different *in silico* methods developed to predict GIs and will discuss how the knowledge acquired on GI networks can be used to design predictive tools with higher performances.

Keywords: genetic interaction, network, conservation, prediction, *Saccharomyces cerevisiae, Caenorhabditis elegans,* genomics

WHAT IS A GENETIC INTERACTION?

GENERAL DEFINITION

The term genetic interaction (GI) covers a group of functional relationships between genes. One kind of these relationships, called epistasis, was first defined by Bateson and Mendel (1909). Biological epistasis was then described as the effect of one allele masking the effect of another one (Moore, 2003). Nine years later statistical epistasis, originally called "epistacy," was described by Fisher (1919) as a significant deviation of the phenotype of a double mutant from what is expected considering the phenotypes of the single mutants.

This statistical epistasis enabled the identification of an array of different GIs. One popular classification of these GIs consists of dividing them in two main classes: the negative and the positive interactions. The negative GIs, called also aggravating or synergistic interactions, refer to an observed phenotype higher than expected when considering the phenotypes of single mutants and assuming that the mutated genes function independently one from the other (**Figure 1**). A synthetic lethal interaction, which is an extreme case of negative GI, occurs when both single mutants are viable but the double mutant is lethal (**Figure 1**). At the opposite, the positive GIs can be subdivided in buffering/alleviating interactions where the biological effect of an allele is mitigated by a second one, and also the suppressive interactions in which the double mutant is healthier than the sickest single mutant (**Figure 1**).

As mention above, identification of statistical epistasis depends on the calculation of the expected phenotype of the double mutant considering the phenotype of the single mutants and assuming a functional independency of the two mutated genes. Several models exist and are used to estimate this expected value. For developmental and population geneticists, the quantitative assessment of a phenotype involves the statistical assessment of its penetrance – the statistical occurrence of a phenotype in a group of known genotypes – considering its expressivity. A threshold is then usually set for the expressivity of the phenotype – the degree to which the phenotype expression differs among individuals – to measure the penetrance (Miko, 2008).

The development of additive, multiplicative, Min and Log models to calculate the expected phenotype of double mutants was mostly motivated by the development of systematic and large-scale

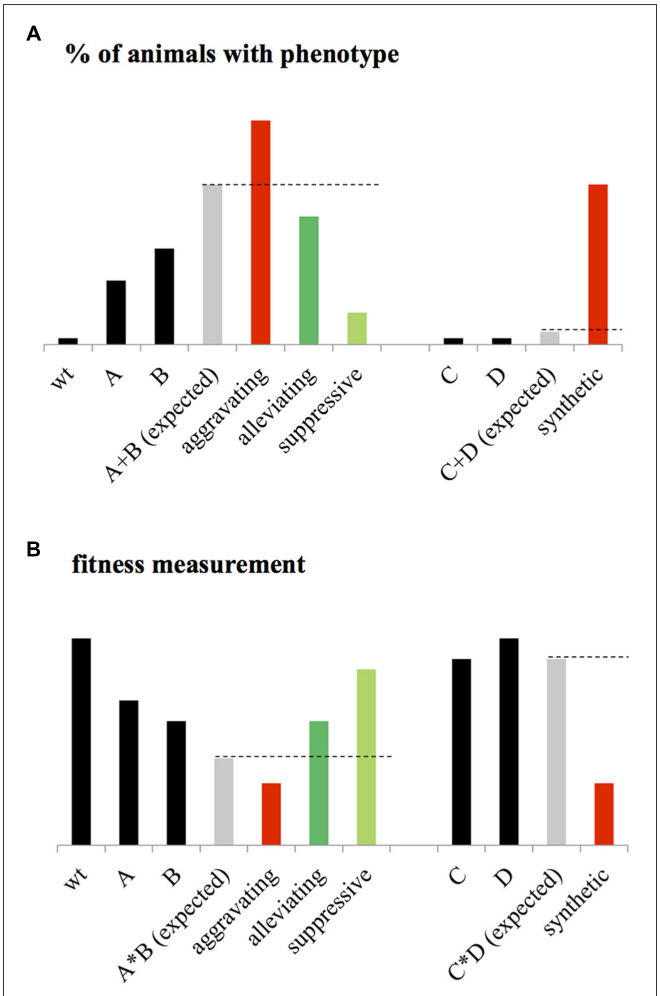


FIGURE 1 | Statistical epistasis. (A) When considering the penetrance of a given phenotype as the percentage of animals expressing this phenotype at a given "significative" level, genetic interactions (GIs) are usually identified using the additive model. Considering the phenotype of wild-type (*wt*) animals, close to zero, the expected phenotype of the double mutant AB corresponds to the sum of the phenotypes of mutant A and B. An aggravating GI between A and B is then identified if the phenotype of AB is significantly higher than the expected. An Alleviating GI is identified if the phenotype of AB is significantly lower than expected. A suppressive interaction is identified if the phenotype of AB is lower than the single mutant with the highest penetrance. When considering two mutants C and D with no observable phenotype, a synthetic interaction is identified if the double mutant CD expresses a significant phenotype. **(B)** When fitness is measured as a phenotype, the *wt* animals present high fitness rate, the expected phenotype of the double mutant AB is calculated using the multiplicative phenotype (it could also be the Log or Min) as the product of the fitness level of A and B. An aggravating interaction is then identified if AB is significantly lower than expected. Alleviating is identified if the fitness of AB is significantly higher than expected. Suppressive interaction is identified or if the double mutant is more viable than the sickest single mutants. A synthetic interaction is identified if the double mutant presents a significant fitness defect while the two single mutants are fit.

screening of GIs, especially in the yeast *Saccharomyces cerevisiae* (Tong et al., 2001; Collins et al., 2007; Jasnos and Korona, 2007; Costanzo et al., 2010). These studies identified GIs based on fitness measurements (**Figure 1B**), a class of phenotype that is measured in terms of population allele frequency (Wolf et al., 2000; Otto and Lenormand, 2002; Puniyani et al., 2004), growth rate, or number of progeny of mutant strain relative to wild-type (Elena and Lenski, 1997; Szafraniec et al., 2003; Segre et al., 2005; Sanjuan and Elena, 2006; St Onge et al., 2007). The additive and multiplicative models, originally used by developmental geneticists (**Figure 1A**) and fitness measurements in yeast (**Figure 1B**) respectively, consider the expected phenotype of a double mutant to be the sum (or the product) of the phenotypes measured for the single mutants if the two mutated genes function independently one from the other (Mani et al., 2008). The Log model has been specifically designed to identify GIs from measurements on a logarithmic fitness scale (Mani et al., 2008). The Min model considers that for non-interacting genes, the fitness of the double mutant should be similar to the fitness of the less-fit single mutant. Although these models agree under certain circumstances, they often diverge dramatically (Mani et al., 2008). For example, while the Min model appears to be highly suitable for pairs of genes with more extreme single-mutant defects, this model is clearly not ideal for defining alleviating interactions and more particularly, several epistatic interactions for which a double mutant phenotype is similar to that of the single mutant with the most severe phenotype (St Onge et al., 2007). Unfortunately, GIs identified using this model account for a large part of all GIs found in interaction databases. This tends to bias the yeast genetic interactome against this later kind of GIs (Mani et al., 2008). Identification of GIs considering several of these models would then be an appropriate approach to enable fair comparison and integration of GIs from different screening pipeline into a homogeneous GI interactome.

LEVELS OF ABSTRACTION IN BIOLOGICAL SYSTEMS

Mapping of GI networks is an endeavor that attracted more attention with the emergence of network and systems biology approaches. Network biology consists in simplifying complex biological systems into different layers of graphical representations in which nodes correspond to physical elements (genes, protein, metabolites, RNA, etc.) and edges refer to different relationships between these elements. Systems biology, and more particularly integrative genomics, aims to better understand the structure and the functioning of the system through integration of these different networks (Ge et al., 2003).

In computer sciences, organization of systems into several abstraction levels aims to hide a certain level of detail to allow the programmer to focus on a given problem. For a computer, the lower level of abstraction would contain details on the hardware while the higher level will represent the logic of the program. In agreement with this approach, a systems biologist will consider a biological system with all its complexity and identify, from the genomic sequence to the phenotype, different levels of abstractions. At the lower level of this conceptual structure, we would find several networks representing the physical structure and organization of the genome. In these networks, nodes could be genes/coding sequences, single-nucleotide polymorphisms (SNPs) or coding sequences linked by edges representing their physical proximity and organization within chromosomes, their homology etc. (**Figure 2**, level I). The second level of abstraction would represent the expression of that genome into physical components: proteins and RNA. Edges between these elements would indicate that they are co-expressed in different contexts or that their expression profiles throughout multiple experimental conditions are highly correlated (**Figure 2**, level II; Ge et al., 2003; Vidal et al., 2011). The third level of abstraction would represent physical interactions between different elements – protein–protein (PPI), protein-DNA (PDI) or protein-RNA (PRI) interactions (**Figure 2**, level III; Vidal et al., 2011). The fourth level of abstraction will allow the visualization of the functional relationships linking these physical elements. This level would contain GI networks, signaling and metabolic pathways (**Figure 2**, level IV). The fifth level would represent biological processes. This level would contain networks where proteins implicated in the same biological process would be linked by an edge (**Figure 2**, level V). The sixth and last level of abstraction would represent phenotypes and show the relationships between elements associated with similar phenotypes and diseases (**Figure 2**, level VI). Breaking down through the different levels of abstraction aims to understand the molecular basis of higher levels. A huge amount of effort is being made to enable such a breaking down and to establish the links and the dynamics underlying the relationships between networks located at the different levels. The relationship between the second (gene expression) and the third level (mainly PPI and PDI) has been well documented. Some studies showed that interacting proteins are more likely to be encoded by genes with similar expression profiles than non-interacting proteins (Ge et al., 2001; Grigoriev, 2001; Mrowka et al., 2001; Jansen et al., 2002; Kemmeren et al., 2002). Similarly, expression profiles can be used to understand the organization and dynamics of protein interaction networks through functional characterisation of highly connected nodes (Hubs). For example, Hubs have been divided into "party" and "dating" Hubs. The former class of Hubs corresponds to proteins that tend to be co-expressed with their protein partners while the later ones are not (Han et al., 2004). Party Hubs have then been proposed to interact with all their protein partners in all biological conditions, while dating Hubs may interact with subgroups of their protein partners in certain conditions and/or environments (Han et al., 2004). PPIs and PDIs can also be used to understand the molecular basis of co-expression (Lee et al., 2002; Segal et al., 2003; Yu et al., 2003; Luscombe et al., 2004).

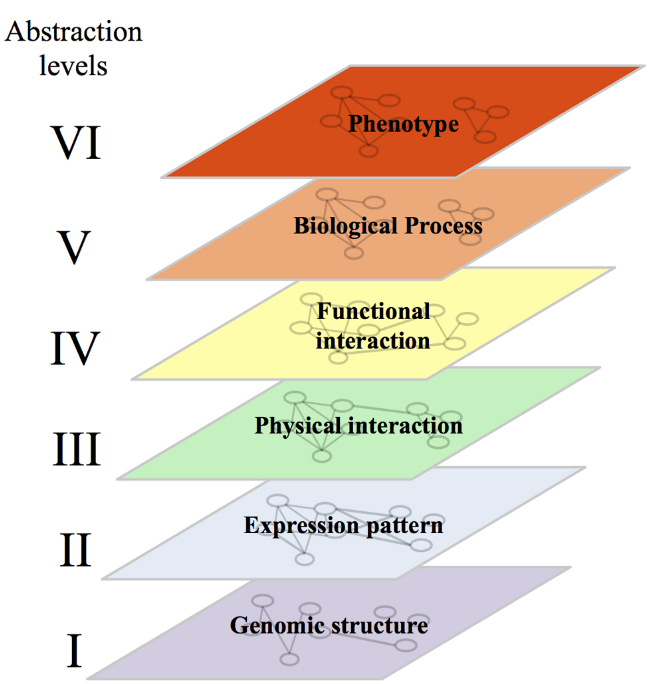


FIGURE 2 | Representation of the six levels of abstraction in biological systems. Note that, while each gene/protein can be followed from one abstraction level to another, the relationships linking it with its neighbors are different at each level. The conservation of links between two levels of abstraction in a given system and between orthologous genes/proteins in different systems are discussed in the main text of this review.

The link between the third (molecular interactions) and the fourth level (functional interactions) has also been investigated. Notably, signaling and metabolic pathways were shown to be enriched in PPIs and PDIs (Vidal et al., 2011). It is important to notice that, as detailed in the third chapter of this review, the term pathway has been assimilated in several papers as PPI and PDI modules – PPI/PDI subnetworks with a high density of links – or as dense GI network structures (Kelley and Ideker, 2005; Bellay et al., 2011a). Here, signaling and metabolic pathways will be described as a group of molecules functioning together and most of the time, in cascade to control a biological function. As detailed in the following chapters, GI networks are also linked to PPI and PDI networks (see *In Silico* Mapping of GIs). This link is however less evident than the link between PPI/PDI networks and signaling/metabolic pathways (see *In Silico* Mapping of GIs).

The relationship existing between the level six (phenotypes and diseases) and the level four (functional interactions) motivated the construction of pathway databases such as Reactome (Joshi-Tope et al., 2005) or the kyoto encyclopedia of genes and genomes (KEGG; Kanehisa and Goto, 2000), and is at the forefront of the research effort to identify therapeutic targets and pharmaceutical compounds (Yuryev, 2012).

The link between the levels four (functional interactions) and five (biological processes) is clear for signaling and metabolic pathways. Each signaling pathway, for example the EGF receptor/Ras/MAP kinase pathway, involves proteins that can be grouped based on their implication in the control of various biological processes, e.g., endocytosis, Ras regulation, actin cytoskeleton remodeling, kinase activity/phosphorylation, etc.

Abstractions levels can also be linked to distant levels. For example, GIs are shown to be enriched in co-expressed genes (Zhong and Sternberg, 2006; Lee et al., 2010a; link between the fourth and the second level). Similarly, integration of the sixth level (phenotype) to the third (PPI) permitted the construction of the human disease interactome. This interactome was proposed to support the existence of disease specific functional modules and also to help the molecular characterization of the protein products of disease genes (Goh et al., 2007).

Integration of different networks within or across abstraction levels brings substantial information on the structure of the system, and to some extent, information about its dynamics (Han et al., 2004). These pieces of information constitute, as described in this review, the baseline for the construction of predictive tools used to enrich and complete sparse networks.

We will focus, in this review, on the fourth level and more particularly, on GI networks. While this kind of functional relationship is linked to higher and lower levels of abstraction, most of these links appear much less clear than those involving signaling and metabolic pathways. We can then wonder if mapping such a network is of biological interest: would it bring complementary information to those brought from pathways dissection and significantly help understanding the functioning of the system?

WHY CONSTRUCTING A CATALOG OF GENETIC INTERACTIONS?

There are two main reasons why mapping GI networks is of biological interest. The first one is to understand the mechanisms underlying the robustness of biological systems. How the system compensate for the loss or alteration of a biological function or the alteration of its environment?

Unnecessary genes do not exist in biological systems and would be eliminated through evolutionary processes (Stern and Orgogozo, 2009). So, why 73% of these necessary genes appears not to be essential (Giaever et al., 2002)? Because compensatory relationships exist between genes, pathways, and biological processes. Therefore, mapping of GIs appears to be the best way to identify these compensatory phenomena. In addition to the high contribution this mapping will bring to basic sciences, it is also of high interest for translational research. Biological robustness is indeed, a major problem in the pharmaceutical industry with the development of resistance to therapeutic agents, particularly to anti-cancer chemotherapies (Edelman et al., 2010). Identification of compensatory relationships between genes and pathways, through mapping of GIs, appears then as an avenue that needs to be explored in parallel with the dissection of the pathways themselves.

The second reason is associated with the still mysterious relationship existing between genotype and phenotype. Population geneticists highlighted the intricate complexity of genetic variations and how positive and negative relationships between alleles influence phenotypical outcome (Gibson, 2010). Cancer modifier loci, including "susceptibility" or "resistance" alleles, are good examples of genetic variations affecting a patient phenotype, here the aggressiveness of the tumor phenotype (Dragani, 2003). Similarly, GIs and more particularly digenic synthetic GIs may underlie many common diseases that are familial but not Mendelian in their inheritance, such as glaucoma, type II diabetes, lupus erythematosus and schizophrenia (Tong et al., 2004). Exploring GI networks in model organisms, through screening of low order (between two alleles) and high-order interactions (between more than two alleles), may then help understanding the genetic networks underlying phenotypical variations and multigenic diseases (Lehner, 2011).

MAPPING GENETIC INTERACTOMES IN MODEL ORGANISMS

IN YEAST

Quantitative studies of synthetic sick or lethal (SSL) interactions in the baker's yeast *S. cerevisiae* represent most of the GIs screens done to date. The existence of mutation libraries for both essential and non-essential genes is regarded as the main reason for the development of large-scale GI studies (Giaever et al., 2002). Non-essential gene mutant libraries contain strains where single gene coding sequences are substituted by a drug-resistance marker (Giaever et al., 2002) while essential genes mutant libraries consist in a collection of conditional alleles (Tong et al., 2001; Davierwala et al., 2005; Schuldiner et al., 2005; Costanzo et al., 2010). These libraries have been extensively used in an automated methodology called synthetic genetic array (SGA; Tong et al., 2001, 2004). SGA screening consists in using single mutated yeasts as query against a whole deletion library for the construction of double mutants in a high-throughput fashion (Tong et al., 2001, 2004). The fitness defects of double mutants are then scored to uncover SSL interactions for non-essential genes (Tong et al., 2004; Sharifpoor et al., 2012) and essential genes (Tong et al., 2001; Davierwala et al., 2005; Schuldiner et al., 2005; Costanzo et al., 2010).

In parallel, the epistatic mini-array profile (E-MAP) – another variant of SGA – takes colony size measurements (based on imaging) as a basis for the detection of GIs (Schuldiner et al., 2005). GIs are then identified through measurement of a slower (SSL, negative GIs) or faster (alleviating, positive GIs) growth rate of the double mutants than what is expected from each single mutant growth rate. This allowed the identification of both positive and negative GIs while SGA was set originally to detect negative SSL GIs only. E-MAP was also used to map GIs in different yeast species such as *Schizosaccharomyces pombe* (Ryan et al., 2012).

Among the other high-throughput methods to discover GIs in yeast, diploid-based synthetic lethality analysis with microarrays (dSLAM), uses a library of barcoded mutants and barcode microarrays to measure the relative abundance of each barcoded double mutants in pooled populations to identify digenic SSL interactions (Pan et al., 2006; Lin et al., 2008). Optical density measurements (St Onge et al., 2007), biomass quantification analysis termed flux balance analysis (FBA) (Segre et al., 2005), quantitative phenotype (Drees et al., 2005) and gene expression data (Van Driessche et al., 2005) have also been employed to map GIs in specific biological processes. However, these studies remain restricted in terms of genome coverage.

IN *C. elegans*

Screening a large amount of GIs in the nematode requires the utilization of RNA interference (RNAi) through soaking animals in a solution containing RNAi molecules or feeding them with *E. coli* strains expressing the RNAi (Maeda et al., 2001; Timmons et al., 2001). This approach induces a downregulation of the expression of targeted gene, instead of a deletion. This has to be taken into consideration when comparing the *Caenorhabditis elegans* and yeast genetic interactomes (Lehner, 2007; Dixon et al., 2009). To identify a GI, either both genes are targeted using RNAi or a genetic mutant strain containing either a hypomorphic or a null allele can be submitted to RNAi targeting the other gene (Kamath et al., 2003; Lehner et al., 2006; Byrne et al., 2007). Both approaches have been used to map a quite limited area of the *C. elegans* genetic interactome (<2,000 GIs) when compared to genetic studies in yeast (>200,000 GIs; Lehner et al., 2006; Byrne et al., 2007; Tischler et al., 2008; Costanzo et al., 2010).

IN HUMAN

To identify GIs in human, apart from the RNAi treatment of specifically mutated cell lines (reviewed in Dixon et al., 2009), Lin et al. (2010) suggested an interesting method that uses radiation hybrid (RH) genotyping data sets. This approach, while being fast and inexpensive, is different than standard RNAi screening in that RH panels are used in order to "simulate" a double mutations. The simulation is done with medium-selected cells that possess extra copies of two genes and "attractive" or "repulsive" interactions are then identified whether the promoting effect of the extra copies is death or survival of the cell line respectively. The results obtained using this approach could not be easily compared to negative and positive interactions observed through gene deletion and/or expression reduction. By joining several data sets of RH panels, a network of ~6.7 million potential GIs were extracted and covered ~3.4% of all human gene pairs (Lin et al., 2010).

IN SILICO MAPPING OF GIS

Only few organisms, mainly unicellular, are amenable to an experimental mapping of GIs through genome-wide screening. Mapping of genetic interactomes in higher organisms requires development of predictive tools that allow a significant reduction of the number of gene pairs to be tested experimentally.

During the last decade, numerous strategies have been used to infer GIs in unicellular and multicellular organisms (**Table 1**; reviewed in Steen, 2012). However, to date, only *S. cerevisiae* and *C. elegans* genetic networks have gained substantial information from large-scale machine learning studies. Numbers of tools were developed to predict PPIs, co-essentiality, genes with similar functions, genes functioning in the same molecular complex and GIs. The design of these tools highlighted the intimate link existing between different networks – GI networks being used to infer PPIs and co-functionality (Tong et al., 2004; Ye et al., 2005a) and inversely PPI networks, phenotypic profiles and GO annotations being used to predict GIs as detailed below. These different predictors present also cross-specificities – GIs occurring to some extend between genes coding for interacting or non-interacting proteins, between or within-pathways/molecular modules, between genes involved in the same biological process or being involved in different and compensatory processes as discussed below.

Intuitively, we expect that the GI world constitutes a patchwork of functional relationships with distinctive properties. Predictive tools capturing different properties will then be able to identify a portion of the GI interactome and will be complementary one to another. Ultimately, acquiring a good knowledge on the molecular particularities of subclasses of GIs will lead to the design of specific and accurate predictors. To make an informed choice on the different elements that could be employed to design these predictors, we will review here the different structural and functional particularities of GIs, and detail how they have been used or could be used to generate predictor for GIs.

EXPLOITING THE PROTEIN–PROTEIN AND GENETIC INTERACTION NETWORK DENSITY AND STRUCTURE

A primary attribute of biological interaction networks, including GI networks, is a scale-free/power law distribution of connections, where most nodes are sparsely connected ("non-Hub" nodes) and few ones are highly connected ("Hub" nodes) (Watts and Strogatz, 1998; Jeong et al., 2001; Wagner, 2001; Tong et al., 2004). These networks appear also to exhibit a small-world organization – dense interacting modules are sparsely connected to other modules but with a short average path length (Watts and Strogatz, 1998; Jeong et al., 2001; Wagner, 2001).

There is a clear connection between PPI- and GI-Hubs since a protein with many interactions in the physical network (PPI-Hub) typically has also many interactions in the genetic network (GI-Hub; Ozier et al., 2003; Kafri et al., 2008). Both kinds of Hubs tend to be essential or associated with severe fitness defects, and to genetically interact with each other (Ozier et al., 2003; Davierwala et al., 2005; Lehner et al., 2006; Goh et al., 2007; Baryshnikova et al., 2010; Costanzo et al., 2010; Sharifpoor et al., 2012). Intuitively, we may see essential Hubs as a direct association with human diseases. However, it is important to notice that, while PPI-Hubs tend to be ubiquitously expressed, disease genes (such as inherited disease genes) tend to encode for PPI-non-Hubs and to be tissue specific (Goh et al., 2007; Vidal et al., 2011).

Comparative analysis of the yeast interactome networks also revealed that the "non-essential" SSL network is at least four times denser than the PPI network (Tong et al., 2004), while the "essential“ SSL network is five times denser than the "non-essential" SSL (Tong et al., 2001, 2004; Davierwala et al., 2005). The higher density of essential when compared to non-essential GI networks, suggests that essential genes are highly connected Hubs within GI networks, and that essential pathways may be connected to number of compensatory pathways (Davierwala et al., 2005; Costanzo et al., 2010). Given that 18% of all yeast genes are essential (Giaever et al., 2002; Christie et al., 2004), this also suggests that most yeast GIs may involve at least one essential gene (Davierwala et al., 2005). The higher density of GI network, when compared to PPI network, may reflect the fact that in the case of two compensatory pathways, PPIs may occur between proteins of a linear pathway, while any member of each pathway may genetically interact with any component of its own pathway or of its compensatory pathway (Tong et al., 2004).

As shown for PPI networks, the interaction density is not homogenously distributed within GI networks that are composed of dense modules (Tong et al., 2004). These structures, as detailed above and in the following sections, are enriched in interactions occurring within functional modules (such as signaling pathways or protein complexes) or between functional modules. This property of dense GI modules could directly be used to predict novel GIs within a non-saturated network. Tong et al. (2004) showed for three specific GI modules, that ~20% of genes that interact with a high number of common partners – being part of the same dense GI module – also genetically interact one with the others. This was significantly higher than what was measured in random networks (approximately 1%; Tong et al., 2004). Qi et al. (2008) extended this network analysis by including neighbors of interacting genes from any distances and by classifying those distances by the parity of the path lengths. They employed a graph diffusion kernel that uses weighted sums for different path lengths and found that odd-length kernels were better at predicting GIs while even-length kernels were more effective in finding new PPI partners (Qi et al., 2008).

Several methods have been developed to dissect complex networks into functionally meaningful modules. Using various clustering techniques, some studies reordered the GI matrix to sort genes according to the similarity of their GI profiles. Congruent genes are then defined as genes with similar GI profiles (Schuldiner et al., 2005; Ye et al., 2005b; Collins et al., 2007; Costanzo et al., 2010, 2011). The resulting map has a modular structure that distinguishes between major biological processes, such as transcription and chromatin remodeling, DNA replication and repair or sister chromatid segregation. These GI profiles then provide a powerful way to identify sets of genes functioning in the same biological process (Tong et al., 2004; Schuldiner et al., 2005; Ye et al., 2005b; Pan et al., 2006). Some of these methods have used the complex and pathway (COP) scores for finding sets of genes that are both highly correlated and that lack an aggravating GI (Schuldiner et al., 2005; Collins et al., 2006, 2007). The top-scoring gene pairs using this method included several sets of known complex or linear pathway components, as well as several predictions of novel ones (Schuldiner et al., 2005). Mutual clustering coefficient (MCC) was also employed to measure the neighborhood sharing of connections in the GI network – called congruence score (Ye et al., 2005a,b). A high score indicates that two genes share more GI partners than expected by chance. The resulting scores are then used as weight for non-directed edges linking genes within a congruence network (Ye et al., 2005b). By comparing path lengths in three types of networks (GI, genetic congruence, and protein interaction), they showed that high genetic congruence exhibits correlation with direct PPI linkage and also exhibits proportionate distance with the PPI network (Ye et al., 2005b). This congruent score can then be used to predict PPIs.

Altogether, these studies showed that the structure of the GI network contains enough information to predict novel GIs and also to predict novel PPI, highlighting the intricate relationship existing between PPI and GI networks.

By further exploiting the relationship between PPI and GI networks, Paladugu et al. (2008) showed that PPI network graph-theory properties could also be used to predict GIs. They showed that proteins coded by SSL gene pairs, as compared to non-SSL ones, tend to have higher average degree, closeness centrality, information centrality and number of mutual neighbors within PPI network (Paladugu et al., 2008). When combined, these graph-theory properties of PPI network provided a powerful tool to predict SSL GIs (Paladugu et al., 2008). Moreover, this approach showed that the PPI network alone contains enough valuable information to predict SSL interactions. This approach appears particularly useful to predict GIs in higher organisms which are hardly amenable to systematic screening of GIs while having their PPIs at least partially mapped.

Few methods used GI and PPI networks to observe the distribution of GIs within or between dense modules of physical interactions (PPI and PDI), called in these studies "pathways" (**Figures 3A,B**; Kelley and Ideker, 2005; Ulitsky and Shamir, 2007). Canonical "within and between pathway models" were originally identified by Kelley and Ideker (2005). They found that the "between pathway model," consisting of GIs occurring between dense modules of molecular interactions (**Figure 3B**), can explain three-and-a-half times as many GIs as the "within pathway" involving GIs within dense molecular interaction modules (**Figure 3A**; Kelley and Ideker, 2005). Further arguments for the prevalence of between-pathway GIs in yeast were given by Ye et al. (2005a) and Pan et al. (2006) who postulated that genes in the same pathway are expected to share common GI partners. The between and within pathway models were however shown to explain only 40% of all yeast GIs (Kelley and Ideker, 2005). Ulitsky and Shamir (2007) extended this interactome coverage by defining "pathways" as connected subnetworks within the physical interaction network rather than a dense interaction module (**Figure 3C**). This study provided a significant increase from the number of interactions explained by the Kelley and Ideker models (Ulitsky and Shamir, 2007).

Kelley and Ideker (2005) used their within and between pathway models to predict novel GIs. A five-fold cross validation technique was used to investigate the accuracy of predicting GIs using both the "within pathway model" – genes within a given pathway genetically interact more frequently than expected by chance – or using the "between pathway model" – genes in one pathway genetically interact with many of the same partners in a second pathway. They showed that both models are efficient for predicting GIs while the "between-pathway" model appears to outperform the "within-pathway model" (Kelley and Ideker, 2005).

Deeper studies on the "between and within pathways models" showed that they were often monochromatic, meaning that they were composed almost exclusively of a single type of GIs, either all negatives or all positives (Segre et al., 2005; Costanzo et al., 2010; Michaut et al., 2011). Monochromatic patterns have been used to identify biological processes and other functional modules (Segre et al., 2005; Pu et al., 2008; Jaimovich et al., 2010). Monochromatic processes are functionally diverse, but also biased (Michaut et al., 2011; Szappanos et al., 2011). For instance, microautophagy and histone exchange are monochromatic positives whereas protein import and small GTPase mediated signal transduction are monochromatic negatives (Michaut et al., 2011). Importantly, those studies showed that protein complexes are often monochromatic (Bandyopadhyay et al., 2008; Costanzo et al., 2010) and that monochromatic patterns, identified within and between biological processes, are mainly dependant on protein complexes (Michaut et al., 2011). The distinction between negative and positive interactions, when considering the relationship between PPIs and GIs, has not yet been exploited to predict GIs to the best of our knowledge. The monochromaticity and the functional bias of this monochromaticity pattern have not been exploited neither.

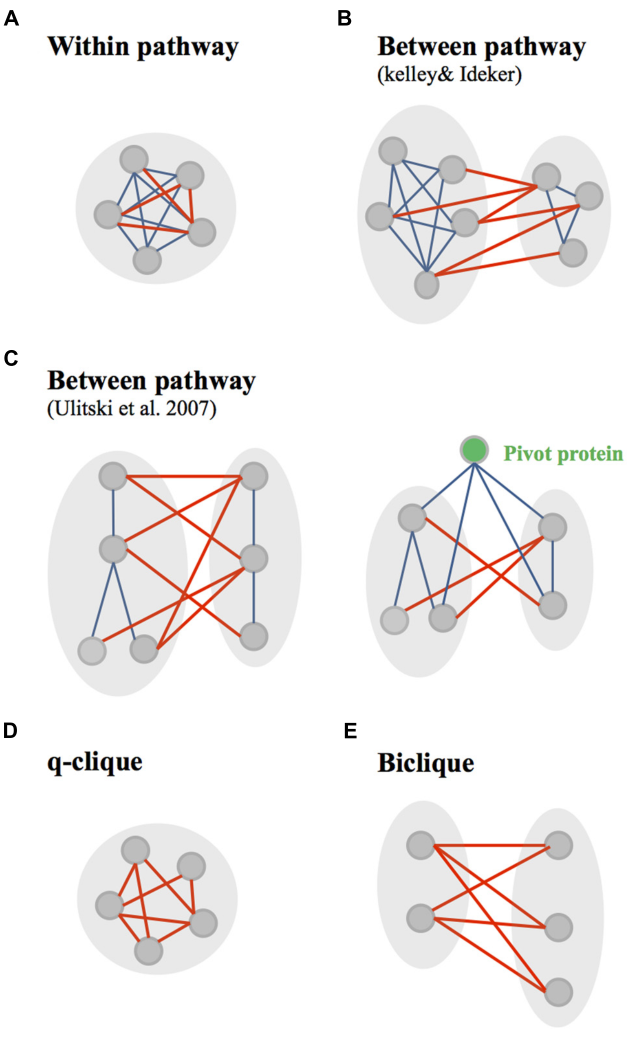


FIGURE 3 | Canonical and non-canonical within/between pathway models. (A) The canonical within pathway model as described by Kelley and Ideker, consists in genetic interactions (GIs, red edges) occurring within a dense module of protein interactions (bleu edges). **(B)** The canonical between pathway models, as described by Kelley and Ideker (2005) , consists in GIs occurring between dense modules of protein interactions. **(C)** The canonical between pathway models, as defined by Ulitsky et al., 2008, consist in GIs occurring between connected subnetworks/graph modules of protein interactions. This study also identified pivot proteins as proteins highly connected at the molecular level with component of two subnetworks connected through between pathway GIs. **(D)** Non-canonical within pathway model, are quasi-cliques (q-cliques), biclusters of highly connected genes. **(E)** The non-canonical between pathway models consist in bicliques – biclusters in which prays and target genes of GIs do not overlap.

In contrast to what was shown in yeast, the "within pathway model" tends to be more prevalent when compared to the "between pathway model" in the *C. elegans* interactome (Lehner et al., 2006; Lehner, 2007). It was suggested that this difference might come from experimental screening methodologies employed to generate the GI interactomes in different organisms (Lehner, 2007). While in yeast most of the mutations used to disrupt genes are null, in *C. elegans*, they are mainly hypomorphic. The highest number of "within pathway" interactions in *C. elegans* when compared to yeast may then be explained by the fact that hypomorphic alterations of genes functioning within the same protein complex or signaling pathway, may lead to a significant aggravation of the phenotype (synthetic interaction) while this would not be the case for null mutations (Lehner, 2007). Also, we cannot exclude the possibility that this difference might come from the intrinsic difference existing between unicellular and multicellular organisms. "Within and between-pathway models" have not been used directly to predict novel GIs in the nematode.

While it is clear that signaling pathways are enriched in molecular interaction modules, it is important to notice the potential ambiguity created by the denomination of GIs occurring between dense molecular interaction modules as "between pathways" interactions. To the best of our knowledge, it has not been clearly proved that two densely connected molecular networks may not participate to the same signaling pathway – defined as a cascade of molecular events controlling a biological function. This possibility is supported by the fact that a high number of "pathways"/molecular interaction modules defined by Kelley and Ideker (2005) as well as Ulitsky and Shamir (2007), are very small (Ma et al., 2008). Consequently, we cannot exclude the possibility that some "between pathways/molecular modules" interactions may actually occur within signaling or metabolic pathways. This taken into consideration, the fact that most GIs in yeast occurs between molecular modules and presumably pathways constitutes a golden avenue to identify compensatory pathways responsible for the cellular homeostasis and development of resistance to therapeutic agents (Tucker and Fields, 2003; Szappanos et al., 2011). This hypothesis was validated experimentally using, for example, the Cdc14 early anaphase release (FEAR) and the mitotic exit network (MEN), two parallel pathways required for the release of the essential protein phosphatase Cdc14p from nucleolus during yeast cell cycle (Stegmeier et al., 2002).

Other approaches were used to study the modularity of GI networks. The decomposition of these networks using a biclustering technic recalled the idea of congruence. This technic was used to clusters groups of genes based on their GI profiles. However, in addition to clustering, biclustering helped the identification of two kinds of motif within the GI network: bicliques and q-cliques. This decomposition of the GI networks in absence of any integration of molecular networks gave also a bright new perspective to the within/between pathway models (Bellay et al., 2011a). In this study, the between pathway model implies that GIs occurs in "bicliques" – defined as biclusters in which the query genes (first cluster of genes) and the array genes (set of genes interacting with the query genes) do not overlap (**Figure 3E**). Following the same reasoning, within pathway interactions occur in "cliques/quasi-cliques/q-cliques" – defined as biclusters in which query and array genes have significant overlap (**Figure 3D**; Bellay et al., 2011a). Interestingly, both positive and negative interactions were mainly found in bicliques (Bellay et al., 2011a), similarly to what was shown using the canonical "between pathway" model (Costanzo et al., 2011). In addition, negative q-cliques – q-cliques composed of negative interactions – which corresponded to only 9%

of negative biclusters (versus 91% of negative bicliques), did not appear to represent single protein complexes or pathways (Bellay et al., 2011a). This constitutes a major difference with the canonical "within pathway" model defined by the overlap of genetic and molecular modules (Kelley and Ideker, 2005). The genes found in negative q-cliques were found to be expressed in a coordinated manner and to be enriched for chromosome segregation and cell cycle processes (Bellay et al., 2011a). Bellay et al. (2011a) suggested that this particular functional enrichment might arise due to general sensitivity to perturbation in fragile systems such as cell division.

Altogether, these studies support the idea that different techniques used to decompose GI networks help revealing different categories of GIs. They suggest that predictive tools developed based on any of these models (the canonical "within /between pathway" model or the "biclique/q-clique" model) may be complementary to models built on the other one. The functional bias observed for different GI modules also suggests that predictive tools may gain in performance if they specifically target GIs associated to a subset of biological functions alongside homogenous particularities with respect to GI network modularity.

Network decompositions using biclustering techniques also help to provide critical information on duplicated genes (Bellay et al., 2011a). Duplicate genes were previously shown to display negative GIs with each other and exhibit fewer GIs than other genes because they tend to buffer one another functionally (VanderSluis et al., 2010). They were also shown to exhibit numerous unique GIs, suggesting that duplicated genes are functionally redundant but have divergent roles (Ihmels et al., 2007; VanderSluis et al., 2010). While, we would expect duplicated genes to be part of the isolated group of GIs within the biclustering array, a significant amount of them were fund to exhibit negative GIs with each other as part of larger modular structures (biclusters; Bellay et al., 2011a). Interestingly, this subgroup of duplicates was significantly more divergent in terms of sequence identity. It was suggested by Bellay et al. (2011a) that duplicates with a high degree of functional similarity specifically compensate for the loss of one another (isolated GIs in biclustering array), while in the second case, they appeared to have diverged into entirely different functional modules with compensatory properties (GIs being part of large biclusters). This study opens the door to predictive avenues that consider using protein sequence homology to identify compensatory genes and modules.

EXPLOITING RELATIONSHIPS BETWEEN NETWORKS AT DIFFERENT ABSTRACTION LEVELS

Networks at different abstraction levels were used to infer GIs in yeast and *C. elegans* as detailed in **Table 1** and below. These studies also brought a deeper understanding of the molecular basis of GIs (Avery and Wasserman, 1992; Guarente, 1993; Thomas, 1993).

Genetic interaction in yeast, *C. elegans* and in human, were significantly more abundant between genes sharing mutant phenotypes (abstraction level VI) or gene ontology (GO) annotations (level V), and between genes encoding proteins in the same subcellular localization (level V) and/or within the same protein complex (level III) or pathway (level IV; Lee et al., 2004, 2008; Tong et al., 2004; Kelley and Ideker, 2005; Lin et al., 2010). In agreement with the general idea that synthetic GIs may occur between genes with redundant functions, the SSL yeast network was also found to be enriched in gene pairs encoding homologous proteins (level I).

A link between two genes or their protein products within networks located at different levels of abstraction is then informative of a potential GI. An important class of predictive methodologies used these diverse sources of data to discriminate interacting from non-interacting genes. The first of these studies used decision tree learning to integrate various types of data along with a "2hop" network topology assessment for various genomic relationships (**Table 1**; Wong et al., 2004). The "2hop" method considers gene pairs linked to a common partner by a functional relationship (e.g., physical interaction and sequence homology) to be informative of a potential SSL interaction between them in yeast. In total, 123 functional relationships (26 "major" categories) were used (Wong et al., 2004). The most powerfull predictive informations were selected using a Bayesian information criterion (BIC; similar to the Akaike information criterion, AIC).

For multicellular organisms, Zhong and Sternberg (2006) integrated multiple types of data from yeast, fly and nematodes to predict 18,183 GIs in the nematode *C. elegans* (**Table 1**). Here, a logistic regression was used to integrate features (or "attributes") defined as the relative weight of a single type of data according to its predictive power. The positive set of elements used to train the model consisted in 1,816 validated GIs and 2,878 PPIs while negative examples were made of 3,296 paired *cis* markers. These makers are used in genetic mapping experiments and are assumed to have less probability of interacting together than pairs of genes randomly picked from the genome. The utilization of yeast/fly data to obtain greater genome coverage for a multitude of data sources appears to positively contribute to the predictive power of the developed tool (Zhong and Sternberg, 2006). We will discuss the limitation brought by data from other organisms in the following chapter considering evolutionary conservation of PPI and GI networks. In this study, the predictive interaction network was submitted to experimental validation using as bait *let-60*/Ras and *itr-1*/ITPR (two human disease-related genes) with a high success rate – 44 and 60% of true positive predictions respectively (Zhong and Sternberg, 2006). Although it is still unpublished, a new version of Zhong and Sternberg (2006) predictor, called "GeneOrienteer," is available online (geneorienteer.org). This model employs a naïve Bayes classifier and integrates more than 20 features to predict GIs in several species.

Another approach, developed by Chipman and Singh (2009) , used a random walks algorithm to calculates the topological similarity of two genes in many types of biological networks, including genetic and physical interactions, co-expression and GO annotation networks, for both *S. cerevisiae* and *C. elegans* (**Table 1**). This topological similarity is then used to predict negative GIs. In this study, the decision tree classifier was shown to outperform the logistic regression classifier (Chipman and Singh, 2009). The good performances of this approach, tested using cross-validation technics, was unfortunately not supported by any experimental validations.

Other studies using the likelihood scoring of gene pairs for the prediction of GIs in the nematode *C. elegans* were generated

Table 1 | *In silico* methodologies for the predictions of genetic interactions.

Reference	Type	Data	Training	Method	Number of predictive features	Experimental validation	Note
Tong et al. (2004)	Yeast SSL	Genetic interactions (GI)	~4,000 GIs	Network connectivity	1	No	GIs for ~20% of query genes
Wong et al. (2004)	Yeast SSL	Protein function and localization, gene expression, protein–protein interactions (PPI), phenotype, sequence homology	795,732 gene pairs (incl. 4,598 SSL)	Decision tree	26	Yes	Network topology and "2hop" relationships
Kelley and Ideker (2005)	Yeast SSL	PPI, protein-DNA, protein-reaction	4,849 SSLs and 27,604 PPIs for "naïve predictions"	Network connectivity	2	No	Between-pathway and within-pathway models
Zhong and Sternberg (2006)	*C. elegans* GIs	GIs/PPIs orthologs from yeast/fly; anatomical expression, phenotype, GO term, mRNA co-expression along with with orthologs (yeast/fly)	1,816 GIs; 2,878 PPIs; 3,296 *cis* markers as negatives	Logistic regression	5	Yes	18,183 high-confidence GIs covering 2,254 genes
Qi et al. (2008)	Yeast SSL	GIs	13,022 SSL	Graph diffusion kernel	1	Yes	Non-restricting distance measures to find new interacting partners
Paladugu et al. (2008)	Yeast SSL	PPIs	6,074 SSL; 400,473 negatives	Support vector machine	13	No	Graph-theoretic features using only PPIs as a network
Chipman and Singh (2009)	SSL for *S. cerevisiae* and *C. elegans*	Yeast: GO, PPI; worm: PPI with yeast/human orthologs, gene expression	Yeast: 22,432 SSL and 726,457 negatives; worm: 3,863 SSL and 58,579 negatives	Decision tree	5–6	No	Random walks algorithm to achieve topological similarity measurements between gene pairs
Lee et al. (2010b)	*C. elegans* GIs	mRNA co-expression, PPI, cocitation of genes name, phylogenetic profile analysis	626,342 GO-annotated gene pairs	Weighted sum	21	Yes	Functional network-guided predictions of genetic modifiers
Lee et al. (2010a)	*C. elegans* GIs	mRNA co-expression, PPI, phenotype	1,522 GIs	Logistic regression	6	Yes	Extended multi-species interactome and new phenotype/ PPI network-based features
Szappanos et al. (2011)	*S. cerevisiae GI degree*	*GIs*	3,572 SSL and 1,901 positive GIs	Flux balance analysis	1	Yes	Prediction of GI connectivity for metabolic genes
Koch et al. (2012)	*S. cerevisiae GI degree*	*Several types including mRNA expression, GO, PPI, copy number, phylogenetic profile analysis*	~3,500 GIs with 63.2% of all genes used in training	Decision tree	16	Yes	Prediction of GI degrees in *S. pombe*
Hoehndorf et al. (2013)	Fish, yeast, fly, worm, mouse GIs	GO, GI, PPI, phenotype	GIs and PPIs as positives	Semantic similarity (with the Jaccard index)	2	No	Prediction of GIs from infererred gene functions

soon after (**Table 1**; Lee et al., 2010a,b). The first approach, called "WormNet," is used to infer the shared function of two genes, which is also indicative of a possible GI (Lee et al., 2008). This model was trained on thousands of gene pairs sharing GO annotations. A second version of this model, called "WormNet2," employs a weighted sum instead of a naïve Bayes classifier and integrates many "updated" features derived from log likelihood scores of various functional data (Lee et al., 2010b). Contrarily to Zhong and Sternberg (2006) methodology where functional data are more intuitive (e.g., co-expression of genes), WormNet2 included some "less-common" types of data (e.g., co-citation of gene names) as features to infer shared functions (Lee et al., 2010b). Although they did not use any feature selection methodology (e.g., BIC or AIC), several examples of resulting predicted interactions by the weighted sum model showed that most features contributed to the final scores. They also succeeded in validating several GI for three signaling genes via RNAi screening but the validation success rate for individual genes appears to be low ranged from only 4% to a maximum of 15% achieved for the gene *vab*-1 (Lee et al., 2010b).

Considering the environment of genes/proteins in networks at different level of abstractions, we built an additional model: "GIFinder" (**Table 1**; Lee et al., 2010a). This tool used logistic regression and six features to predict GIs with a positive training set composed uniquely of validated GIs. This model also used novel attributes that consider the enrichment of phenotypic features in the co-expression/physical network environment of a gene. This kind of attribute integrates data from three abstraction levels (level II, III, and VI) to assess whether two genes may be part of the same functional module instead of relying only on evidences of direct interactions. These attributes also reduced the negative effect of using biological datasets with poor genome coverage and were shown to highly contribute to the overall performance of the predictor (Lee et al., 2010a). This approach would be appropriate when trying to integrate sparse data such as tissue expression profiles and subcellular localization, to other datasets with high genome coverage such as expression data. Experimental validations of predicted GIs for *gdi-1*/GDI1 – a Rho GTPase regulator associated with non-syndromic forms of mental retardation in human – supported the idea that such methodology could be useful to identify therapeutic targets for monogenic diseases from predictive GI networks of lower organisms (Lee et al., 2010a). With a success rate of at least 42%, the performance in experimental validations was comparable to that of similar approaches.

Recently, Hoehndorf et al. (2013) created a predictor of GIs for 4 different species by inferring the function of many genes using semantic similarity measurements of phenotypes and GO annotations. The semantic similarity – a measure of the distance or relatedness between two terms – was done using the Jaccard index. Unfortunately, the GIs obtained from their inferred gene functions were not validated experimentally. This later methodologies exploit only biological information located at the highest level of abstraction (level V and VI). We expect that this methodology – ignoring co-expression and molecular interaction levels – would then be able to predict GIs occurring between genes controlling a given biological process from distant environments (cell non-autonomous interactions). However, this possibility has not been investigated by the authors (Hoehndorf et al., 2013).

When trying to compare the relative performances of predictive tools, it is important to note, that while experimental validation of predictions highly contribute to the demonstration of the validity of the method, the heterogeneity of link density within the GI network and the experimental methods used to validate the interactions may highly influence the success rate of the validation. Therefore, it is extremely difficult to compare the relative performance of individual methods just by comparing the success rate of validation experiments, using one or two genes as bait, and different validation methods (mutant and RNAi, mutant and double mutant, or RNAi and double RNAi).

To assess how different integration designs impact the prediction of GIs for a given organism, we compared the predictions obtained for GeneOrienteerv2.12, GIFinder and WormNet2. Interestingly, these predictors appear to be highly complementary with more than 90% of predicted interactions by the three models being unique – i.e., predicted by only one approach (**Figure 4A**). This suggests that these three predictors capture different areas of the GI interactome covered by sets of experimentally identified GIs leaving more than 57% of it untouched (**Figure 4B**). GeneOrienteerv2.12 performed extremely well when tested on a set of 1,514 GIs obtained from interaction databases. This set of GIs, being used as a predictive feature or training sets in GIFinder and GeneOrienteerv2.12 (see "geneorienteer.org"; Lee et al., 2010a), we tested the three models on a set of recently published interactions (curated manually and absent from the databases) and observed a significant reduction in the performance of GeneOrienteer when compared to the two other models (**Figure 4C**). The deprived overlaps of predictions generated using the three predictors could be explained by the different integration methodologies used to generate the predictors (naïve Bayes classifier vs. linear regression) or by the different training sets used. The major difference of GIFinder when compared to others tools comes from the utilization of validated GIs as the only positive training examples as opposed to the two other ones that also employed physical interactions or functional annotations (Zhong and Sternberg, 2006; Lee et al., 2010a,b). While PPI and GI networks may have some overlap (some interactions occurring within protein modules), training a model using PPIs as a positive training set may bias the model toward within protein module GIs. Similar reasoning would be also valid for functional annotations. While functional annotations, such as GO annotations, are enriched between interacting genes, a large number of GIs are expected to occur between genes with different functions as discussed earlier. Interestingly, and as discussed in the following chapter, within protein module and within biological process GI appear to be more conserved that between modules or process GIs. We may then postulate that the bias induced through training the models using PPIs and GO annotations may increase the rate of evolutionary conserved interactions in the predictions. This taken into consideration, the fact that the training sets, constituted by the union of GIs and PPIs and/or pairs of genes with similar functions, is larger than validated sets of GIs only may improve the performance of predictive models using machine-learning techniques (Babyak, 2004).

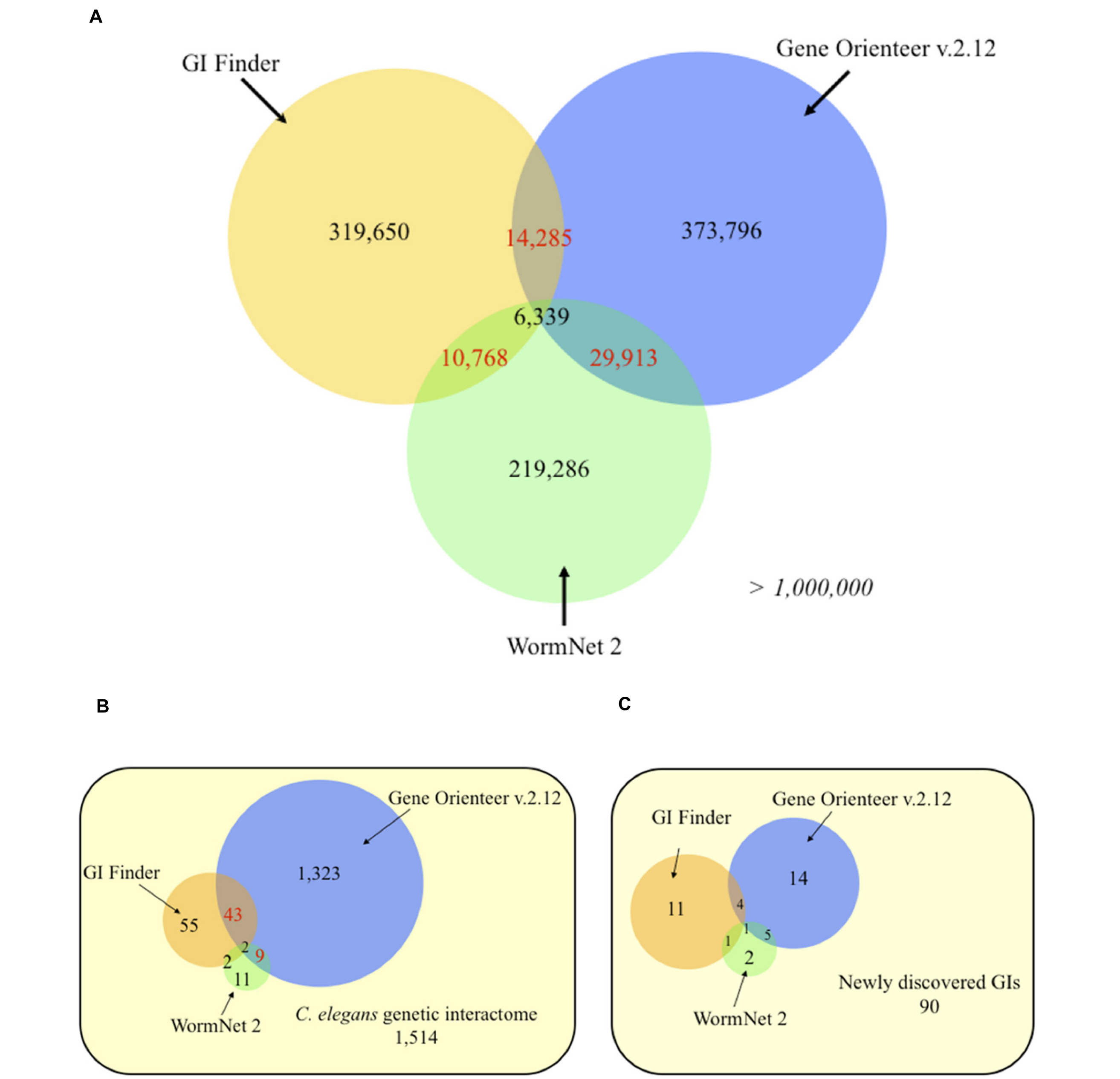


FIGURE 4 | Venn diagrams of *C. elegans* predicted genetic interactions from three different approaches. (A) Genome-wide predictions. **(B)** Experimentally validated genetic interactions taken from Lee et al. (2010a). **(C)** Experimentally validated genetic interactions (GIs) collected from recent studies (2009–2012). Numbers in red indicates statistically significant overlaps ($P < 0.05$), evaluated using an exact hypergeometric probability test. The selected score thresholds used to predict GIs yield the same false positive rate (FPR) for all three predictors. Each FPR was evaluated using a negative set consisting of 10,000 random gene pairs free of any gene present in validated interactions. Predicted GIs or functional links, for GeneOrienteer and WormNet respectively, were downloaded in October 2010.

While the existence of an edge between two genes/proteins in a network at a given level of abstraction is now confirmed as a useful information to infer a missing edge between these two genes/proteins at another level, it is important to realize that the conservation of links between two genes/proteins in different networks is a relatively rare event. For example, approximately 1% of SSL pairs (0.4% of negative and 0.5% of positive GIs in E-MAP datasets) codes for physically linked proteins (conservation of links between networks at levels III and IV) and 1% for homologous proteins (conservation of links between networks at levels I and IV; Tong et al., 2004; Costanzo et al., 2010). Cumulating these evidences of direct links between genes and proteins may increase the sensitivity of predictive tools for GIs. Considering only these direct links may also contribute to their relative poor performances. These tools would then gain in performance if integrating attributes that consider the environment of the genes in networks and the network modularity as shown by GIFinder (Lee et al., 2010a).

CONSIDERING EVOLUTION OF PROTEIN–PROTEIN AND GENETIC INTERACTION NETWORKS

Several tools used data from evolutionary distant species to predict GIs. The evolutionary conservation of these data along with the structure of interaction networks between species is then of a critical interest when considering using this information to design a powerful predictive tool. In addition, while GI interactomes are extensively mapped in certain organisms such as yeast, the utilization of these networks to predict GIs in higher organisms mainly depends on the evolutionary conservation of GIs and of the GI network structure.

Genetic interaction are known to play a critical role in evolutionary processes (Yukilevich et al., 2008; Stern and Orgogozo, 2009). In opposition to what was initially thought, all genes are not equal in the eyes of evolution, and evolutionarily relevant mutations tend to accumulate in hotspot genes at specific positions of these genes (Stern and Orgogozo, 2009). A mutation in a gene, having a high number of GI partners, would not be advantageous in a context of adaptive evolution since it will increase the phenotypic variance associated with this mutation and therefore, will cause an increased fitness fluctuation dependent on the genetic background (Stern and Orgogozo, 2009). In addition, mutations generating specific phenotypic changes are more likely to contribute to adaptive evolution than pleiotropic mutations altering several seemingly unrelated traits (Stern and Orgogozo, 2009). Genetic Hubs, being by definition connected to a large number of genes and highly enriched for pleiotropic and multifunctional genes (Costanzo et al., 2010), would then be less touched by mutations associated with adaptive evolution. As expected, GI-Hubs are highly evolutionary conserved (Bellay et al., 2011b) as are PPI-Hubs (Wuchty et al., 2006).

When considering PPIs, interactions within modules are conserved at a higher level than interactions occurring outside modules (Zinman et al., 2011). This suggests that there might be a much higher selective pressure to maintain interactions within a single module than between modules (Zinman et al., 2011). PPI networks from distant species were used in number of studies to predict GIs (**Table 1**; Zhong and Sternberg, 2006; Chipman and Singh, 2009; Lee et al., 2010a,b; Hoehndorf et al., 2013). These studies, however, did not discriminate dense modules of PPIs from non-modular interactions. Since within complex/modules PPIs were shown to be more conserved than extra-modular PPIs, it would be interesting to assess whether the utilization of modular components of PPI interactomes from distant species, instead of the complete interactome, would improve performances of predictive tools for GIs.

While the evolutionary conservation of PPI- and GI-Hubs, as well as PPIs within protein complexes/modules has been clearly established, the overall conservation of GIs between evolutionary distant species is still controversial. Comparison of the *S. cerevisiae* and *S. pombe* E-MAPs showed that negative and positive GIs of two yeast species, distant of approximately 400 million years, were significantly conserved (Sipiczki, 2000). Also, essentiality in genes appears to be highly conserved between the yeast and nematode (Kamath et al., 2003), the extent of the GI conservation between these organisms appears to be very low, and not reported as significant in all studies (Pan et al., 2004; Lehner, 2007; St Onge et al., 2007; Mani et al., 2008; Tischler et al., 2008). The difference in methodologies used to generate the GI networks between yeast and nematodes, the fact that some GIs in nematodes may not be cell autonomous because of its multi-cellularity and the poor genome coverage of *C. elegans* vs. yeast genetic interactomes may be part of the reasons behind the poor conservation of GIs observed between these organisms.

Since we expect the majority of GIs not to be conserved across species, GI-Hubs, on the other hand, appear to be well conserved throughout evolution (Lehner et al., 2006; Costanzo et al., 2010). Predicting genetic Hubs are of biological importance because of their tendency to influence fitness defects when they are individually mutated (Costanzo et al., 2010). Some high-end methodologies have been developed to predict GI degrees – the number of GIs involving a given gene – in the yeast, *S. cerevisiae* (Szappanos et al., 2011; Koch et al., 2012). The first study successfully predicts negative and positive interaction degrees for genes implicated in yeast metabolism (Szappanos et al., 2011). Using only SSL and positive GIs as training sets, they showed that only a small fraction of interacting genes shares the majority of the interactions in both empirical and *in silico* data. They also provided a mechanistic explanation for genetic "Hubs" in relation with their tendency to be multifunctional and found that the predicted negative interaction degree of a gene correlates with its multifunctionality (Szappanos et al., 2011). In another work, Koch et al. (2012) drove the analysis furthermore to predict the GI degrees of many genes in *S. cerevisiae* and also in the distantly related species *Schizosaccharomyces pombe*. They integrated 16 features; covering mRNA expressions, GO terms, PPIs and other functional data, via a decision-tree learning to predict GI degrees with only interacting genes as training sets. Among some interesting findings, they confirmed the general consensus that the GI network structure is conserved across species (Koch et al., 2012). In fact, they found retaining high conservation of GI degrees between *S. cerevisiae* and *S. pombe* for specific genes sharing a significant amount of functional information. It would be extremely interesting to carry on such study to assess whether, despite the poor conservation of GIs between yeast and nematodes, the GI network structures may also be conserved between the two organisms.

As the conservation of GI-degrees, conservation of GIs between *S. cerevisia* and *S. pombe* was significantly increased when the analysis was restricted to genes that shared the same functional annotations and when the analysis was restricted to pairs of genes coding for interacting proteins (Roguev et al., 2008). This indicates that GIs between two genes is more evolutionary conserved if these two genes are also linked in networks located at lower and higher abstraction levels. Several studies also suggested that both positive and negative GIs within functional modules (protein complexes, gene belonging to the same biological process) are significantly more conserved between *S. cerevisiae* and *S. Pombe*, than wiring between these modules (Dixon et al., 2008; Roguev et al., 2008;

Ryan et al., 2012). This suggests that not only the dependencies, but also the buffering relationships within complexes are highly conserved (Ryan et al., 2012).

While the conservation of GIs between functional modules/biological processes appears to be limited, the overall number of GIs between biological processes appears to be highly retained (Ryan et al., 2012). For example, while a significantly high number of GIs links genes controlling chromatin/transcription and those controlling mitosis and chromosome segregation in distant species, the level of conservation for individual interactions between these processes remains low (Ryan et al., 2012). This suggests that, although there is flexibility at the level of individual GIs and consequently significant rewiring between functional modules/processes in distant species, there may exist a biological selective pressure and requirements for the conservation of a high of low linking strength between particular processes (Ryan et al., 2012). Importantly, biological processes interacting with a larger amount of biological processes than expected – called here "process-Hubs" – suggest that these processes are important for mediating cross-process connections in genetic networks of several organisms (Lehner et al., 2006; Costanzo et al., 2010). For example, process-hubs such as chromatin/transcription, secretion and membrane trafficking, have been identified in *S. cerevisiae* (Costanzo et al., 2010) and *C. elegans* (Lehner et al., 2006). Conversely, some processes, such as amino acid metabolism and trans-membrane transport, have very few GIs linking them to other processes, suggesting a high degree of functional independency among these modules with less impact on other cellular processes than process-Hubs (Ryan et al., 2012).

Altogether, these data suggest that the conservation of the overall structure of GI networks still needs further characterization in distant organisms to identify the selective pressure applied on GI networks, not necessarily at the level of individual genes, but at the level of functional modules. Conclusions from such studies would bring important information that could be exploited in order to use GI networks from lower organisms to predict GIs in higher ones.

CONCLUSION AND PERSPECTIVES

Mapping of GI networks and extensive study of their structures, conservation in different species and relationships with other functional and molecular interaction networks has already provided us with a better understanding of the biological robustness and phenotypical manifestation of genomic codes. Some of these pieces of information have also been exploited to generate predictors for GIs as detailed in this review. However, to date, these tools show limited performances and gave predictions, for example in *C. elegans,* for less than 50% of the expected GI interactome. These studies also opened some paths that could be followed to improve predictive tools for GIs.

The first path suggests that tools should consider GIs in their structural context instead of considering them in isolation. This comes from several observations. The first one showed that similarity of GI profiles of two genes is more indicative of a co-functionality (sharing GO annotations, involvement in the same protein complex, etc.) than a direct GI between these genes. This comes along with the other observations that – irrespective of the method used to decompose the genetic interactome into modules – GI tends to segregate into two categories following either a "within-" or a "between-pathway" model (**Figure 5**). These two kinds of GIs, based on structural properties of the network, have also different particularities. The "between-pathway" GIs tend to be less evolutionary conserved than the "within-pathway" GIs. Similarly, at a lower level of abstraction, "between protein modules" PPIs tend to be less conserved than "within protein modules" PPIs. Overall, these data suggests that "within and between pathways" GIs may have to be assessed using different approaches. This also suggests that data used to predict GIs, such as PPIs, may also have to be considered in their modular context.

The second path of improvement for predicting GIs consists in considering GIs from a higher level of abstraction when attempting to predict GIs using data from distant species. This comes from the observation that the overall level of GIs between biological processes appears to be much more conserved between distant species than independent GIs between genes involved in

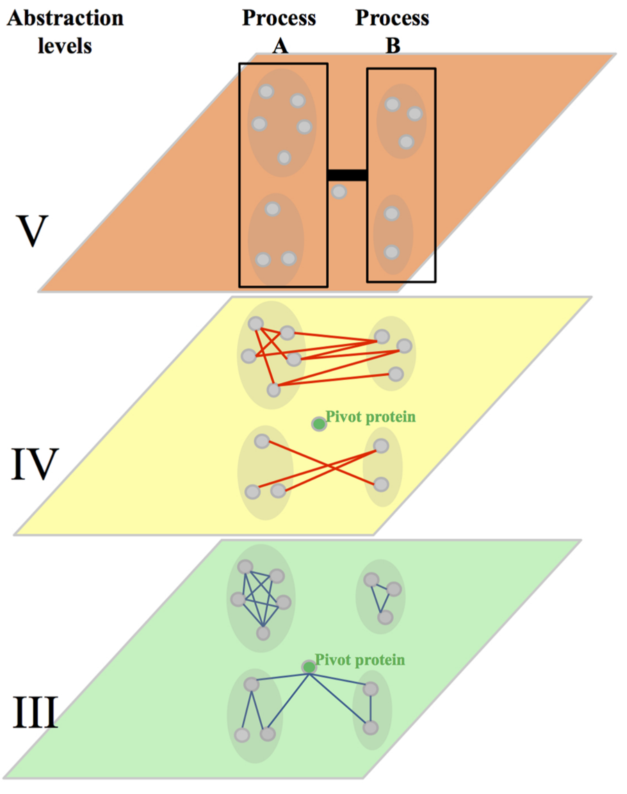


FIGURE 5 | Integration of the abstraction level III, IV, and V. Abstraction level III shows protein–protein interactions (PPIs, blue edges) within highly connected protein interaction modules. It represents also a pivot proteins highly connected with proteins of two dense modules. The abstraction level IV shows the connection of dense protein modules through genetic interactions (GIs, red edges, between pathway model). It shows also the approximate rate of within pathway and between pathways GIs observed in yeast. The level V shows the clustering of dense modules in biological processes and the link brought by GIs between these processes. The strength of that link is more evolutionary conserved than individual GIs at the abstraction level IV.

different processes (**Figure 5**). Considering GIs at the level of the biological processes (abstraction level V) instead of individual genes (abstraction level IV), may then significantly improve our ability to accurately predict functional relationships between genes and group of genes. Such approach may also open exciting opportunities. Studying the monochromaticity of GI modules also showed that the monochromatic within and between pathways interactions were biologically biased. This suggests that biological processes have either compensating or synergistic relationships one with another, but also that many components of a given biological process have predominantly either compensating or synergistic relationships. These data suggest that considering GIs from a higher level of abstraction may also be a good avenue to specifically identify synergistic and compensating/antagonistic relationships between functional biological modules. This avenue is particularly attractive when considering the need of such predictive tools in translational research and more particularly when trying to identify compensatory mechanisms leading to therapeutic drug resistance.

The last proposed path to improve GI predictions, in particular in higher organisms, is to try to better understand the structural differences that may exist between lower/unicellular and higher organisms. The fact that the within pathway model may be prevalent over the between pathway model in *C. elegans,* as opposed to yeast, need to be confirmed and the reason why this trend might be different in several organisms needs to be explained. In conclusion, while an extensive characterization of genetic networks in yeast has brought precious information about the still mysterious genetic interactome, its apparent plasticity requires similar studies to be done in higher organisms. These studies would then open the door to the design of well-informed and highly performing predictors for GIs in higher organisms such as human.

REFERENCES

Avery, L., and Wasserman, S. (1992). Ordering gene function: the interpretation of epistasis in regulatory hierarchies. *Trends Genet.* 8, 312–316. doi: 10.1016/0168-9525(92)90263-4

Babyak, M. (2004). What you see may not be what you get: a brief, nontechnical introduction to overfitting in regression-type models. *Psychosom. Med.* 66, 411–421. doi: 10.1097/01.psy.0000127692.23278.a9

Bandyopadhyay, S., Kelley, R., Krogan, N. J., and Ideker, T. (2008). Functional maps of protein complexes from quantitative genetic interaction data. *PLoS Comput. Biol.* 4:e1000065. doi: 10.1371/journal.pcbi.1000065

Baryshnikova, A., Costanzo, M., Kim, Y., Ding, H., Koh, J., Toufighi, K., et al. (2010). Quantitative analysis of fitness and genetic interactions in yeast on a genome scale. *Nat. Methods* 7, 1017–1024. doi: 10.1038/nmeth.1534

Bateson, W., and Mendel, G. (1909). *Mendel's Principles of Heredity*. Cambridge: University Press.

Bellay, J., Atluri, G., Sing, T. L., Toufighi, K., Costanzo, M., Ribeiro, P. S., et al. (2011a). Putting genetic interactions in context through a global modular decomposition. *Genome Res.* 21, 1375–1387. doi: 10.1101/gr.117176.110

Bellay, J., Han, S., Michaut, M., Kim, T., Costanzo, M., Andrews, B. J., et al. (2011b). Bringing order to protein disorder through comparative genomics and genetic interactions. *Genome Biol.* 12, R14. doi: 10.1186/gb-2011-12-2-r14

Byrne, A. B., Weirauch, M. T., Wong, V., Koeva, M., Dixon, S. J., Stuart, J. M., et al. (2007). A global analysis of genetic interactions in *Caenorhabditis elegans. J. Biol.* 6, 8. doi: 10.1186/jbiol58

Chipman, K. C., and Singh, A. K. (2009). Predicting genetic interactions with random walks on biological networks. *BMC Bioinformatics* 10:17. doi: 10.1186/1471-2105-10-17

Christie, K. R., Weng, S., Balakrishnan, R., Costanzo, M. C., Dolinski, K., Dwight, S. S., et al. (2004). *Saccharomyces* Genome Database (SGD) provides tools to identify and analyze sequences from *Saccharomyces cerevisiae* and related sequences from other organisms. *Nucleic Acids Res.* 32, D311–D314. doi: 10.1093/nar/gkh033

Collins, S. R., Miller, K. M., Maas, N. L., Roguev, A., Fillingham, J., Chu, C. S., et al. (2007). Functional dissection of protein complexes involved in yeast chromosome biology using a genetic interaction map. *Nature* 446, 806–810. doi: 10.1038/nature05649

Collins, S. R., Schuldiner, M., Krogan, N. J., and Weissman, J. S. (2006). A strategy for extracting and analyzing large-scale quantitative epistatic interaction data. *Genome Biol.* 7, R63. doi: 10.1186/gb-2006-7-7-r63

Costanzo, M., Baryshnikova, A., Bellay, J., Kim, Y., Spear, E. D., Sevier, C. S., et al. (2010). The genetic landscape of a cell. *Science* 327, 425–431. doi: 10.1126/science.1180823

Costanzo, M., Baryshnikova, A., Myers, C. L., Andrews, B., and Boone, C. (2011). Charting the genetic interaction map of a cell. *Curr. Opin. Biotechnol.* 22, 66–74. doi: 10.1016/j.copbio.2010.11.001

Davierwala, A. P., Haynes, J., Li, Z., Brost, R. L., Robinson, M. D., Yu, L., et al. (2005). The synthetic genetic interaction spectrum of essential genes. *Nat. Genet.* 37, 1147–1152. doi: 10.1038/ng1640

Dixon, S. J., Costanzo, M., Baryshnikova, A., Andrews, B., and Boone, C. (2009). Systematic mapping of genetic interaction networks. *Annu. Rev. Genet.* 43, 601–625. doi: 10.1146/annurev.genet.39.073003.114751

Dixon, S. J., Fedyshyn, Y., Koh, J. L., Prasad, T. S., Chahwan, C., Chua, G., et al. (2008). Significant conservation of synthetic lethal genetic interaction networks between distantly related eukaryotes. *Proc. Natl. Acad. Sci. U.S.A.* 105, 16653–16658. doi: 10.1073/pnas.0806261105

Dragani, T. A. (2003). 10 years of mouse cancer modifier loci: human relevance. *Cancer Res.* 63, 3011–3018.

Drees, B. L., Thorsson, V., Carter, G. W., Rives, A. W., Raymond, M. Z., Avila-Campillo, I., et al. (2005). Derivation of genetic interaction networks from quantitative phenotype data. *Genome Biol.* 6, R38. doi: 10.1186/gb-2005-6-4-r38

Edelman, L. B., Eddy, J. A., and Price, N. D. (2010). In Silico Models of Cancer. *Wiley Interdiscip. Rev. Syst. Biol. Med.* 2, 438-459. doi: 10.1002/wsbm.75

Elena, S. F., and Lenski, R. E. (1997). Test of synergistic interactions among deleterious mutations in bacteria. *Nature* 390, 395–398. doi: 10.1038/37108

Fisher, R. A. (1919). XV—The correlation between relatives on the supposition of mendelian inheritance. *Trans. R. Soc. Edinb.* 52, 399–433. doi: 10.1017/S0080456800012163

Ge, H., Liu, Z., Church, G. M., and Vidal, M. (2001). Correlation between transcriptome and interactome mapping data from *Saccharomyces cerevisiae. Nat. Genet.* 29, 482–486. doi: 10.1038/ng776

Ge, H., Walhout, A. J., and Vidal, M. (2003). Integrating 'omic' information: a bridge between genomics and systems biology. *Trends Genet.* 19, 551–560. doi: 10.1016/j.tig.2003.08.009

Giaever, G., Chu, A. M., Ni, L., Connelly, C., Riles, L., Veronneau, S., et al. (2002). Functional profiling of the *Saccharomyces cerevisiae* genome. *Nature* 418, 387–391. doi: 10.1038/nature00935

Gibson, G. (2010). Hints of hidden heritability in GWAS. *Nat. Genet.* 42, 558–560. doi: 10.1038/ng0710-558

Goh, K. I., Cusick, M. E., Valle, D., Childs, B., Vidal, M., and Barabasi, A. L. (2007). The human disease network. *Proc. Natl. Acad. Sci. U.S.A.* 104, 8685–8690. doi: 10.1073/pnas.0701361104

Grigoriev, A. (2001). A relationship between gene expression and protein interactions on the proteome scale: analysis of the bacteriophage T7 and the yeast *Saccharomyces cerevisiae. Nucleic Acids Res.* 29, 3513–3519. doi: 10.1093/nar/29.17.3513

Guarente, L. (1993). Synthetic enhancement in gene interaction: a genetic tool come of age. *Trends Genet.* 9, 362–366. doi: 10.1016/0168-9525(93)90042-G

Han, J. D., Bertin, N., Hao, T., Goldberg, D. S., Berriz, G. F., Zhang, L. V., et al. (2004). Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature* 430, 88–93. doi: 10.1038/nature02555

Hoehndorf, R., Hardy, N. W., Osumi-Sutherland, D., Tweedie, S., Schofield, P. N., and Gkoutos, G. V. (2013). Systematic analysis of experimental phenotype data reveals gene functions. *PLoS ONE* 8:e60847. doi: 10.1371/journal.pone.0060847

Ihmels, J., Collins, S. R., Schuldiner, M., Krogan, N. J., and Weissman, J. S. (2007). Backup without redundancy: genetic interactions reveal the cost of duplicate gene loss. *Mol. Syst. Biol.* 3, 86. doi: 10.1038/msb4100127

Jaimovich, A., Rinott, R., Schuldiner, M., Margalit, H., and Friedman, N. (2010). Modularity and directionality in genetic interaction maps. *Bioinformatics* 26, i228–i236. doi: 10.1093/bioinformatics/btq197

Jansen, R., Greenbaum, D., and Gerstein, M. (2002). Relating whole-genome expression data with protein-protein interactions. *Genome Res.* 12, 37–46. doi: 10.1101/gr.205602

Jasnos, L., and Korona, R. (2007). Epistatic buffering of fitness loss in yeast double deletion strains. *Nat. Genet.* 39, 550–554. doi: 10.1038/ng1986

Jeong, H., Mason, S. P., Barabasi, A. L., and Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature* 411, 41–42. doi: 10.1038/35075138

Joshi-Tope, G., Gillespie, M., Vastrik, I., D'eustachio, P., Schmidt, E., De Bono, B., et al. (2005). Reactome: a knowledgebase of biological pathways. *Nucleic Acids Res.* 33, D428–D432. doi: 10.1093/nar/gki072

Kafri, R., Dahan, O., Levy, J., and Pilpel, Y. (2008). Preferential protection of protein interaction network hubs in yeast: evolved functionality of genetic redundancy. *Proc. Natl. Acad. Sci. U.S.A.* 105, 1243–1248. doi: 10.1073/pnas.0711 043105

Kamath, R. S., Fraser, A. G., Dong, Y., Poulin, G., Durbin, R., Gotta, M., et al. (2003). Systematic functional analysis of the *Caenorhabditis elegans* genome using RNAi. *Nature* 421, 231–237. doi: 10.1038/nature01278

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

Kelley, R., and Ideker, T. (2005). Systematic interpretation of genetic interactions using protein networks. *Nat. Biotechnol.* 23, 561–566. doi: 10.1038/nbt1096

Kemmeren, P., Van Berkum, N. L., Vilo, J., Bijma, T., Donders, R., Brazma, A., et al. (2002). Protein interaction verification and functional annotation by integrated analysis of genome-scale data. *Mol. Cell.* 9, 1133–1143. doi: 10.1016/S1097-2765(02)00531-2

Koch, E. N., Costanzo, M., Bellay, J., Deshpande, R., Chatfield-Reed, K., Chua, G., et al. (2012). Conserved rules govern genetic interaction degree across species. *Genome Biol.* 13, R57. doi: 10.1186/gb-2012-13-7-r57

Lee, A. Y., Perreault, R., Harel, S., Boulier, E. L., Suderman, M., Hallett, M., et al. (2010a). Searching for signaling balance through the identification of genetic interactors of the Rab guanine-nucleotide dissociation inhibitor gdi-1. *PLoS ONE* 5. doi: 10.1371/journal.pone.0010624

Lee, I., Lehner, B., Vavouri, T., Shin, J., Fraser, A. G., and Marcotte, E. M. (2010b). Predicting genetic modifier loci using functional gene networks. *Genome Res.* 20, 1143–1153. doi: 10.1101/gr.102749.109

Lee, I., Date, S. V., Adai, A. T., and Marcotte, E. M. (2004). A probabilistic functional network of yeast genes. *Science* 306, 1555–1558. doi: 10.1126/science.1099511

Lee, I., Lehner, B., Crombie, C., Wong, W., Fraser, A. G., and Marcotte, E. M. (2008). A single gene network accurately predicts phenotypic effects of gene perturbation in *Caenorhabditis elegans. Nat. Genet.* 40, 181–188. doi: 10.1038/ng. 2007.70

Lee, T. I., Rinaldi, N. J., Robert, F., Odom, D. T., Bar-Joseph, Z., Gerber, G. K., et al. (2002). Transcriptional Regulatory Networks in *Saccharomyces cerevisiae. Science* 298, 799–804. doi: 10.1126/science.1075090

Lehner, B. (2007). Modelling genotype-phenotype relationships and human disease with genetic interaction networks. *J. Exp. Biol.* 210, 1559–1566. doi: 10.1242/jeb.002311

Lehner, B. (2011). Molecular mechanisms of epistasis within and between genes. *Trends Genet.* 27, 323–331. doi: 10.1016/j.tig.2011.05.007

Lehner, B., Crombie, C., Tischler, J., Fortunato, A., and Fraser, A. G. (2006). Systematic mapping of genetic interactions in *Caenorhabditis elegans* identifies common modifiers of diverse signaling pathways. *Nat. Genet.* 38, 896–903. doi: 10.1038/ng1844

Lin, A., Wang, R. T., Ahn, S., Park, C. C., and Smith, D. J. (2010). A genome-wide map of human genetic interactions inferred from radiation hybrid genotypes. *Genome Res.* 20, 1122–1132. doi: 10.1101/gr.104216.109

Lin, Y. Y., Qi, Y., Lu, J. Y., Pan, X., Yuan, D. S., Zhao, Y., et al. (2008). A comprehensive synthetic genetic interaction network governing yeast histone acetylation and deacetylation. *Genes Dev.* 22, 2062–2074. doi: 10.1101/gad.1679508

Luscombe, N. M., Babu, M. M., Yu, H., Snyder, M., Teichmann, S. A., and Gerstein, M. (2004). Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature* 431, 308–312. doi: 10.1038/nature02782

Ma, X., Tarone, A. M., and Li, W. (2008). Mapping genetically compensatory pathways from synthetic lethal interactions in yeast. *PLoS ONE* 3:e1922. doi: 10.1371/journal.pone.0001922

Maeda, I., Kohara, Y., Yamamoto, M., and Sugimoto, A. (2001). Large-scale analysis of gene function in *Caenorhabditis elegans* by high-throughput RNAi. *Curr. Biol.* 11, 171–176. doi: 10.1016/S0960-9822(01)00052-5

Mani, R., St Onge, R. P., Hartman, J. L. T., Giaever, G., and Roth, F. P. (2008). Defining genetic interaction. *Proc. Natl. Acad. Sci. U.S.A.* 105, 3461–3466. doi: 10.1073/pnas.0712255105

Michaut, M., Baryshnikova, A., Costanzo, M., Myers, C. L., Andrews, B. J., Boone, C., et al. (2011). Protein complexes are central in the yeast genetic landscape. *PLoS Comput. Biol.* 7:e1001092. doi: 10.1371/journal.pcbi.1001092

Miko, I. (2008). Phenotype variability: penetrance and expressivity. *Nat. Educ.* 1.

Moore, J. H. (2003). The ubiquitous nature of epistasis in determining susceptibility to common human diseases. *Hum. Hered.* 56, 73–82. doi: 10.1159/000073735

Mrowka, R., Patzak, A., and Herzel, H. (2001). Is there a bias in proteome research? *Genome Res.* 11, 1971–1973. doi: 10.1101/gr.206701

Otto, S. P., and Lenormand, T. (2002). Resolving the paradox of sex and recombination. *Nat. Rev. Genet.* 3, 252–261. doi: 10.1038/nrg761

Ozier, O., Amin, N., and Ideker, T. (2003). Global architecture of genetic interactions on the protein network. *Nat. Biotechnol.* 21, 490–491. doi: 10.1038/nbt0503-490

Paladugu, S. R., Zhao, S., Ray, A., and Raval, A. (2008). Mining protein networks for synthetic genetic interactions. *BMC Bioinformatics* 9. doi: 10.1186/1471-2105-9-426

Pan, X., Ye, P., Yuan, D. S., Wang, X., Bader, J. S., and Boeke, J. D. (2006). A DNA integrity network in the yeast *Saccharomyces cerevisiae. Cell* 124, 1069–1081. doi: 10.1016/j.cell.2005.12.036

Pan, X., Yuan, D. S., Xiang, D., Wang, X., Sookhai-Mahadeo, S., Bader, J. S., et al. (2004). A robust toolkit for functional profiling of the yeast genome. *Mol. Cell.* 16, 487–496. doi: 10.1016/j.molcel.2004.09.035

Pu, S., Ronen, K., Vlasblom, J., Greenblatt, J., and Wodak, S. J. (2008). Local coherence in genetic interaction patterns reveals prevalent functional versatility. *Bioinformatics* 24, 2376–2383. doi: 10.1093/bioinformatics/btn440

Puniyani, A., Liberman, U., and Feldman, M. W. (2004). On the meaning of non-epistatic selection. *Theor. Popul. Biol.* 66, 317–321. doi: 10.1016/j.tpb.2004.05.001

Qi, Y., Suhail, Y., Lin, Y. Y., Boeke, J. D., and Bader, J. S. (2008). Finding friends and enemies in an enemies-only network: a graph diffusion kernel for predicting novel genetic interactions and co-complex membership from yeast genetic interactions. *Genome Res.* 18, 1991–2004. doi: 10.1101/gr.077693.108

Roguev, A., Bandyopadhyay, S., Zofall, M., Zhang, K., Fischer, T., Collins, S. R., et al. (2008). Conservation and rewiring of functional modules revealed by an epistasis map in fission yeast. *Science* 322, 405–410. doi: 10.1126/science. 1162609

Ryan, C. J., Roguev, A., Patrick, K., Xu, J., Jahari, H., Tong, Z., et al. (2012). Hierarchical modularity and the evolution of genetic interactomes across species. *Mol. Cell.* 46, 691–704. doi: 10.1016/j.molcel.2012.05.028

Sanjuan, R., and Elena, S. F. (2006). Epistasis correlates to genomic complexity. *Proc. Natl. Acad. Sci. U.S.A.* 103, 14402–14405. doi: 10.1073/pnas.0604543103

Schuldiner, M., Collins, S. R., Thompson, N. J., Denic, V., Bhamidipati, A., Punna, T., et al. (2005). Exploration of the function and organization of the yeast early secretory pathway through an epistatic miniarray profile. *Cell* 123, 507–519. doi: 10.1016/j.cell.2005.08.031

Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., et al. (2003). Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.* 34, 166–176. doi: 10.1038/ng1165

Segre, D., Deluna, A., Church, G. M., and Kishony, R. (2005). Modular epistasis in yeast metabolism. *Nat. Genet.* 37, 77–83.

Sharifpoor, S., Van Dyk, D., Costanzo, M., Baryshnikova, A., Friesen, H., Douglas, A. C., et al. (2012). Functional wiring of the yeast kinome revealed by global analysis of genetic network motifs. *Genome Res.* 22, 791–801. doi: 10.1101/gr.129213.111

Sipiczki, M. (2000). Where does fission yeast sit on the tree of life? *Genome Biol.* 1, REVIEWS1011. doi: 10.1186/gb-2000-1-2-reviews1011

Steen, K. V. (2012). Travelling the world of gene-gene interactions. *Brief. Bioinform.* 13, 1–19. doi: 10.1093/bib/bbr012

Stegmeier, F., Visintin, R., and Amon, A. (2002). Separase, polo kinase, the kinetochore protein Slk19, and Spo12 function in a network that controls Cdc14 localization during early anaphase. *Cell* 108, 207–220. doi: 10.1016/S0092-8674(02)00618-9

Stern, D. L., and Orgogozo, V. (2009). Is genetic evolution predictable? *Science* 323, 746–751. doi: 10.1126/science.1158997

St Onge, R. P., Mani, R., Oh, J., Proctor, M., Fung, E., Davis, R. W., et al. (2007). Systematic pathway analysis using high-resolution fitness profiling of combinatorial gene deletions. *Nat. Genet.* 39, 199–206. doi: 10.1038/ng1948

Szafraniec, K., Wloch, D. M., Sliwa, P., Borts, R. H., and Korona, R. (2003). Small fitness effects and weak genetic interactions between deleterious mutations in heterozygous loci of the yeast *Saccharomyces cerevisiae. Genet. Res.* 82, 19–31. doi: 10.1017/S001667230300630X

Szappanos, B., Kovacs, K., Szamecz, B., Honti, F., Costanzo, M., Baryshnikova, A., et al. (2011). An integrated approach to characterize genetic interaction networks in yeast metabolism. *Nat. Genet.* 43, 656–662. doi: 10.1038/ng.846

Thomas, J. H. (1993). Thinking about genetic redundancy. *Trends Genet.* 9, 395–399. doi: 10.1016/0168-9525(93)90140-D

Timmons, L., Court, D. L., and Fire, A. (2001). Ingestion of bacterially expressed dsRNAs can produce specific and potent genetic interference in *Caenorhabditis elegans. Gene* 263, 103–112. doi: 10.1016/S0378-1119(00)00579-5

Tischler, J., Lehner, B., and Fraser, A. G. (2008). Evolutionary plasticity of genetic interaction networks. *Nat. Genet.* 40, 390–391. doi: 10.1038/ng.114

Tong, A. H., Evangelista, M., Parsons, A. B., Xu, H., Bader, G. D., Page, N., et al. (2001). Systematic genetic analysis with ordered arrays of yeast deletion mutants. *Science* 294, 2364–2368. doi: 10.1126/science.1065810

Tong, A. H., Lesage, G., Bader, G. D., Ding, H., Xu, H., Xin, X., et al. (2004). Global mapping of the yeast genetic interaction network. *Science* 303, 808–813. doi: 10.1126/science.1091317

Tucker, C. L., and Fields, S. (2003). Lethal combinations. *Nat. Genet.* 35, 204–205. doi: 10.1038/ng1103-204

Ulitsky, I., and Shamir, R. (2007). Pathway redundancy and protein essentiality revealed in the *Saccharomyces cerevisiae* interaction networks. *Mol. Syst. Biol.* 3, 104. doi: 10.1038/msb4100144

Ulitsky, I., Shlomi, T., Kupiec, M., and Shamir, R. (2008). From E-MAPs to module maps: dissecting quantitative genetic interactions using physical interactions. *Mol. Syst. Biol.* 4, 209. doi: 10.1038/msb.2008.42

Van Driessche, N., Demsar, J., Booth, E. O., Hill, P., Juvan, P., Zupan, B., et al. (2005). Epistasis analysis with global transcriptional phenotypes. *Nat. Genet.* 37, 471–477. doi: 10.1038/ng1545

VanderSluis, B., Bellay, J., Musso, G., Costanzo, M., Papp, B., Vizeacoumar, F. J., et al. (2010). Genetic interactions reveal the evolutionary trajectories of duplicate genes. *Mol. Syst. Biol.* 6, 429. doi: 10.1038/msb.2010.82

Vidal, M., Cusick, M. E., and Barabasi, A. L. (2011). Interactome networks and human disease. *Cell* 144, 986–998. doi: 10.1016/j.cell.2011.02.016

Wagner, A. (2001). The yeast protein interaction network evolves rapidly and contains few redundant duplicate genes. *Mol. Biol. Evol.* 18, 1283–1292. doi: 10.1093/oxfordjournals.molbev.a003913

Watts, D. J., and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature* 393, 440–442. doi: 10.1038/30918

Wolf, J. B., Brodie, E. D., and Wade, M. J. (2000). *Epistasis and the Evolutionary Process.* Oxford: Oxford University Press.

Wong, S. L., Zhang, L. V., Tong, A. H., Li, Z., Goldberg, D. S., King, O. D., et al. (2004). Combining biological networks to predict genetic interactions. *Proc. Natl. Acad. Sci. U.S.A.* 101, 15682–15687. doi: 10.1073/pnas.0406614101

Wuchty, S., Barabasi, A. L., and Ferdig, M. T. (2006). Stable evolutionary signal in a yeast protein interaction network. *BMC Evol. Biol.* 6:8. doi: 10.1186/1471-2148-6-8

Ye, P., Peyser, B. D., Pan, X., Boeke, J. D., Spencer, F. A., and Bader, J. S. (2005a). Gene function prediction from congruent synthetic lethal interactions in yeast. *Mol. Syst. Biol.* 1, 2005 0026. doi:10.1038/msb4100034

Ye, P., Peyser, B. D., Spencer, F. A., and Bader, J. S. (2005b). Commensurate distances and similar motifs in genetic congruence and protein interaction networks in yeast. *BMC Bioinformatics* 6:270. doi: 10.1186/1471-2105-6-270

Yu, H., Luscombe, N. M., Qian, J., and Gerstein, M. (2003). Genomic analysis of gene expression relationships in transcriptional regulatory networks. *Trends Genet.* 19, 422–427. doi: 10.1016/S0168-9525(03)00175-6

Yukilevich, R., Lachance, J., Aoki, F., and True, J. R. (2008). Long-term adaptation of epistatic genetic networks. *Evolution* 62, 2215–2235. doi: 10.1111/j.1558-5646.2008.00445.x

Yuryev, A. (2012). Contextual data integration in drug discovery. *Expert Opin. Drug Discov.* 7, 659–666. doi: 10.1517/17460441.2012.691877

Zhong, W., and Sternberg, P. W. (2006). Genome-wide prediction of C. *elegans* genetic interactions. *Science* 311, 1481–1484. doi: 10.1126/science.1123287

Zinman, G. E., Zhong, S., and Bar-Joseph, Z. (2011). Biological interaction networks are conserved at the module level. *BMC Syst. Biol.* 5:134. doi: 10.1186/1752-0509-5-134

Conflict of Interest Statement: 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.

Received: 15 September 2013; paper pending published: 03 October 2013; accepted: 28 November 2013; published online: 17 December 2013.

*Citation: Boucher B and Jenna S (2013) Genetic interaction networks: better understand to better predict. Front. Genet. **4**:290. doi: 10.3389/fgene.2013.00290*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 23 December 2013
doi: 10.3389/fgene.2013.00289

Current composite-feature classification methods do not outperform simple single-genes classifiers in breast cancer prognosis

Christine Staiger[1,2], **Sidney Cadot**[2], **Balázs Györffy**[3], **Lodewyk F. A. Wessels**[2,4,5]* and **Gunnar W. Klau**[1,6]*

[1] Life Sciences, Centrum Wiskunde & Informatica, Amsterdam, Netherlands
[2] Computational Cancer Biology, Division of Molecular Carcinogenesis, Netherlands Cancer Institute, Amsterdam, Netherlands
[3] Research Laboratory of Pediatrics and Nephrology, Hungarian Academy of Sciences, Budapest, Hungary
[4] Cancer Systems Biology Center, Netherlands Cancer Institute, Amsterdam, Netherlands
[5] Delft Bioinformatics Lab, Faculty of Electrical Engineering, Mathematics and Computer Science, TU Delft, Delft, Netherlands
[6] Operations Research and Bioinformatics, Faculty of Sciences, VU University Amsterdam, Amsterdam, Netherlands

Edited by:
Benjamin Haibe-Kains, Institut de Recherches Cliniques de Montréal, Canada

Reviewed by:
Hongying Dai, Children's Mercy Hospital, USA
Yang Dai, University of Illinois at Chicago, USA
Patrick Breheny, University of Iowa, USA
Ancha Baranova, George Mason University, USA

****Correspondence:***
Lodewyk F. A. Wessels, Computational Cancer Biology, Division of Molecular Carcinogenesis, Netherlands Cancer Institute, Plesmanlaan 121, 1066CX Amsterdam, Netherlands
e-mail: l.wessels@nki.nl;
Gunnar W. Klau, Life Sciences, Centrum Wiskunde and Informatica, Science Park 123, 1098 XG Amsterdam, Netherlands
e-mail: gunnar.klau@cwi.nl

Integrating gene expression data with secondary data such as pathway or protein-protein interaction data has been proposed as a promising approach for improved outcome prediction of cancer patients. Methods employing this approach usually aggregate the expression of genes into new composite features, while the secondary data guide this aggregation. Previous studies were limited to few data sets with a small number of patients. Moreover, each study used different data and evaluation procedures. This makes it difficult to objectively assess the gain in classification performance. Here we introduce the Amsterdam Classification Evaluation Suite (ACES). ACES is a Python package to objectively evaluate classification and feature-selection methods and contains methods for pooling and normalizing Affymetrix microarrays from different studies. It is simple to use and therefore facilitates the comparison of new approaches to best-in-class approaches. In addition to the methods described in our earlier study (Staiger et al., 2012), we have included two prominent prognostic gene signatures specific for breast cancer outcome, one more composite feature selection method and two network-based gene ranking methods. Employing the evaluation pipeline we show that current composite-feature classification methods do not outperform simple single-genes classifiers in predicting outcome in breast cancer. Furthermore, we find that also the stability of features across different data sets is not higher for composite features. Most stunningly, we observe that prediction performances are not affected when extracting features from randomized PPI networks.

Keywords: outcome prediction, breast cancer, classification, feature selection, networks, evaluation

1. INTRODUCTION

During the past decade several algorithms for predicting outcome in breast cancer based on gene expression data were developed. The first predictors used single-genes approaches that extracted genes, which were differentially expressed between the "good" outcome (metastasis-free for at least 5 years) and "poor" outcome patients (metastasis within 5 years). Two prominent gene signatures that were determined by such approaches are the gene signatures by van 't Veer et al. (2002) and Wang et al. (2005). Although these gene signatures can predict outcome, they vary substantially between data sets, and could thus not provide a homogeneous biological interpretation of the data. Moreover, Ein-Dor et al. (2005) showed in their study that there exist many other signatures that perform as well as the suggested gene signatures. This indicates that the signal is distributed over many genes which in turn makes it difficult to pinpoint one predictive network or gene signature from expression data alone. One explanation for this lies in the data. Since the underlying data are high-dimensional gene expression studies that contain many genes but only few patients, the extraction of predictor genes is prone to overtraining and may fit the noise in the data rather than explaining the underlying disease/phenotype.

Integrating gene expression data with secondary data such as pathway or protein-protein interaction (PPI) data has been proposed to address these problems and to improve outcome prediction of cancer patients (Chuang et al., 2007; Lee et al., 2008; Taylor et al., 2009; Abraham et al., 2010; Dao et al., 2010; Ma et al., 2010). These methods infer disease or subtype specific subnetworks and subpathways and use their status as features in classification. In the context of classification we call these subnetworks and subpathways composite features. In the single-genes approaches, each gene is represented by a gene expression vector across the patients, composite features carry a vector in which for each patient the expression values of the feature's member genes are aggregated. Employing composite features reduces the

Network Inference

The article describes a novel framework for evaluating network inference methods in the context of breast cancer. The inferred networks are specific for the outcome of breast cancer patients with respect to the endpoints "5-year distant metastasis free survival" and "5-year recurrence free survival." We tested the classification performance of classifiers employing the inferred networks as features and compared the performances to classifiers employing single genes. Our results show that the tested classifiers employing network-based features do not perform better than simple single-genes classifiers on the breast cancer data. However, we find evidence that network inference methods are more sensitive to the quality of the underlying data and are thus less noisy.

feature space. The underlying biological hypothesis that motivates the data integration and aggregation of genes is that genes do not act alone, and complex diseases, such as cancer, are caused by the activation or inactivation of whole pathways and protein complexes.

Previous studies exploring the use of such features were limited to few data sets with a small number of patients. Moreover, each study used different data and evaluation procedures. This makes it difficult to objectively assess the gain in classification performance and shows the need for a standardized evaluation procedure.

To overcome these problems we recently suggested a classification protocol and showed on a breast cancer cohort of ∼900 samples that current composite-feature classification methods do not outperform simple single-genes classifiers in predicting outcome in breast cancer (Staiger et al., 2012). Similar findings have been reported in (Cun and Fröhlich, 2012). Furthermore, we showed that the gene signatures defined by composite features are not more stable across different data sets than single genes. We found that, unexpectedly, classifiers employing composite features extracted from randomized PPI networks and pathway databases performed as well as those employing features extracted from unperturbed secondary data. In our evaluation we strictly separated between the training and the testing data by using different gene expression studies for the two steps.

Since the publication of the first composite classifiers, more gene expression data has become available. In addition, procedures to remove batch effects and merge data sets have become available. This allows the creation of much larger breast cancer gene expression data sets, resulting in more statistical power in the analyses. According to the findings by Ein-Dor et al. (2006) thousands of samples are required to generate stable gene lists for classification. In our work we pooled twelve studies to form a data set of 1600 patients. To account for the fact that we now only have one data set, we employ a double loop cross validation (DLCV) protocol (Wessels et al., 2005) that also ensures strict separation between the testing and training data. All classifications are performed by nearest mean classifiers (NMC). We chose the NMC for the following reasons: (i) the NMC provides performances comparable to other classifiers on expression data (Wessels et al., 2005; Popovici et al., 2010), (ii) the NMC is a simple base-line classifier, and (iii) compared to other non-linear classifiers it offers an easier way to biologically interpret the use of features.

In this work, we introduce the Amsterdam Classification Evaluation Suite (ACES), an implementation of the DLCV protocol. ACES is a Python package to objectively evaluate classification and feature-selection methods and contains methods for pooling and normalizing Affymetrix gene expression microarray data from different studies. In the provided software package both schemes, the DLCV and the previously published pipeline (Staiger et al., 2012), can be applied in the evaluation procedure.

ACES is simple to use and therefore facilitates the comparison of new approaches to best-in-class approaches. In addition to the methods described in (Staiger et al., 2012), we include here the well-established prognostic gene signatures proposed by van 't Veer et al. (2002) and Wang et al. (2005), the recent composite-feature selection method by Dao et al. (2010) and two network-based gene-ranking methods by Morrison et al. (2005) and by Winter et al. (2012). To analyse classification performances we employ a much larger cohort of patients. In contrast to the paired data set evaluation in Staiger et al. (2012) we describe here an evaluation framework that makes use of a DLCV, which facilitates the evaluation of classifiers on one large data set. Furthermore, we provide a concise correction for batch effects. In addition to the above-mentioned NMC, the software package contains an implementation of the logistic regression and the *k*-nearest neighbor classifier. To account for new developments in the field we provide detailed information on how to add new data to the package. Furthermore, we dedicate a tutorial on how to insert new feature-selection methods into ACES.

Applying ACES to a large breast cancer cohort confirms the findings of our previous study, that is, (i) none of the evaluated methods performs better than a simple single-genes classifier; (ii) features extracted by the methods are as stable as single genes, and (iii) randomizing the secondary data source has no effect on the classification performance.

The software package ACES, the normalization and merging package for gene expression data and all raw results can be downloaded from http://ccb.nki.nl/aces/.

2. MATERIALS AND METHODS

2.1. CLASSIFICATION

Classifiers were trained by a double-loop cross validation (see **Figure 1**). Since the gene signatures (Wang et al., 2005) and (van 't Veer et al., 2002) consist of a fixed set of genes, it was not necessary to run the inner CV. Hence, only one classifier for each training data set was trained employing all genes in the gene signatures. All other feature selection methods provide a ranking of the features. We trained classifiers with increasing number of features up to 400 features. Features were added sequentially to the classifiers according to the order in the ranking.

The package provides the nearest mean classifier (NMC) with four different scoring metrics based on the cosine distance and the Euclidean distance. Here we use a metric (V1), that projects the sample to the straight line connecting the two class means and normalizes the value; points that project closer to the mean of the poor outcome patients μ_{poor} are scored as zero, points that project closer to the mean of the good outcome μ_{good} patients are scored as one. The three other metrics are described in Supplement section 8. We also provide the code for a k-nearest neighbor classifier and the logistic regression.

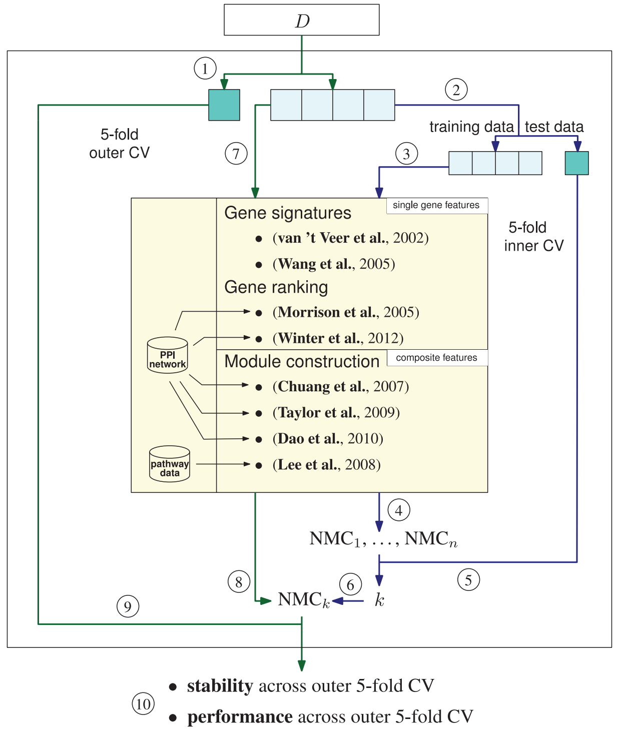


FIGURE 1 | Amsterdam classification evaluation suite (ACES). ACES executes a double-loop cross validation (DLCV) to train classifiers with features extracted by different feature extraction methods. The DLCV consists of two nested fivefold cross-validations (CV), an outer and an inner CV. **1**, Data set D is split into five parts of which one is reserved as test data and four parts are used as the training data for the outer CV loop. To determine the best number of features an inner CV is executed. The blue arrows denote the connection between the outer and the inner cross validation. Inner CV (blue arrows): **2**, The training data of the outer CV is split into five parts, four parts serve as training data, the remaining part is used as test data for the inner CV. **3**, Features are determined with one of the methods listed in the yellow box. The methods returns n ranked features—either single genes or composite features. Note that due to significance testing the number of returned features is not known in advance for some of the methods. We set the maximum for n to 400. **4**, Nearest mean classifiers (NMC) are trained by sequentially adding the features according to their ranking. The index corresponds to the number of features employed in the training. Thus, NMC_i is an NMC trained on the top i features. **5**, The performance of the NMCs is tested on the reserved test data of the inner CV. **6**, Steps 3 and 4 are repeated until each of the five splits was used as test data yielding five performances for each number of features. The index k of the best performing NMC gives the number of features that will be employed in the outer CV. Outer CV (green arrows): **7**, k features are extracted on the four training splits, i.e., the training data of the outer CV. **8**, An NMC with the top k ranked features is trained on the training data. **9**, The classifier's performance is tested on the fifth split. **10**, After completing the outer CV, i.e., each split was employed once as test data, we receive five performances and five sets of features.

2.2. EXPRESSION DATA

We compiled a large cohort of breast cancer samples from NCBI's Gene Expression Omnibus (GEO) (see **Table 1**) as it was suggested in (Györffy and Schäfer, 2009). We only took samples from the U133A platform into account and removed duplicate samples, that is, samples that occur in several studies under the same GEO id. Array quality checks were executed for all samples belonging to the same study by the R package `arrayQualityMetrics`. Due to high memory demands of this package, studies containing more than 400 samples had to be divided into two parts. Samples that were classified as outliers in the RLE or NUSE analysis were discarded. Finally, all samples across all studies were normalized together using R's `justRMA` function yielding for each sample and each probe a log(intensity) value. This normalization also included a quantile normalization step. Subsequently, probe intensities were mean centered, yielding for each sample and each probe p a $\log(\frac{\text{intensity}}{\mu(\text{intensity}_p)})$ value.

We found batch effects within single studies, where samples have been collected from different locations and batch effects between studies. Specifically for breast cancer, samples also form

Table 1 | Datasets.

Label	Data set	Geo accession (GSE)	No. of poor	No. of good
DMFS	Ivshina	4922	6	29
	Hatzis-Pusztai	25066	102	48
	Desmedt-June07	7390	36	146
	Miller	3494	7	33
	Schmidt	11121	24	145
	Loi	6532	15	32
	Total		190	433
RFS	Ivshina	4922	30	72
	Hatzis-Pusztai	25066	102	48
	Desmedt-June07	7390	56	127
	Minn	2603	21	44
	Miller	3494	21	68
	WangY-ErasmusMC	2034	88	169
	Schmidt	11121	24	145
	Pawitan	1456	33	114
	Symmans	17705	37	187
	Loi	6532	24	33
	Zhang	12093	9	112
	WangY	5327	10	42
	Total		455	1161

Shown are the original studies from which the two data sets U133A-DMFS and U133A-RFS were compiled. The patient labels "good" and "poor" correspond to 5 year distant metastasis free survival (DMFS) and recurrence free survival (RFS).

batches according to the five subtypes of breast cancer: luminal A, luminal B, Her2 enriched, normal like and basal like. To account for these effects we employed R's `combat`, where the cancer subtype was modeled as an additional covariate to maintain the variance associated with the subtypes. To do so we needed to stratify the patients according to the subtype. Since this variable is not always available in the annotation of the patients, we predict the subtype employing the PAM50 marker genes as documented in R's `genefu` package.

Principal component analysis of the batch corrected data revealed pairs of samples with a very high correlation (>0.9). Those pairs were regarded as replicate samples. For each pair of replicate samples one sample was removed randomly. Affymetrix probe IDs were mapped to Entrez Gene IDs via the mapping files provided by Affymetrix. Only probes that mapped to exactly one Gene ID were taken into account and probes starting with AFFX were discarded. If an Entrez Gene ID mapped to several Affymetrix probe IDs, probes were considered in the following order according to their suffix (Gohlmann and Talloen, 2010): "_at," "s_at," "x_at," "i_at," and "a_at." When there were still several probes valid for one Gene ID, the Affymetrix probe with the higher variance of expression values was chosen.

The patients' class labels corresponding to recurrence free or distant metastasis free survival were calculated with respect to a 5-year threshold. The final cohort is shown in **Table 1**. We derived two data sets: one labeled according to recurrence free survival (RFS) and one labeled according to distant metastasis free survival (DMFS). Note, that the DMFS data set is a subset of the RFS data set.

We provide all of the code, data, secondary data and the procedure for normalization, sample selection and batch correction as a package at http://ccb.nki.nl/aces/.

2.3. SECONDARY DATA

2.3.1. KEGG

We collected all pathway information for *Homo sapiens* (hsa) from the KEGG database (Kanehisa et al., 2010), version December 2010. The considered pathways are metabolic pathways, pathways involved in genetic information processing, signal transduction in environmental information processing, cellular processes and pathways active in human disease and drug development. We obtained 215 pathways. In this way we obtained a network composed of 200 pathways containing 4066 nodes and 29972 interactions of which 3249 nodes are also contained in the expression sets.

2.3.2. MsigDB

As second pathway database we used the C2 collection of the MsigDB (Subramanian et al., 2005) (version 3.0), which was also used in Lee et al. (version 1.0). It contains gene sets from pathway databases such as KEGG, gene sets made available in scientific publications and expert knowledge. We obtained 3272 gene sets of which 3000 could be entirely or partially covered by genes in the expression data. The MsigDB does not contain any edges, thus this database was only usable for the algorithm by Lee et al. (2008).

2.3.3. HPRD9

The protein-protein interactions were derived from the literature. We employed the HPRD version 9 (Prasad et al., 2009). The HPRD contains 9231 proteins and 35853 interactions. The protein ids were mapped to their corresponding Entrez Gene IDs. There are 7728 genes contained in both the HPRD and the expression sets.

2.3.4. OPHID/I2D

The OPHID/I2D database, downloaded in April 2011, combines protein-protein interactions from BIND, HPRD and MINT as well as predicted interactions from yeast, mouse and *C. elegans*. The database contains 12643 nodes and 142309 edges. 10018 of the nodes are also present in the breast cancer studies examined here.

2.3.5. PPI network curated by Chuang et al. (NetC)

Chuang et al. (2007) gathered a PPI of 57228 interactions and 11203 nodes of which 8572 are contained in the cohort. The source of the interactions are yeast two hybrid experiments and interactions predicted from co-citation.

2.4. FEATURE SELECTION METHODS

Let $\mathcal{E}$ be the expression data matrix where $\mathcal{E}_{pj}$ is the expression of gene j in patient p. The set of genes is denoted by G. We denote the patient's class label by c_p where $c_p = 0$ indicates a "good" outcome patient and $c_p = 1$ indicates a "poor" outcome patient. Similarly, we denote the patient's survival time as $t_p \in \mathbb{R}$.

A PPI network is defined as a graph $\mathcal{N} = (G, E)$ where G is the set of genes and edges E denote interactions between genes. A pathway is an unsorted set of genes $G' \subseteq G$.

2.4.1. Gene signatures Wang et al., 2005 and van 't Veer et al., 2002

We included two gene signatures for predicting distant metastasis free survival based on gene expression data, the signature by van 't Veer et al. (2002) and by Wang et al. (2005). Each gene j is used as one feature in the classifier and the value for each of these features is the gene's expression value for a patient p. Both signatures are actually probe signatures. The signature by Wang et al. (2005) (Erasmus) was determined on the Affymetrix U133A array, thus all probes are also present in the two data sets we generated. The 76 probes map to only 66 unique geneIDs.

The signature by van 't Veer et al. (2002) (NKI) was determined on an Agilent platform. This required the probes to be matched to gene IDs and then mapped to the data. Here we employed the gene ID collection from the MsigDB 'VANTVEER_BREAST_CANCER_POOR_PROGNOSIS' pathway as gene signature. From this pathway 41 genes were also present in the two data sets.

2.4.2. Single-genes and random genes—the benchmark methods

The single-genes approach ranks all genes G by their t-statistic between the good and poor outcome patients. The top n genes are used in an NMC and the expression values of the top n genes serve as the feature values for each patient. To determine the genes to be employed in a random single-genes classifier we simply randomly selected n genes from the total set of genes.

2.4.3. GeneRank (Morrison et al., 2005) and Winter et al., 2012

The GeneRank algorithm (Morrison et al., 2005) and the method by Winter et al. (2012) are based on Google's page rank algorithm (Page et al., 1999). The vector of gene ranks r is calculated as follows:

$$(I - dW^tD^{-1})r = (1 - d)r^0 \tag{1}$$

where I is the identity matrix, W^t is the transpose of the PPI network's adjacency matrix, $D = \text{diag}(\deg(j) + 1)$ for $j \in G$ and r^0 is the vector of initial ranks. The vector r contains for each gene the resulting rank. The degree of genes was incremented by one to allow singleton genes to be included in the calculation. The parameter d is called the damping factor and regulates the influence of the network on the rank. If $d = 1$ gene ranks are determined by the network only whereas with $d = 0$ each gene keeps its initial rank.

As initial ranks for GeneRank we chose the absolute difference of average expression between the "poor" outcome patients and the "good" outcome patients, as it was suggested in the original paper. Additionally, we calculated classification performances with the initial ranks being the t-statistic between the two patient groups.

The original Winter method proposed the correlation coefficient between the survival times of the patients and the genes' expression values. Additionally, we considered the correlation between the patients' class labels.

2.4.4. Chuang et al., 2007

This method determines subnetworks with the aim to distinguish between "good" and "poor" outcome patients. The discriminatory power of a subnetwork is evaluated by the mutual information score between the discretized average gene expression (Equation 2) and the patients' class labels. Given a subnetwork induced by $G' \subseteq G$, its activity score a for a patient p is given by

$$a_{G',p} = \sum_{j \in G'} \frac{e_{pj}}{\sqrt{|G'|}} \tag{2}$$

To calculate the mutual information of a subnetwork we need to calculate the activity scores for each patient and subsequently discretize them. Let a' be the vector of discretized activity scores for the network induced by G' and let c be the vector of class labels. The mutual information score for the subnetwork is defined as

$$s_{\text{MI}}(a', c) = \sum_{x \in a'} \sum_{y \in c} \rho(x, y) \log \frac{\rho(x, y)}{\rho(x)\rho(y)} \tag{3}$$

where ρ denotes the joint and marginal probability density functions.

All subnetworks are subjected to statistical tests assessing the significance with respect to the local and global null distribution of the activity scores and with respect to the null distribution of mutual information scores. We used the java package PinnacleZ as an implementation of the algorithm. PinnacleZ performs a z-normalization prior to the subnetwork search, which is depreciated in a fivefold cross validation. Therefore, we implemented a patch that skips this normalization step.

2.4.5. Taylor et al., 2009

This algorithm identifies differentially coordinated hub proteins in the PPI network. As measure for coordination the Pearson correlation is used. The coordination of a hub and one of its interactors is defined as the Pearson correlation $PC(h, i)$ between the hub's expression h and the interactor's expression i. To assess the different coordination of a hub across the two patient groups the average hub difference is calculated

$$d(h) = \frac{\sum_{i \in n(h)} |PC^0(h, i) - PC^1(h, i)|}{|n(h)|} \tag{4}$$

given the two sample classes, indicated by the superscript 0 and 1, $n(h)$ denotes the set of neighbors. All hubs are subjected to a statistical test, testing the significance of the hub difference. Only hubs with a significant hub difference are selected as features. Feature values for each patient are given by the average difference of expression between the hub and its interactors.

2.4.6. Dao et al., 2010

This method defines subnetworks that obey two criteria: they are (i) maximally densely connected and (ii) show deregulation in at least L poor outcome patients. To decide whether a gene is deregulated the expression matrix is discretized, i.e., each pair of patient and gene is assigned one of the three signs $\{+, -, 0\}$, where $+$ means the gene is overexpressed, $-$ indicates underexpression and 0 indicates that patient does not show an aberrant gene expression with respect to the cohort. Given a PPI network and a gene expression data set the algorithm first enumerates all connected subnetworks that obey the above-mentioned two criteria such that no subnetwork is a subgraph of any other subnetwork. The subnetworks are ranked based on their information gain. The parameter L was set such that at least 5% of the poor outcome patients were covered by each subnetwork. In the classification step these subnetworks served as features. To classify patients the average expression across all member genes of each subnetwork was calculated for each patient to obtain feature values.

2.4.7. Lee et al., 2008

This method extracts sub-pathways as features from a pathway database. The member genes of each pathway are ranked by their t-statistic between the "good" and "poor" outcome patients. Then the top n genes are combined by Equation 2 and their combined expression is again tested by the t-statistics. The search for the subpathway starts with the highest ranking gene and successively adds the next genes in the ranking as long as the t-statistic increases.

3. TUTORIALS

To enable a wider use of ACES and to keep the package flexible to new developments in the field we provide tutorials on how to include more expression data, PPI networks and pathway data. Further, we dedicate one tutorial to the topic of including more feature extraction methods, including methods that are developed in programming languages different from Python, and show how to create a wrapper that links the new software to ACES.

3.1. INTEGRATING NEW DATA

We created Python objects to represent the expression data, PPI networks and pathways. The class `ExpressionDataset` contains the expression matrix, patient labels and the patient class labels. PPI networks are represented by the class `EdgeSet`. Each edge is represented by a `frozenset` containing the start and end node of the edge. Weights on the edges can be stored as a `dictionary` in `EdgeSet.edgeweights`, where the key is the edge and the value is the weight. Pathways are represented by the class `GeneSetCollection`. The whole pathway database is represented as a list of lists, `GeneSetCollection.geneSets`, where each pathway itself is stored as a list of genes. The names of the pathways are stored as a list in `GeneSetCollection.geneSetsNames`.

3.1.1. Expression data

The Python script `NewDatasets.py` provides code and information on how to convert external data files into an `ExpressionDataset` and subsequently saves it in hdf5 format.

3.1.2. Network and pathway data

New PPI data should be provided as SIF formatted file and can be read in by `EdgeSet.ReadSIF`. Similarly pathway data can be read in by `GeneSetCollection. ReadGeneSetCollection`. The file format is as follows. Each line contains one gene set, and genes in a gene set are space-separated. If you want to attach names to each gene set, insert a line starting with "NAME" directly before the gene set. Examples are provided in the folder "experiments/data" in the ACES package.

3.2. INTEGRATING A NEW FEATURE SELECTION METHOD

We assume that any new feature selection method written in some programming language is provided as software that is called from command line. We further assume that all input is read in from files and all output is written to files.

To integrate a new feature selection method you will need to provide the code for the two classes `Feature ExtractionFactory` and `FeatureExtractor`. The `FeatureExtractorFactory` determines the features on a training data set and a secondary data source, whereas the `FeatureExtractor` maps the input genes from the data set to the feature space and scores each feature for each sample in the data set. We clearly divided between these two classes since they correspond to different steps in the pipeline.

3.2.1. The FeatureExtractorFactory

In the `FeatureExtractorFactory` the code that defines features is provided. When the actual feature extraction algorithm is given as an independent software package in a different language the `FeatureExtractionFactory` serves as a wrapper to connect the software to ACES. To initialize a new `FeatureExtractorFactory` the location of the executable of the software is passed to the constructor—the `__init__` function:

```
def __init__(self, softwareExecutable):
     self.executable = softwareExecutable
```

The method `train` receives all necessary data instances to extract the features. To ensure that several instances of the `FeatureExtractionFactory` can be run at the same time on the same machine we first create a temporary directory to which the input files are written. The input files can be directly created from the data instances, which contain functions to write the data as space- or tab-separated files. The format for pathways is as follows: each line contains all genes belonging to one pathway separated by spaces. The name of each pathway, if present in the `GeneSetCollection` instance, is printed in the line preceding the member genes and is indicated by the keyword "NAME." Instances of the type `EdgeSet` can be written to a space-separated sif-file or a file where each line consists of the start node, end node and the edge weight. The function `ExpressionDataset.writeToFile` writes the gene expression matrix to a tab-separated file, while all patients' class labels are saved in a separate file by the function `ExpressionDataset.writeClasslabels`.

In the example below the expression matrix is written to the file "matrix_file.txt," the patients' class labels are written to "classlabels_file.txt" and the network is written to "network_file.sif":

```
def train(self, dataset, network):
    tempdir = tempfile.mkdtemp()

    MatrixFilename = os.path.join(tempdir,
                  'matrix_file.txt')
    dataset.writeToFile(MatrixFilename)
    ClassesFilename = os.path.join(tempdir,
                  'classlabels_file.txt')
    dataset.writeClasslabels(ClassesFilename)
    NetworkFilename = os.path.join(tempdir,
                  'network_file.sif')
    network.writeSIF(NetworkFilename)
```

Next, we create the command that calls the executable with the input files. Note that the executable lies in a different directory than the input files. To achieve that also the output is written to the temporary directory we either need to copy the executable to the new location or create an option for the output in the executable. The `shutil` module provides several functions for copying files to a different location from within python. For now, we assume the executable is located in the temporary directory and the output is written to a file called "output.txt" that contains the features. The list `args` contains the complete call of the executable. You can check the correctness by `print ' '.join(args)`. The command is executed as subprocess in the temporary directory:

```
def train(self, dataset, network):

    tempdir = tempfile.mkdtemp()

    ...

    args = []
    args.extend([yourCompiler+' '+os.path.basename
        (executable)])
    args.extend([MatrixFilename, ClassesFilename,
        NetworkFilename])

    proc = subprocess.Popen (args, cwd=os.path.
      dirname(tempdir))
```

Finally, the generated output.txt needs to be read in and formatted as a list of lists, where each sublist contains the genes belonging to one feature. This is accomplished by `modules = readOutput(tempdir+'/output.txt')`, which must be provided by the user. In ACES we assume that the genes belonging to the features are not given by their name but by their index with respect to the data set used in the function `train`. Thus, if genes are given by name in the output file, we need to map them to their indices:

```
def train(self, dataset, network):

    ...

    modules = readOutput
              (tempdir+'/output.txt')
    geneLabelsToIndex = dict(zip(dataset,
                        geneLabels, xrange(len(dataset.
                        geneLabels))))
    features = [frozenset([geneLabels
                ToIndex[gene] for gene in module if
                gene in geneLabelsToIndex]) for module
                in modules]

        return NewFeatureExtractor
        (dataset.geneLabels, features)
```

The output of the `FeatureExtractorFactory` is an instance of the `FeatureExtractor` that maps an expression data set with the same genes and the same ordering of the genes as the data set employed in the `train` function to the feature space.

3.2.2. The FeatureExtractor

In the `FeatureExtractor` an input data set is mapped to the feature space and each feature is scored for each patient of the data set. Features are defined over the indices of the genes in the data set employed to determine the features. The `FeatureExtractor` is initialized with the gene space and the features it maps to. Only data sets with the same genes and the same ordering of genes can be mapped to the features:

```
def __init__(self, geneLabels, features):

        self.geneLabels          = geneLabels
        self.features            = features
        self.validFeatureCounts = range(1,
                   len(self.features) + 1)
```

The method `extract` maps the data to the first *k* features. We ensure here that there are *k* features and that the data set is defined on the correct genes:

```
def extract(self, dataset, k):

        assert all(dataset.geneLabels ==
    self.geneLabels)
        assert k in self.validFeatureCounts

        return numpy.transpose(numpy.array ([self.score
            (dataset.expressionData, feature)
            for feature in self.features[:k]]))
```

The function `score` attaches a score to each feature for each patient. In the case of single genes this would be the gene's expression value for the patients. In case of a feature consisting of multiple genes the function score needs to provide information on how to merge the genes' expression to one value. We show here an example of how to average over the genes' expression that belong to the same feature:

```
@staticmethod
def score(expressionData, feature):
    return numpy.sum(expressionData[:, list(feature)],
    axis = 1) /len(feature)
```

To store and reload a feature extractor efficiently, we provide a function `toJsonExpression` which stores all the information in a `json` document:

```
def toJsonExpression(self):
    return json.dumps((self.__class__.__name__,
        [geneLabel for geneLabel in self.geneLabels],
        [sorted(feature) for feature in self.features]))
```

The full example code is shown in Supplementary section 6.

4. RESULTS AND DISCUSSION

4.1. NETWORK AND PATHWAY-BASED METHODS DO NOT OUTPERFORM THE BENCHMARK METHODS

We evaluated the performances of nearest mean classifiers (NMC) employing the benchmark feature-selection methods “single genes,” “random genes” and gene signatures specific for breast cancer outcome, “NKI” and “Erasmus,” and compared them with the performances of classifiers employing composite features.

All classifiers were trained in the double-loop cross validation (DLCV) procedure described in **Figure 1**. The DLCV consists of two nested fivefold cross validations. In the outer CV we determine the training and testing data. From the inner CV we obtain the parameters for the outer CV's classifier and feature selection method (number of features and the damping factor for the Page Rank based algorithms Morrison et al., 2005; Winter et al., 2012). Once the inner CV is completed we use its best performing parameters to train the outer CV classifier. Thus, although having only one initial data set for training and evaluating classifiers, we strictly separate the data employed in these two steps, which ensures an unbiased evaluation.

Figures 2, **3** and Supplementary figure S1 show the results for the NMC using the V1 metric. There are no differences in performances between the different versions of the NMC. From this we conclude that the distance measure does not play a major role (the raw data for all NMCs can be downloaded at http://ccb.nki.nl/aces/). None of the composite-features classifiers significantly outperforms the single-genes classifier (see Table S1). In the Supplement sections 2.1–2.19 we show that changing the number of features does not lead to a change in performance. The feature selection proposed by Winter et al. (2012) and the GeneRank algorithm are also influenced by the damping factor. Supplementary section 3, however, shows that the classifiers performances do not vary significantly across different damping factors. This suggests that the network only has a marginal influence on the classification result.

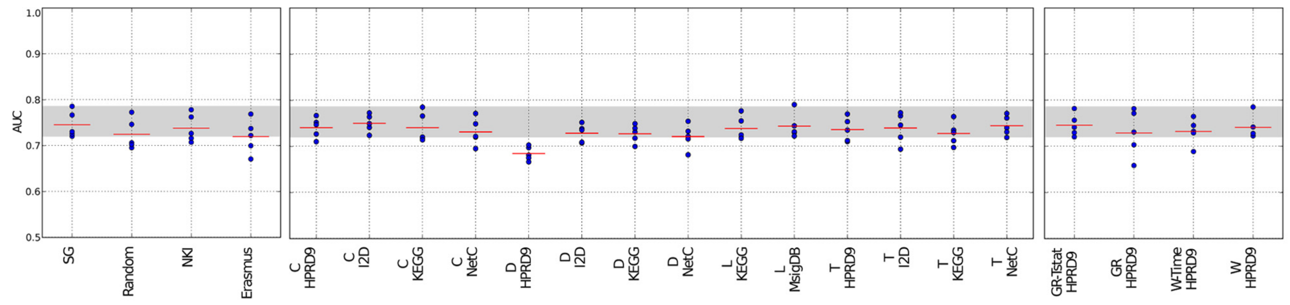


FIGURE 2 | Classification results of the NMC V1 on the DMFS data set. The number of features was trained in the inner CV. Shown are the five performances received in the outer CV. The individual AUC values are shown as blue dots. The red lines indicate the mean performances. The gray area indicates the performance interval of the single-genes classifier. SG, Single genes; L, Lee; C, Chuang; T, Taylor; D, Dao; W, Winter using association between class labels and gene expression as initial gene ranks; W-time, Winter using association between survival times and gene expression as initial gene ranks; GR, GeneRank with absolute expression difference between "good" and "poor" outcome groups as initial rank; GR-Tstat, GeneRank employing the *t*-statistic as initial gene ranks.

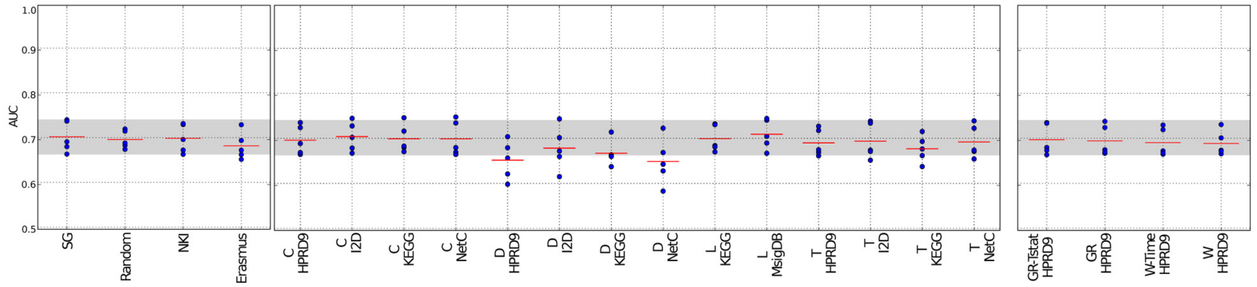


FIGURE 3 | Classification results of the NMC V1 on the RFS data set. Description as in **Figure 2**. SG, Single genes; L, Lee; C, Chuang; T, Taylor; D, Dao; W, Winter using association between class labels and gene expression as initial gene ranks; W-time, Winter using association between survival times and gene expression as initial gene ranks; GR, GeneRank with absolute expression difference between "good" and "poor" outcome groups as initial rank; GR-Tstat, GeneRank employing the *t*-statistic as initial gene ranks.

The method by Dao et al. (2010) performs worse than the benchmark methods. The reason might be that not necessarily all patients are considered during extraction of predictive network markers. In the algorithm a minimum number of "poor" outcome patients is required to be covered by each network. However, there is no constraint reinforcing that each patient is covered by the networks. This allows that the same group of poor outcome patients determines all the features and good outcome patients are neglected in this step. Thus, valuable information about patients might be lost, which, in turn, leads to higher misclassification rates.

Previously, we have shown that classifiers employing the features by Taylor et al. (2009) perform worse than the single-genes classifiers (Staiger et al., 2012). In our earlier interpretation of the algorithm each edge was regarded as a single feature. This led to an enormous feature space and to poor classification performances. Here, we keep the selection of hubs and their interactors, but in contrast to the previous classifier, we score each hub by the average expression difference between itself and all of its interactors. This decreases the feature space and leads to much better classification results. Still, the method does not outperform the benchmark methods.

4.2. NETWORK AND PATHWAY-BASED METHODS DO NOT PRODUCE MORE STABLE GENE SETS THAN THE BENCHMARK METHODS

In addition to the claim that using composite features increases classification performance, it is often stated that these features are by far more stable than single genes. Here, we analyze the overlap of composite features by means of Fisher's exact test and compare them to the overlap of single genes. Since composite features consist of many genes we considered all genes belonging to the k best performing features. Thus, the overlap of two composite-feature sets is determined by the overlap of the corresponding gene sets. Composite features are calculated from PPI and pathway data, which contain different numbers of genes and fewer genes than there are genes in the expression data. These differences have to be taken into account when comparing the overlap between gene sets. For example, when determining two composite feature sets from the KEGG database for two different data sets the overlap between the two sets is very likely to be higher than generating two feature sets for the same data on the I2D network due to the difference in size of the two PPIs. Fisher's exact test takes these differences into account. We illustrate the use of the test in Supplementary section 7. Moreover, to compare the overlap of the composite features' gene sets to single genes we have to correct for the size of the composite features since a single composite feature can contain many genes. For each training data set and each feature selection approach we obtain the *n* highest ranking features containing *m* genes. We then determine the size-matched highest scoring single genes on the same training data set of the outer CV.

Figures 4, **5** show that none of the composite features produces more stable gene sets than the single genes. In many cases the control-for-size single genes are more stable than the corresponding composite features. The overlap of randomly drawn genes is very low, as expected. Although their performance in classification is equally good as single genes, the experiment clearly shows that overlap and performance in classification are not related to each other. The method by Taylor et al. (2009) produces highly stable gene sets. The method selects hub proteins and a feature consists of the hub and all of its interactors. Thus, a large number of genes contribute to a feature. This seems to be enough to ensure a high overlap as the corresponding box of the control-for-size single genes also indicate highly stable gene sets.

From the results we can not conclude that composite-features ensure more stable gene signatures from expression data than single-genes classifiers where the genes were selected on an individual basis, given that a sufficiently large number of genes are selected.

4.3. RANDOMIZATION OF THE SECONDARY DATA SOURCES DOES NOT DECREASE CLASSIFICATION PERFORMANCES OF NETWORK AND PATHWAY-BASED METHODS

To find out whether the quality of the PPI networks have a major influence on the performance we executed randomization experiments. The nodes in each network were shuffled. By this the network topology stayed the same, but nodes that were originally hubs may now have only few neighbors and nodes with few neighbors might become hubs. For each PPI network we did this random shuffling 25 times, resulting in 25 PPI networks. Each network dependent method was then executed on each of these 25 networks using the DLCV ACES protocol. Thus, the network provides only non-sensical biological information, which in turn should hinder the methods to extract useful features. We would expect that the classification performances drop dramatically when employing these features.

Figures 6, **7** show the results for the network dependent methods executed on the shuffled I2D PPI network, the performance interval employing the original networks is depicted in gray. Figures S2, S3 show the results for the randomized HPRD9.

The methods by Chuang et al. (2007), Dao et al. (2010) and Taylor et al. (2009) do not always find features for some combinations of data set and randomized networks, i.e., the algorithms do not return features. This indicates that these methods are indeed sensitive to the quality of the network data.

The method by Taylor et al. (2009) searches for significantly altered hubs across the two conditions. Shuffling the nodes in the networks disrupts the connection between significantly altered genes and hubs. Previous hub genes might no longer be hubs or may be shifted to a neighborhood in which their interactors do not show high (anti-)correlation with it. Under these circumstances the method cannot define features. A confirmation of this effect provides the analysis of the features. Supplement section 5 clearly shows that the algorithm finds fewer features with fewer member genes on the randomized PPI networks. The effect becomes stronger when a small network is randomized (cf. Taylor on I2D and HPRD9) or when the data set size is small (cf. Taylor on the DMFS data set and the RFS data set). Thus, searching for altered hubs might offer a good biological interpretation of the data in context of outcome prediction. However, it is important to note that the algorithm is sensitive to the network size and data set size. When features can be defined by the method, they perform

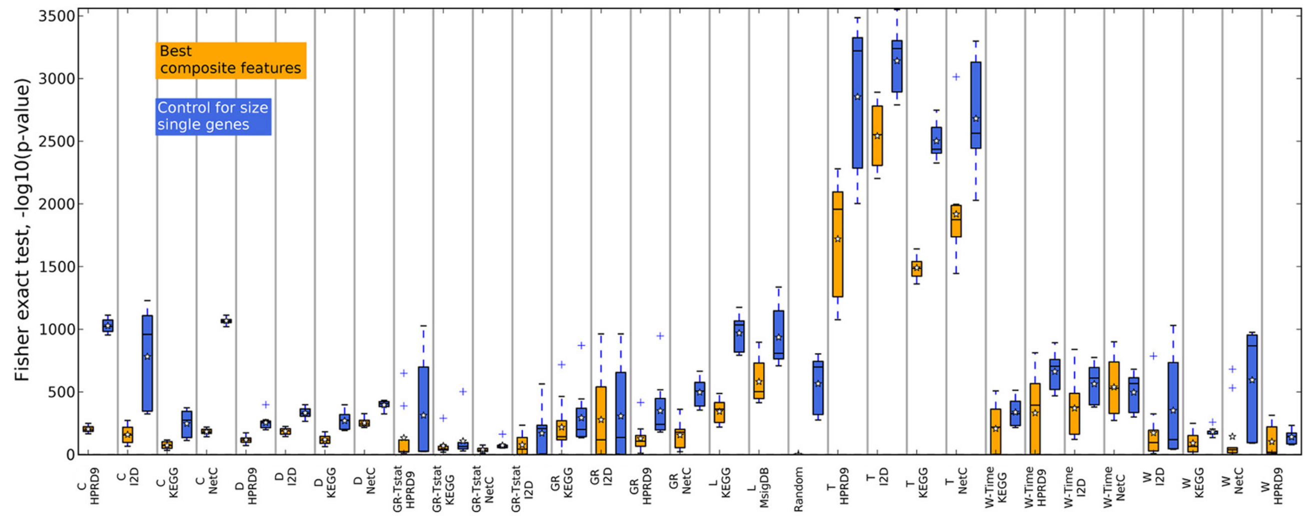


FIGURE 4 | Overlap of the gene sets corresponding to the best number of features on the DMFS data set. Genes belonging to the best number of composite features were extracted for each training set (outer CV). The overlap between the gene sets was calculated with Fisher's exact test. The corresponding number of top ranking single genes for each training data set were drawn and the overlap was calculated between these gene sets to control the influence of the gene set sizes.

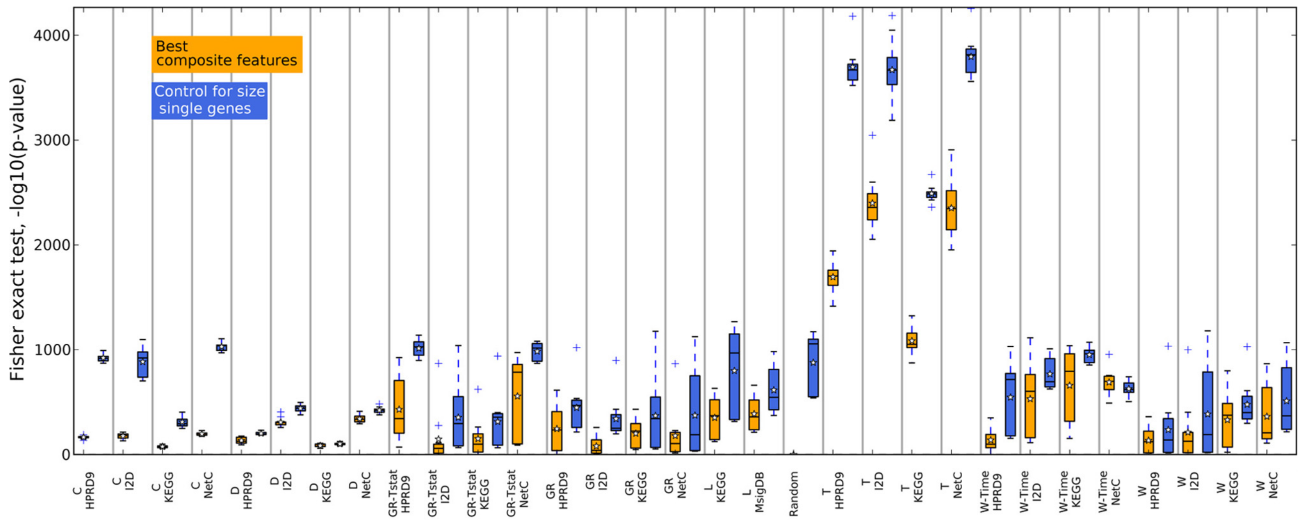


FIGURE 5 | Overlap of the gene sets corresponding to the best number of features on the RFS data set.

as well in classification as features determined on the real PPI networks. Thus, the major factor contributing to a good classification performance is the expression data.

Chuang et al. (2007) determines subnetworks whose mutual information between the member genes' expression and the class labels is high. This link is certainly disrupted by randomizing the PPI network. The algorithm includes statistical tests to only return significantly altered subnetworks, which should prevent returning randomized features. Thus, for some combinations of randomized networks and expression data no subnetworks can be found whose mutual information score is significantly high. However, if features are found we observe that the classification performance is as good as with features extracted from the real networks. Moreover, Supplement section 5 shows that the number of features increases on the randomized PPI networks. One reason for that could be that many genes are involved in breast cancer and many of them also show a significant differential expression (Ein-Dor et al., 2005). Thus, by shuffling the nodes there is still a high chance that subsets of these genes again form a subnetwork that is then identified by the algorithm as a feature. Since genes are no longer grouped according to their pathway, the information is scattered over the network. Thus, features extracted from randomized networks with the method by Chuang et al. may contain a lot of redundant information. As above this leads to the conclusion that the main factor in classification is the expression data.

In contrast to the above mentioned algorithms, the features by Dao et al. (2010) perform significantly worse in classification when they were determined on random PPI networks. We also observe that no features are found for some combinations of input data and in general fewer features are found (Supplement section 5). Since it is required in the method that a certain percentage of "poor" outcome patients show deregulation for each of the features, the number of member genes in the features can not decrease. The method searches for maximally densely connected subnetworks that cover at least 5% of the poor outcome patients. As noted before, looking for features that only describe one condition and do not consider information about all training samples might lead to a poor performance. The effect is worsened when giving the algorithm non-sensical biological information, as we do with the randomized networks. However, comparing the results obtained on the I2D network and the HPRD9 network and on the two different expression data sets, it seems that this effect is also linked to network and data set size. Since the methods by Taylor et al. (2009) and Dao et al. (2010) are more sensitive to the underlying quality of the data we can conclude that they are less prone to extract noise from the underlying data.

Also the GeneRank algorithm (Morrison et al., 2005) and the method by Winter et al. (2012) do not suffer from randomizing the networks. Both methods determine the rank for each gene by an initial rank and the diffused ranks of the genes in the vicinity. Having many differentially expressed genes in a network may contribute to selecting genes that can well distinguish between the patient classes. This is also confirmed by the fact that the damping factor, and thus the network, has only a minor influence on the classification when employing real PPI networks (see Section 3 in the supplement).

4.4. COMPOSITE FEATURES EXTRACTED FROM RANDOMIZED NETWORKS ARE LESS STABLE

In previous studies the overlap between features, i.e., in case of composite features the genes contained in the features, has been used as an indicator for biological meaningful features. When genes are chosen as features or as a part of composite features on different data sets, they might contain valuable biological information. We now analyze the overlap between features generated on the randomized PPI networks. For each training data set in the outer CV we determined the best performing features on one randomized network. We then calculated the overlap between

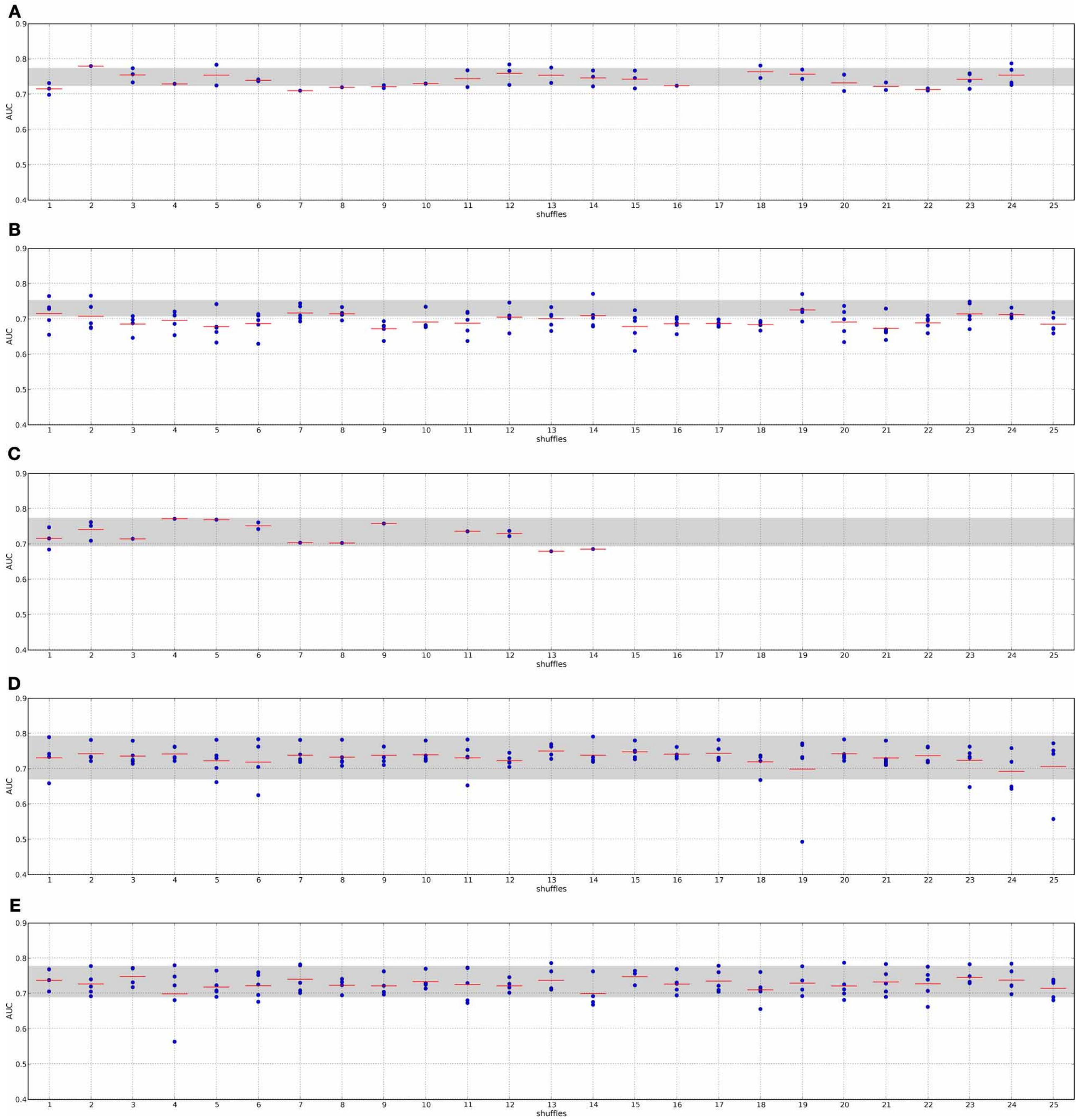


FIGURE 6 | Classification performance when employing randomized networks, I2D and DMFS data. The nodes in the I2D PPI network were shuffled 25 times yielding 25 different randomized networks. Each network dependent feature selection method was applied to each of the randomized networks and classifiers were trained using the double-loop CV protocol. The gray area indicates the AUC value interval employing the original PPI network. Panel **(A)**, Chuang; panel **(B)**, Dao; panel **(C)**, Taylor; panel **(D)**, GeneRank with absolute expression difference between "good" and "poor" outcome groups as initial rank; panel **(E)**, Winter using association between survival times and gene expression as initial gene ranks.

the genes contained in the features for the five training data sets as above. Thus, we only compared features that were generated using the same algorithm and the same randomization of the network. The boxes in **Figures 8**, **9** summarize all values across the 25 randomizations. Overlap for gene sets determined on random networks is always significantly worse than the overlap of features determined on the real networks when employing the method by Dao et al. (2010). Apparently, looking for maximally densely connected subnetworks is an adequate mathematical translation to define marker genes for breast cancer outcome. Taylor always

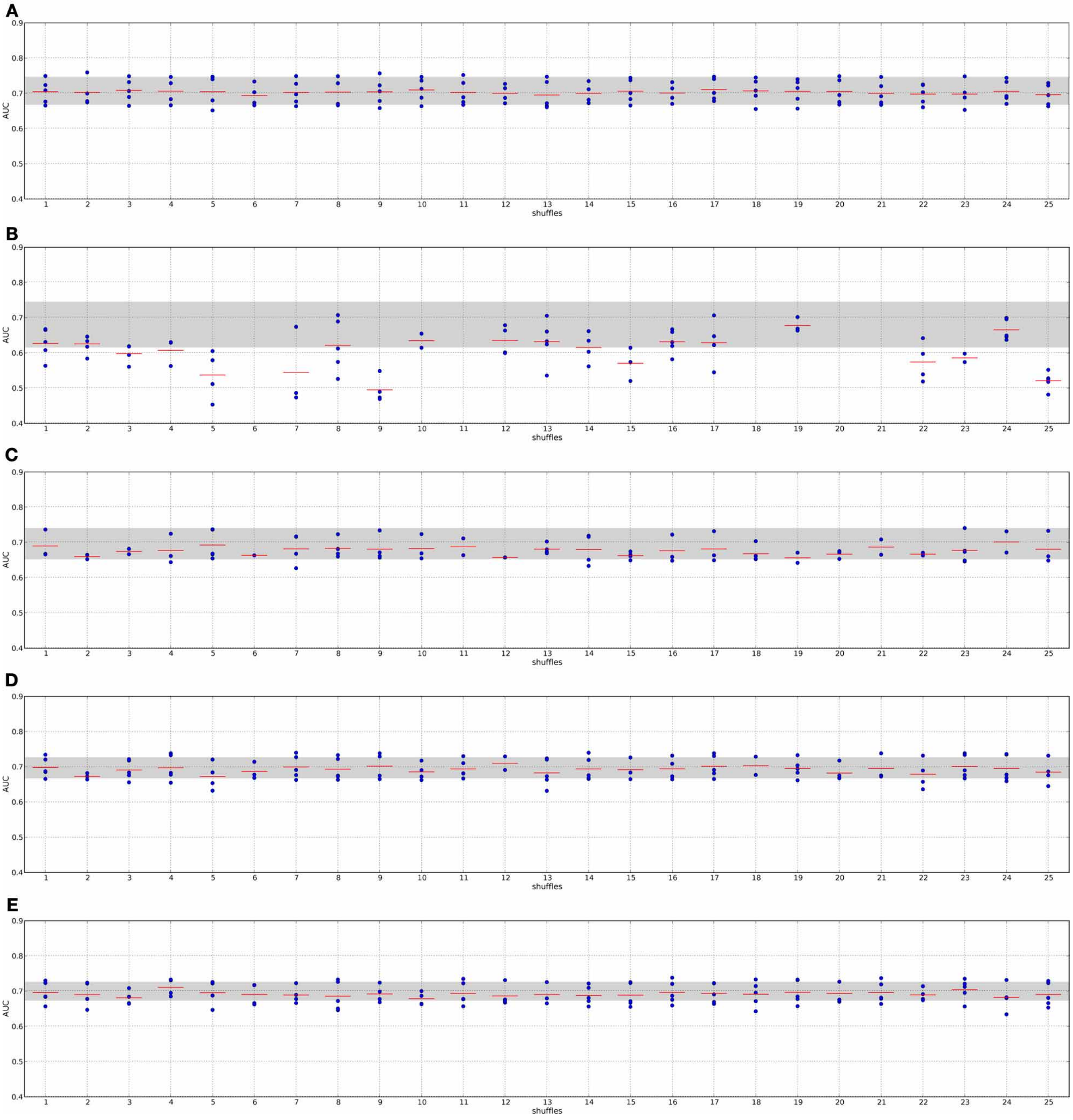


FIGURE 7 | Classification performance when employing randomized networks, I2D and RFS data. Description as in **Figure 6**. Panel **(A)**, Chuang; panel **(B)**, Dao; panel **(C)**, Taylor; panel **(D)**, GeneRank with absolute expression difference between "good" and "poor" outcome groups as initial rank; panel **(E)**, Winter using association between survival times and gene expression as initial gene ranks.

produces an equally stable overlap. The only exception on the NetC PPI network is due to the small number of features that could be determined on this network. This confirms that the high overlap is merely due to the algorithm. Selecting many genes leads to stable gene sets. The results for Chuang, Winter and the GeneRank algorithm are mixed. Here, the stability of features seems to depend on the combination of network and expression dataset. To conclude, we showed that randomizing the subnetworks leads to a loss of information that is important to extract gene sets that are stable across different data sets. However, the lost information is irrelevant for the classification as shown in the previous section.

4.5. SUMMARY

Previously many feature selection methods were put forward for better classification of breast cancer outcome. The novel

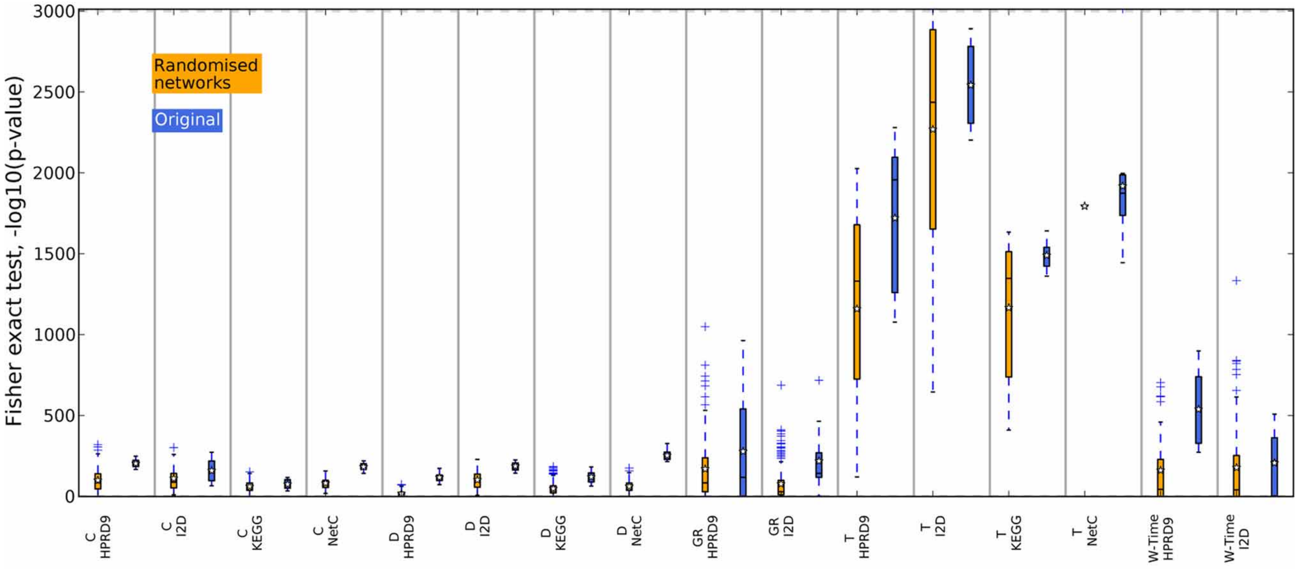


FIGURE 8 | Overlap of the gene sets determined on the randomized PPI networks, DMFS data set. The overlap between the gene sets was calculated with Fisher's exact test. The blue boxes show the overlap of the corresponding features determined on the original networks.

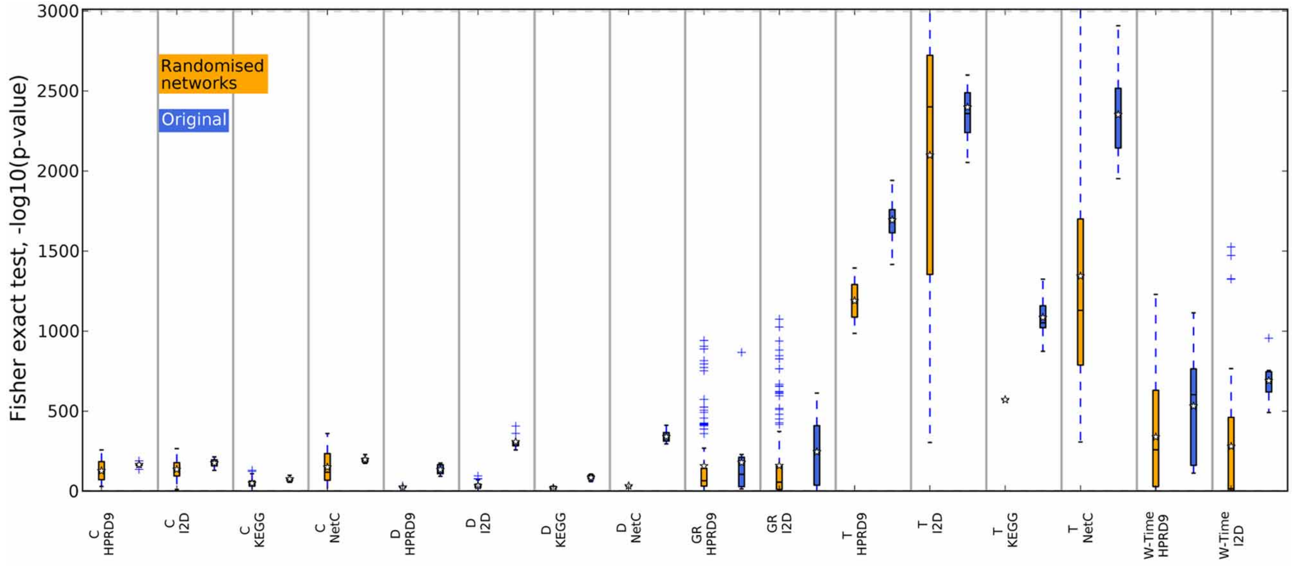


FIGURE 9 | Overlap of the gene sets determined on the randomized PPI networks, RFS data set. The overlap between the gene sets was calculated with Fisher's exact test. The blue boxes show the overlap of the corresponding features determined on the original networks.

methods claimed that integrating gene expression data and secondary data, such as PPI networks and pathway data, improves the classification performance and provides more stable features. We evaluated the methods based on two large breast cancer data sets and a variety of PPI networks and pathway databases. Our results do not confirm any of these claims.

To facilitate an easy and unbiased evaluation of more methods on more networks, pathways and expression data, we have proposed the Amsterdam Classification Evaluation Suite (ACES), a novel evaluation framework. In the implemented pipeline, we strictly separate between the training data and the testing data by employing a double-loop cross validation procedure. We provide tutorials which make it very easy to extend the described pipeline with additional data. Furthermore, we provide a tutorial and in depth instructions how to include new feature selection methods. ACES is freely available.

We conclude that it remains difficult to evaluate whether the composite-features selection methods draw any useful information from the secondary data sources, such as PPI networks and pathway data. We showed here and in our previous work (Staiger et al., 2012) that the methods by Chuang et al. (2007), Winter et al. (2012) and Lee et al. (2008) and the GeneRank algorithm (Morrison et al., 2005) do indeed perform as well on randomized PPI networks as on the real PPI networks. In contrast, the methods by Dao et al. (2010) and Taylor et al. (2009) are more dependent on the subnetwork structure when selecting features and fail to provide useful features on randomized network data. However, we also observe that in some cases these two methods perform worse on the original PPI networks than the single-genes classifiers, suggesting that some specific combinations of gene expression data and network data delivers less information for the classification task than the expression data alone. This suggests that the most predictive power for outcome is derived from the gene expression data and that the PPI network and pathway data only provides some means to reduce the feature space but adds little to the predictive accuracy of the classifiers. To this end it is extremely difficult to decide whether networks in general add little information to the classification task or whether the tested methods are not able to successfully leverage this information.

There are two independent goals when creating feature selection methods for outcome prediction in breast cancer: (i) to correctly classify the patients and (ii) to find genes or combinations of genes that carry some biological meaning. We have shown that currently the first goal can best be achieved by applying simple single-gene approaches and not by applying elaborate methods that use network or pathway data. However, for the definition of gene signatures specific for certain phenotypes, such methods seem to be more reliable to extract less noisy features—and thus possibly biological meaningful genes—than single-gene approaches.

AUTHOR CONTRIBUTIONS

Christine Staiger, Lodewyk F. A. Wessels, and Gunnar W. Klau conceived and designed the experiments. Christine Staiger and Balázs Györffy acquired and analyzed the data. Christine Staiger and Sidney Cadot performed the experiments. Christine Staiger, Lodewyk F. A. Wessels, and Gunnar W. Klau analyzed the experimental results. Christine Staiger, Lodewyk F. A. Wessels, and Gunnar W. Klau wrote the manuscript. All authors read and approved the final version of the manuscript.

ACKNOWLEDGMENTS

We would like to thank Jan Bot for providing help to perform the vast amount of calculations. This work was carried out on the Dutch national e-infrastructure with the support of the SURF Foundation.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://www.frontiersin.org/journal/10.3389/fgene.2013.00289/abstract

REFERENCES

Abraham, G., Kowalczyk, A., Loi, S., Haviv, I., and Zobel, J. (2010). Prediction of breast cancer prognosis using gene set statistics provides signature stability and biological context. *BMC Bioinformatics* 11:277. doi: 10.1186/1471-2105-11-277

Chuang, H.-Y., Lee, E., Liu, Y.-T., Lee, D., and Ideker, T. (2007). Network-based classification of breast cancer metastasis. *Mol. Syst. Biol.* 3, 140. doi: 10.1038/msb4100180

Cun, Y., and Fröhlich, H. F. (2012). Prognostic gene signatures for patient stratification in breast cancer - accuracy, stability and interpretability of gene selection approaches using prior knowledge on protein-protein interactions. *BMC Bioinformatics* 13:69. doi: 10.1186/1471-2105-13-69

Dao, P., Colak, R., Salari, R., Moser, F., Davicioni, E., Schönhuth, A., et al. (2010). Inferring cancer subnetwork markers using density-constrained biclustering. *Bioinformatics* 26, i625–i631. doi: 10.1093/bioinformatics/btq393

Ein-Dor, L., Kela, I., Getz, G., Givol, D., and Domany, E. (2005). Outcome signature genes in breast cancer: is there a unique set? *Bioinformatics* 21, 171–178. doi: 10.1093/bioinformatics/bth469

Ein-Dor, L., Zuk, O., and Domany, E. (2006). Thousands of samples are needed to generate a robust gene list for predicting outcome in cancer. *Proc. Natl. Acad. Sci. U.S.A.* 103, 5923–5928. doi: 10.1073/pnas.0601231103

Gohlmann, H., and Talloen, W. (2010). *Gene expression studies using Affymetrix microarrays*. Boca Raton, FL: CRC Press.

Györffy, B., and Schäfer, R. (2009). Meta-analysis of gene expression profiles related to relapse-free survival in 1,079 breast cancer patients. *Breast Cancer Res. Treat.* 118, 433–441. doi: 10.1007/s10549-008-0242-8

Kanehisa, M., Goto, S., Furumichi, M., Tanabe, M., and Hirakawa, M. (2010). Kegg for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Res.* 38(Database issue), D355–D360. doi: 10.1093/nar/gkp896

Lee, E., Chuang, H.-Y., Kim, J.-W., Ideker, T., and Lee, D. (2008). Inferring pathway activity toward precise disease classification. *PLoS Comput. Biol.* 4:e1000217. doi: 10.1371/journal.pcbi.1000217

Ma, S., Shi, M., Li, Y., Yi, D., and Shia, B.-C. (2010). Incorporating gene co-expression network in identification of cancer prognosis markers. *BMC Bioinformatics* 11:271. doi: 10.1186/1471-2105-11-271

Morrison, J. L., Breitling, R., Higham, D. J., and Gilbert, D. R. (2005). Generank: using search engine technology for the analysis of microarray experiments. *BMC Bioinformatics* 6:233. doi: 10.1186/1471-2105-6-233

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). *The PageRank Citation Ranking: Bringing Order to the Web.* Technical Report 1999-66, Stanford InfoLab.

Popovici, V., Chen, W., Gallas, B. G., Hatzis, C., Shi, W., Samuelson, F. W., et al. (2010). Effect of training-sample size and classification difficulty on the accuracy of genomic predictors. *Breast Cancer Res* 12, R5. doi: 10.1186/bcr2468

Prasad, T. S. K., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S., et al. (2009). Human protein reference database–2009 update. *Nucleic Acids Res.* 37(Database issue), D767–D772. doi: 10.1093/nar/gkn892

Staiger, C., Cadot, S., Kooter, R., Dittrich, M., Müller, T., Klau, G. W., et al. (2012). A critical evaluation of network and pathway-based classifiers for outcome prediction in breast cancer. *PLoS ONE* 7:e34796. doi: 10.1371/journal.pone.0034796

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., et al. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. U.S.A.* 102, 15545–15550. doi: 10.1073/pnas.0506580102

Taylor, I. W., Linding, R., Warde-Farley, D., Liu, Y., Pesquita, C., Faria, D., et al. (2009). Dynamic modularity in protein interaction networks predicts breast cancer outcome. *Nat. Biotechnol.* 27, 199–204. doi: 10.1038/nbt.1522

van 't Veer, L. J., Dai, H., Van De Vijver, M. J., He, Y. D., Hart, A. A., Mao, M., et al. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature* 415, 530–536. doi: 10.1038/415530a

Wang, Y., Klijn, J. G. M., Zhang, Y., Sieuwerts, A. M., Look, M. P., Yang, F., et al. (2005). Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet* 365, 671–679. doi: 10.1016/S0140-6736(05)17947-1

Wessels, L. F. A., Reinders, M. J. T., Hart, A. A. M., Veenman, C. J., Dai, H., He, Y. D., et al. (2005). A protocol for building and evaluating predictors of disease state based on microarray data. *Bioinformatics* 21, 3755–3762. doi: 10.1093/bioinformatics/bti429

Winter, C., Kristiansen, G., Kersting, S., Roy, J., Aust, D., Knösel, T., et al. (2012). Google goes cancer: improving outcome prediction for cancer patients by network-based ranking of marker genes. *PLoS Comput. Biol.* 8:e1002511. doi: 10.1371/journal.pcbi.1002511

Conflict of Interest Statement: 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.

Received: 15 September 2013; paper pending published: 13 October 2013; accepted: 28 November 2013; published online: 23 December 2013.

Citation: Staiger C, Cadot S, Györffy B, Wessels LFA and Klau GW (2013) Current composite-feature classification methods do not outperform simple single-genes classifiers in breast cancer prognosis. Front. Genet. ***4****:289. doi: 10.3389/fgene.2013.00289*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

ORIGINAL RESEARCH ARTICLE
published: 24 December 2013
doi: 10.3389/fgene.2013.00303

Experimental assessment of static and dynamic algorithms for gene regulation inference from time series expression data

Miguel Lopes[1,2]* **and Gianluca Bontempi**[1,2]

[1] *Machine Learning Group, Computer Science Department, Universite Libre de Bruxelles, Bruxelles, Belgium*
[2] *Interuniversity Institute of Bioinformatics in Brussels (IB)^2, Brussels, Belgium*

Edited by:
Benjamin Haibe-Kains, Institut de Recherches Cliniques de Montréal, Canada

Reviewed by:
Vanni Bucci, Memorial Sloan-Kettering Cancer Center, USA
Victor P. Andreev, University of Miami, USA

****Correspondence:***
Miguel Lopes, Machine Learning Group, Computer Science Department, Universite Libre de Bruxelles, Campus de la Plaine, Boulevard du Triomphe, B-1050 Bruxelles, Belgium
e-mail: mlopes@ulb.ac.be

Accurate inference of causal gene regulatory networks from gene expression data is an open bioinformatics challenge. Gene interactions are dynamical processes and consequently we can expect that the effect of any regulation action occurs after a certain temporal lag. However such lag is unknown a priori and temporal aspects require specific inference algorithms. In this paper we aim to assess the impact of taking into consideration temporal aspects on the final accuracy of the inference procedure. In particular we will compare the accuracy of static algorithms, where no dynamic aspect is considered, to that of fixed lag and adaptive lag algorithms in three inference tasks from microarray expression data. Experimental results show that network inference algorithms that take dynamics into account perform consistently better than static ones, once the considered lags are properly chosen. However, no individual algorithm stands out in all three inference tasks, and the challenging nature of network inference tasks is evidenced, as a large number of the assessed algorithms does not perform better than random.

Keywords: gene network inference, causality inference, temporal models, static models, experimental assessment

1. INTRODUCTION

The measurement of gene expression levels, by using microarrays or high throughput technologies, makes it possible to infer statistical dependencies (e.g., correlations) between the expression of two genes. Some of these dependencies can be seen as a result of causal interactions, as the expression of a gene can influence the future expression of another gene (these causal interactions are known as gene regulatory interactions). Several methods have been proposed to infer gene regulatory interactions from measured gene expression levels. Some of them are static, in the sense that they do not take temporal aspects into consideration, while others are designed in order to learn the dynamical aspects of the dependencies. Since gene interactions are not instantaneous, we expect that temporal aspects should shed light on the causal dependencies between genes. In other terms if two genes are part of a regulatory interaction, their expression levels over time are expected to be correlated with a certain lag and the time order is expected to elucidate the respective promoter/target roles. However, unfortunately such lag is unknown a priori and should be properly learned from data. If on one hand dynamic approaches may appear as more powerful than static ones because of the temporal representation, on the other hand they are more sensitive to the accuracy of the adopted lag. In machine learning jargon, this is known as a bias/variance trade-off. The adoption of temporal dynamic models makes the learner less biased but necessarily more exposed to high variance. In spite of this intuition, and although there are some comparisons between dynamic and static methods in the literature on gene regulatory networks, these are not systematic or extensive.

For this reason, we propose in this paper an experimental setting to assess the role of dynamics on the accuracy of the inferred regulatory network. To this aim, we compare a number of state-of-the-art static and dynamic approaches on three challenging inference tasks. As state-of-the-art static approaches, we consider Bayesian networks (Balov and Salzman, 2010; Kalisch et al., 2012) and directed graphical Gaussian models (GGM) (Schäfer and Strimmer, 2005). These two methods are based on the estimation of conditional dependencies between genes. The first infers a directed network using the rules of d-separation, the latter is an undirected graphical model (an edge indicates the presence of a conditional linear correlation between the respective nodes), but that can be made directed by making ad hoc assumptions. As dynamic approaches we consider: Vector AutoRegressive models (VAR) (Charbonnier et al., 2010), Dynamic Bayesian networks (DBN) (Lebre, 2009) and adaptive lag methods (Zoppoli et al., 2010; Lopes et al., 2012). VAR models are linear models where the target variable at a time point is modeled as a linear combination of predictor variables at previous time points (typically one). DBN are graphical models where variables at different time points are represented by different nodes and edges are allowed only from variables at time t to variables at time superior than t. Adaptive lag models are dynamic approaches which include an automatic estimation of a temporal lag for each pair of genes, e.g., by maximizing some dependence score. In order to make a fair comparison, all the assessed approaches (static and dynamic) are causal, in the sense that they infer directed interactions.

Our experimental study makes an assessment of static and dynamic algorithms by comparing the accuracy of the networks

Network Inference

- Q: Which kinds of biological networks have been inferred in the paper?
- A: 500 gene regulatory networks of 5 nodes were inferred for three species (*E.coli*, yeast, fruit fly). Networks were inferred from time series gene expression datasets.
- Q: How was the quality/utility of the inferred networks assessed. How were these networks validated?
- A: The gold standard was defined as being interactions reported in the literature. A precision recall curve, and the respective area under (AUPRC) was assigned to each inferred network. The AUPRC values of the 500 networks predicted by an inference method were averaged, and this value was used to score that method.
- Q: What are the main results described in the paper?
- A: The general performance of state of the art network inference methods on the proposed task is weak (in two species, most of the methods do not have a performance significantly better than random). However, methods that take into account temporal information tend to perform better than static, non-temporal methods. The performance of temporal methods is expected to depend on the temporal sampling interval and on the sample size of the used time series. This fact is confirmed in our experiments and we infer general conclusions on the proper use of temporal network inference methods.

inferred from three microarray time series. These datasets have different characteristics, in terms of biological species, time length and sampling period (5, 10, and 30 min). The first outcome of the study is that dynamic models perform consistently better than static ones. The second outcome is an interesting insight on the most probable interaction lag between gene expressions. Our results suggest that this lag can take values in the range of a few hours, and that temporal network inference models should be adjusted to incorporate this information. In the next chapter we will present the assessed network inference algorithms, the third chapter describes the experimental setting and is followed by the results and discussion.

2. MATERIALS AND METHODS

Two family of network inference algorithms, static and dynamic, are considered in this study and will be discussed in the following section. **Table 1** summarizes the differences between the used models.

2.1. STATIC MODELS

Static network inference models do not take into account any information related to the temporal nature of the gene expression data. Two well-known examples are Bayesian networks and GGM.

A Bayesian network is a graphical representation by directed acyclic graph of a multivariate probability distribution, where nodes denote variables and edges variable dependencies. Under the faithfulness assumption for the probability distribution, there exists a bijective mapping between the conditional independencies of variables in the distribution and topological properties (d-separation) in the graph. The main advantages of a Bayesian Network representation are its sparsity (i.e., use of few parameters), the ease of interpretation and the availability of several inference algorithms. For further references on the estimation of Bayesian networks from biological data see Needham et al. (2007) or Margaritis (2003).

A GGM is an undirected graph, where the presence of an edge indicates a non zero partial correlation between two nodes given all the others (Dempster, 1972; Lauritzen, 1996). Partial correlations can be obtained by inverting the covariance matrix, but this is problematic if the covariance matrix does not have full rank. One solution is a positive definitive estimation of the covariance matrix (Opgen-Rhein and Strimmer, 2007). Another approach estimates partial correlations using the eigenvectors of the covariance matrix associated with non-zero eigenvalues (Lezon et al., 2006). It has been shown that partial correlations emerge, under the assumption that the variables are Gaussian-distributed, when maximizing the entropy of the system conditioned on the empirical mean and covariance of the variables (Lezon et al., 2006). Below we describe three implementations of static models, available in R packages: two estimations of Bayesian networks and one estimation of a GGM with an extension to direct some of its edges.

The R package *catnet* (Balov and Salzman, 2010) infers categorical Bayesian networks from categorical data (the variables have to be discrete, taking only a finite number of values). The maximum likelihood criterion is used to assess different possible networks. This package implements a stochastic search in the network space, using a simulated annealing algorithm. In the experiments here presented, we defined the number of categories to be three (corresponding to different levels of gene expression). The output of this algorithm is a number of networks (represented by adjacency matrices) of increasing complexity each annotated with a likelihood. In order to obtain a final score matrix we made a weighted sum (based on likelihood) of all adjacency matrices.

The package *pcalg* (Kalisch et al., 2012) infers Bayesian networks from continuous data, and is based on the PC algorithm (Spirtes et al., 1993). The PC algorithm starts by considering a fully connected graph and progressively removes edges, depending on the conditional dependencies between the respective genes. The size of the conditioning sets is one at the beginning and then gradually increased. The existence and the direction of the edges is inferred using the rules of d-separation. In our experiments, the conditional dependence is measured by partial correlation, which is equivalent to assume that the variables are Gaussian distributed and their dependencies linear. The Fisher transformation is used to compute the significance level of the partial correlation value. By defining a set of decreasing threshold values for the significance level, we obtained a number of inferred

Table 1 | Assessed network inference models.

Method	Type	Lags	Category	Features
catnet	Static	–	Bayesian network	– Categorization of data – Stochastic search (simulated annealing) in the network space
pcalg	Static	–	Bayesian network	– Progressive removal of edges (backwards selection) – Conditional dependence estimated with partial correlation
GeneNet	Static	–	Graphical Gaussian Model	– Full partial correlations estimated through shrinkage – Edges are directed from the most to the less exogenous variable
VAR l +lars	Dynamic	Fixed (first)	VAR	–VAR(l) model subject to a Ll penalty term – Regression coefficients estimated with least angle regression (lars)
simone	Dynamic	Fixed (first)	VAR	–VAR(l) model subject to a variable penalty term (to favor the selection of transcription factors) – Regression coefficients estimated through optimization
Gl DBN	Dynamic	Fixed(first)	Dynamic Bayesian network	– Estimation of a number of first order partial regression coefficients,for each possible interaction – Predictors and target are lagged by l time point
Time Delay ARACNE	Dynamic	Estimated(one)	Information–theoretic	– Mutual information used to infer dependencies (MI estimated with a copula–based approach) – Estimation of the lag between two genes – Use of the DPI to break up fully connected triplets
Time lagged MRNET	Dynamic	Estimated(one)	Information–theoretic	– Mutual information used to infer dependencies (Gaussian assumption) – Estimation of the lag between two genes – mRMR feature selection
Time lagged CLR	Dynamic	Estimated(one)	Information–theoretic	– Mutual information used to infer dependencies (Gaussian assumption) – Estimation of the lag between two genes – Normalization of MI

networks with an increasing number of edges. Then we associated to each possible interaction a score equal to the average number of times that this interaction is inferred in the returned networks.

GeneNet (Opgen-Rhein and Strimmer, 2007) estimates partially directed GGM. Once the positive definitive covariance matrix is estimated (using a shrinkage technique Schaefer et al., 2006), it computes the concentration matrix (the inverse of the covariance matrix) and a partial correlations matrix. An undirected GGM is created by selecting the edges associated to the highest partial correlations. GeneNet infers the directionality of the interactions by comparing, for each pair of connected nodes, the partial variances of the respective variables. The partial variance of a variable is its variation that cannot be modeled, or predicted, in a linear regression model using the other variables in the set as predictors. The ratio between the partial variance and the variance gives the percentage of the variation that corresponds to unexplained variation. These relative values of unexplained variation are used as indicators of how much of the variable variation can be explained from within the system (using all the other variables). An edge between two nodes is directed from the one with higher unexplained variation to the one with lower. Each edge is given a p-value (the null hypothesis is that the partial correlation between its nodes, or genes, is zero). For each edge we assigned a score equal to 1 minus the respective p-value.

2.2. DYNAMIC MODELS

We will distinguish dynamic models according to the approach used to define the lag between variables. In what follows p is the number of genes and X^t is used to denote the value of the variable X at time t.

2.2.1. Fixed lag models

Vector autoregressive models of order $l_{\max}$ (VAR($l_{\max}$)) models each gene X^t, at time t, as a linear function of all the genes at time $t - l$, where $l = 1, .., l_{\max}$.

$$X_i^t = c + \sum_{l=1}^{l_{\max}} \sum_{j=1}^{p} \beta_{l,j} X_j^{t-l} + \epsilon_i \qquad (1)$$

Therefore VAR(1) denotes a lag-one model where the value of $l_{\max}$ is set to 1. The coefficients β in (1) can be estimated by Ordinary Least Squares algorithm (OLS), provided that there are enough samples. Alternatively, β can be returned by a regularization algorithm, such as the *Lasso* (Tibshirani, 1994), which adds a penalty term in the OLS solution equation, that is proportional to the L_1 norm of β. In other words, the Lasso minimizes the sum of squares of the residuals, given that the sum of the absolute value of the coefficients β is less than a constant. This approach imposes scarcity in the number of returned non-zero coefficients and can be used to detect the most relevant coefficients.

Another fixed lag model is the Dynamic Bayesian Network (DBN). DBN are modifications of Bayesian networks to model time series: each gene is represented by different nodes, at different time points (Perrin et al., 2003). An edge is allowed to go from a node X^{t-l} to a node Y^t. In our study we assessed three lag-one models, two of them penalty-constrained implementations of VAR(1) models, and one of them an implementation of a DBN. They are described below.

Our implementation *VAR(1)+lars* models the data from a VAR(1) perspective: a variable X_i^t is regressed using all the variables lagged by one time point: $X_j^{t-1}, j = 1 \ldots p$. As with the Lasso, a penalty term proportional to the L_1 norm of the regressor coefficients is added to the model. The coefficients of the model are estimated using the *lars* algorithm [(Efron et al., 2004), available in the R package *lars*]. The lars algorithm computes in a fast manner the coefficients of the lasso path, when the regularization penalty term goes from infinity (where there is no non-zero returned coefficients) to 0 (corresponding to the OLS solution). Using lars, for each gene we computed the coefficients of its predictors at the points (in the lasso path) where the coefficient of a predictor becomes non-zero and enters the model. We then computed the average of the coefficients of each predictor variable and used it as the directed dependence score between the predictor and the target gene.

The R package *simone* (Charbonnier et al., 2010) estimates the coefficients of a VAR1 model subject to a L_1 norm penalty term. Here, a weighted lasso is used, a modification of the Lasso to allow different penalty terms for different regressors. Genes are grouped into two main groups: *hubs*, which are genes that show a high level of connectivity probability to all the other genes, and *leaves*, which are only connected to hubs. It is suggested that hubs will correspond to transcription factors (genes whose expression levels influence the transcription of other genes). Every gene is assigned to the group of hubs or to group of leaves, from an initial estimation (or optionally, from expert knowledge if available). This initial estimation is done by computing a matrix of coefficients using the standard Lasso, and then group genes into hubs or leaves according to the L_1 norm of the respective rows in the estimated coefficients matrix. The regressors are assigned one of two different weights, one for hubs and the other for leaves, which multiply the respective coefficients before they are used in the calculation of the penalty term. The idea behind this implementation is that interactions coming from hubs (transcription factors) should be less penalized than interactions coming from leaves. Simone returns a list of inferred networks for different values of the penalty weights. In the experiments here reported, we defined the score for an interaction as the number of times the interaction is associated with a non-zero coefficient in all the returned networks.

G1DBN is a R package (Lebre, 2009) that estimates dynamic Bayesian networks, using first order conditional dependencies. G1DBN is designed to work with time series and implements a lag-one model. Each gene is represented by two nodes lagged by one time point. Interactions are only allowed from nodes at time $t-1$ to nodes at time t. It is a two-step method: the first step computes all possible regression coefficients, of each gene X_j^{t-1} to each gene X_i^t, conditioned on each other gene X_k^t, $k \neq j, i$. This way, each directed interaction is assigned a number of coefficients, one for each conditioning variable. Each of these coefficients is subject to a statistical test based on the student's t distribution (the null hypothesis is that the value is zero) and a *p*-value is returned. The maximum of these *p*-values is considered as a score for the respective interaction. A threshold α_1 is defined, and edges with scores lower than it are selected. The second step of the algorithm starts with this graph and removes more edges: for each gene, it is calculated the regression coefficient of it toward one of its parents, given all the other parents. To each of these coefficients is assigned a *p*-value, in an analogous way as in the first step. A new threshold α_2 is defined, and only edges with *p*-values lower than α_2 are kept. In our experiments, we defined $\alpha_1 = 0.7$, as it was the value used in the method's original proposal. We used several values for α_2, and for each of them an adjacency matrix was returned, with the estimated *p*-values for each possible interaction. For each interaction, the subtraction 1 minus the average of the respective final *p*-values, was used as the final score.

2.2.2. Adaptive lag models

Adaptive lag models are models where each possible interaction is assigned a score which is a function of an estimated temporal lag, that hypothetically characterizes the interaction. This lag is estimated as the one which maximizes some score S. The lag between two genes X and Y is estimated as:

$$\mathrm{lag}^{XY} = \arg\max_{l} \left(S\left(X^t, Y^{t-l} \right) \right) l = -l_{\max}, .., -1, 0, 1, .., l_{\max} \tag{2}$$

The parameter $l_{\max}$ is the maximum allowed lag. The adaptive lag methods implemented are based on the measure of mutual information (which is represented as $I(X; Y)$, between two variables X and Y).

The *Time-Delay ARACNE* (Zoppoli et al., 2010) is an extension of the information theoretic algorithm ARACNE (Margolin et al., 2005). It is based on three steps: the first step estimates the times at which each gene starts to be differentially expressed (and the set of possible interactions is restricted to the directed interactions where the target gene has a start-of-regulation time higher than the start-of-regulation time of the source gene). The second step of the algorithm lags the temporal expression of each pair of genes, and finds the lag which maximizes the mutual information between the genes. The mutual information is estimated through a copula based approach. A copula transformation (a rank based empirical copula) is applied to the distribution, and a kernel density estimator is used to estimate the bivariate marginal distribution $\hat{p}(X_i^t, X_j^{t-l})$, for each gene X_i^t and each gene X_j^{t-l}. The directed edges whose lagged mutual information is higher than a defined threshold are kept in the graph. The third and final step of the algorithm applies the data processing inequality (DPI) property to break up fully connected triplets. A binary adjacency matrix, indicating the predicted interactions, is returned. We defined various values for the threshold and obtained different adjacency matrices. Each interaction is assigned a score equal to the number of times the interaction has been predicted in the returned adjacency matrices. The parameter $l_{\max}$ was set to 6 time points.

The *Time-lagged MRNET* is the dynamic extension of the MRNET algorithm (Meyer et al., 2007) which is based on the *minimum-Redundancy Maximum-Relevance* (mRMR) feature selection method (Ding and Peng, 2005). For each gene Y, it selects all other genes in a sequential manner. The first selected gene (added to the group of selected genes S) is the one that has the highest mutual information toward the target gene. The next gene to be selected, X_j^{mRMR}, is defined as the one which maximizes the following mRMR score, $u - r$:

$$X_j^{\mathrm{mRMR}} = \arg\max_{X_j \notin S} \left(u_j - r_j\right) \tag{3}$$

where u_j and r_j are defined as follows:

$$u_j = I(X_j; Y) \tag{4}$$

$$r_j = \frac{1}{\mid S \mid} \sum_{X_k \in S} I\left(X_j, X_k\right) \tag{5}$$

The term u_j represents the relevance of X_j toward Y and the term r_j represents the redundancy of X_j with the previously selected genes in S. This process is repeated for all genes. To any pair of genes the MRNET algorithm assigns a score which is equal to the maximum between two mRMR scores: the mRMR score of the first when the second is the target, and the mRMR score of the second when the first is the target. The time-lagged MRNET is a modification of the MRNET algorithm (Lopes et al., 2012). Here, the mutual information considered by the algorithm (between each pair of genes) is a lagged mutual information. The lag is the one which maximizes the mutual information, as in the Equation (2). The estimation of lags allows to direct interactions, as the sign of the lags provide information on the direction of interactions. Therefore, the time-lagged MRNET returns directed interactions, as opposed to the standard undirected MRNET.

The *Time-lagged CLR* is the dynamic version of the *context likelihood of relatedness* (CLR) inference algorithm (Faith et al., 2007). CLR takes into account the fact that some genes exhibit, on average, a relatively high, or low, mutual information toward all the other genes. Each possible interaction between X and Y is assigned a score equal to $w_{xy} = \sqrt{z_x^2 + z_y^2}$, where:

$$z_x = \max\left(0, \frac{I(X, Y) - \mu_x}{\sigma_x}\right) \tag{6}$$

μ_x and σ_x are the empirical mean and the standard deviation of the mutual information between X and all the other genes. This way, the CLR score for an interaction between genes X and Y is higher for situations when both X and Y, or any of them, exhibit a low mutual information toward the majority of remaining genes in the dataset, compared with the otherwise situation. The time-lagged CLR is a modification of CLR (Lopes et al., 2012), just as the time-lagged MRNET is relative to MRNET.

On the implementations here described, the mutual information used by the time-lagged MRNET and CLR was estimated with the Pearson correlation. The value for the maximum allowed lag parameter, $l_{\max}$, was set to be 6, 12, and 18 time points. In the following results, the time-lagged MRNET and CLR of a certain $l_{\max}$ are referred as TL $l_{\max}$ MRNET and TL $l_{\max}$ CLR, respectively (e.g., TL12 MRNET).

We note that the assessment here presented does not constitute an extensive review of all the causal network inference models found in the literature. These include dynamic models based on ordinary differential Equations, such as the *Inferelator* (Bonneau et al., 2006) or the *TSNI* (Bansal et al., 2006), and other implementations of Bayesian and Dynamic Bayesian networks, such as *Banjo* (Smith et al., 2006).

2.3. THE DATASETS

Three time series datasets, from different species were collected. All these datasets are available in the Gene Expression Omnibus (GEO) database repository.

- A time series dataset of the gene expression of *Drosophila melanogaster*, of length 22 h (Hooper et al., 2007). The number of observations is 28 and the time between observations is 1 h after the 10 first observations, and approximately 30 min in the first 10 observations. We will refer to this dataset as *dataset Fly*.
- A time series dataset of the gene expression of *Escherichia coli*, of length 5 h and 20 min (Traxler et al., 2006). The number of observations is 17 and the time between observations changes between 10 and 50 min. We will refer to this dataset as *dataset E.coli*.
- A time series dataset of the gene expression of *Saccharomyces cerevisiae*, of length 2 h (Pramila et al., 2006). The number of observations is 25 and the time between observations is 5 min. We will refer to this dataset as *dataset Yeast*. This dataset is composed of two time series, and we averaged the samples of equal time points.

In the datasets Fly and *E.coli* we interpolated linearly the data, to obtain time series with a constant step: 30 min in the first and 10 min in the second. After this operation, the dataset *E.coli* has 32 time points and the dataset Fly has 45 time points.

2.4. PERFORMANCE ASSESSMENT

Adjacency matrices with documented interactions for the three different species were obtained in Gallo et al. (2010), Gama-Castro et al. (2011) and Abdulrehman et al. (2011) (for the Fly, *E.coli* and Yeast datasets). Only strong evidence interactions were selected. From these adjacency matrices, we generated small regulatory networks, containing only genes whose expression levels are measured in the respective dataset. For each dataset, 500 sub-networks of 5 nodes were randomly generated. Using the algorithms in the way that was described in the previous section, we obtained for each algorithm and network, a square matrix of scores for all possible directed interactions (the element (i, j) represents the score of the interaction from gene i to gene j). For any pair of genes, only one interaction was kept, corresponding to the strongest direction. To assess the performance of an algorithm on a given network we used the AUPRC (area under the precision recall curve). Interactions were incrementally selected (from the highest to the lowest ranked), and at each selection, precision and recall values were computed. We assigned to each recall its highest associated precision (there can be multiple precision values for a given recall). The AUPRC was estimated as the average precision, for all values of recall. For each algorithm and dataset, we averaged the AUPRC obtained for the 500 networks. The random baseline was estimated as being the expected average AUPRC of a random ranking, on all networks. **Figure 1** shows some examples of precision recall curves, in blue with an higher AUPRC than the expected random baseline, and in red with lower (the number of instances is 20, and the number of positives is 5).

3. RESULTS

The average AUPRC values for each algorithm and dataset can be seen in the **Figure 2**. The **Figure 3** represents the existence (black), or not (white) of a significant difference between the performance of any two algorithms. All pairs of algorithms were subject to a paired t-test (two-sided, different variances) to test for a significant difference in their performance. The algorithms' AUPRC values were given as the input to the test and a difference was considered significant is the returned p-value was lower than 0.05. Of particular interest are the differences relative to the random ranking of interactions. Relative to the dataset Fly, dynamic models clearly outperform static models, which do not perform better than random. In the dataset *E.coli*, the best performers are the time lagged-MRNET and the time lagged-CLR when $l_{\max}$ is set to 18 time points (corresponding to 3 h). Fixed lag models and static models perform similarly, with only one method performing better than random (VAR1+lars). Relative to the dataset Yeast, the best performers are G1DBN and Time-Delay ARACNE, and are the only ones with a performance significantly better than random. As a control procedure, the ordering of the time points in the datasets was randomized, and the dynamic network inference methods were rerun (static models do not depend on the ordering of the samples). As expected, on all occasions the performance drops to the random level.

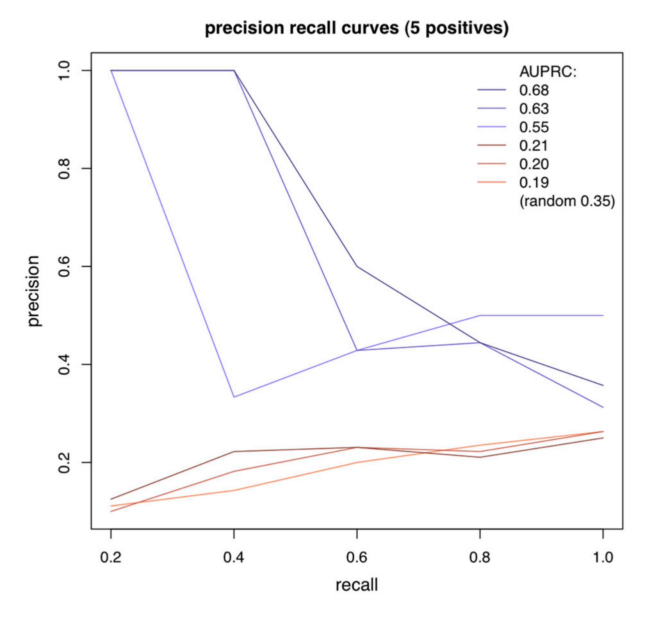


FIGURE 1 | Precision-recall curves.

4. DISCUSSION

Some points can be drawn from the results presented:

- The performance of some methods can be poor. On the dataset *E.coli* only three methods are better than random, and on the dataset Yeast there are only two. On the dataset Fly no static method performs better than random (the dynamic methods, on the contrary, perform well). This poor performance may be a result of the low number of samples of the datasets, or with the way the networks are generated and assessed, using gene regulatory interactions as a ground-truth that may not be adequate, or representative of the interactions that are regulating gene expression.
- The best performers on all datasets are dynamic models. This suggests that incorporating temporal information is beneficial to the inference of gene regulatory interactions. On all datasets, static models do not perform better than random. The fact that the assessed dynamic models are computationally simpler than the static algorithms (particularly the ones estimating Bayesian networks) is another reason to prefer dynamic models over static ones when inferring networks from time series.
- Most of the temporal models perform better on the dataset Fly than on the datasets Yeast and *E.coli* (see the comparison with random in **Figure 3**). This difference is possibly due to the temporal characteristics of the datasets (Fly is a 30-min interval dataset, of duration of 24 h; Yeast is a 5-min interval dataset, of duration 2 h). It seems natural that the gain in performance using dynamic models depends on the temporal characteristics of the dataset. On the dataset Fly, the dynamic performers also exhibit a significant difference between them. On the contrary, on the dataset Yeast, most of the models perform similarly (at the random level) and do not exhibit such difference.
- On the dataset Fly, the best performers are fixed lag methods. These methods directly estimate conditional dependencies, as opposed to the adaptive lag methods that only estimate pairwise dependencies. This aspect may play a role in the observed differences in performance.
- The performance of the adaptive lag models changes with the parameter $l_{\max}$. On the datasets Fly and Yeast there is a decrease in the performance of the time-lagged MRNET and CLR as $l_{\max}$ increases. On the dataset *E.coli*, on the contrary, there is a large performance boost when $l_{\max}$ is set to 18 time points.
- On the dataset Fly, a long time series where each time point corresponds to 30 min, setting $l_{\max}$ to too high values can be unrealistic (a lag of 18 time points corresponds to 9 h). If we estimate lagged dependencies over a long and unrealistic range of lags, it may happen that some genes that do not interact, are

eventually found to be correlated at some lag. This may be the reason behind the decrease in performance, when l_{max} is set to high values.

- On the dataset *E.coli*, setting l_{max} to 18 time points greatly improves the performance. Here, 18 time points correspond to 3 h. This number may be an indication of the true range of values of gene interaction lags.
- Relative to the dataset Yeast, the performance decrease that is seen when setting l_{max} to 18 time points is likely to be a result of the fact that this dataset is composed of only 25 points. The number of samples used to estimate dependencies between genes varies from n to $n - l_{max}$ where n is the number of samples in the dataset. On datasets of a low n, setting l_{max} to a high value may greatly reduce the number of samples used in the estimations, and if this number is too low, the variance of the algorithm increases, which causes the estimation of high correlations between genes that in reality do not interact. This may be happening in the case of the dataset Yeast, of 25 time points. When l_{max} is 90 min, the number of points used is only 7. If we compare with the dataset *E.coli*, when l_{max} is set to the maximum of 180 min, the number of samples used is still 14. When it comes to the dataset Fly, the number of samples used in the maximum l_{max}, of 9 h, is 27.
- The performance of fixed lag models (lag being one time point) should be influenced by the interval length of the time series. These models should perform, relatively to static models, better on time series with interval lengths similar to the true lags of interactions. It can be seen that fixed lag models perform consistently better than static models on the dataset Fly. The same cannot be said regarding the other two datasets, where static and fixed lag models perform similarly. This may indicate that fixed lag, with lag equal to one, models are more appropriate to model time series with a temporal step relatively high, in the order of 30 min, than to model time series of shorter steps.

4.1. ANALYSIS OF LAG DISTRIBUTIONS

Adaptive lag algorithms are based on the estimation of lags between pairs of genes. These should reflect in some way the true

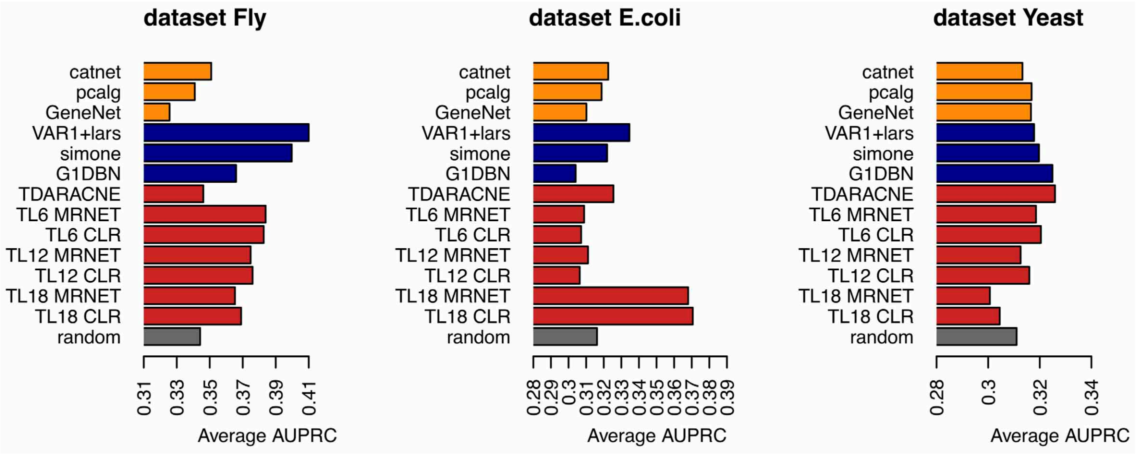


FIGURE 2 | Average AUPRC for the three datasets and different algorithms.

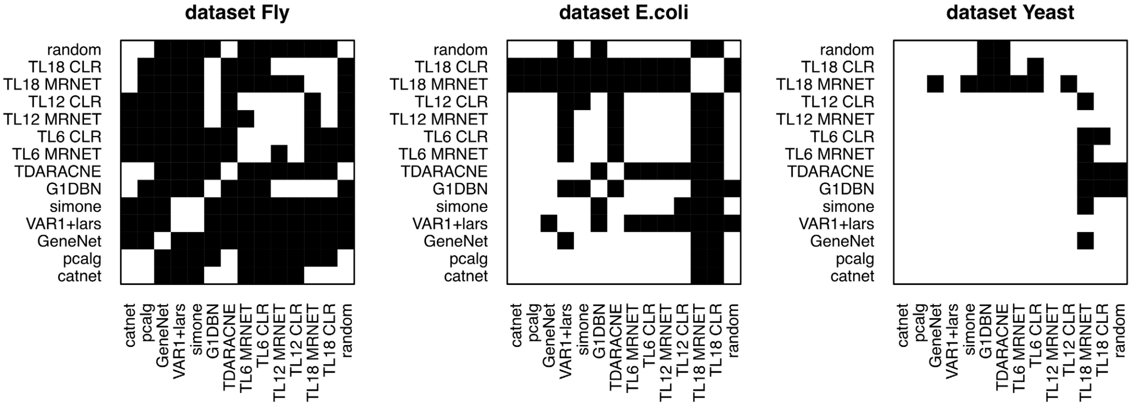


FIGURE 3 | Existence (black) or not (white) of a significant difference between the algorithms performance.

lags of the interactions. The **Figures 4**, **5** show the distribution of the estimation of lags of true interactions, done by the algorithms time-lagged MRNET and time-lagged CLR, when the maximum allowed lag is set to 6 time points and 18 time points. There is a relatively high value of lags estimated to be 0, on all datasets. An explanation may be that a number of assumed interactions (taken from the regulatory interactions lists) are not correct, and that the respective genes, instead of one regulating the other, are in fact co-regulated. These results may provide insights on the temporal lag of gene interactions. Different interactions are possibly characterized by different lags, and these can depend on the biological function of the interacting genes. Also, it is likely that different species have different gene interaction lag times. On the dataset Fly, adaptive lag models see their performance decrease when $l_{\max}$ is set to 9 h. We suggest that this is due to the fact that, when setting $l_{\max}$ to such a high value, some interaction lags are estimated to be unrealistically high. This is confirmed in the **Figure 5**, when we see that there is a relatively large proportion of interaction lags estimated to be between 7 and 9 h. We also note a peak on estimated lag values between 1 and 2 h, that can be an indication of some of the true interaction lags. On the dataset *E.coli*, there is a large proportion of interaction lags estimated to be between 130 and 180 min. The fact that there is a great performance increase, when $l_{\max}$ is set to 180 min, suggests that maybe some interactions are characterized by these large lag values. However, it is possible that these high estimated lag values are a result of a decrease in the number of samples used to estimate the lagged dependencies. This phenomenon is certainly happening in the dataset Yeast, when the number of samples used to estimate dependencies reduces to 25% of the time series length (7 samples, or 30 min), when $l_{\max}$ is set to 90 min, and increasing the variance of the algorithm.

4.2. STUDY LIMITATIONS

Only three gene expression datasets were used, each with its own distinct characteristics. Further validation of the results here presented should be made using other datasets, preferably with higher number of samples, as they become more available to biostatisticians. The inference of regulatory interactions was done on networks of 5 genes. All things equal, the network inference models here presented will return lower AUPRC scores if the number of genes increases, and the ratio true edges/possible edges decreases - the inference task becomes more challenging. Network inference was assessed using interactions reported in the literature, which means some true interactions may be missing, and some reported interactions may be biologically inexistent in the used datasets.

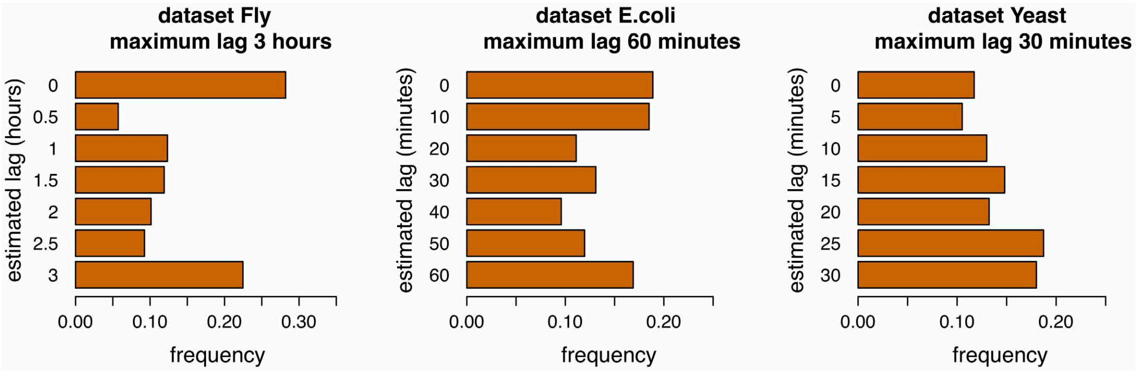


FIGURE 4 | Distribution of lags for the three datasets, maximum allowed lag is 6 time points.

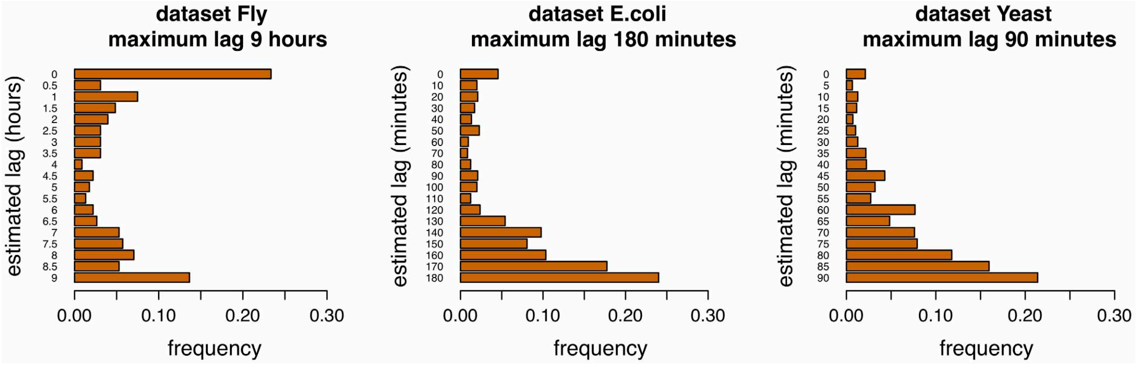


FIGURE 5 | Distribution of lags for the three datasets, maximum allowed lag is 18 time points.

5. CONCLUSION

Results obtained using three different datasets show that dynamic models perform better on the inference of gene regulatory interactions from time series, than static models such as Bayesian networks. This is explained by the inclusion of beneficial temporal information. Nevertheless, the overall performance of the assessed models is poor: only three and two models outperformed random in the *E.coli* and Yeast datasets, respectively. The differences in the results obtained in the datasets (a much higher performance variation in Fly, with most of the methods performing better than random) are likely due to the characteristics of the time series, such as the temporal interval. Regarding the dynamic models, the advantage of the considered fixed lag models is that they directly estimate conditional dependencies, instead of being based on pairwise dependencies, as the considered adaptive lag models are. On the other hand, the advantage of the adaptive lag models is that they can potentially infer interactions characterized by higher and variable lags. Their performance depends on the maximum allowed lag, $l_{\max}$, and care should be taken when defining this parameter: if it is set to an unrealistic high value, in the range of many hours, eventually interactions will be estimated at that range, hurting the network inference performance (we argue that this is seen in the results regarding the dataset Fly). If $l_{\max}$ is set to be equal to a high fraction of the length of the time series, lagged dependencies between genes will be estimated with a small number of samples, increasing the variance of the algorithm and decreasing its performance (this is seen in the results regarding the dataset Yeast). Relative to the lag of regulatory gene interactions, the fact that lag-one models (the fixed lag models) perform, compared with static models, better on a dataset with a temporal interval of 30 min than in datasets with lower temporal intervals (10 and 5 min) suggests that the range of lags of gene interactions is likely to be closer to 30 min than to 10 or 5 min. The experimental results also suggest that there may exist gene interactions characterized by a longer lag, in the order of a couple of hours. As a general set of rules, we conclude from the experiments here reported that dynamic methods should be used to predict interactions in time series; fixed lag methods (estimating conditional dependencies) should be used when the interval scale is high (30 min to hours); adaptive lag methods should be used when the maximum allowed lag is set to high values (order of a couple of hours), and, in order to prevent an excessive algorithm variance, the number of samples minus the maximum allowed lag is still high (the results obtained on the *E.coli* dataset suggest this value to be at least 14 samples).

AUTHOR CONTRIBUTIONS

Miguel Lopes designed and implemented the experimental run, and contributed to the writing of the paper. Gianluca Bontempi supervised the study and contributed to the writing of the paper.

ACKNOWLEDGMENTS

This work was supported by grants from the Communaute Francaise de Belgique Actions de Recherche Concertees (ARC).

REFERENCES

Abdulrehman, D., Monteiro, P. T., Teixeira, M. C., Mira, N. P., Lourenco, A. B., dos Santos, S. C., et al. (2011). Yeastract: providing a programmatic access to curated transcriptional regulatory associations in saccharomyces cerevisiae through a web services interface. *Nucleic Acids Res.* 39(Suppl. 1), D136–D140. doi: 10.1093/nar/gkq964

Balov, N., and Salzman, P. (2010). *How to Use Catnet Package.* Comprehensive R Archive Network (CRAN), R package vignette (version 1.13.8).

Bansal, M., Gatta, G. D., and Di Bernardo, D. (2006). Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics* 22, 815–822. doi: 10.1093/bioinformatics/btl003

Bonneau, R., Reiss, D., Shannon, P., Facciotti, M., Hood, L., Baliga, N., et al. (2006). The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets *de novo*. *Genome Biol.* 7:R36+. doi: 10.1186/gb-2006-7-5-r36

Charbonnier, C., Chiquet, J., and Ambroise, C. (2010). Weighted-lasso for structured network inference from time course data. *Stat. Appl. Genet. Mol. Biol.* 9:1. doi: 10.2202/1544-6115.1519

Dempster, A. (1972). Covariance selection. *Biometrics* 28, 157–175. doi: 10.2307/2528966

Ding, C., and Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *J. Bioinform. Comput. Biol.* 3, 185–205. doi: 10.1109/CSB.2003.1227396

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). *Ann. Stat.* 32, 407–499. doi: 10.1214/009053604000000067

Faith, J. J., Hayete, B., Thaden, J. T., Mogno, I., Wierzbowski, J., Cottarel, G., et al. (2007). Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles. *PLoS Biol.* 5:e8. doi: 10.1371/journal.pbio.0050008

Gallo, S. M., Gerrard, D. T., Miner, D., Simich, M., Des Soye, B., Bergman, C. M., et al. (2010). REDfly v3.0: toward a comprehensive database of transcriptional regulatory elements in Drosophila. *Nucleic Acids Res.* 39(Suppl. 1), D118–D123. doi: 10.1093/nar/gkq999

Gama-Castro, S., Salgado, H., Peralta-Gil, M., Santos-Zavaleta, A., Muniz-Rascado, L., Solano-Lira, H., et al. (2011). RegulonDB version 7.0: transcriptional regulation of *Escherichia coli* K-12 integrated within genetic sensory response units (Gensor Units). *Nucleic Acids Res.* 39(Suppl. 1), D98–D105. doi: 10.1093/nar/gkq1110

Hooper, S., Boue, S., Krause, R., Jensen, L., Mason, C., Ghanim, M., et al. (2007 (data accessible at NCBI GEO database (Edgar et al., 2002), accession GSE6186)). Identification of tightly regulated groups of genes during Drosophila melanogaster embryogenesis. *Mol. Syst. Biol.* 3:72. doi: 10.1038/msb4100112

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., and Bühlmann, P. (2012). Causal inference using graphical models with the R package pcalg. *J. Stat. Softw.* 47, 1–26.

Lauritzen, L. (1996). *Graphical Models*. Oxford: Oxford University Press. ISBN 0-19-852219-3

Lebre, S. (2009). Inferring dynamic genetic networks with low order independencies. *Stat. Appl. Genet. Mol. Biol.* 8:9. doi: 10.2202/1544-6115.1294

Lezon, T. R., Banavar, J. R., Cieplak, M., Maritan, A., and Fedoroff, N. V. (2006). Using the principle of entropy maximization to infer genetic interaction networks from gene expression patterns. *Proc. Natl. Acad. Sci. U.S.A.* 103, 19033–19038. doi: 10.1073/pnas.0609152103

Lopes, M., Meyer, P., and Bontepi, G. (2012). Estimation of temporal lags for the inference of gene regulatory networks from time series. in *BeneLearn 2012: Proceedings of the 21st Belgian-Dutch Conference on Machine Learning* (Ghent), 19–26.

Margaritis, D. (2003). *Learning Bayesian Network Model Structure from Data.* PhD thesis, School of Computer Science, Carnegie-Mellon University, Pittsburgh, PA.

Margolin, A. A., Nemenman, I., Basso, K., Klein, U., Wiggins, C., Stolovitzky, G., et al. (2005). ARACNE: an Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context. *BMC Bioinformatics* 7(Suppl 1):S7+. doi: 10.1186/1471-2105-7-S1-S7

Meyer, P. E., Kontos, K., Lafitte, F., and Bontempi, G. (2007). Information-theoretic inference of large transcriptional regulatory networks. *EURASIP J. Bioinf. Syst. Biol.* 2007. doi: 10.1155/2007/79879. Available online at: http://bsb.eurasipjournals.com/content/2007/June/2007

Needham, C. J., Bradford, J. R., Bulpitt, A. J., and Westhead, D. R. (2007). A primer on learning in bayesian networks for computational biology. *PLoS Comput. Biol.* 3:e129+. doi: 10.1371/journal.pcbi.0030129

Opgen-Rhein, R., and Strimmer, K. (2007). From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. *BMC Syst. Biol.* 1:37. doi: 10.1186/1752-0509-1-37

Perrin, B. E., Ralaivola, L., Mazurie, A., Bottani, S., Mallet, J., and Buc, F. D. (2003). Gene networks inference using dynamic Bayesian networks. *Bioinformatics* 19(Suppl. 2), II138–II148. doi: 10.1093/bioinformatics/btg1071

Pramila, T., Wu, W., Miles, S., Noble, W., and Breeden, L. (2006 (data accessible at NCBI GEO database (Edgar et al., 2002), accession GSE4987)). The forkhead transcription factor hcm1 regulates chromosome segregation genes and fills the s-phase gap in the transcriptional circuitry of the cell cycle. *Genes Dev.* 20, 2266–2278. doi: 10.1101/gad.1450606

Schaefer, J., Opgen-Rhein, R., and Strimmer, K. (2006). Reverse engineering genetic networks using the geneNet package. *R News* 6/5, 50–53.

Schäfer, J., and Strimmer, K. (2005). An empirical Bayes approach to inferring large-scale gene association networks. *Bioinformatics* 21, 754–764. doi: 10.1093/bioinformatics/bti062

Smith, A. V., Yu, J., Smulders, T. V., Hartemink, A. J., and Jarvis, E. D. (2006). Computational inference of neural information flow networks. *PLoS Comput. Biol.* 2:e161+. doi: 10.1371/journal.pcbi.0020161

Spirtes, P., Glymour, C., , and Scheines, R. (1993). "Lecture Notes in Statistics," in *Causation, prediction, and search*. Vol. 81 (Springer).

Tibshirani, R. (1994). Regression shrinkage and selection via the lasso. *J. R. S. Soc. Ser. B* 58, 267–288.

Traxler, M. F., Chang, D.-E., and Conway, T. (2006 (data accessible at NCBI GEO database (Edgar et al., 2002), accession GSE7265)). Guanosine 3′ 5′-bispyrophosphate coordinates global gene expression during glucose-lactose diauxie in *Escherichia coli*. *Proc. Natl. Acad. Sci. U.S.A.* 103, 2374–2379. doi: 10.1073/pnas.0510995103

Zoppoli, P., Morganella, S., and Ceccarelli, M. (2010). Timedelay-aracne: reverse engineering of gene networks from time-course data by an information theoretic approach. *BMC Bioinformatics* 11:154. doi: 10.1186/1471-2105-11-154

Conflict of Interest Statement: 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.

Received: 30 July 2013; paper pending published: 24 August 2013; accepted: 10 December 2013; published online: 24 December 2013.

Citation: Lopes M and Bontempi G (2013) Experimental assessment of static and dynamic algorithms for gene regulation inference from time series expression data. Front. Genet. ***4****:303. doi: 10.3389/fgene.2013.00303*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 26 December 2013
doi: 10.3389/fgene.2013.00291

Network statistics of genetically-driven gene co-expression modules in mouse crosses

Marie-Pier Scott-Boyer[1], Benjamin Haibe-Kains[2] and Christian F. Deschepper[1]*

[1] Cardiovascular Biology Research Unit, Institut de Recherches Cliniques de Montréal, Montreál, QC, Canada
[2] Bioinformatics and Computational Genomics Research Unit, Institut de Recherches Cliniques de Montréal, Montreál, QC, Canada

Edited by:
Frank Emmert-Streib, Queen's University Belfast, UK

Reviewed by:
Xia Yang, University of California Los Angeles, USA
Matthias Dehmer, UMIT - University for Health Sciences, Medical Informatics and Technology, Austria

***Correspondence:**
Christian F. Deschepper, Cardiovascular Biology Research Unit, Institut de Recherches Cliniques de Montréal, Université de Montréal, 110 Avenue des Pins, Montreál, QC H2W 1R7, Canada
e-mail: christian.deschepper@ircm.qc.ca

In biology, networks are used in different contexts as ways to represent relationships between entities, such as for instance interactions between genes, proteins or metabolites. Despite progress in the analysis of such networks and their potential to better understand the collective impact of genes on complex traits, one remaining challenge is to establish the biologic validity of gene co-expression networks and to determine what governs their organization. We used WGCNA to construct and analyze seven gene expression datasets from several tissues of mouse recombinant inbred strains (RIS). For six out of the 7 networks, we found that linkage to "module QTLs" (mQTLs) could be established for 29.3% of gene co-expression modules detected in the several mouse RIS. For about 74.6% of such genetically-linked modules, the mQTL was on the same chromosome as the one contributing most genes to the module, with genes originating from that chromosome showing higher connectivity than other genes in the modules. Such modules (that we considered as "genetically-driven") had network statistic properties (density and centralization) that set them apart from other modules in the network. Altogether, a sizeable portion of gene co-expression modules detected in mouse RIS panels had genetic determinants as their main organizing principle. In addition to providing a biologic interpretation validation for these modules, these genetic determinants imparted on them particular properties that set them apart from other modules in the network, to the point that they can be predicted to a large extent on the basis of their network statistics.

Keywords: genetics, network inference, mouse recombinant inbred strains, gene co-expression modules, chromosome domain

INTRODUCTION

In recent years, new technologies such as microarrays have made it possible to generate large numbers of gene expression datasets. To understand how genes interact with one another, methods have been developed to construct gene co-expression networks, and then identify modules of highly connected genes. "Weighted Gene Co-expression Network Analysis" (WGCNA) is the most established and widely used of such methods (Langfelder and Horvath, 2008). Several studies have used these methods to construct (on the basis of gene expression datasets) gene co-expression networks, and then identify modules of highly connected genes (Califano et al., 2012; Cho et al., 2012; Weiss et al., 2012). One common premise of such analyses is that co-expressed genes within modules are more likely to share biological functions. Accordingly, it has been reported several times that some modules detected by gene co-expression analysis show enrichment for genes originating from a particular biologic pathway (Gargalovic et al., 2006; Yang et al., 2009; Rhinn et al., 2013).

The properties of gene co-expression modules can be analyzed in several ways. Eigengenes are values that represent the first principal component of all expression profiles in modules. When networks are constructed using expression data from individuals in a genetic cross, genetic mapping can be performed to test whether the eigengenes of modules show linkage to quantitative trait loci (QTLs), the latter being called "module QTLs" (mQTLs). For instance, mQTLs have been detected in some mouse F2 genetic crosses, with some of them having profiles matching that of phenotypic QTLs (Davis et al., 2012; Leduc et al., 2012). Such findings suggest that the same genetic determinants may link to both a phenotype and the expression levels of genes within the associated module. This suggests that genetic linkage, rather than function, may contribute to coexpression modules detected in genetic crosses However, it is currently not known whether the contributions of genetic determinants to gene co-expression modules represent a common phenomenon, and/or whether corresponding modules have distinctive properties.

Recombinant inbred strain (RIS) are organisms derived from the progenies of crosses of parental inbred strains, and where recombination events between parental chromosomes have been made permanent by long-term inbreeding. When tissue gene expression is measured in RIS by using several animals per strain (to provide both biologic and technical replicates), genetic variations constitute the main cause of variance in gene expression level. Moreover, RIS are homozygous at all loci, which maximizes the potential effect of genetic variation on gene expression. Panels of RIS therefore constitute sensitive backgrounds to study links

Network Inference

- Type of Biological Networks
  The analyzed networks correspond to gene-co-expression networks constructed from gene expression data obtained in mouse genetic crosses, where genetic variants are the main cause of gene expression variance.
- Utility of the Inferred Networks
  We focused on the detection of gene co-expression network modules showing linkage to quantitative trait loci in multiple independent datasets. We tested the reproducibility of our findings across multiple datasets and across two network inference methods.
- Summary of Results
  In tissues from mouse recombinant inbred strain (RIS) panels, a sizeable portion of gene co-expression modules had genetic determinants as their main organizing principle. These modules had particular properties that set them apart from other modules in the network, to the point that they can be predicted on the sole basis of their gene expression profile characteristics and associated network statistics.

between genomic variants and gene expression. To test to which extent genomic variants may link to coordinate gene expression within gene co-expression modules, we analyzed publicly available gene expression datasets obtained in several tissues from two kinds of mouse RIS panels. In such panels, we found that a sizeable proportion of gene co-expression modules showed linkage to mQTLs. Moreover, such modules had network statistics that set them apart from other modules in the network. Lastly we observed that these network statistics are sufficiently discriminative to predict, solely on the basis of gene expression, which modules are likely to be genetically-driven.

MATERIALS AND METHODS

DATASETS PREPROCESSING

Discovery datasets were used to test whether gene co-expression modules showing linkage to mQTLs had properties and network statistics that set them apart from other modules. In follow-up experiments, validation sets were used to test whether the properties and network statistics of gene co-expression modules (as determined in the validation sets) could be used to predict accurately whether gene co-expression modules corresponded to a particular type of modules. The discovery sets comprised data obtained in five tissues and one purified cell population from BxD mouse RIS, as well as one tissue from AxB/BxA mouse RIS (**Table 1**). The validation sets comprised data obtained in one purified cell population from BxD and one tissue from AxB/BxA mouse RIS (**Table 1**). All data were obtained from the www.genenetwork.org web site, and comprised both gene expression data as well as genomic maps. For gene expression analysis, we used for each gene the one single probe that corresponded to the most variant one. To reduce computation time and facilitate the comparisons between networks, we used the data for the 20,000 most variant genes in each tissue (corresponding to the number of genes that was the smallest common denominator among all datasets used).

NETWORK CONSTRUCTION AND MODULES DETECTION

We used the "Weighted Gene Co-expression Network Analysis" (WGCNA) R package (Langfelder and Horvath, 2008) to construct the gene co-expression networks. To avoid computationally intensive tuning of WGCNA parameters, we used all default parameters as proposed previously (Zhang and Horvath, 2005). Within a network, each gene represents a node, and the connections between nodes are defined as edges. To obtain comparable networks between the different datasets, we utilized the top 25% most significant edges in each network. To define modules (i.e., clusters of highly interconnected genes), we used the dynamic tree cut algorithm implemented in the dynamicTreeCut function. "Eigengenes" are summary values representative of the gene expression profiles in corresponding modules. Accordingly, eigengene values can be used to detect "module-QTLs" (mQTLs), i.e., QTLs showing linkage to entire gene co-expression modules(Davis et al., 2012; Leduc et al., 2012). For each module, we used WGCNA to calculate its corresponding eigengene value, and performed QTL mapping with the "R-QTL" tool (Broman et al., 2003), using a detection threshold corresponding to a "logarithm-of-the-odds" (LOD) score of 3.3 (Lander and Kruglyak, 1995). Modules shown for illustration were drawn using the Cytoscape software (Shannon et al., 2003).

In order to test the robustness of our findings with respect to the network inference approach, we also used the GeneNet R package (Schaefer et al., 2006) to construct the gene co-expression networks. This method uses partial correlation to calculate the link between two genes and has the advantage of not requiring any parameter (with the exception of the correlation threshold used to select the most relevant edges). The results derived from GeneNet are reported in Supplementary Information.

COMPARISONS BETWEEN MODULES

To estimate the contribution of each chromosome to a module, we calculated the percentage of genes that each chromosome contributed to the module. The one chromosome with the highest percentage was considered as the "top contributing" chromosome, and the corresponding percentage value was considered as the "enrichment index for single chromosome contribution." To calculate a normalized index (and thus allow comparisons across modules), the enrichment index value was divided by the mean of the percentages of genes contributed by all other chromosomes in the module.

Each module was also characterized in terms of its "network statistics" (also known as "fundamental network concepts") (Dong and Horvath, 2007). We thus calculated the values

Table 1 | Gene expression datasets from tissues of mouse RIS used for either discovery or validation analyses in the present study.

Discovery datasets	Mouse RIS panel	Tissue	Microarray platform	# of WGCNA modules total/gen	# of GeneNet modules total/gen
GN373	24 AXB-BXA	Liver	Affy	95/10	313/31
GN207	68 BXD	Whole eyes	Affy	49/11	42/16
GN160	47 BXD	Lung	Affy	42/12	124/34
GN389	48 BXD	Pituitary	Affy	52/15	65/21
GN122	33 BXD	Regulatory T cells	Affy	77/11	311/34
GN260	38 BXD	Spleen	Illumina	45/13	177/52
GN323	*46 BXD*	*Brain amygdala*	*Affy*	*34/0*	*168/32*
VALIDATION DATASETS					
GN210	24 AXB-BXA	Whole eyes	Illumina	43/4	74/6
GN319	31 BXD T cell helper	Helper T cells	Affy	68/12	280/39

First column: GeneNetwork ID number of the dataset. Second column: type of mouse RIS and number of strains used in the study. Third column: name of tissue or type of cell used. Fourth column: microarray platform used in the study. Fifth column: number of gene co-expression modules (total and "type 1" genetic) detected in each network using WGCNA for construction of the network. Since no genetic module was detected in dataset GN323 using default parameters, this dataset was not used for analysis of WGCNA modules. Sixth column: number of gene co-expression modules (total and "type 1" genetic) detected in each network using GeneNet for construction of the network.

of heterogeneity, centralization, and density, using the function "fundamentalNetworkConcepts" of WGCNA R package (Langfelder and Horvath, 2008). Comparisons between groups were performed using either the non-parametric Wilcoxon Signed Rank test (for binary comparisons) or the Kruskal Wallis test (for comparisons involving more than 2 classes). Combined P-values were calculated using the Z transform approach (Whitlock, 2005), using the survcomp R package (Schröder et al., 2011).

VALIDATION TESTS

In the datasets used for validation (**Table 1**), we first calculated the values of heterogeneity, centralization, density and normalized enrichment index in order to identify which modules could be considered as being "genetically-driven" (according to our own definition: see below). We then ranked all modules according to corresponding values by grouping them in "top percentile" windows ranging from the top 5% to the top 80% (in successive 5% steps). We then: (1) tested whether modules in the top percentile windows corresponded or not to genetically-driven modules, and (2) calculated the accuracy with which each network statistic value categorized corresponding modules. For the latter tests, we calculated the numbers of modules whose characteristics were truly positively predicted (TP), truly negatively predicted (TN), falsely positively predicted (FP) and falsely negatively predicted (FN), and we calculated the receiving operating characteristics (ROC) curves based on sensitivity and specificity, using the ROCR package in R.

All network statistics (heterogeneity, centralization, density and normalized enrichment index) were analyzed independently.

RESULTS:

GENETICALLY-LINKED AND GENETICALLY-DRIVEN MODULES

Gene co-expression networks were built using WGCNA for seven RIS mouse expression datasets (**Table 1**). Since the datasets were obtained using different microarray platforms for different tissues from different animal crosses, we built gene co-expressions network using the same number of genes (the 20,000 most varying genes) and selected the 25% most significant edges in the networks. This approach allowed us to generate networks with comparable characteristics. For each network, we extracted modules containing at least 30 genes, and found that networks contained in average 56 modules (**Table 1**). Genomic mapping analyses were performed for the eigengenes of all modules to determine whether we could detect linkage of modules to mQTLs.

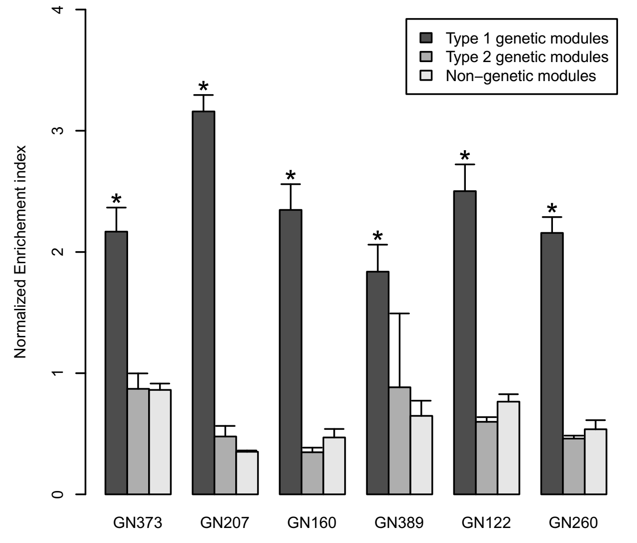


FIGURE 1 | The bar graphs represent normalized enrichment indices (mean ± SD) in the 6 tested discovery datasets. The indices quantify to which extent genes in co-expression network originate from a single chromosome. Black bars: values for "genetically-driven" modules (type 1 genetic modules); gray bars: values for the other "genetic" modules (type 2); white bars: values for "non-genetic modules." *P < 0.05 (Kruskal Wallis tests).

We found that in 6/7 networks, we could detect modules that could be considered as "genetically-linked," on the basis of showing linkage to a mQTL. In these 6 networks, the proportion of such genetically-linked modules averaged 29.3% (sd 8.4%) (with values ranging from 15.7 to 36.7%)., could be. For 74.6% of these genetically-linked modules, the chromosome harboring the mQTL corresponded to the top-contributing chromosome. Since in such cases the location of the mQTL corresponded to the chromosome that contributed most genes to the modules, we considered these particular modules to be "genetically-driven." In further comparisons, we called such modules "type 1 genetic modules"; genetically-linked modules where the top-contributing chromosome was not the same as the one harboring the mQTL were called "type 2 genetic modules." For both types of genetic modules, we calculated the "normalized enrichment index for single chromosome contribution," and compared it to that of other modules that did not show linkage to any mQTL ("non-genetic modules") (**Figure 1**). In all 6 tested WGCNA networks, normalized enrichment index of type 1 genetic modules was significantly higher than that of other types of modules, with type 2 genetic modules showing no difference in comparison to non-genetic modules (**Figure 1**).

NETWORK STATISTICS

For further analyses, we studied the three following network statistics (Dong and Horvath, 2007): (I) density (which corresponds to the mean connectivity of the network); (II) centralization (which takes the value 0 if the network has a star topology and the value 1 if all nodes have the same connectivity); and (III) heterogeneity (which is the coefficient of variation of the connectivity of the network). Within each studied network, we calculated these three values for genetically-driven (type 1 genetic) modules, and compared them to that obtained other modules in the network (including both the type 2 genetic and the non-genetic modules) (**Figure 2**). Density was significantly higher ($P < 0.05$) in genetically-driven modules for all six networks, whereas centralization was significantly higher in genetically-driven modules

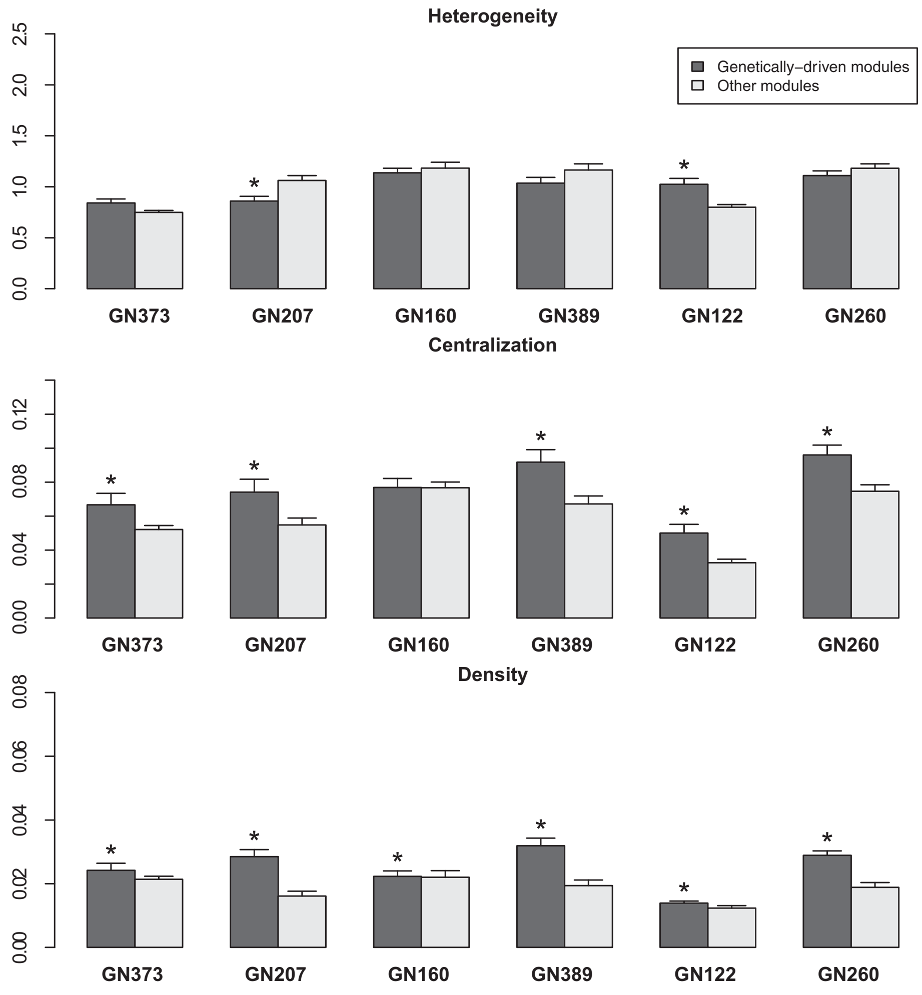


FIGURE 2 | The bar graphs represent the heterogeneity, centralization and density values (mean ± SD) of modules within networks from the 6 tested discovery datasets. Black bars: "genetically-driven" modules; gray bars: other modules. *$P < 0.05$ (Wilcoxon Signed Rank test).

for 5 out of 6 of the studied networks (**Figure 2**). We did not observe a consistent trend for heterogeneity (**Figure 2**). When all six modules were combined to calculate overall *P*-values, the differences between type 1 genetic modules vs. all other modules were significant for centralization ($p = 9.68\text{e-}06$) and density ($p = 2.02\text{e-}08$), but not for heterogeneity ($p = 0.457$). Differences in network statistics were not due to differences in the sizes of the modules since the latter showed no significant difference in genetically-driven networks compared to the other modules.

Given that (I) density was higher in genetically-driven modules; and (II) these modules showed enrichment in genes originating from one single chromosome, we tested in these modules whether the connectivity of genes from the top-contributing chromosome was higher than that of other genes in the modules We found that this was indeed the case, with differences being significant for genetically-driven modules in 5 out of the 6 networks tested (**Figure 3**). When all datasets were combined, the overall *P*-value for connectivity was 5.8e-18.

VALIDATION TESTS

We used two independent validation datasets to test how robustly network statistics values could discriminate genetically-driven modules from the other ones. In the GN319 dataset, the "area under the curve" (AUC) values for ROC curves were all higher than 0.9, with normalized enrichment index and centralization being most predictive (**Figure 4**). Even in GN210 (where the proportion of type 1 genetic networks was <10%), network statistics still had good predictive power, since all AUC values were greater than 0.7 (data not shown).

ALTERNATIVE NETWORK INFERENCE METHOD

To test the robustness of our findings we performed the analyses previously described using GeneNet (Schaefer et al., 2006) as an alternative method to build networks of gene co-expression. Interestingly, whereas the number of modules detected in the WGCNA networks averaged 60 ($sd = 21$), we detected a higher number of modules averaging 172 ($sd = 118$) in the corresponding networks built using GeneNet, although this difference was not significant (*p*-value = 0.06 by two-sided paired Wilcoxon signed rank test). Nonetheless, regardless of the method used for network inference, our observations concerning the differences between genetic and non-genetic modules held true (with in addition heterogeneity also being significantly higher in genetically-driven modules than in non-genetic modules). The various differences in network statistics are further illustrated in two modules of similar sizes detected in the GN122 dataset on the basis of networks constructed with GeneNet (**Figure 5**)

DISCUSSION

Complex genetic quantitative traits result from the many interactions of genetic variants with environmental factors, and only a minority of are believed to result from the dysregulation of only one gene (Plomin et al., 2009). Moreover, biological systems are typically organized as modular networks where genes act synergistically rather than representing the sum of their individuals actions (Cho et al., 2012; Weiss et al., 2012). Consequently, gene co-expression network analyses have been proposed as a means to better understand the mechanisms of complex regulatory biologic processes (Califano et al., 2012; Cho et al., 2012; Weiss et al., 2012). Up until now, much of the interpretation of gene co-expression has relied on empirical observations.

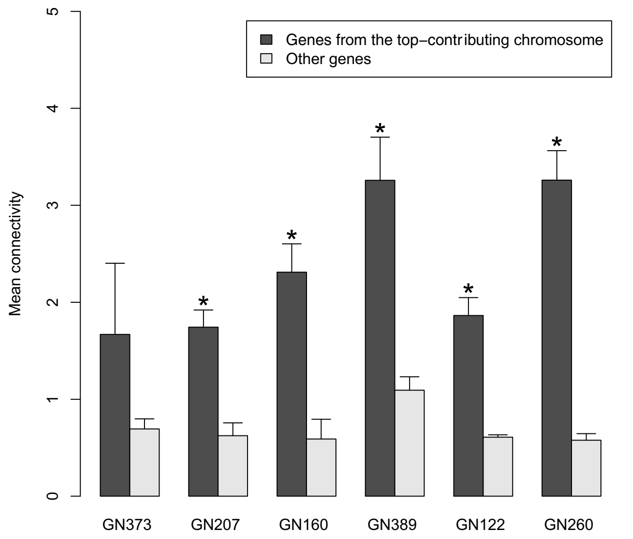


FIGURE 3 | Comparisons (for the genetically driven modules detected in the 6 tested discovery datasets) of the mean connectivity values of genes originating from the top-contributing chromosome vs. that of other genes in the modules. The bars represent mean ± SD. **P* < 0.05 (Wilcoxon Signed Rank test).

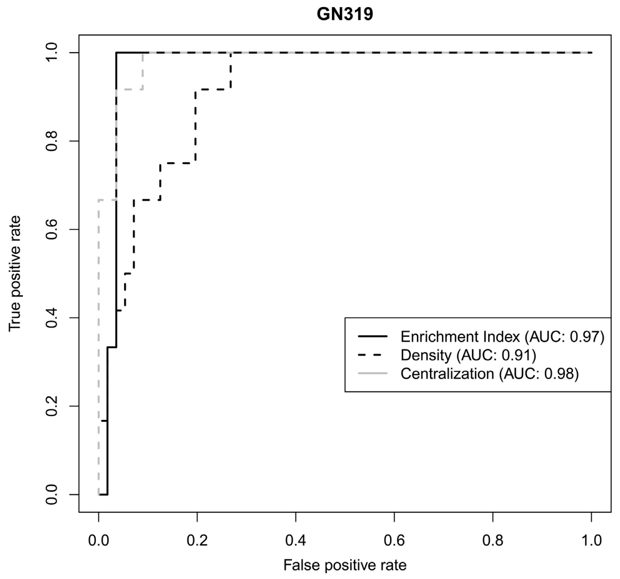


FIGURE 4 | Receiver operating characteristic (ROC) curves illustrating how 3 different network statistics discriminate genetically-driven modules from other modules in a validation set.

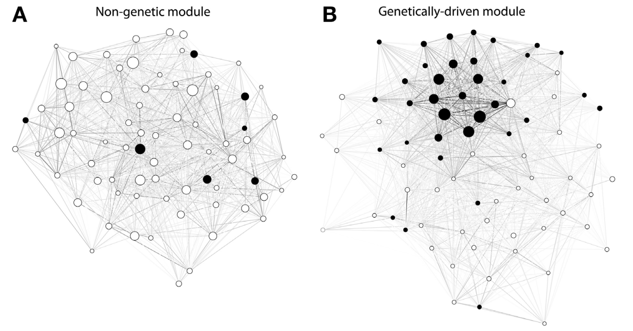


FIGURE 5 | Illustrative examples of gene expression modules detected in the GN122 dataset from regulatory T cells (on the basis of the gene co-expression network being built using GeneNet). Each module was of equal size as they both contained a total of 75 genes; **(A)**: non-genetic module; **(B)**: genetic module. Each node is represented by a circle, either full (when the corresponding gene originates from the top contributing chromosome) or empty (other genes). The edges are colored according to a gray scale, where the darkness of the edge is proportional to the connectivity between 2 nodes. It can be seen that the genetically-driven module contains a higher number of genes from the top-contributing chromosome. Moreover, that module contains a core a several genes displaying connectivity levels that are much higher than other genes in the module, which corresponds to the fact that the values of density and centralization were higher in genetically-driven modules.

For instance, one common strategy has been to rely on annotations (either gene ontology or pathway information) to test whether module show enrichment for genes related to annotated functions. However, the drawbacks are that: (1) "canonical" pathways are often still incomplete, and in fact represent "over-simplifications"; and (2) enrichment analyses are biased toward what we already know (Carro et al., 2010; Farber, 2013).

In some instances, gene co-expression modules have shown linkage to mQTLs in genetic animal crosses, with some of them having profiles matching that of phenotypic QTLs (Davis et al., 2012; Leduc et al., 2012). In such cases, it is likely that a valid biologic process drives gene co-expression in the module. To test to which extent such mechanisms could underlie the organization of gene co-expression modules in genetic crosses, we performed gene co-expression network analyses of datasets originating from eight different tissues and two different panels of mouse RIS. We found (on the basis of detection of mQTLs) evidence of genetic contributions for an average of 29% of the modules. For about 73% of these genetically-linked modules, the influence of the genetic determinants appeared to be even stronger, as the mQTL was located on the same chromosome that was the highest contributor of genes to the module. In such modules, the normalized enrichment index for single chromosome contribution was significantly higher than in other types of modules. Given this clustering of co-expressed genes around mQTLs, we considered such modules as being "genetically-driven." These modules also appear to have specific features in terms of network statistics: (1) their density was higher, indicating that their mean connectivity was higher than that of other modules; (2) their centralization value was higher, which is compatible with the presence of a core several highly connected genes (in opposition to the presence of one main hub gene regulating all others in the module). Since genetically-driven modules show enrichment for genes originating from one chromosome, these differences in network statistics might be explained if these genes showed higher connectivity than that of other genes in the module. We thus tested this possibility, and found that within genetically-driven modules, connectivity of genes from the top-contributing chromosome was in average 2.25 higher than that of other genes in the module. Our observations did not depend on network inference approaches, as similar conclusions were reached using either WGCNA or GeneNet.

Thus, the gene composition and network statistics of genetically-driven modules indicate that one of their main component is constituted by several highly connected genes originating from one chromosome. In mammals, co-expressed genes have been reported to cluster both at either short-range (1 Mb) or long-range (>10 Mb) levels (Woo et al., 2010). Moreover, we have recently reported in mouse RIS the existence of clusters of co-expressed genes that all show linkage to one common QTL (Scott-Boyer and Deschepper, 2013). Corresponding genomic regions showed a greater abundance of polymorphic SINE retrotransposons, the latter showing enrichment for the motifs of binding sites for various regulators of transcription. We postulate that such mechanisms may account (at least in part) for the presence of several high co-expressed genes within chromosome domains, which constitute the core of gene co-expression modules that have characteristics that set them apart from other kinds of modules.

In mouse RIS, genetically-driven modules are not a rare occurrence, since they constitute in average 21% of all modules. Their network statistics differ substantially from that of other modules, with high AUC values being obtained for the normalized enrichment index as well as the density and centralization valuesThis suggests that genetically-driven modules can, to some extent, be predicted solely on the basis of their gene expression patterns.

In summary, genetic determinants constitute one main organizing principle of a sizeable portion of gene-co-expression modules detected in mouse RIS panels, which provides a biologic validation for corresponding modules. In addition, these modules appear to derive from cores of highly inter-connected genes clustering on one chromosome. This may constitute one particular mechanism driving gene co-expression, which imparts on genetically-driven modules particular properties. These properties set them apart from other modules in their network, to the point that they can be predicted to a large extent on the basis of their network statistics. Of note, it is possible that RIS panels provide a background that is particularly appropriate for the detection of genetically-driven modules. It remains to be seen to which extent they will be detectable in other types of genetic crosses.

AUTHOR CONTRIBUTIONS

All authors participated to the writing of the manuscript and design of the experiments, with most analyses having been performed by Marie-Pier Scott-Boyer.

ACKNOWLEDGMENTS

This work was supported by grant MOP-93583 from the Canadian Institutes for Health Research (CIHR).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://www.frontiersin.org/Journal/10.3389/fgene.2013.00291/abstract

REFERENCES

Broman, K. W., Wu, H., Sen, S., and Churchill, G. A. (2003). R/qtl: QTL mapping in experimental crosses. *Bioinformatics* 19, 889–890. doi: 10.1093/bioinformatics/btg112

Califano, A., Butte, A. J., Friend, S., Ideker, T., and Schadt, E. (2012). Leveraging models of cell regulation and GWAS data in integrative network-based association studies. *Nat. Genet.* 44, 841–847. doi: 10.1038/ng.2355

Carro, M. S., Lim, W. K., Alvarez, M. J., Bollo, R. J., Zhao, X., Snyder, E. Y., et al. (2010). The transcriptional network for mesenchymal transformation of brain tumours. *Nature* 463, 318–325. doi: 10.1038/nature08712

Cho, D.-Y., Kim, Y.-A., and Przytycka, T. M. (2012). Chapter 5: network biology approach to complex diseases. *PLoS Comput. Biol.* 8:e1002820. doi: 10.1371/journal.pcbi.1002820

Davis, R. C., Nas, A., van Castellani, L. W., Zhao, Y., Zhou, Z., Wen, P., et al. (2012). Systems genetics of susceptibility to obesity-induced diabetes in mice. *Physiol. Genomics* 44, 1–13. doi: 10.1152/physiolgenomics.00003.2011

Dong, J., and Horvath, S. (2007). Understanding network concepts in modules. *BMC Syst. Biol.* 1:24. doi: 10.1186/1752-0509-1-24

Farber, C. R. (2013). Systems-level analysis of genome-wide association data. *G3* 3, 119–129. doi: 10.1534/g3.112.004788

Gargalovic, P. S., Imura, M., Zhang, B., Gharavi, N. M., Clark, M. J., Pagnon, J., et al. (2006). Identification of inflammatory gene modules based on variations of human endothelial cell responses to oxidized lipids. *Proc. Natl. Acad. Sci. U.S.A.* 103, 12741–12746. doi: 10.1073/pnas.0605457103

Lander, E., and Kruglyak, L. (1995). Genetic dissection of complex traits: guidelines for interpreting and reporting linkage results. *Nat. Genet.* 11, 241–247. doi: 10.1038/ng1195-241

Langfelder, P., and Horvath, S. (2008). WGCNA: an R package for weighted correlation network analysis. *BMC Bioinformatics* 9:559. doi: 10.1186/1471-2105-9-559

Leduc, M. S., Blair, R. H., Verdugo, R. A., Tsaih, S.-W., Walsh, K., Churchill, G. A., et al. (2012). Using bioinformatics and systems genetics to dissect HDL-cholesterol genetics in an MRL/MpJ × SM/J intercross. *J. Lipid Res.* 53, 1163–1175. doi: 10.1194/jlr.M025833

Plomin, R., Haworth, C. M. A., and Davis, O. S. P. (2009). Common disorders are quantitative traits. *Nat. Rev. Genet.* 10, 872–878. doi: 10.1038/nrg2670

Rhinn, H., Fujita, R., Qiang, L., Cheng, R., Lee, J. H., and Abeliovich, A. (2013). Integrative genomics identifies APOE ε4 effectors in Alzheimer's disease. *Nature* 500, 45–50. doi: 10.1038/nature12415

Schaefer, J., Opgen-Rhein, R., and Strimmer, K. (2006). Reverse engineering genetic networks using the genenet package. *R News* 6/5, 50–53. Available online at: http://uni-leipzig.de/~strimmer/lab/publications/misc/GeneNet-Rnews2006.pdf

Schröder, M. S., Culhane, A. C., Quackenbush, J., and Haibe-Kains, B. (2011). survcomp: an R/Bioconductor package for performance assessment and comparison of survival models. *Bioinformatics* 27, 3206–3208. doi: 10.1093/bioinformatics/btr511

Scott-Boyer, M.-P., and Deschepper, C. F. (2013). Genome-wide detection of gene coexpression domains showing linkage to regions enriched with polymorphic retrotransposons in recombinant inbred mouse strains. *G3* 3, 597–605. doi: 10.1534/g3.113.005546

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., et al. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.* 13, 2498–2504. doi: 10.1101/gr.1239303

Weiss, J. N., Karma, A., MacLellan, W. R., Deng, M., Rau, C. D., Rees, C. M., et al. (2012). "Good enough solutions" and the genetics of complex diseases. *Circ. Res.* 111, 493–504. doi: 10.1161/CIRCRESAHA.112.269084

Whitlock, M. C. (2005). Combining probability from independent tests: the weighted Z-method is superior to Fisher's approach. *J. Evol. Biol.* 18, 1368–1373. doi: 10.1111/j.1420-9101.2005.00917.x

Woo, Y. H., Walker, M., and Churchill, G. A. (2010). Coordinated expression domains in mammalian genomes. *PLoS ONE* 5:e12158. doi: 10.1371/journal.pone.0012158

Yang, X., Deignan, J. L., Qi, H., Zhu, J., Qian, S., Zhong, J., et al. (2009). Validation of candidate causal genes for obesity that affect shared metabolic pathways and networks. *Nat. Genet.* 41, 415–423. doi: 10.1038/ng.325

Zhang, B., and Horvath, S. (2005). A general framework for weighted gene co-expression network analysis. *Stat. Appl. Genet. Mol. Biol.* 4:17. doi: 10.2202/1544-6115.1128

Conflict of Interest Statement: 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.

Received: 11 September 2013; paper pending published: 11 October 2013; accepted: 29 November 2013; published online: 26 December 2013.

Citation: Scott-Boyer M-P, Haibe-Kains B and Deschepper CF (2013) Network statistics of genetically-driven gene co-expression modules in mouse crosses. Front. Genet. ***4****:291. doi: 10.3389/fgene.2013.00291*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 03 February 2014
doi: 10.3389/fgene.2014.00012

Utility of network integrity methods in therapeutic target identification

Qian Peng[1,2*†] **and Nicholas J. Schork**[1,2*†]

[1] Department of Molecular and Experimental Medicine, The Scripps Research Institute, La Jolla, CA, USA
[2] Scripps Genomic Medicine, The Scripps Translational Science Institute, La Jolla, CA, USA

Edited by:
Frank Emmert-Streib, Queen's University Belfast, UK

Reviewed by:
Teresa N. De Villavicencio Díaz, Center for Genetic Engineering and Biotechnology, Cuba
Peter Langfelder, University of California, Los Angeles, USA

***Correspondence:**
Qian Peng and Nicholas J. Schork, Department of Molecular and Experimental Medicine, The Scripps Research Institute, 3344 North Torrey Pines Court, Suite 300, La Jolla, CA 92037, USA
e-mail: qpeng@scripps.edu; nschork@scripps.edu

†Present address:
Qian Peng and Nicholas J. Schork, J. Craig Venter Institute, La Jolla, CA, USA

Analysis of the biological gene networks involved in a disease may lead to the identification of therapeutic targets. Such analysis requires exploring network properties, in particular the importance of individual network nodes (i.e., genes). There are many measures that consider the importance of nodes in a network and some may shed light on the biological significance and potential optimality of a gene or set of genes as therapeutic targets. This has been shown to be the case in cancer therapy. A dilemma exists, however, in finding the best therapeutic targets based on network analysis since the optimal targets should be nodes that are highly influential in, but not toxic to, the functioning of the entire network. In addition, cancer therapeutics targeting a single gene often result in relapse since compensatory, feedback and redundancy loops in the network may offset the activity associated with the targeted gene. Thus, multiple genes reflecting parallel functional cascades in a network should be targeted simultaneously, but require the identification of such targets. We propose a methodology that exploits centrality statistics characterizing the importance of nodes within a gene network that is constructed from the gene expression patterns in that network. We consider centrality measures based on both graph theory and spectral graph theory. We also consider the origins of a network topology, and show how different available representations yield different node importance results. We apply our techniques to tumor gene expression data and suggest that the identification of optimal therapeutic targets involving particular genes, pathways and sub-networks based on an analysis of the nodes in that network is possible and can facilitate individualized cancer treatments. The proposed methods also have the potential to identify candidate cancer therapeutic targets that are not thought to be oncogenes but nonetheless play important roles in the functioning of a cancer-related network or pathway.

Keywords: network analysis, centrality, cancer, pathway, drug targets, personalized treatment, gene expression

1. INTRODUCTION

Treating many forms of cancer effectively is notoriously difficult as most tumors have complex cellular dysfunctions replete with compensatory and redundancy mechanisms that contribute to tumor growth despite some aspect of the tumor being targeted for destruction by an anti-cancer therapeutic agent. Thus, while many cancer treatments seem effective when first administered, relapses often occur, particularly in later stages of tumor development. This general "robustness" of biological networks in tumor cells presents true challenges for cancer treatments and cures, especially if treatments administered only target a single gene. To reduce the likelihood of resistance and the risk of relapse, it may be important to target multiple pathways and oncogenes simultaneously, but the best way to do this has not been established (Hughes, 2007; Petrelli and Giordano, 2008; Dar et al., 2012).

While many tumors have certain pathologies and dysfunctional pathways in common, the specific mechanisms contributing to the growth of any one tumor are often distinctive and subtle. However, the identification of these mechanisms and the characterization of their contributions to individual tumor growth and treatment resistance can be greatly aided through the use of modern genomic assays and pathway analyses. Assays such as DNA sequencing, RNA sequencing, copy number variation assays, and proteomic profiling can reveal phenomena such as damaging mutations in oncogenes, resistance gene amplifications, and abnormal silencing of tumor suppressor genes. In conjunction with these assays, network and pathway analyses methods can reveal connections between different perturbations in tumors and may suggest interactions between genes that, if targeted simultaneously with different therapeutic compounds, could disrupt the network integrity of the tumor cells and lead to more effective interventions.

The best way to assess connections between multiple perturbations in tumors that could be targeted simultaneously is an open question. However, analyses of the principal properties, behavior and structures associated with biological networks within tumors may lead to the identification of more optimal therapeutic targets. Of the measures that one could consider in evaluating the properties of a tumor gene network, those focusing on network integrity are of particular interest. Network integrity analysis can lead to the identification of central gene nodes or gene *hubs* within

Network Inference

- Q: What types of biological networks have been inferred in the paper?
- A: We use gene expression data in conjunction with cancer-related signaling pathways to infer tumor–specific networks. The extracted tumor-specific networks help us further infer critical nodes (genes) and potential therapeutic targets for specific types of tumors or tumor cells.
- Q: How was the quality/utility of the inferred networks assessed?
- A: We compare and contrast the predictions with those derived using canonical pathways. We further compare the predictions on various normal tissues, tumor types and tumor cells. We also assess the results using multiple pathway/network databases.
- Q: How were these networks validated?
- A: Many of the targets predicted from the networks have supporting evidences in the literatures: they are either implicated as oncogenes or known targets of cancer treatments.

the network that contribute to the maintenance and growth of a tumor in critical ways (Jeong et al., 2001; Ágoston et al., 2005; Perumal et al., 2009; Horvath, 2011; Li et al., 2011). For example, genes that are critical to the formation and growth of tumors have been observed to code for proteins that have increased levels of *connectedness* with other genes as well as greater *centrality* (i.e., occupying a more central place in the network rather than being on the periphery of the network) than genes that do not contribute to tumor growth and formation (Jonsson and Bates, 2006; Sun and Zhao, 2010; Xia et al., 2011). However, it has also been shown that most disease genes do not necessarily code for proteins that are hubs within a network, suggesting that some network characteristics may be better indicators of optimal therapeutic gene targets than others (Goh et al., 2007). In addition, most network analyses have been performed on comprehensive and generic interaction information rather than on networks or pathways specific to individual tumors, calling into question which type of network topology or representation an analysis should be pursued with. It is noteworthy, however, that network centralities have also been used to derive integrated gene signatures for breast cancer (Wang et al., 2011) and, in the context of signaling pathways, centrality-based analysis approaches have been used to identify enriched pathways from gene expression data (Gu et al., 2012), suggesting that different data types and approaches may provide complementary insights.

We assess the properties and characteristics of a cancer network topology based on gene expression data across a variety of tumors with subsequent analyses confined to specific types of tumors or tumor cells. We contrast the results of the use of different measures of network integrity on the ability to identify therapeutically meaningful gene targets in cancer networks. Our ultimate goal was to determine if it is possible to make compelling claims about the existence of gene targets that might be optimal for therapeutic intervention based on the network characteristics. We rank genes (i.e., nodes in the network) and edges based on their influences on network function and topology defined by various measures, and illustrate that centrality analysis on signaling pathways may provide additional insights to that based on protein-protein interaction (PPI) networks. One of the potential uses of network topology analyses like those we pursued is to identify targets that are not necessarily known to be directly cancer-related but may influence tumor growth nonetheless. Thus, in addition to common measures of network centrality which focus on cancer-related genes, we also investigate the utility of centralities based on spectral graph theory, including spectral gap centrality, that consider network function in a broader context and that have not been explored in the context of biological networks to date.

The remainder of the manuscript is organized as follows. Section 2 describes several centrality measures based on both graph theory and spectral graph theory, as well as the construction of network centralities based on gene expression data. Section 3 contrasts the critical nodes (i.e., genes) and edges defined and determined by different measures in cancer PPI subnetworks and pathways, pathways from different sources, and pathways conditioned on specific tissues and tumor cell lines. Section 4 summarizes the main observations and issues, and makes recommendations. We note that some of the terminology used in the literature and ways of referring to network components are often ambiguous. We use *network* and *pathway* interchangeably, although *network* often corresponds to the actual topology associated with a biological *pathway*. Also, when referring to *nodes* in a network (pathway) we are referring to individual *genes* and their place in the topology associated with a network (pathway).

2. MATERIALS AND METHODS

2.1. CRITICAL NODES IN A NETWORK

Network centralities are important structural attributes of a network. They can be exploited in analyses evaluating network robustness and reflect how much a network is connected and, importantly, how network functionality might be affected locally or globally if certain nodes or connections in the network are disrupted. There are many types of centrality measures (Freeman, 1978/1979; Koschützki and Schreiber, 2008; Horvath, 2011) and they are often used in different contexts. In biological network or pathway analysis, potential drug targets are expected to be highly influential nodes such that perturbing these nodes will have a major effect on network integrity and the flow of information through that network. These nodes might correspond to genes that affect many other genes in the network, or they could be associated with network fragility in the sense that if they are perturbed the network cannot function as a whole. Such highly influential nodes in a network or pathway might also be toxic to the entire network and lead to a complete inability of the network to function if perturbed. Such complete dysfunction might induce more harm than good if it is a network that normal, non-tumor cells require in order to function properly. In this light, it might be better to target nodes or genes that influence the most critical nodes in a network and not the actual critical nodes themselves. Among the various measures of network centrality that have been proposed in the literature, we primarily focused on the four measures described briefly below.

2.1.1. Degree centrality

The simplest and the most common measure of node importance in the context of a specific network topology is degree centrality. Consider a network defined as a simple graph $G = (V, E)$ with $n = |V|$ nodes and $|E|$ edges. The degree of node $v \in V$ is the number of edges incident to v. Mathematically, the graph G can be represented as an adjacency matrix $A(G)$, defined as

$$A_{ij} = \begin{cases} 1 \text{ if } i, j \in V, \{i, j\} \in E, \\ 0 \text{ otherwise,} \end{cases}$$

where $1 \leq i, j \leq n$. Note that in discussions of the adjacency matrix, we will often refer to node v_i as node i and use these two notations interchangeably. The degree centrality of node i is then defined as $c_d(i) = \sum_j a_{ij}$ and reflects how well a node is connected as well as its likely direct influence on its neighbors.

2.1.2. Betweenness centrality

The betweenness centrality is defined as the frequency with which a node is on the shortest path between two other nodes (Freeman, 1978/1979). It reflects the likely *control of communication* between other nodes by the node in question. There are definitional and operational differences between two types of betweenness centrality measures: *node betweenness* and *edge betweenness*. Betweenness for node k is defined as following,

$$c_b(k) = \sum_{i<j} \frac{g_{ikj}}{g_{ij}}$$

where g_{ij} denotes the number of shortest paths between nodes i and j, and g_{ikj} denotes the number of shortest paths between i, j through node k. Betweenness for edge e is similarly defined as,

$$e_b = \sum_{i<j} \frac{g_{iej}}{g_{ij}}$$

where g_{iej} denotes the number of shortest paths between nodes i, j through edge e. In contrast to the local effect of degree centrality, betweenness captures local connectivity as well as a node's global importance to the network. A node or edge of high betweenness essentially serves as a gatekeeper that could control the flow of information across the network.

2.1.3. Eigenvector centrality

The eigenvector centrality is defined as the centrality of a node that is proportional to the sum of the centralities of the nodes it is connected to Bonacich (1972). The eigenvector centrality of node i is

$$c_e(i) = \frac{1}{\lambda} \sum_j a_{ij} c_e(j)$$

where λ is the largest eigenvalue of the adjacency matrix A. It reflects how well a node is connected to the well-connected nodes and how differences in node degrees propagate through a network. Both Google's PageRank measures and Katz centrality are variants of the eigenvector centrality.

2.1.4. Spectral gap centrality

Another measure derived from spectral graph theory was proposed by Wehmuth and Ziviani (2011). As it is based on the spectral gap of sub-networks, we will refer to it as *spectral gap centrality*. The *diagonal degree* matrix of G, denoted $D(G)$, is defined as

$$D_{ij} = \begin{cases} d_k \text{ if } i = j = k, \\ 0 \quad \text{otherwise,} \end{cases}$$

where d_k is the degree of node k. The normalized Laplacian matrix (Chung, 1997) of graph G, denoted $L(G)$, is defined as

$$L_{ij} = \begin{cases} 1 & i = j, \\ -\frac{1}{\sqrt{d_i d_j}} & \{i, j\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

All eigenvalues of $L(G)$ are between 0 and 2, i.e., $0 = \lambda_1(L) \leq \lambda_2(L) \leq \cdots \leq \lambda_n(L) \leq 2$. If G is a single connected component, $\lambda_2(L)$ (referred to as the *spectral gap*) is the smallest non-zero eigenvalue and is less than 1 if the graph is not complete. λ_2 approaches 0 as the graph becomes less connected. The critical nodes are nodes with high spectral gap centrality. The spectral gap centrality of node i is defined as

$$c_s^h(i) = \begin{cases} \frac{\lambda_2^i}{log_2(d_i)} & d_i > 1, \\ \infty & d_i = 1. \end{cases}$$

where λ_2^i is the spectral gap of the h-neighborhood of node i, i.e., the subgraph induced by all nodes within h edges from node i, and d_i is the degree of node i. The lower the value $c_s^h(i)$, the more critical the node i is to the network. The spectral gap centrality thus reflects the neighborhood connectivity, and captures both degree and betweenness to some extent depending on the value of h.

The four centrality measures are chosen primarily for their representative characteristics of networks, their direct relevance to potential biological functions that we are interested in, and the intuitive interpretation of the results. Among other measures that might be of interest, closeness (Sabidussi, 1966) and radiality (Valente and Foreman, 1998) centralities reflect how quickly a node can reach another, which represents a different type of functionality. Closeness centrality requires network to be strongly connected which is often not the case for pathways. PageRank (Page et al., 1999) and Katz status index (Katz, 1953) are variants of eigenvector centrality. Another class of centralities are motif-based (Koschützki and Schreiber, 2008), which represent functional substructures, thus are more likely employed in specific contexts.

2.2. THE ORIGINS AND RELEVANCE OF NETWORK AND PATHWAY TOPOLOGIES

The question of which centrality measure yields a better prediction for therapeutic targets is only one of many important questions associated with biological network analyses. A more fundamental question is which biological network to interrogate. Network analysis can be applied to protein-protein interaction (PPI) networks, often derived empirically through experimentation, or biological pathways that have been described over the years. The choice of a particular pathway is also complicated, since there are multiple versions and subcomponents of pathways to choose from. One option is to derive a protein-protein interaction subnetwork from the genes of relevance to a particular, e.g., phenotype that are grounded in a pathway. An alternative is to analyze the pathway topology directly without considering the elements associated with a protein-protein interaction subnetwork. Different choices of a network or pathway representation—even if chosen to address the same overarching questions—will undoubtedly yield different results due to intrinsic differences between PPI subnetwork definitions and pathways. In addition, the same pathway defined from different database sources, or compiled based on different readings or reviews of the literature, may also yield different results due to topological differences between the network representations. Further compounding these issues is the fact that all genes in a pathway are not equally expressed in all tissues. Thus, networks constructed from one set of resources or experiments may not represent the true network topologies associated with different tissues. For the identification of critical nodes and genes to be relevant to a particular biological setting, tissue-specific network configurations might need to be considered. Obviously, if a gene is not expressed in a particular tissue of interest, for example, the node in another tissue-derived gene expression-based network corresponding to that gene and its associated edges must perforce be deleted from the network, thus altering the network topology.

To evaluate the effects of different network representations and different network centrality measures on the identification of critical nodes in that network, we analyzed MAPK and EGFR signaling pathways and configurations obtained from different sources. We treated these signaling pathway representations as true networks. We obtained pathway information from the KEGG (Kanehisa and Goto, 2000) and WikiPathways (Pico et al., 2008) databases. We obtained a human PPI network from the STRING (Mering et al., 2003) database. In order to have the pathway representations comparable to PPI network representations, we treated them as undirected graphs. Note that the PPI subnetwork from a pathway is a subgraph of the entire PPI network limited to the nodes corresponding to the intersection between genes implicated in the pathway and those present in the PPI network.

To compare and contrast tissue-specific pathways (based on the genes expressed in that tissue) and more generic, non-expression-based pathways, we analyzed cancer-related pathways based on expression patterns obtained from the NCI60 tumor cell lines (Scherf et al., 2000). To determine expression patterns in the NCI60 cell lines, we applied the Gene expression barcode algorithm (McCall et al., 2011) to the Affymetrix gene expression data of each cell line, which yielded an expression state (i.e., expressed/unexpressed) for each gene in each cell line. In addition, we analyzed pathways conditioned on a set of gene expression states and levels obtained from normal tissues. RNA-Seq data for eleven human tissues were obtained from RNA-Seq Atlas (Krupp et al., 2012). A threshold on gene expression value RPKM (reads per kilobase of transcript per million mapped reads (Mortazavi et al., 2008)) was used to filter genes such that genes with expression levels having an RPKM$<$ 0.5 were considered unexpressed. Tissue or cell-specific pathway information was obtained by removing genes (i.e., nodes in the network) corresponding to unexpressed genes from the default pathway.

For each pathway (represented as a network) and each network centrality measure, the nodes (i.e., genes) within them were ranked in two ways: (i) by their centrality values; (ii) by the order that they were removed based on an iterative procedure to identify their importance in the network. This iterative procedure worked by removing top-ranked nodes based on centrality value (along with edges incident to the node), reassessing the nodes in the network and repeating this process until all nodes were assessed.

3. RESULTS

3.1. THE EGFR AND MAPK PATHWAYS IN CANCER

We ultimately analyzed two different pathways known to have pronounced roles in oncogenesis: The epidermal growth factor receptor (EGFR) pathway and the mitogen-activated protein kinase (MAPK) pathway. Both EGFR and MAPK signaling pathways are well-studied and comprehensively curated, making them ideal for our comparison of various methods for assessing node importance and therapeutic target potential. We briefly describe each below.

EGFR, also called ErbB1, is a member of the ErbB family of receptor tyrosine kinases. The EGFR pathway is one of the most important pathways regulating cell growth, differentiation and survival (Holbro and Hynes, 2004). Abnormally high levels of the EGFR protein are frequently found on the surface of many types of cancer cells, facilitating the excessive cell division that is the hallmark of cancer. The defective regulation of the EGFR signal transduction pathway is also known to be associated with oncogenesis. EFGR and its signaling components therefore offer promising therapeutic targets for various cancers (Citri and Yarden, 2006; Scaltriti and Baselga, 2006).

The MAPK superfamily includes well-conserved kinase genes known to be involved in various cellular functions including cell growth, proliferation, differentiation, migration and apoptosis. They are regulated by four distinct groups of genes in mammals: ERK1/2, JNK, p38 and ERK5. While ERK1/2 and ERK5 pathways are relatively insulated, JNK and p38 kineses share many of their activators, thus the two cascades are more entangled (Chen et al., 2001; Yang et al., 2003). It has been well-established that aberrations in MAPK signaling play critical roles in cancer development and progression (Dhillon et al., 2007).

3.2. PPI SUB-NETWORK VERSUS SIGNALING PATHWAY ANALYSES

The EGFR network we derived from WikiPathways [EGFR signaling pathways (Pico et al., 2008; Kandasamy et al., 2010)] has 235 nodes and 249 edges. The average node degree is 1.06, and the graph density (i.e., the fraction of possible edges) is 0.01. The PPI subnetwork induced by EGFR pathway has 119 nodes and 4638 edges. Its average node degree is 39, and the graph density is 0.66. We applied the different centrality measures discussed above to each network and ranked the nodes (genes) on the basis of these measures. **Tables 1**, **2** list the top-ranked genes (ranked between 1 and 10 for at least one measure) obtained from the PPI subnetwork and the EGFR pathway, respectively. The left five columns reflect degree centrality, node betweenness, eigenvector centrality, spectral gap centrality with $h = 2$ and spectral gap centrality with $h = 3$; the right five columns provide corresponding measures with top-ranked nodes removed and the remaining subgraphs re-evaluated iteratively. The upper triangular matrix at the bottom half of the table gives Spearman's rank correlation coefficients assessing the relationship between the results of each pair of metrics. This is computed using the actual rankings of all genes listed in the table, including those ranked beyond ten.

As shown by Spearman's ρ in **Table 1**, the rankings from different metrics are highly correlated among genes in the PPI subnetwork. This can be explained by the properties of the network. The PPI network, like many biological networks (Lima-Mendez and van Helden, 2009), has the following properties: (i) High-degree nodes tend to be connected with other high-degree nodes; (ii) The network diameter (i.e., the length of the longest of the shortest paths between any two nodes) is usually small. A subnetwork shares these properties if it is induced on nodes of high degrees. Genes corresponding to high-degree nodes in a PPI network usually have systemwide effects and are involved in multiple pathways including cancer-related pathways (Han et al., 2004; Barabási et al., 2011). Spectral gap centralities in such a subnetwork are largely dominated by node degrees, and eigenvector and betweenness centralities also track the degrees for the high-degree nodes. Consequently, different centrality metrics on a pathway-induced PPI subnetwork are unlikely to yield significant insights beyond what is already coded in node degrees. As noted, although high-degree nodes in a PPI network may serve as effective drug targets, they are also likely to be toxic if perturbed in severe ways, due to their system-wide influence, i.e., their likely being involved in many cellular functions as they influence many pathways simultaneously. In this light, Wang et al. (2013) showed that the number of side effects of a drug is positively correlated with the degree and betweenness centralities of that drug's targets in the protein-protein interaction network. This observation was found to be the case for both cancer and non-cancer drugs.

In contrast to an analysis of the PPI network, the different centrality measures produced different node rankings when applied to the pathway information, as described in **Table 2**. Thus, while some nodes ranked high in multiple metrics indicating their overall importance, there are groups of nodes that rank high based on one or another measure, especially with respect to the

Table 1 | Top ranking genes in EGFR PPI subnetwork by various centrality measures.

Gene	c_d	c_b	c_e	c_s^2	c_s^3	c_d^r	c_b^r	c_e^r	c_s^{2r}	c_s^{3r}
AKT1	1	1	1	1	1	1	1	1	1	1
EGF	2	3	2	2	2	2	3	2	2	2
EGFR	3	2	3	3	3	3	2	3	3	3
GRB2	4	5	4	4	4	4	5	5	5	4
MAPK1	5	4	5	5	5	5	4	4	4	5
RAC1	6	7	6	6	6	6	8	6	6	6
CDC42	7	6	7	7	7	7	6	8	7	7
MAPK3	8	10	8	8	8	8	10	7	8	8
STAT3	9		9	9	9	9		9	9	9
ERBB2	10	8		10	10	10	7		10	10
FOS		9					9			
PTEN			10					10		
c_d		0.92	0.97	1	1	1	0.9	0.96	0.99	1
c_b			0.87	0.92	0.92	0.92	0.99	0.85	0.93	0.92
c_e				0.97	0.97	0.97	0.83	0.99	0.97	0.97
c_s^2					1	1	0.9	0.96	0.99	1
c_s^3						1	0.9	0.96	0.99	1
c_d^r							0.9	0.96	0.99	1
c_b^r								0.8	0.9	0.9
c_e^r									0.97	0.96
c_s^{r2}										0.99

c_d, degree centrality; c_b, node betweenness; c_e, eigenvector centrality; c_s^2, c_s^3, spectral gap centrality h = 2, 3. $c^r_{\{d,b,e,s\}}, c_s^{\{2,3\}r}$, node ranking are obtained by consecutively removing the top ranked nodes. The bottom matrix is Spearman's rank correlation coefficients.

Table 2 | Top ranking genes in EGFR pathway by various centrality measures.

Gene	c_d	c_b	c_e	c_s^2	c_s^3	c_d^r	c_b^r	c_e^r	c_s^{2r}	c_s^{3r}
SRC	1			2	5	1	2	3	2	5
STAT3	2		1	3	2	2		1		2
EGFR	4		3	1	6	7	7		1	10
HRAS	6	3			8	5	9	4	6	
MAPK1	7	9		4	7	6	6		3	4
MAPK3	8				9	9		2		
GRB2	9	1		7	1	8	1	5	5	1
MAPK7	10					10				8
SOS1		2		6	3					
RAF1		4		5			3		4	
REPS2		5								
ASAP1		6					4			
MAP2K1		7			4					3
MAP2K2		10						6		
STAT1			2							
JAK2			4							
JAK1			5							
PIAS3			6							
COX2			7							
GRIM19			8							
PLCG1				9					8	9
GAB1							8			
PLD1							10			
CBLC								9		
MAPK8								10		
SH3KBP1									7	7
JUN									9	
JUND									10	
c_d		0.34	−0.27	0.76	0.68	0.87	0.53	0.76	0.66	0.69
c_b			−0.35	0.65	0.78	0.3	0.44	0.26	0.4	0.39
c_e				−0.29	−0.25	−0.19	−0.31	−0.24	−0.16	−0.21
c_s^2					0.81	0.53	0.5	0.49	0.71	0.51
c_s^3						0.65	0.44	0.49	0.48	0.7
c_d^r							0.65	0.8	0.48	0.79
c_b^r								0.6	0.54	0.45
c_e^r									0.35	0.5
c_s^{r2}										0.51

Spearman's rank correlation coefficients are computed on genes in the table with their actual rankings (ranking > 10 not shown). Nodes that do not directly correspond to genes are omitted.

betweenness and eigenvector centrality measures. High ranking nodes based on the betweenness and eigenvector centrality measures appear to be exclusive to each other in the pathways we have analyzed. The spectral gap centrality measure tends to capture a few nodes ranked high by each of the other three metrics. Similar results were observed when we analyzed the MAPK pathways, as described in the next section.

We note that genes (nodes) ranked high exclusively by the eigenvector centrality measure (i.e., STAT1, JAK2, JAK1, PIAS3, COX2, GRIM19) are all neighbors (directly downstream or upstream in the pathway) of STAT3, which plays a leading role in cancer inflammation and immunity, and is a validated target for cancer therapy (Yu et al., 2009). JAK-STAT signaling is a well understood cascade as its aberrant activation has been implicated in various types of leukemias, as well as solid tumors (Ferrajoli et al., 2006; Sansone and Bromberg, 2012). In addition, it has been established that STAT1 overexpression is associated with anticancer drug resistance (Khodarev et al., 2012). Interestingly, the FDA-approved drug ruxolitinib is a JAK1 and JAK2 inhibitor, and more JAK inhibitors are in development (Verstovsek et al., 2012). Also, PIAS3 overexpression has been shown to inhibit cell growth and increase drug sensitivity in lung cancer (Ogata et al., 2006), and several studies have indicated that COX2 inhibitors (NSAIDs and celecoxib) have protective effects against colorectal cancers and breast cancers (Gupta and DuBois, 2001; Arun and Goss, 2004; Brown and DuBois, 2005). Finally, Okamoto et al. (2010) demonstrated that overexpression of GRIM19 in cancer cells suppresses STAT3-mediated cancer growth.

As emphasized, the analysis of singular nodes that may be logical drug targets in a network is tremendously important in cancer therapeutic development. However, targeting multiple signaling pathways simultaneously is an essential strategy in managing cancer and reducing the possibility of an individual tumor developing drug resistance. It is therefore important to identify critical genes in multiple cascades within a network. By removing top-ranked nodes that appear to be the most critical for drug response and then re-evaluating the remaining subnetworks, additional critical nodes that may act as redundancy and compensatory mechanisms and contribute to drug resistance can be identified. Interestingly, when the betweenness and spectral gap centralities are applied to a network in such fashion, the first few critical nodes often reside on different paths (cascades) in that network. This phenomenon is not as pronounced for the node degree and eigenvector centrality measures, as their values are affected primarily by a node's nearest neighbours in the network and the properties that these neighboring nodes have. For example, consider c_s^r as applied to the EGFR pathway (**Table 2**): for $h = 2$, the top three nodes are EGFR, SRC, and MAPK1 (ERK), belonging to two paths; for $h = 3$, the top nodes are GRB2, STAT3 and MAP2K1 (MEK), also on two cascades, the classical MAPK and Jak-STAT cascades. We observed similar effects in the analysis of the MAPK pathway as detailed in the next section. In this light, nodes ranked high by c_s^r alone, such as SH3KBP1, PLCG1, JUN, JUND, might also serve as potential therapeutic targets. SH3KBP1 has been implicated in cell death and shown to mediate down regulation of EGFR (Soubeyran et al., 2002; Feng et al., 2011), and JUND has been shown to reduce tumor angiogenesis (Gerald et al., 2004). We consider the betweenness centrality measure in a separate section.

3.3. DIFFERENT REPRESENTATIONS OF THE SAME NETWORK

There are often multiple sources for the same biological network or pathway. Variations in the topology of a network associated with different representations of that network can be attributed to, among other things: what genes or proteins (nodes) are included in the network; what types of interactions are included (e.g., gene-protein, protein-protein, interactions derived from correlations in expressions values of genes) and how they are represented as edges in the network; and how protein complexes are represented. The MAPK pathway can be used to illustrate this. The MAPK pathway from KEGG (Kanehisa and Goto, 2000) has 129 unique nodes and 161 edges (average node degree = 1.25; graph density = 0.02), while the same pathway from WikiPathways (Pico et al., 2008) is made of 186 nodes and 168 edges (average node degree = 0.90; graph density = 0.01). Note that a pathway is not always represented as one connected component. A main difference between the KEGG and WikiPathways representations is that protein-gene complexes are shown as single nodes in the former, while various components of the complex appear individually in the latter, and additional nodes, referred to as compound nodes subsequently, are used to group the complex together. Although the different representations of the MAPK pathway have biological appeal, since they exploit and incorporate different data types and ways of integrating them, the resulting topologies are quite different and obviously affect the ability to identify critical nodes in that pathway. For instance, nodes connecting a complex (i.e., compound nodes) in WikiPathways often have high degrees. Consequently, the significance of the individual nodes in the complex, as well as other nodes, will be affected in the identification of critical nodes. Pathways involving different data sources may be represented as compound graphs for perhaps a clearer layout and to facilitate more modularized modeling (Dogrusoz et al., 2005). However, it is unclear how best to treat differences between pathway representations in a network analysis, especially with respect to what makes the most biological sense, as well as how to interpret the different results. We considered analyses involving both the KEGG and WikiPathways representations to highlight differences that may result from their use.

Tables 3, **4** list critical nodes identified from the MAPK pathway as derived from the KEGG and WikiPathways representations. While some nodes ranked high in one pathway but not the other, most top-ranked nodes are shared. Their rankings, however, are rather different. Of the compound nodes in the MAPK pathway from WikiPathways, CASP*, PPP3* and PRKC* rank high essentially because they are each connected to multiple individual genes (7, 5, 5 genes, respectively) of a complex, thus having relatively high degrees. While compound nodes highly affect the degree centrality ranking, other measures, especially the spectral gap centrality measures, are less affected (unless average node-degree is high, as shown in the analysis of the PPI subnetworks), making them more informative and reliable. In addition, the spectral gap centrality measure, when applied with a higher h, captures nodes with more global rather than local importance. For instance, the top three nodes by $c_s^r(h = 3)$ in the KEGG MAPK pathway representation are RAF1, ASK1 and MEKK1, which are on the ERK1/2, p38, and JNK cascades respectively. Similarly the top three nodes in WikiPathways MAPK pathway representation are ERK, MEKK1 and MKK7, which are on the ERK1/2 and JNK cascades. As shown in the previous section, nodes captured by eigenvector centrality are especially interesting, particularly if they are not captured by other measures, since they are often connected to otherwise critical nodes, thus suggesting that these nodes have the potential of being a direct influence on the behavior of the network. For instance, the MKP (from the DUSPs gene family) and PTP genes are ranked high by c_e alone and ranked 2 and 3 based on an analysis of the KEGG MAPK pathway representation, and as the top two c_e nodes in the WikiPathways representation of the MAPK pathway as well. These genes are known to be inhibitors of ERK, JNK and p38, thus covering three out of four potential cascades or crucial subcomponents of the MAPK pathway. Indeed, PTP genes have emerged as drug targets for cancer (Jiang and Zhang, 2008), and MKP-DUSP genes have been found to be involved in cancer progression and resistance, and have thus also become potential drug targets (Bermudez et al., 2010).

The PPI subnetworks associated with the MAPK pathway, based on both KEGG and WikiPathways representations, share

Table 3 | Top ranking genes in MAPK pathway (KEGG) by various centrality measures.

Gene	c_d	c_b	c_e	c_s^2	c_s^3	c_d^r	c_b^r	c_e^r	c_s^{2r}	c_s^{3r}
MEKK1	1	2	7	4	4	1	4	2	9	3
JNK	2	6	1	3	9	2	2	1	4	4
ASK1	3	7			7	3	3	7		2
Ras	4	4			3	4	8	4		5
ERK	5	5	6	10	5	5	6	6	7	7
Elk1	6		4			10		8		
p38	7	10	5	7	8	6	5	3		6
GRB2	8	9			6	7		5		10
MAPKAPK	9			8					6	
MKK7	10		10	9			10			
MEK2		1		1	2		1		1	
Raf1		3			1					1
SOS		8							10	
MKP			2							
PTP			3							
MKK4			8	2					2	
Sap1a			9							
MKK3				5	10				5	
cJUN				6						
TNFR							7			9
IL1R								9		
TRAF2									3	
TAK1									8	

similar properties with those from EGFR pathways: (i) the average node degrees are high; (ii) the graph densities are roughly 2/3; (iii) the node degrees dominate the critical node rankings of various metrics. As a result, the top-ranked nodes are the usual suspects, such as AKT1, P53, and RAF1. And for brevity's sake, we do not provide detailed descriptions of the results of this analysis.

3.4. THE IMPORTANT ROLE OF BETWEENNESS CENTRALITY IN NETWORK ANALYSES

Since attacking multiple networks and pathways therapeutically in cancer is appropriate and necessary, it is imperative to find the critical signaling and major parallel cascade subnetworks. Betweenness centralities have the potential to reveal gatekeeping nodes or edges that control the flow of signal transduction along the cascades. In addition to node betweenness, edge betweenness may reveal targets that confer distinct functional advantages. It is noteworthy that although many genes/nodes in a network can be linked to multiple functions, it may be the case that only one of such links is disease-related (Zhong et al., 2009). Thus, blocking or perturbing a node with multiple functions may have unanticipated effects. The edge betweenness measure may offer more information than node importance in this regard. In addition, it could identify edges connecting major cascades involved in multiple functions. Since it is known that some cancer-related genes and proteins are difficult to target with small molecules, for example the p53 gene, drugs targeting an edge/interaction for which such genes are connected may offer ways of indirectly targeting and influencing those genes (Arkin and Wells, 2004).

Our analyses involving betweenness centrality with consecutive removal of top-ranked nodes or edges is more revealing. For instance, in **Table 2**, the following nodes with high betweenness, GRB2, SOS1, HRAS, RAF1, MAP2K1, MAP2K2, and MAPK1 are all on the same path. If the top-ranked node is removed and betweenness is re-evaluated on the remaining network, we immediately recognize the critical importance of nodes GRB2 and SRC, which are involved in multiple signaling paths in the network. Similarly, for analyses involving the edge betweenness centrality for the EGFR pathway, while four out of the top five edges by e_b are on the same path, the top five edges by e_b^r are on four distinct paths (**Table 5**).

This phenomenon of nodes and edges gaining or losing importance depending on the measure used is even more pronounced in the analysis of the MAPK pathways (**Tables 3**, **6**). The MAPK pathways include four cascades: classical MAPK pathway (also known as ERK1/2 pathway), JNK and p38 MAPK pathway, and ERK5 pathway. The top three nodes by c_b^r, MEK2, JNK, ASK1, are on three of these cascades (**Table 3**). **Table 6** suggests that while four of the top five edges by e_b are on the same ERK1/2 cascade, the top three edges by e_b^r are each on one cascade: Raf1—MEK2 on ERK1/2, ASK1—MKK2 on p38, MKK—JNK on JNK, while the fourth edge MEKK1—MEK2 connects the JNK and ERK1/2 paths (see Xu et al., 1995 for details on this link). Thus, these nodes and edges do indeed essentially capture the main paths in the MAPK network. Note that the ERK5 cascade is presented as a separate component and the subgraph is a linear graph. Consequently, none of its nodes or edges ranked high in this particular analysis.

Table 4 | Top ranking genes in MAPK pathway (WikiPathways) by various centrality measures.

Gene	c_d	c_b	c_e	c_s^2	c_s^3	c_d^r	c_b^r	c_e^r	c_s^{2r}	c_s^{3r}
CASP*	1	7		2	8	1		1		
DUSP*(MKP)	2		2		9	2		4		
TGFBR1/2	3			10		3	10	7	5	
IL1R1/2	4	1		1	4		1	5	1	5
PPP3*	5					4	9	8		7
K/N/MRAS	6		6			5	3	6	9	
PRKC*(PKC)	7					6				
GRB2	8					7	7	10	7	
MAP3K1(MEKK1)	9	2		4	2	8	4	3		2
MAPK8-10(JNK)	10		4	7		9				
TRAF2		3		6	3					
MAP3K7(MKK7)		4			6					3
MAP3K7IP1		5								
RAC1/2,CDC42		8		5	10				3	
PTPN5/7,PTPRR(PTP)		9	1	3	5		2	2	2	6
MAP2K4(MKK4)		10							10	
MAPK1/3/4/6(ERK)			3		1		5			1
PTPN5			5	8						
MAPK12-14(p38)			8			10	8			
MAPK13			9							
PTPN7			10							
NRAS					7					4
IKBKB/G,MAP3K14								9		10
MAPK1									8	
KRAS										8
MAPK10										9

Nodes that do not correspond to genes directly are omitted.

Table 5 | High betweenness edges in EGFR pathway.

Rank	e_b (Edge)	Path	e_b^r (Edge)	Path
1	GRB2–SOS1	Classical MAPK	GRB2–SOS1	Classical MAPK
2	SOS1–HRAS	Classical MAPK	SRC–PLCG1	Calcium
3	GRB2–REPS2		SRC–GAB2	SRC/GAB2/PI3K/AKT (Phagocytosis)
4	HRAS–RAF1	Classical MAPK	RAF1–MAP2K1	Classical MAPK
5	RAF1–MAP2K1	Classical MAPK	ASAP1–ARF6	PAG3/ARF6 (Phagocytosis)

3.5. PATHWAY ANALYSES CONDITIONED ON EXPRESSED GENES IN TISSUES AND TUMOR CELLS

Not all genes are expressed in all tissues and cells. In tumor cells, certain genes are amplified, others silenced, often abnormally so. Not only do tumor cells differ from normal cells in this regard, but they also differ from each other. As such, the same pathway manifests differently in different cell types: if a gene is unexpressed, the encoded protein should be considered non-functional, and should be factually deleted from the pathway for an analysis. While analyzing the default pathway topology yields invaluable insights, tissue or cell-specific pathway topology needs be considered for network analysis to be more relevant. The best way to construct appropriate networks for cell or tissue-specific analyses is an open question, but might be achieved best by constructing them de novo from relevant experimental data (Ranola et al., 2013).

3.5.1. EGFR pathway restricted by gene expression levels in the NCI60 cell lines

There are sixty unique cell lines of nine tumor types in NCI60 database. We applied the gene expression barcode algorithm (McCall et al., 2011) to the microarray gene expression data of NCI60 cell lines to filter out unexpressed genes. The gene expression barcode is essentially a normalization method leveraging microarray data in the public domain to answer the question: "given an individual microarray experiment of a cell type, is a gene expressed or unexpressed in that cell?" Unexpressed genes are deleted from the default pathway. For each NCI60 cell

Table 6 | High betweenness edges in MAPK pathway (KEGG) .

Rank	e_b (Edge)	Path	e_b^r (Edge)	Path
1	Raf1–MEK2	Classical MAPK	Raf1–MEK2	Classical MAPK
2	Ras–Raf1	Classical MAPK	ASK1–MKK3	p38 MAPK
3	MEKK1–MEK2		MKK4–JNK	JNK MAPK
4	MEK2–ERK	Classical MAPK	MEKK1–MEK2	
5	SOS–Ras	Classical MAPK	MKK4–MKK7	JNK MAPK

line, between 40% and 60% of the 235 nodes in the default network made from EGFR pathway were removed after this simple analysis. We then evaluated the importance of nodes or edges in each individual topology.

Figure 1A shows gene rankings for the spectral gap centrality measure $c_s^r(h = 2)$ averaged over cell lines for each tumor type, where the top row labeled as *def* provides the rankings in the default pathway. While all tumor types are different from each other, the patterns of genes expressed and unexpressed in them suggest that a network derived from these genes would be very different from the default pathway. Essentially, each individual cell line presents unique gene expression patterns as well. This diversity requires pathway analyses specific for each individual tumor. **Figures 1B,C** show the gene rankings for individual melanoma and breast cancer cell lines respectively. **Figures 2**, **3** show top-ranked nodes by eigenvector centrality and top-ranked edges by betweenness for each tumor type as well as the individual melanoma and breast cancer cell lines.

Melanoma cell lines can be clustered into three groups by $c_s^r(h = 2)$ top-ranked genes, one with EGFR and RAF1 ranked high exclusively, the other with MAPK1 at top rank, and the third with a mixture of EGFR/RAF1/MAPK1 (**Figure 1B**). Eigenvector centrality c_e clusters melanoma cell lines quite differently from $c_s^r(h = 2)$ (**Figure 2B**). Among breast cancer cell lines, MCF7 is unique when the measure $c_s^r(h = 2)$ is used to assess the EGFR pathway with GRB2 and MAPK1 ranking highest, while all the other cell lines have EGFR and/or RAF1 as top ranking genes (**Figure 1C**). T47D appears unique by both c_e and e_b (**Figures 2C**, **3C**). Eigenvector centrality c_e yields unique sets of genes for each cell line except for genes STAT1/3 and CBL, each shared by two cell lines as the top candidates (**Figure 2C**). CBL protein family has been implicated in a number of human cancers and indeed shown to enhance breast tumor formation by inhibiting tumor suppressive activity of TGF-β signaling (Kang et al., 2012). The application of the edge betweenness measure again clusters melanoma and breast cancer cell lines into two to three groups, but in different ways than those derived with other metrics (**Figures 3B,C**).

3.5.2. Analysis of the EGFR pathway restricted to eleven normal tissues

The RNA-Seq Atlas (Krupp et al., 2012) has RNA-Seq data for eleven normal human tissues. Hebenstreit et al. (2011) suggests that there are two major classes of gene expression levels in most cells: lowly expressed, which are likely non-functional, and highly expressed, which are likely to be biologically meaningful. The distribution of log_2(RPKM) gene expression values across the eleven human tissues is indeed bimodal, suggesting these two major classes. Determining a simple threshold for defining unexpressed genes, however, is still somewhat arbitrary. We considered a 0.5 RPKM value as a threshold for differentiating unexpressed vs. expressed gene, which is not only often suggested as a conservative threshold, but also seems reasonable in this dataset. **Figures 4A,C** show the top ranked genes by spectral gap centrality $c_s^r(h = 2)$ and top edges by the betweenness measure e_b^r for each tissue. With the exception of liver, and to a lesser extent skeletal muscle, the rankings of the most critical genes in the other nine tissues are quite similar to those from the default pathway, and even more similar to each other. The critical edges in tissues differ from those from the default pathway, but they are very similar to each other with the exception of those of skeletal muscle. Even though the data set cannot be compared directly to the NCI tumor cell lines for purely technical reasons, the general patterns of node importance are markedly different (see **Figures 1**, **3**). With the use of a threshold of 0.5 RPKM, around a quarter nodes are deleted from the default EGFR pathway. To make the number of nodes more comparable to the tumor cell lines, we set a more aggressive threshold of 3 RPMK so that between 40% and 60% nodes are filtered out. **Figure 4B** shows the result for $c_s^r(h = 2)$ (edge betweenness e_b^r is omitted due to space limitation). Even though there are considerable differences and variations, they are still less varied than the tumor cell types (**Figures 1A**, **2A**, **3A**). We note that expression patterns of a tissue could be the averaged expressions over different cell types within the tissue.

3.5.3. Integrated breast cancer pathway restricted by NCI60 breast tumor cells

The Integrated Breast Cancer pathway incorporates the most important proteins for breast cancer. It has 190 unique nodes and 348 edges (mean node degree = 1.83; graph density = 0.02). **Figure 5** shows the top ranking nodes and edges by different measures for each NCI60 breast cancer cell line.

While BRCA1 ranked highest by c_d (not shown), $c_s^r(h = 2, 3)$ ($h = 3$ not shown) for cell lines MCF7, MDA_MB_231 and BT_549, MAX (Myc associated factor X) ranked highest by c_d and $c_s^r(h = 3)$ for HS578T and T47D. It is known that MYC deregulation contributes to breast cancer development and progression. Loss of BRCA1 coupled with MYC overexpression leads to the development of breast cancer (Xu et al., 2010) and recent evidence has shown that MYC is druggable (Pourdehnad et al., 2013).

Smad2 ranked high by at least one measure for each cell line. Smad genes are highly ranked by c_e for all but MCF7 and BT_549, for which STAT1 and AR emerged more important (in addition to BRCA1). Although Smad2/3/4 signaling plays a

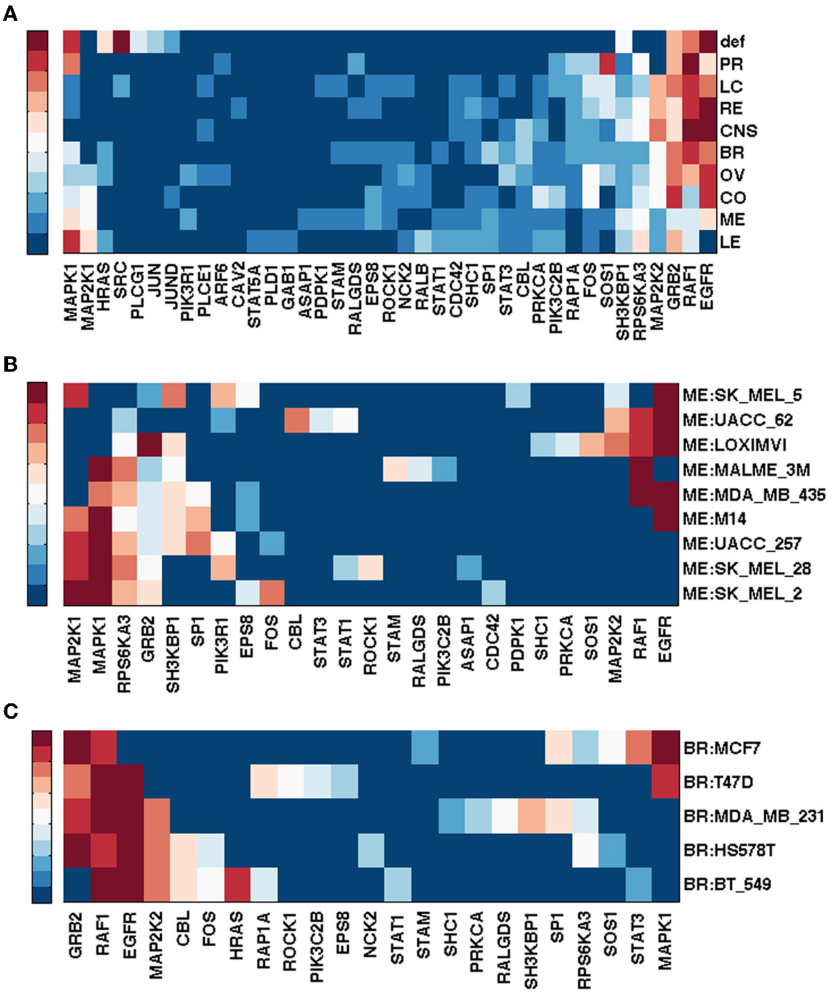


FIGURE 1 | Top ranked genes by spectral gap centrality with node removal $c_s^r(h = 2)$ of EGFR pathway conditioned on NCI60 cell line gene expression. Ranks range from 1 (dark red) to 10 (blue), and > 10 (the darkest blue). **(A)** Rankings are averaged for each tumor type: BR, breast; CNS, central nervous system; CO, colon; LC, non-small cell lung; LE, leukemia; ME, melanoma; OV, ovarian; PR, prostate; RE, renal. def, default pathway with all nodes. **(B)** Gene ranking by $c_s^r(h = 2)$ for NCI60 melanoma cell lines. **(C)** Gene ranking $c_s^r(h = 2)$ for NCI60 breast cancer cell lines.

tumor suppressor role, it also exhibits a pro-metastatic function in breast cancer (Kang et al., 2005). It is also believed that Smad-dependent pathway is involved in TGF-β tumor suppressor functions. Various TGF-β inhibitors are in development and preclinical studies have shown their promises in cancer treatments (Nagaraj and Datta, 2010).

Evidence has correlated up-regulation of STAT1 activity with increased breast tumor progression and immune suppression in tumor microenvironment, thus STAT1 inhibition is a promising immune therapeutic target (Hix et al., 2013). Androgen receptor (AR) is commonly expressed in breast cancers. It ranked high by c_e for cell lines MCF7 and BT-549. There is a history of targeting AR for therapy in breast cancer, although the efficacy of AR targeted treatments is moderate (Garay and Park, 2012) probably due to a lack of clear understanding of the AR signaling mechanism. For MCF7 cell line though, inhibitory effects of androgens targeting AR have indeed been shown in multiple studies (Greeve et al., 2004; Macedo et al., 2006).

Notice that since the analysis of the breast cancer pathway is conditioned on the gene expression patterns in each cell line, major tumor suppressor genes such as P53 and BRCA2 are deleted. The exome data of NCI60 (Abaan et al., 2013) cell lines shows that each of the five breast cancer cell lines has between one to four missense or silencing TP53 mutations, and two to five missense or silencing mutations in BRCA2. Only MDA_MB_231 has a silencing BRCA1 mutation. If we analyze the default breast cancer pathway instead of the pathways built only from genes expressed in the cell lines, the top three gene nodes are P53, AKT1 and BRCA1 based on the c_d or c_e measures, or CERK1, SMAD2 and AKT1 by the $c_s^r(h = 2)$ measure, respectively. The top two ranked edges based on the betweenness measure (with edge removal) are the TGFR1-SMAD4 and P53-C9JNK1 edges.

4. DISCUSSION

The identification of genes that are optimal or logical therapeutic targets in tumors based on genomic information is crucial for

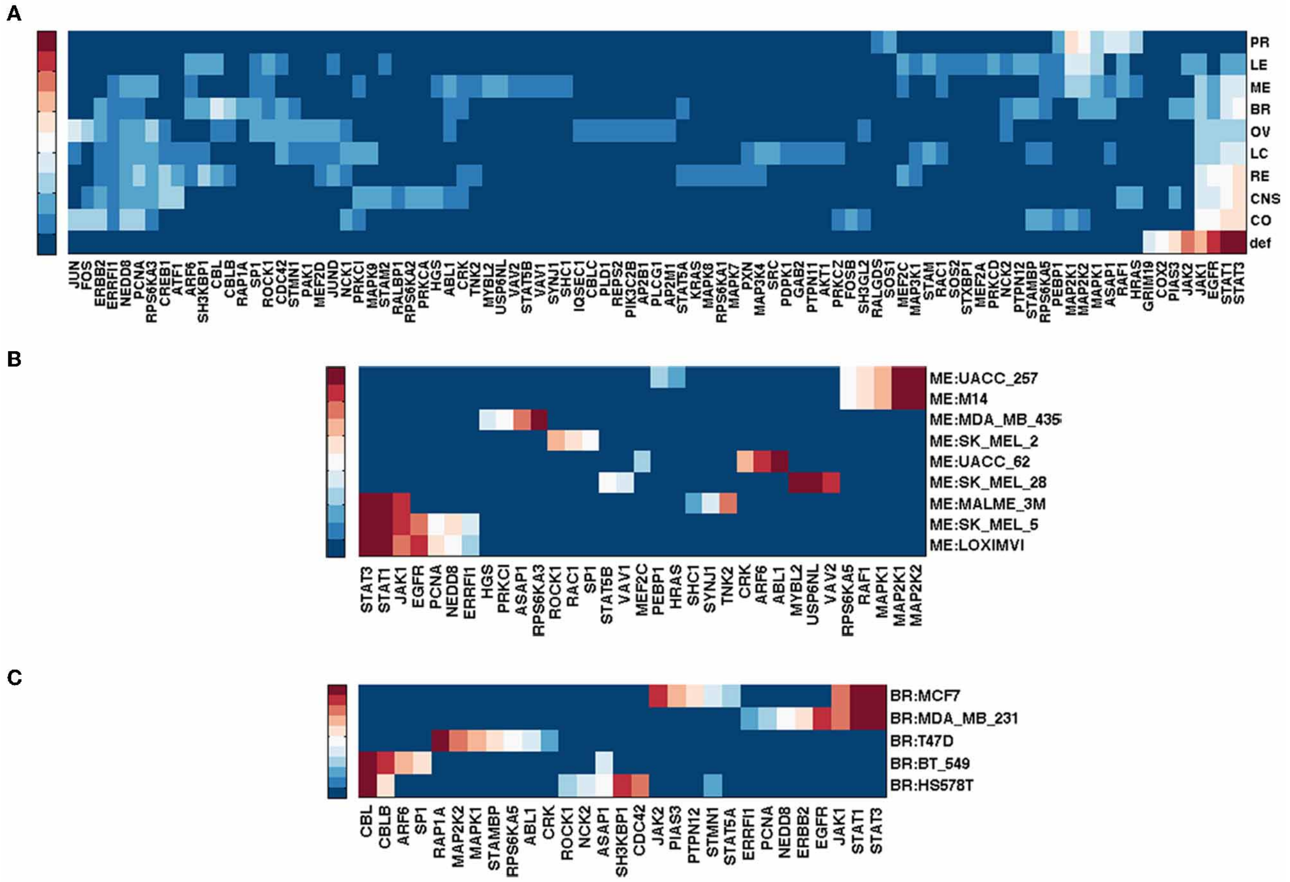


FIGURE 2 | Top ranked genes by eigenvector centrality c_e of EGFR pathway conditioned on NCI60 cell line gene expression. Ranks range from 1 (dark red) to 10 (blue), and > 10 (the darkest blue). **(A)** Gene ranking by c_e for tumor types. **(B)** Gene ranking c_e for NCI60 melanoma cell lines. **(C)** Gene ranking c_e for NCI60 breast cancer cell lines.

individualizing cancer treatments. We explored the utility of network centrality analysis of standard pathways and pathways based on gene expression information in identifying potential therapeutic targets for a tumor. We also described the complexity of, and issues associated with, such analysis. We considered ranking genes in a network or pathway either by their centrality values or by iteratively recording the top-ranked node and reevaluating the remaining subnetwork with the highest ranked node removed. When analyses are performed on PPI subnetwork created from genes associated with a specific pathway, the top ranked genes based on different node importance measures are highly positively correlated. We observed a similar phenomenon when PPI subnetworks derived from genes that have been implicated in particular types of cancers were assessed, both when using the genes in these PPI subnetworks alone and by expanding these subnetworks by including nodes one or two edges from the seed genes used to create the PPI subnetwork (data not shown). The high-degree nodes in a PPI network are critical to the functioning of that network, and thus are likely to be important drug targets. However, such nodes are not likely to be specific to a particular pathway and as such targeting them therapeutically could also be potentially toxic to a patient.

When applied to a signaling pathway, various measures of centrality yield different sets of important genes and the rankings of these genes across different node importance measures are much less correlated. This lack of correlation among node importance measures may provide more insight into the functioning of a network or pathway since the different measures may be capturing different aspects of information flow through the network. However, a possible confounding factor in the analysis of node importance in networks is that the same pathway may be represented in different ways across different databases, leading to different network topologies. It is unclear how to determine which topology is the best representation of a pathway in such cases.

In the context of different measures of node importance, eigenvector centrality has the potential to reveal nodes that may impact other highly influential nodes (for instance nodes of high degree). These other nodes may reflect genes that could serve as alternative therapeutic targets when the highest ranked nodes or genes are hard to target or possibly be toxic to the system as a whole if targeted therapeutically directly. Identifying these alternative important nodes using eigenvector centrality should be done on the pathway without iteratively deleting nodes or those

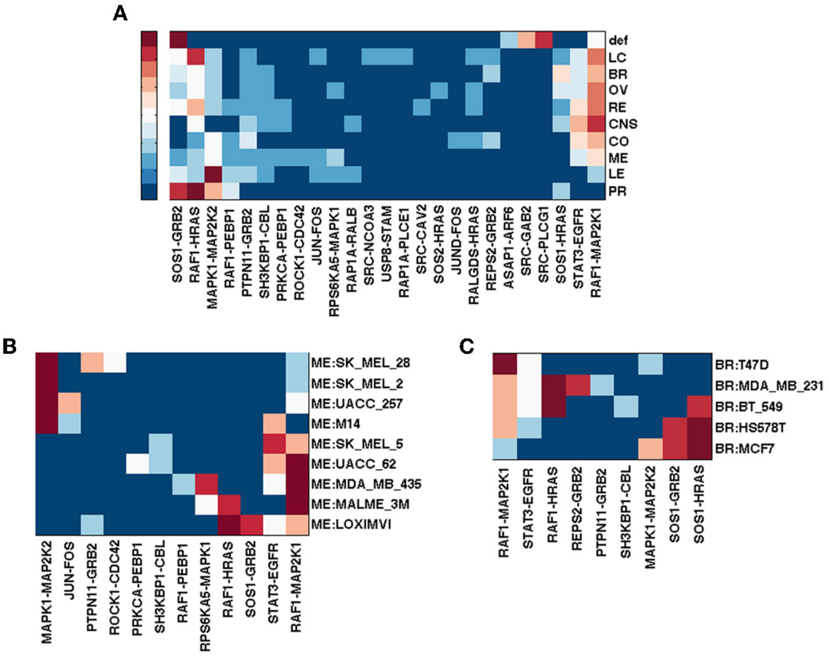


FIGURE 3 | Top ranked edges by betweenness with edge removal e_b^r of EGFR pathway conditioned on NCI60 cell line gene expression. Ranks range from 1 (dark red) to 5 (blue), and > 5 (the darkest blue). **(A)** Edge ranking by e_b^r for tumor types. **(B)** Edge ranking by e_b^r for NCI60 melanoma cell lines. **(C)** Edge ranking by e_b^r for NCI60 breast cancer cell lines.

alternative nodes are not likely to be discovered. For instance, while SRC, STAT3, EGFR and GRB2 ranked the highest in the EGFR pathway by two or more measures, STAT1, JAK1/2, PIAS3, COX2 and GRIM19, all being neighbors of STAT3, ranked within top ten exclusively by the eigenvector centrality. Each of these genes has been implicated in some type of cancers and some are known targets of cancer treatments. As mentioned in Section 3.2, Ruxolitinib, an FDA-approved drug for treatment of a type of bone marrow cancer, is a JAK1/2 inhibitor (Mesa, 2010). NSAIDs and Celecoxib are COX2 inhibitors and have protective effects against colorectal and breast cancers (Gupta and DuBois, 2001; Arun and Goss, 2004; Brown and DuBois, 2005). In addition, Hide et al. (2011) showed that the combination of a PTGS2 (COX2) inhibitor and an EGFR inhibitor prevented tumorgenesis of oligodendrocyte lineage-derived glioma-initiating cells. Finally, Li et al. (2013) demonstrated that microRNA-26b might act as a tumor suppressor in breast cancer by targeting PTGS2.

Nodes ranked high by the betweenness measure with iterative node removal are often on parallel cascades in the pathway, which are important for simultaneously targeting multiple pathways in cancer treatment. The top three nodes identified in this fashion in MAPK pathways, for example, are MEK2, JNK and ASK1, which reside on ERK1/2, JNK and p38 cascades respectively. Edge betweenness generates potential edge-specific, or edgetic targets, which are more specific to a particular pathway and the nodes implicated in these edges might provide an alternative for therapeutic targeting if the highest ranked individual nodes are hard to target. Similarly, edges identified as important by iterative edge removal tend to reside on separate paths.

Although high degree nodes are very important to the functioning of a network, they are also more prone to differ if local changes in a network topology are made. The spectral gap centrality measure, on the other hand, is less sensitive to local degree changes, and is more reliable if slightly different network topologies are considered. The spectral gap centrality measure also captures both degree and betweenness phenomena simultaneously, thus complementing betweenness measures when used in isolation in an important way. This is particularly true in the context of signaling pathways where the betweenness measures tend to capture fragile nodes and edges. The choice of the parameter h in the spectral gap centrality measure calculation is more complicated and is likely best approached empirically. Smaller values of h tend to capture local node importance while larger values of h tend to capture more global node importance. For typical pathways and PPI networks, setting $h = 2$ or 3 is a reasonable choice. The spectral gap centrality measure node rankings are also more informative when computed with iterative node removal.

Ultimately, in the context of finding potential therapeutic targets for tumors, we firmly believe that network analysis should consider cell or tissue specific pathways and networks and not rely on generalized or tissue independent canonical pathways and networks. In order to assess tissue-specific networks and pathways, we considered the use of the expression levels of genes in tissues to filter out unexpressed genes. We did this by using either gene expression barcodes based on available array data or a RPKM threshold based on RNA-Seq data. When different measures of node importance are applied to tissue or cell-specific pathways obtained in this way, the resulting top-ranked genes

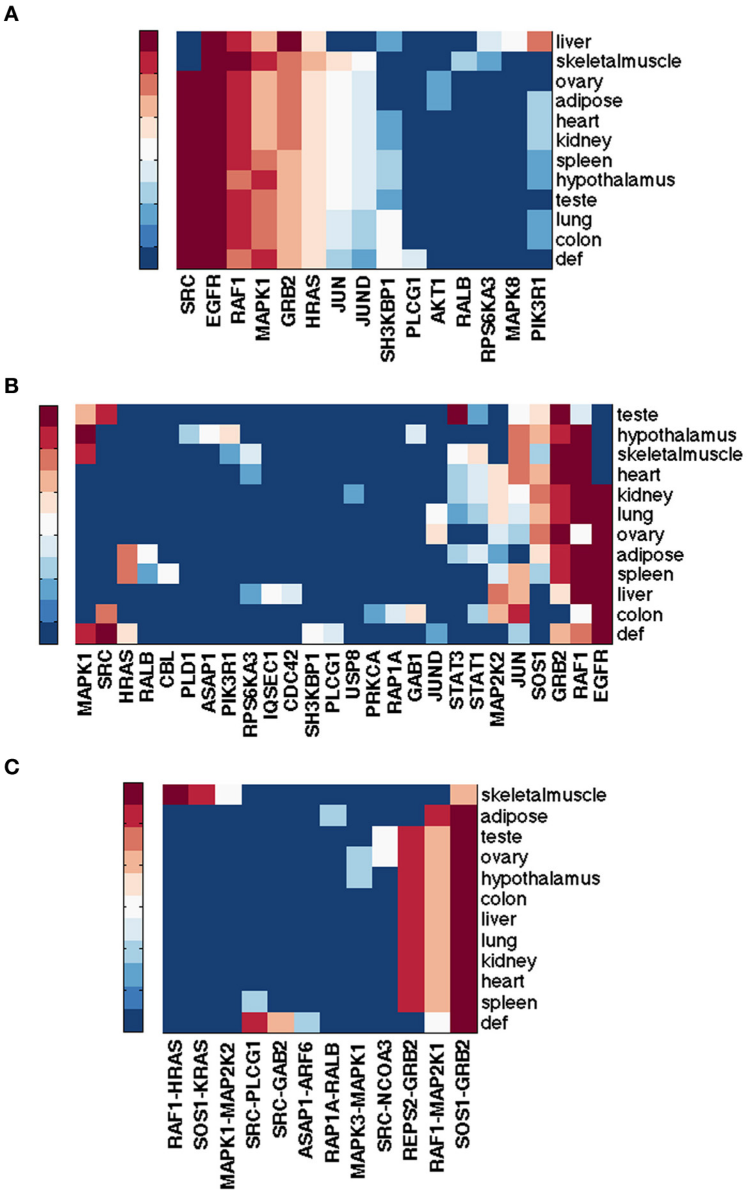


FIGURE 4 | Top ranked nodes of EGFR pathway conditioned on eleven normal human tissues from RNA-Seq Atlas. Ranks range from 1 (dark red) to 10 (blue), and > 10 (the darkest blue) for **(A,B)**; from 1 (dark red) to 5 (blue), and > 5 (the darkest blue) for **(C)**. **(A)** Node ranking by spectral gap centrality $c_s^r(h = 2)$ with RPKM ≥ 0.5. **(B)** Node ranking by spectral gap centrality $c_s^r(h = 2)$ with RPKM ≥ 3.0. **(C)** Edge ranking by betweenness e_b^r with RPKM ≥ 0.5

varied significantly among different cell types. We found that variations in node importance between different tumor types are generally larger than those variations between different normal tissues. This is to be expected given the complex rearrangements and perturbations in tumors. For a particular tumor type, analysis of different tumor cell lines or subtypes results in different nodes deemed crucial or important to a particular pathway. For instance, when the integrated breast cancer pathway is restricted by the five NCI60 breast tumor cell lines based on their respective gene expressions, BRCA1 ranked highest by degree and spectral gap centralities for cell lines MCF7, MDA_MB_231 and BT_549, while MAX ranked highest by the same measures for cell lines HS578T and T47D. SMAD2 ranked high by at least one centrality measure for each of the five cell lines. While SMAD genes were highly ranked by eigenvector centrality for MDA_MB_231, HS578T and T47D, STAT1 and AR appeared more important for MCF7 and BT_549.

We recognize that there are limitations and caveats in our analyses. As more and more RNA sequencing studies are being pursued on tumors, a simple threshold used to differentiate expressed and unexpressed genes in these tumors will be harder to define. Thus, better methods need be explored to determine

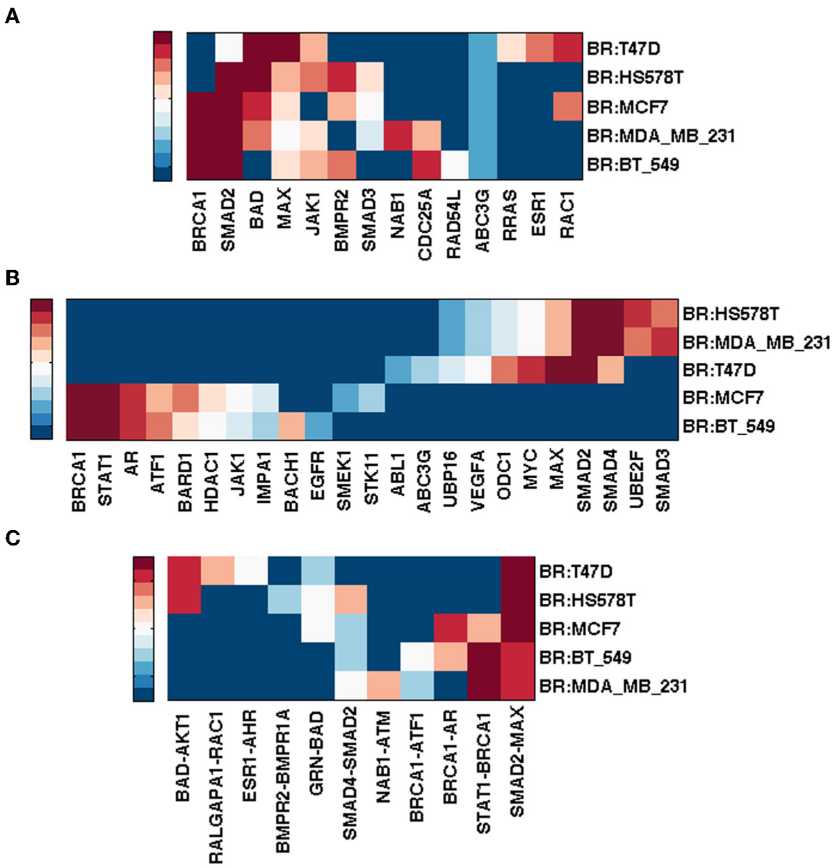


FIGURE 5 | Top ranked nodes and edges of integrated breast cancer pathway conditioned on NCI60 breast cancer cell lines. Ranks range from 1 (dark red) to 10 (blue), and > 10 (the darkest blue) for **(A,B)**; from 1 (dark red) to 5 (blue), and > 5 (the darkest blue) for **(C)**. **(A)** Node ranking by spectral gap centrality $c_s^r(h = 2)$. **(B)** Node ranking by eigenvector centrality c_e. **(C)** Edge ranking by betweenness e_b^r.

which genes might need to be filtered out or included in a pathway analysis. While capturing important relevant oncogenes or genes impacting oncogenes in a pathway, filtering genes based on whether they are expressed or unexpressed in a cell type naturally filters out abnormally silenced genes, thus potentially excluding malfunctioned tumor suppressor genes in analysis, such as the p53 gene. This can be salvaged by analyzing the default pathway to some extent. In this light, given the extremely complex nature of cancers, finding critical genes in specific pathways is just a tiny piece of a puzzle to determine how best to treat cancers. Not only will an analysis of critical nodes in a network need be approached with caution, but it should also be used in conjunction with other information, such as the analysis of DNA sequence mutations, copy number variations and other bio-markers. In addition, treating gene expression as a binary factor to construct a network's topology for use in an analysis of node importance is admittedly a simplistic approach. Rather, expression levels and rates of gene amplifications can also be incorporated into network analysis. Also, in addition to analyzing tumor cells alone, it will likely be more informative to compare normal and tumor samples to better quantify tumor-specific genomic perturbations. Ultimately, we believe our analyses shed light on the utility of measures of node and edge importance in an analysis of gene networks and pathways in tumor biology and cancer treatment choice and hope that they may motivate further research in this area.

ACKNOWLEDGMENTS

FUNDING

Nicholas J. Schork and Qian Peng are supported in part by NIH grants 5 UL1 RR025774, R21 AI085374, 5 U01 DA024417, 5 R01 HL089655, 5 R01 DA030976, 5 R01 AG035020, 1 R01 MH093500, 2 U19 AI063603, 2 U19 AG023122, 5 P01 AG027734, 1 R21 DA033813 as well as grants from Johnson and Johnson, the Veteran's Administration, the Viterbi Foundation, the Stand-Up-to-Cancer organization and the Price Foundation.

REFERENCES

Abaan, O. D., Polley, E. C., Davis, S. R., Zhu, Y. J., Bilke, S., Walker, R. L., et al. (2013). The exomes of the nci-60 panel: a genomic resource for cancer biology and systems pharmacology. *Cancer Res.* 73, 4372–4382. doi: 10.1158/0008-5472.CAN-12-3342

Ágoston, V., Csermely, P., and Pongor, S. (2005). Multiple weak hits confuse complex systems: a transcriptional regulatory network as an example. *Phys. Rev. E* 71:051909. doi: 10.1103/PhysRevE.71.051909

Arkin, M. R., and Wells, J. A. (2004). Small-molecule inhibitors of protein-protein interactions: progressing towards the dream. *Nat. Rev. Drug Discov.* 3, 301–317. doi: 10.1038/nrd1343

Arun, B., and Goss, P. (2004). The role of cox-2 inhibition in breast cancer treatment and prevention. *Semin Oncol.* 31(2 Suppl. 7), 22–29. doi: 10.1053/j.seminoncol.2004.03.042

Barabási, A.-L., Gulbahce, N., and Loscalzo, J. (2011). Network medicine: a network-based approach to human disease. *Nat. Rev. Genet.* 12, 56–68. doi: 10.1038/nrg2918

Bermudez, O., Pagès, G., and Gimond, C. (2010). The dual-specificity MAP kinase phosphatases: critical roles in development and cancer. *Am. J. Physiol. Cell Physiol.* 299, C189–C202. doi: 10.1152/ajpcell.00347.2009

Bonacich, P. (1972). Factoring and weighting approaches to clique identification. *J. Math. Sociol.* 2, 113–120. doi: 10.1080/0022250X.1972.9989806

Brown, J. R., and DuBois, R. N. (2005). Cox-2: a molecular target for colorectal cancer prevention. *J. Clin. Oncol.* 23, 2840–2855. doi: 10.1200/JCO.2005.09.051

Chen, Z., Gibson, T. B., Robinson, F., Silvestro, L., Pearson, G., Xu, B.-E., et al. (2001). MAP kinases. *Chem. Rev.* 101, 2449–2476. doi: 10.1021/cr000241p

Chung, F. R. (1997). *Spectral Graph Theory*, Vol. 92. Providence, RI: AMS Bookstore.

Citri, A., and Yarden, Y. (2006). EGF-ERBB signalling: towards the systems level. *Nat. Rev. Mol. Cell. Biol.* 7, 505–516. doi: 10.1038/nrm1962

Dar, A. C., Das, T. K., Shokat, K. M., and Cagan, R. L. (2012). Chemical genetic discovery of targets and anti-targets for cancer polypharmacology. *Nature* 486, 80–84. doi: 10.1038/nature11127

Dhillon, A. S., Hagan, S., Rath, O., and Kolch, W. (2007). MAP kinase signalling pathways in cancer. *Oncogene* 26, 3279–3290. doi: 10.1038/sj.onc.1210421

Dogrusoz, U., Giral, E., Cetintas, A., Civril, A., and Demir, E. (2005). "A compound graph layout algorithm for biological pathways," in *Graph Drawing*, Vol. 3383 of *Lecture Notes in Computer Science*, (New York, NY: Springer Berlin Heidelberg), 442–447.

Feng, L., Wang, J.-T., Jin, H., Qian, K., and Geng, J.-G. (2011). Sh3kbp1-binding protein 1 prevents epidermal growth factor receptor degradation by the interruption of c-cbl-cin85 complex. *Cell Biochem. Funct.* 29, 589–596. doi: 10.1002/cbf.1792

Ferrajoli, A., Faderl, S., Ravandi, F., and Estrov, Z. (2006). The JAK-STAT pathway: a therapeutic target in hematological malignancies. *Curr. Cancer Drug Targets* 6, 671–679. doi: 10.2174/156800906779010227

Freeman, L. C. (1978/79). Centrality of social networks conceptual clarification. *Soc. Netw.* 1, 215–239. doi: 10.1016/0378-8733(78)90021-7

Garay, J. P., and Park, B. H. (2012). Androgen receptor as a targeted therapy for breast cancer. *Am. J. Cancer Res.* 2, 434–445.

Gerald, D., Berra, E., Frapart, Y. M., Chan, D. A., Giaccia, A. J., Mansuy, D., et al. (2004). Jund reduces tumor angiogenesis by protecting cells from oxidative stress. *Cell* 118, 781–794. doi: 10.1016/j.cell.2004.08.025

Goh, K.-I., Cusick, M. E., Valle, D., Childs, B., Vidal, M., and Barabási, A.-L. (2007). The human disease network. *Proc. Natl. Acad. Sci. U.S.A.* 104, 8685–8690. doi: 10.1073/pnas.0701361104

Greeve, M., Allan, R., Harvey, J., and Bentel, J. (2004). Inhibition of mcf-7 breast cancer cell proliferation by 5alpha-dihydrotestosterone; a role for p21(cip1/waf1). *J. Mol. Endocrinol.* 32, 793–810. doi: 10.1677/jme.0.0320793

Gu, Z., Liu, J., Cao, K., Zhang, J., and Wang, J. (2012). Centrality-based pathway enrichment: a systematic approach for finding significant pathways dominated by key genes. *BMC Syst. Biol.* 6:56. doi: 10.1186/1752-0509-6-56

Gupta, R. A., and DuBois, R. N. (2001). Colorectal cancer prevention and treatment by inhibition of cyclooxygenase-2. *Nat. Rev. Cancer* 1, 11–21. doi: 10.1038/35094017

Han, J.-D. J., Bertin, N., Hao, T., Goldberg, D. S., Berriz, G. F., Zhang, L. V., et al. (2004). Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature* 430, 88–93. doi: 10.1038/nature02555

Hebenstreit, D., Fang, M., Gu, M., Charoensawan, V., van Oudenaarden, A., and Teichmann, S. A. (2011). Rna sequencing reveals two major classes of gene expression levels in metazoan cells. *Mol. Syst. Biol.* 7:497. doi: 10.1038/msb.2011.28

Hide, T., Takezaki, T., Nakatani, Y., Nakamura, H., Kuratsu, J.-I., and Kondo, T. (2011). Combination of a ptgs2 inhibitor and an epidermal growth factor receptor-signaling inhibitor prevents tumorigenesis of oligodendrocyte lineage-derived glioma-initiating cells. *Stem Cells* 29, 590–599. doi: 10.1002/stem.618

Hix, L. M., Karavitis, J., Khan, M. W., Shi, Y. H., Khazaie, K., and Zhang, M. (2013). Tumor stat1 transcription factor activity enhances breast tumor growth and immune suppression mediated by myeloid-derived suppressor cells. *J. Biol. Chem.* 288, 11676–11688. doi: 10.1074/jbc.M112.441402

Holbro, T., and Hynes, N. E. (2004). Erbb receptors: directing key signaling networks throughout life. *Annu. Rev. Pharmacol. Toxicol.* 44, 195–217. doi: 10.1146/annurev.pharmtox.44.101802.121440

Horvath, S. (2011). *Weighted Network Analysis: Applications in Genomics and Systems Biology*. New York, NY; Springer. doi: 10.1007/978-1-4419-8819-5

Hughes, B. (2007). Cancer: multiple targets to tackle tough tumours. *Nat. Rev. Drug Discov.* 6, 867–867. doi: 10.1038/nrd2449

Jeong, H., Mason, S. P., Barabasi, A. L., and Oltvai, Z. N. (2001). Lethality and centrality in protein networks. *Nature* 411, 41–42. doi: 10.1038/35075138

Jiang, Z.-X., and Zhang, Z.-Y. (2008). Targeting PTPs with small molecule inhibitors in cancer treatment. *Cancer Metast. Rev.* 27, 263–272. doi: 10.1007/s10555-008-9113-3

Jonsson, P. F., and Bates, P. A. (2006). Global topological features of cancer proteins in the human interactome. *Bioinformatics* 22, 2291–2297. doi: 10.1093/bioinformatics/btl390

Kandasamy, K., Mohan, S. S., Raju, R., Keerthikumar, S., Kumar, G., Venugopal, A., et al. (2010). Netpath: a public resource of curated signal transduction pathways. *Genome Biol.* 11:R3. doi: 10.1186/gb-2010-11-1-r3

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

Kang, J. M., Park, S., Kim, S. J., Hong, H. Y., Jeong, J., Kim, H.-S., et al. (2012). Cbl enhances breast tumor formation by inhibiting tumor suppressive activity of tgf-Îš signaling. *Oncogene* 31, 5123–5131. doi: 10.1038/onc.2012.18

Kang, Y., He, W., Tulley, S., Gupta, G. P., Serganova, I., Chen, C.-R., et al. (2005). Breast cancer bone metastasis mediated by the smad tumor suppressor pathway. *Proc. Natl. Acad. Sci. U.S.A.* 102, 13909–13914. doi: 10.1073/pnas.0506517102

Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika* 18, 39–43. doi: 10.1007/BF02289026

Khodarev, N. N., Roizman, B., and Weichselbaum, R. R. (2012). Molecular pathways: interferon/stat1 pathway: Role in the tumor resistance to genotoxic stress and aggressive growth. *Clin. Cancer Res.* 18, 3015–3021. doi: 10.1158/1078-0432.CCR-11-3225

Koschützki, D., and Schreiber, F. (2008). Centrality analysis methods for biological networks and their application to gene regulatory networks. *Gene Regul. Syst. Bio.* 2, 193–201.

Krupp, M., Marquardt, J. U., Sahin, U., Galle, P. R., Castle, J., and Teufel, A. (2012). RNA-Seq Atlas—a reference database for gene expression profiling in normal tissue by next-generation sequencing. *Bioinformatics* 28, 1184–1185. doi: 10.1093/bioinformatics/bts084

Li, J., Kong, X., Zhang, J., Luo, Q., Li, X., and Fang, L. (2013). Mirna-26b inhibits proliferation by targeting ptgs2 in breast cancer. *Cancer Cell Int.* 13:7. doi: 10.1186/1475-2867-13-7

Li, M., Wang, J., Chen, X., Wang, H., and Pan, Y. (2011). A local average connectivity-based method for identifying essential proteins from the network level. *Comput. Biol. Chem.* 35, 143–150. doi: 10.1016/j.compbiolchem.2011.04.002

Lima-Mendez, G., and van Helden, J. (2009). The powerful law of the power law and other myths in network biology. *Mol. BioSyst.* 5, 1482–1493. doi: 10.1039/b908681a

Macedo, L. F., Guo, Z., Tilghman, S. L., Sabnis, G. J., Qiu, Y., and Brodie, A. (2006). Role of androgens on mcf-7 breast cancer cell growth and on the inhibitory effect of letrozole. *Cancer Res.* 66, 7775–7782. doi: 10.1158/0008-5472.CAN-05-3984

McCall, M. N., Uppal, K., Jaffee, H. A., Zilliox, M. J., and Irizarry, R. A. (2011). The gene expression barcode: leveraging public data repositories to begin cataloging the human and murine transcriptomes. *Nucleic Acids Res.* 39(Suppl. 1), D1011–D1015. doi: 10.1093/nar/gkq1259

Mering, C. v., Huynen, M., Jaeggi, D., Schmidt, S., Bork, P., and Snel, B. (2003). STRING: a database of predicted functional associations between proteins. *Nucleic Acids Res.* 31, 258–261. doi: 10.1093/nar/gkg034

Mesa, R. A. (2010). Ruxolitinib, a selective jak1 and jak2 inhibitor for the treatment of myeloproliferative neoplasms and psoriasis. *IDrugs* 13, 394–403.

Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., and Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Meth.* 5, 621–628. doi: 10.1038/nmeth.1226

Nagaraj, N. S., and Datta, P. K. (2010). Targeting the transforming growth factor-beta signaling pathway in human cancer. *Expert Opin. Invest. Drugs* 19, 77–91. doi: 10.1517/13543780903382609

Ogata, Y., Osaki, T., Naka, T., Iwahori, K., Furukawa, M., Nagatomo, I., et al. (2006). Overexpression of PIAS3 suppresses cell growth and restores the drug sensitivity of human lung cancer cells in association with PI3-K/Akt inactivation. *Neoplasia* 8, 817–825. doi: 10.1593/neo.06409

Okamoto, T., Inozume, T., Mitsui, H., Kanzaki, M., Harada, K., Shibagaki, N., et al. (2010). Overexpression of grim-19 in cancer cells suppresses stat3-mediated signal transduction and cancer growth. *Mol. Cancer Ther.* 9, 2333–2343. doi: 10.1158/1535-7163.MCT-09-1147

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). *The Pagerank Citation Ranking: Bringing Order to the Web.* Technical Report 1999-66, Stanford InfoLab.

Perumal, D., Lim, C. S., and Sakharkar, M. K. (2009). A comparative study of metabolic network topology between a pathogenic and a non-pathogenic bacterium for potential drug target identification. *Summit. Trans. Bioinf.* 2009, 100–104.

Petrelli, A., and Giordano, S. (2008). From single- to multi-target drugs in cancer therapy: when aspecificity becomes an advantage. *Curr. Med. Chem.* 15, 422–432. doi: 10.2174/092986708783503212

Pico, A. R., Kelder, T., van Iersel, M. P., Hanspers, K., Conklin, B. R., and Evelo, C. (2008). WikiPathways: pathway editing for the people. *PLoS. Biol.* 6:e184. doi: 10.1371/journal.pbio.0060184

Pourdehnad, M., Truitt, M. L., Siddiqi, I. N., Ducker, G. S., Shokat, K. M., and Ruggero, D. (2013). Myc and mtor converge on a common node in protein synthesis control that confers synthetic lethality in myc-driven cancers. *Proc. Natl. Acad. Sci. U.S.A.* 110, 11988–11993. doi: 10.1073/Pnas.1310230110

Ranola, J. M., Langfelder, P., Lange, K., and Horvath, S. (2013). Cluster and propensity based approximation of a network. *BMC Syst. Biol.* 7:21. doi: 10.1186/1752-0509-7-21

Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika* 31, 581–603. doi: 10.1007/BF02289527

Sansone, P., and Bromberg, J. (2012). Targeting the interleukin-6/jak/stat pathway in human malignancies. *J. Clin. Oncol.* 30, 1005–1014. doi: 10.1200/JCO.2010.31.8907

Scaltriti, M., and Baselga, J. (2006). The epidermal growth factor receptor pathway: a model for targeted therapy. *Clin. Cancer Res.* 12, 5268–5272. doi: 10.1158/1078-0432.CCR-05-1554

Scherf, U., Ross, D. T., Waltham, M., Smith, L. H., Lee, J. K., Tanabe, L., et al. (2000). A gene expression database for the molecular pharmacology of cancer. *Nat. Genet.* 24, 236–244. doi: 10.1038/73439

Soubeyran, P., Kowanetz, K., Szymkiewicz, I., Langdon, W. Y., and Dikic, I. (2002). Cbl-cin85-endophilin complex mediates ligand-induced downregulation of egf receptors. *Nature* 416, 183–187. doi: 10.1038/416183a

Sun, J., and Zhao, Z. (2010). A comparative study of cancer proteins in the human protein-protein interaction network. *BMC Genom.* 11(Suppl. 3):S5. doi: 10.1186/1471-2164-11-S3-S5

Valente, T. W., and Foreman, R. K. (1998). Integration and radiality: measuring the extent of an individual's connectedness and reachability in a network. *Soc. Netw.* 20, 89–105. doi: 10.1016/S0378-8733(97)00007-5

Verstovsek, S., Mesa, R. A., Gotlib, J., Levy, R. S., Gupta, V., DiPersio, J. F., et al. (2012). A double-blind, placebo-controlled trial of ruxolitinib for myelofibrosis. *New Eng. J. Med.* 366, 799–807. doi: 10.1056/NEJMoa1110557

Wang, J., Chen, G., Li, M., and Pan, Y. (2011). Integration of breast cancer gene signatures based on graph centrality. *BMC Syst. Biol.* 5(Suppl. 3):S10. doi: 10.1186/1752-0509-5-S3-S10

Wang, X., Thijssen, B., and Yu, H. (2013). Target essentiality and centrality characterize drug side effects. *PLoS Comput. Biol.* 9:e1003119. doi: 10.1371/journal.pcbi.1003119

Wehmuth, K., and Ziviani, A. (2011). "Distributed location of the critical nodes to network robustness based on spectral analysis," in *Network Operations and Management Symposium (LANOMS), 2011 7th Latin American* (Quito), 1–8. doi: 10.1109/LANOMS.2011.6102259

Xia, J., Sun, J., Jia, P., and Zhao, Z. (2011). Do cancer proteins really interact strongly in the human protein-protein interaction network? *Comput. Biol. Chem.* 35, 121–125. doi: 10.1016/j.compbiolchem.2011.04.005

Xu, J., Chen, Y., and Olopade, O. I. (2010). Myc and breast cancer. *Genes Cancer* 1, 629–640. doi: 10.1177/1947601910378691

Xu, S., Robbins, D., Frost, J., Dang, A., Lange-Carter, C., and Cobb, M. H. (1995). Mekk1 phosphorylates mek1 and mek2 but does not cause activation of mitogen-activated protein kinase. *Proc. Natl. Acad. Sci. U.S.A.* 92, 6808–6812. doi: 10.1073/pnas.92.15.6808

Yang, S.-H., Sharrocks, A. D., and Whitmarsh, A. J. (2003). Transcriptional regulation by the MAP kinase signaling cascades. *Gene* 320, 3–21. doi: 10.1016/S0378-1119(03)00816-3

Yu, H., Pardoll, D., and Jove, R. (2009). STATs in cancer inflammation and immunity: a leading role for STAT3. *Nat. Rev. Cancer* 9, 798–809. doi: 10.1038/nrc2734

Zhong, Q., Simonis, N., Li, Q.-R., Charloteaux, B., Heuze, F., Klitgord, N., et al. (2009). Edgetic perturbation models of human inherited disorders. *Mol. Syst. Biol.* 5:321. doi: 10.1038/msb.2009.80

Conflict of Interest Statement: 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.

Received: 25 November 2013; paper pending published: 13 December 2013; accepted: 13 January 2014; published online: 03 February 2014.

*Citation: Peng Q and Schork N J (2014) Utility of network integrity methods in therapeutic target identification. Front. Genet. **5**:12. doi: 10.3389/fgene.2014.00012*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 20 February 2014
doi: 10.3389/fgene.2014.00021

Interactive exploration of integrated biological datasets using context-sensitive workflows

Fabian Horn[1]*, Martin Rittweger[1], Jan Taubert[2], Artem Lysenko[2], Christopher Rawlings[2] and Reinhard Guthke[1]

[1] Systems Biology/Bioinformatics, Leibniz Institute for Natural Product Research and Infection Biology - Hans Knöll Institute, Jena, Germany
[2] Department of Computational and Systems Biology, Rothamsted Research, Harpenden, UK

Edited by:
Frank Emmert-Streib, Queen's University Belfast, UK

Reviewed by:
Marshall Abrams, University of Alabama at Birmingham, USA
Frank Emmert-Streib, Queen's University Belfast, UK

***Correspondence:**
Fabian Horn, Systems Biology/Bioinformatics, Leibniz Institute for Natural Product Research and Infection Biology - Hans Knöll Institute, Beutenbergstraße 11a, 07745 Jena, Germany
e-mail: fabian.horn@hki-jena.de

Network inference utilizes experimental high-throughput data for the reconstruction of molecular interaction networks where new relationships between the network entities can be predicted. Despite the increasing amount of experimental data, the parameters of each modeling technique cannot be optimized based on the experimental data alone, but needs to be qualitatively assessed if the components of the resulting network describe the experimental setting. Candidate list prioritization and validation builds upon data integration and data visualization. The application of tools supporting this procedure is limited to the exploration of smaller information networks because the display and interpretation of large amounts of information is challenging regarding the computational effort and the users' experience. The Ondex software framework was extended with customizable context-sensitive menus which allow additional integration and data analysis options for a selected set of candidates during interactive data exploration. We provide new functionalities for on-the-fly data integration using InterProScan, PubMed Central literature search, and sequence-based homology search. We applied the Ondex system to the integration of publicly available data for *Aspergillus nidulans* and analyzed transcriptome data. We demonstrate the advantages of our approach by proposing new hypotheses for the functional annotation of specific genes of differentially expressed fungal gene clusters. Our extension of the Ondex framework makes it possible to overcome the separation between data integration and interactive analysis. More specifically, computationally demanding calculations can be performed on selected sub-networks without losing any information from the whole network. Furthermore, our extensions allow for direct access to online biological databases which helps to keep the integrated information up-to-date.

Keywords: exploratory analysis, Ondex, data integration, data visualization, information network, *Aspergillus nidulans*, customizable workflow, gold-standard

1. INTRODUCTION

In our study, we developed and applied customizable context-sensitive menus to the data integration and visualization tool Ondex. This allows for the interactive exploration of experimental data that is integrated into an information network. The introduction starts with a survey of network inference methods and qualitative assessment of inferred networks. Exploratory data analysis looks for new patterns and hypothesis in a dataset and it is thus well suited to qualitatively assess network modeling and experimental results within the context of an information network.

1.1. QUALITATIVE ASSESSMENT OF NETWORK INFERENCE

Network inference reconstructs molecular interaction networks on the basis of experimental high-throughput data. Nodes in the resulting network usually represent molecular entities (e.g., genes or proteins), for which concentration or activity has been measured using omics-technology. The edges in the network stand for direct and indirect relationships between the molecular entities, i.e., they symbolize diverse modes of regulation or direct molecular interaction. New molecular relationships may be predicted with the help of network inference modeling techniques. The predictions are new biological hypotheses which result from the given experimental data. A highly diverse variety of network inference modeling techniques have been developed based on differential equation systems or Bayesian networks (as reviewed in Hecker et al., 2009). Each modeling technique utilizes a wide range of modeling parameters which are optimized mainly on the given experimental data. For example, the gene regulatory network inference method NetGenerator (Guthke et al., 2005; Töpfer et al., 2006; Weber et al., 2013) is based on differential equation systems which minimizes both the model fit error, i.e., the difference between the measured and the simulated data of time-series experiments, and the number of model parameters. Additionally, prior knowledge is used to guide the inference process (Linde et al., 2010, 2012).

In order to validate the chosen modeling technique and its parameter optimization, it is necessary to assess the validity of the resulting biological networks. Quantitative measures utilize an error model and model selection criteria, e.g., least square

Network Inference

- Q: How can exploratory analysis be used for the validation of inferred networks?
- A: It needs to be qualitatively assessed if inferred network components describe the experimental setting. Tools like Ondex provide automatic data integration and visualization which facilitate exploratory data analysis as well as the quality control.
- Q: What are the challenges for this kind of qualitative network validation?
- A: The large amount of available information leads to a high computational effort during the data integration and the automatic data visualization. This may result in a non-satisfying users' experience.
- Q: What method is introduced to overcome these limitations?
- A: We introduce the concept of context-sensitive workflows for Ondex. During the data exploration, it allows for the integration of additional information for a set of interesting features. Thus, computational-demanding calculations are only performed for a subnetwork, which greatly improves the usability of the tool for network validation.

error model and Akaike's Information Criterion, which makes use of the experimental data and the inferred model (Rao et al., 2008). The internal validation evaluates whether the model is robust and can be generalized. For these purposes, subsampling (cross-validation), bootstrapping and network perturbations are applied (reviewed by Hecker et al., 2009). Another aspect is the utilization of benchmark data which can be either generated from an artificially constructed gene regulatory network or the experimental data is gathered from a well-researched biological system. As an example, the DREAM challenge provided gene expression data from a synthetically generated network which consisted of five genes (Cantone et al., 2009).

Nevertheless, the parameters of each network inference technique cannot be optimized based on experimental or simulated data alone. The particular outcome of a network reconstruction needs to be qualitatively assessed by verifying that its components describe the experimental setting and that they are in accordance to prior knowledge. For this step, test data ("gold-standard") is required, which was not included in the training dataset for the network inference. It consists of expert knowledge and data which was predicted with the help of bioinformatic tools, e.g., the software tool SiTAR for transcription factor binding sites predictions (Fazius et al., 2011). As a second aspect, network inference methods can infer genome-wide networks which may contain thousands of nodes and relations (Altwasser et al., 2012). Validation of these genome-wide networks is hard because the number of model parameters is very high and the gold-standard used is usually too small to make generalizations about the quality of the whole network. Due to the large size of the resulting network, the experimental validation with a high quality standard is not suitable. Thus, all components and proposed interactions need to be interpreted and prioritized before further experimental analysis.

1.2. EXPLORATORY DATA ANALYSIS

From a methodological point of view, feature selection or candidate prioritization can be performed by two complementary approaches: exploratory and confirmatory data analysis (Tukey, 1977, 1980). Confirmatory data analysis starts with an open, precise question. Usually, it is a fixed procedure and it is, hence, especially suited to be performed by a computer, e.g., using statistics or guided pathway exploration. In contrast, exploratory data analysis (as introduced by Tukey, 1980) does not follow a direct route between question and answer, but allows for iterative cycles between research question, experimental design and gathered data. Computer analysis is needed to search for the right questions and hidden relationships buried within the massive amounts of data coming from high-throughput methods. Another aspect is that data stored within public data repositories usually has been analyzed with respect to a limited number of research questions. Therefore, there is still a considerable potential to gain new insights from this data, but the challenge is to find the right questions in order to perform a successful meta-analysis or re-analysis of the data. Despite the importance of exploratory analysis for research, very few software systems are available that support the requirements to integrate multiple sources of biological data and provide the rich set of analysis methods needed for exploratory data analysis (discussed in Kelder et al., 2010). To help researchers recognize patterns within the data more readily and to enable them to concentrate on the interpretation of the data, software tools should perform all automatable tasks of handling large data amounts, i.e., the data integration and automatic data visualizations.

1.3. ONDEX—A SOFTWARE SOLUTION FOR EXPLORATORY ANALYSIS

Many software tools have been developed for data integration and visualization using network structures. [A good review of data integration methodologies and tools is given by Huttenhower and Hofmann (2010) and Bebek et al. (2012) and tools for visualization of biological network data are described by Pavlopoulos et al. (2008).] In this study, we use the Ondex data integration framework, which combines data integration, analysis, and visualization (Köhler et al., 2006). While Ondex shares many of its features with other tools, its main advantage lies in its flexible data representation and available visualization methods (Taubert et al., 2007). It is very suitable for exploratory data analysis—meaning the exploration of experimental data without prior hypotheses and a pre-defined data analysis workflow. Ondex uses a graph-based core data structure where nodes represent biological entities (e.g., genes or proteins) and edges represent the relationships between them (e.g., "a gene *encodes* a protein"). Using ontologies, Ondex automates the integration of heterogeneous data from diverse sources (e.g., structured data repositories, flat files, or free-text) into a semantically consistent graph representation. The provenance of the data is retained during the integration process. The modular plug-in architecture of Ondex enables the addition of extra functionality, such as parsers for new data sources or complex filtering methods. The Ondex front-end facilitates interactive visualization, searching, and filtering of the datasets. Certain attributes contained in the graph can be associated with the color, glyphs, and visibility of nodes and edges. Ondex is open source, written in platform-independent Java and supports open standards and interfaces. The Ondex data integration framework has already successfully been used for the study of microarray expression data (Köhler et al., 2006), data integration for plant genomics (Lysenko et al., 2009), supporting *in silico* drug discovery (Cockell et al., 2010), and finding genes implicated in plant stress response (Hassani-Pak et al., 2010).

Despite the advantages and the successful application of Ondex, the handling of large amounts of data is still challenging. The system can be used to produce integrated datasets with several millions of entries, which makes efficient querying and visualization difficult. Additionally, the data warehousing approach of Ondex means that some of the data can become out-of-date. This contrasts with the approach used by federated data integration systems, which always query live data resources [e.g., Taverna (Hull et al., 2006)]. To overcome these limitations to some extent, the Ondex front-end already offers the possibility to iteratively explore parts of the graph and link-out to more recent data available in online resources.

In this paper, we present the implementation of interactive context-sensitive workflows in Ondex to improve the analysis of large integrated datasets.

1.4. OUR WORK

We applied Ondex to construct a gold-standard information network for *Aspergillus nidulans* as a basis for the qualitative assessment of reconstructed networks. The network inference for filamentous fungi is challenged by the circumstance that prior knowledge is limited and widely scattered. It needs to be collected from literature and diverse databases, or predictions need to be made with the help of additional bioinformatic tools (Horn et al., 2012). As a consequence, no extensive knowledge network exists which can function as gold-standard. *A. nidulans* is the main model organisms for filamentous fungi and substantial knowledge about the regulation of secondary metabolites exists (Brakhage, 2013). Secondary metabolites may directly contribute to the pathogenicity of fungi, e.g., gliotoxin was found to modulate the immune response and induce apoptosis in cells of *A. fumigatus* (Scharf et al., 2012). The knowledge about the mechanisms of regulation of secondary metabolites of *A. nidulans* can be transferred to other filamentous fungi, especially if filamentous fungi share the same secondary metabolite gene clusters. As an example, the penicillin gene cluster is present in *A. nidulans* and *Penicillium chrysogeneum*. Generally, most fungal gene clusters are silent under standard laboratory conditions, and it is promising for drug target research to systematically determine conditions under which these gene clusters are expressed and secondary metabolites are produced (Walton, 2000; Brakhage and Schroeckh, 2011). Prominent examples are non-ribosomal peptide synthetases (NRPS) and polyketide synthases (PKS), which are two main classes of secondary metabolites that often serve as drug lead structures (Newman and Cragg, 2012). Most gene clusters are currently not functionally annotated (Sanchez et al., 2012) making the investigation of gene clusters challenging.

One of our objectives was to facilitate simultaneous interactive data integration during visual data exploration. This procedure has been implemented and is available for the community (see section 2). In order to demonstrate the practical relevance of our approach, the context-sensitive menus and the invocation of external web services were applied to the integrated network for *A. nidulans*. This also included data from an expression profiling experiment comparing the wild-type and a ΔcnsE-mutant at different developmental stages (Nahlik et al., 2010).

2. MATERIALS AND METHODS

In this section, we present (1) an information network for *A. nidulans* and (2) an extension of the Ondex framework in the form of context-sensitive menus, which are subsequently (3) used to analyze a transcriptome experiment. In this section, the functionality of the menus and generalizable workflows and approaches are presented. Exploration strategies which are based on the specific data, intermediate results, and the research question of the experiment are presented in the results section. The workflow of data integration, the resulting network, and the context-specific menu items are available from http://ondex.rothamsted.ac.uk/anidulans.

2.1. DATA INTEGRATION: AN APPLICATION CASE FOR *Aspergillus nidulans*

Experimentally-derived data for most fungi is scarce, incomplete, and scattered over several resources. Additionally, this data undergoes rapid changes due to newly assigned annotations and new genome assemblies. We integrated several publicly available datasets for *A. nidulans* using the pre-existing plug-ins from the Ondex integrator. (An overview of the data sources used and the extracted data is given in **Table 1** and **Figure 1**). *Gene* concepts are mapped to the Gene Ontology hierarchies (Ashburner et al., 2000) (*Biological_Process, Cellular_Component, Molecular_Function*). Additional functional annotation data from the FunCat (*Functional Categories*) (Ruepp et al., 2004) and KEGG Pathways (*Pathway*) (Kanehisa et al., 2012) was integrated. In order to allow comparative analysis of *A. nidulans*, orthologous gene mappings to *Aspergillus fumigatus* and *Saccharomyces cerevisiae* were included. A mapping between publications and genes was performed if these genes were in the focus of the publication. A list of manually-curated publications was downloaded from the Aspergillus Genome Database (AspGD) (Arnaud et al., 2010). Additionally, a metabolic network (David et al., 2008), reflecting the regulatory relationships of enzymatic reactions by regulatory genes was integrated.

Experimental data from a study from Nahlik et al. (2010) was integrated. This data is available from the Gene Expression Omnibus (GEO identifier: GSE22442). The focus of this experiment was to investigate the impact of the COP9 signalosome complex on the transcriptome. Two genotypes (wild-type and Δ*csnE*-mutant) were compared under four different induced

Table 1 | Data sources for *A. nidulans* information network.

Data source	Web address	Description	Concept class
Aspergillus genome database	www.aspgd.org	Gene ontologies Homologues Literature Synonyms	GO GeneAfu, GeneScer Publication Gene
Gene ontology	www.geneontology.org	Gene ontologies	GO
Ensembl fungi (CADRE)	http://fungi.ensembl.org	Annotation Chromosomal position Identifier mapping	Gene Gene Gene
KEGG (version: 2011)	www.genome.jp/kegg/	Pathways	Pathway
MIPS functional catalog	http://pedant.gsf.de	FunCat ontologies	FunCat
David et al. (2008)	www.biomedcentral.com	Metabolic network	Metabolite, reaction
GEO (GSE22442) (Nahlik et al., 2010)	www.ncbi.nlm.nih.gov/geo/	Expression values	Gene

Several data sources for A. nidulans were parsed and mapped into one Ondex information network. Data was downloaded from several sources. Instead of using publicly available transcriptome data from GEO, in-house experimental data may be mapped.

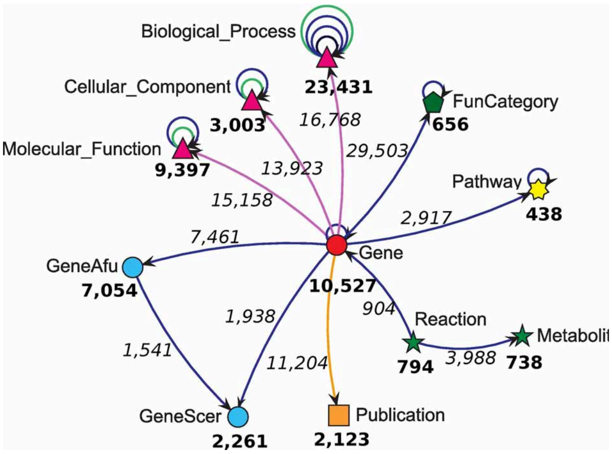


FIGURE 1 | Meta-legend of the integrated dataset for *A. nidulans* depicting all available concept classes. The number of concepts within a concept class is given in bold numbers, whereas the number of mappings or relations between respective concept classes are given in italic numbers.

growth stages—vegetative 14 h, vegetative 20 h, sexual 48 h, asexual 48 h. For each growth stage, two biological replicates each with four technical replicates were measured, summing up to a total of 32 samples. The raw data was downloaded from GEO and the biological replicates were normalized individually with the help of loess and quantile normalization provided by the limma package (Smyth and Speed, 2003). A preceding analysis of variance (ANOVA) showed that the highest systematic variation arises from the biological replicates rather than any other experimental source of variation. Thus, the signals were modeled independently with the linear model provided by the limma-package. Calculated *p*-values were corrected for multiple testing using the method by Benjamini and Hochberg (Benjamini and Yekutieli, 2001). Results from both biological replicates were combined using the z-transformation of the *p*-values suggested by Stouffer (Whitlock, 2005). Probe sequences were mapped to gene definitions of the *A. nidulans* structural genome annotation (Horn et al., 2011). According to this mapping, the experimental data was integrated by Ondex into the *A. nidulans* information network. In order to emphasize the level of regulation, the glyphs of the gene concepts were scaled and colored by Ondex according to the expression values of the corresponding transcripts. For each growth-stage, the resulting information network was further explored separately in order to adequately understand the underlying interactions and correctly interpret the experimental data with respect to the experimental setting.

2.2. ONDEX EXTENSION: CONTEXT-SENSITIVE MENUS

The flexibility and power of the analysis offered by the Ondex system is realized primarily through the notion of customisability, i.e., users are free to build their own application cases from a set of generic re-usable components. Larger integration and analysis tasks are realized as workflow components, whereas the less substantial ones can be completed by calling a set of in-built functions. To that end, the Ondex system incorporates a JavaScript API (based on Mozilla Rhino v1.7) and a rich selection of binding and analysis functions that can be used both to manipulate the graph and to alter its appearance in the Ondex front-end. The binding and functions available via the scripting environment abstract some of the complexity of the Java-based Ondex API and allow for more concise syntax and greater convenience. This additional simplification is made possible by the use of run-time bytecode code generation (powered by the JavaAssist v3.12.0 library) that creates a set of wrappers. This setup allows both easy incorporation of additional external libraries and their seamless integration into the Mozilla Rhino scripting environment by automating the process of creating wrappers that implement additional interface(s) or delegate calls to multiple classes.

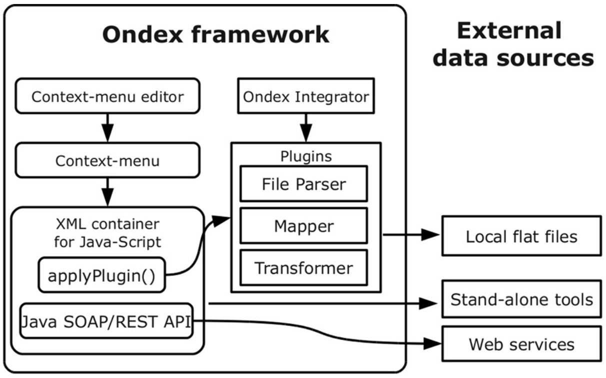


FIGURE 2 | Simplified modular view of context menu extension. The new Ondex framework extension realizes context-sensitive menus which can be edited with the help of a GUI based editor. Concept specific entries are implemented in XML-files that contain Java Script code. Existing plug-ins for data integration from the Ondex library can be directly invoked. Web services and local installations of external programs can be called from the context menu.

The Ondex scripting environment can be accessed interactively using a console environment. In this work, we have extended this functionality further by developing a system of context-specific menus that can dispatch calls to the Ondex scripting APIs. The use of temporary sub-graphs also allows users to define their own JavaScript functions to be added as entries on these menus. For this study, we provide new functions for on-the-fly data integration using InterProScan, Blast, and PubMedCentral full-text search (see next section for more details). Using the modular architecture of Ondex, we extended the framework with context-sensitive pop-up menus which allow integration to be performed on the fly, while the network is being explored visually (see **Figure 2**). Throughout the paper, we keep the official nomenclature that nodes in an Ondex network are referred to as *concepts* (Taubert et al., 2007). The term *gene concept* hence describes one particular node which represents a gene entity. While examining the graph, users can integrate additional data or perform computationally demanding calculations for selected concepts. The menus are sensitive in regard to the concept class. This means that certain operations are only available for certain concept classes, e.g., BLAST operations are only available if the concept is a gene and contains sequence information. Internally, computational operations are either performed directly on the main graph or a temporary sub-graph, which initially consist only of the user-selected concepts. This mechanism also facilitates the re-use of the wide variety of Ondex workflow plug-ins, as individual workflow modules can also be called via particular items in the menu. Context menu functionalities are implemented using JavaScript code and they are stored as Extensible Markup Language (XML) files on the local file system using JavaBeans. Each XML-file represents one context menu item which can be restricted to be applicable only to the nodes of particular Ondex concept classes. The context-specific menus can also be organised hierarchically, where the nesting of the sub-menus is represented by the structure of the directories containing the XML-files in the file system.

The XML-files can either be edited with the help of external tools or with an embedded JavaScript editor. The graphical user interface (GUI) of the editor provides an easy way to specify concept class restrictions, integration of additional Java libraries and syntax highlighting with help of jEdit. The Ondex framework extension has been integrated into the main Ondex project and is freely available at http://www.ondex.org.

2.3. SPECIFIC WORKFLOW AND CUSTOMIZED CONTEXT-SENSITIVE MENUS FOR *Aspergillus nidulans* DATASET

In Ondex, the precise workflow of exploration depends specifically on the integrated data, the research question, and the preferences of the user. Nevertheless, we performed analyses with a generally applicable workflow that gave us a first overview of the information contained in the data, i.e., filtering down to specific genes, biological processes, and underlying interactions. For our study, we provide new functionalities to Ondex through context-sensitive menus, namely the InterProScan, a sequence-based homology search, and a full-text literature search at PubMed Central. During the procedure described above, the log-fold changes of gene expression data from 32 samples were integrated as attributes to the gene concepts in the information network. This allowed us to significantly reduce the number of concepts (i.e., genes and proteins) by applying a filter based on the log-fold change and *p*-value. Thus, the number of interesting concepts which need to be manually checked were reduced. At the same time, all available information can be reconsidered during the analysis by redisplaying previously filtered data. We independently analyzed each contrast, i.e., the differences in transcript abundance between the Δ*csnE*-mutant and the wild-type at different time points (i.e., growth stages), and filtered for either differentially expressed transcripts (DEGs) ($|\text{fold-change}| \geq 4$ and adjusted P-value ≤ 0.05) or for *strongly* differentially expressed transcripts ($|\text{fold-change}| \geq 8$ and adjusted P-value ≤ 0.05). Previously integrated information, i.e., concepts such as gene ontologies and publications, was included in the visualization if it is associated with the resulting gene sets.

The DEGs were subject to further exploratory analysis and the integrated dataset was used to identify which of the functional categories were predominantly up or down-regulated. For this purpose, only gene concepts which are differentially regulated were made visible and all connected gene ontology concepts have been visualized while retaining their hierarchical network structure (see **Figure 3**). The functional categories and their associated differentially expressed genes were arranged using a hierarchical layout (see **Figure 3B**). In order to make our results comparable to the publication from Nahlik et al. (2010), we additionally grouped and named our functional categories according to the terminology adopted by that paper (For details see **Supplementary File S1**). Ondex automatically sorts the networks by its size, i.e., the number of connected concepts. Thus, it is immediately possible to identify and further explore the annotation-orientated sub-networks where many DEGs have been mapped.

A second approach is to explore known characteristics of the species in focus. In fungi, it is known that genes belonging to a single secondary metabolite pathway tend to cluster on the chromosome (Brakhage and Schroeckh, 2011). The *A. nidulans*

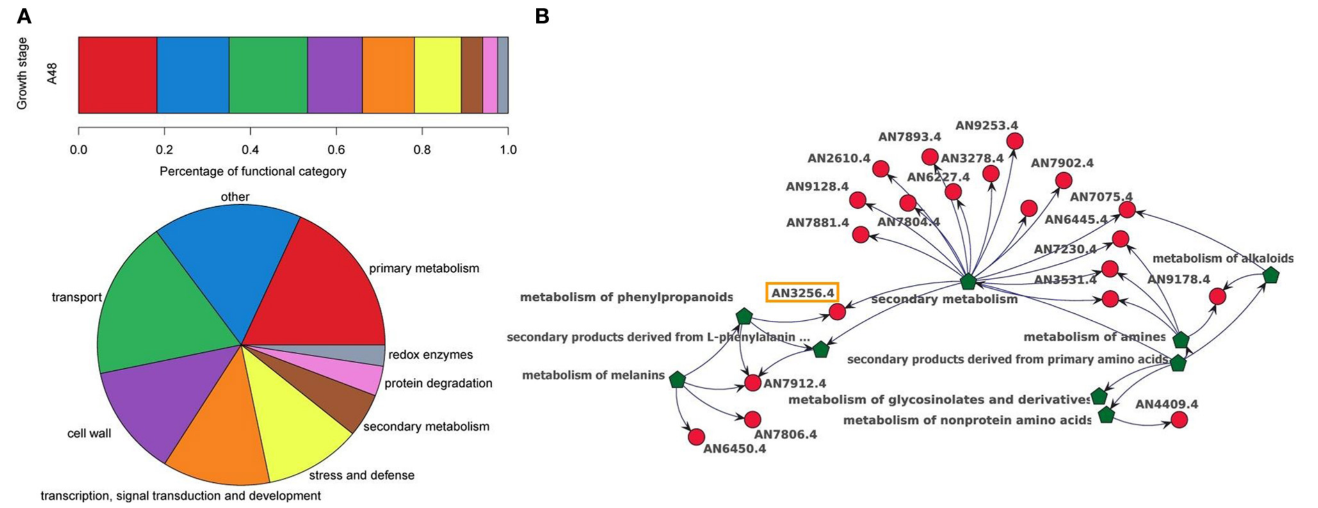


FIGURE 3 | Different visualization types of the distribution of functional annotations. (A) Bar and pie chart diagram representing the percentage of prominent biological processes within the strongly differentially expressed genes (|fold-change| ≥8, adjusted P-value ≤0.05). Example charts are shown for the asexual developmental growth stage after 48 h (A48). **(B)** A detail of the sub-network representing the mapping between functional categories and strongly differentially expressed genes during the sexual development after 48 h. Green polygonic concepts represent gene ontology concepts, whereas red circular concepts represent genes. An arrow between two gene ontologies reflects the hierarchical structure of the directed acyclic graph of the gene ontologies. An arrow pointing from an ontology concept to a gene concept shows which genes have been assigned to the respective gene ontology concept. In contrast to commonly used pie and bar chart diagrams, the visualization within Ondex allows to explore possible multiple mappings between genes and ontologies without losing the information about the level of detail at which the mapping occurs.

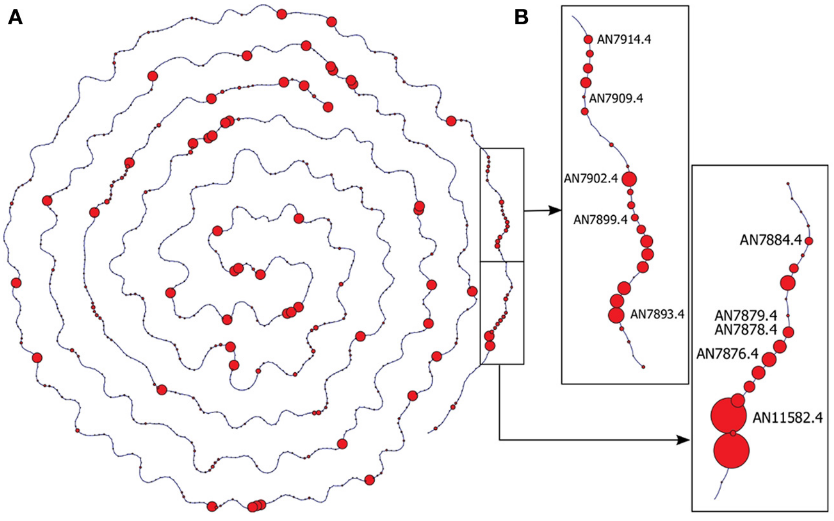


FIGURE 4 | Chromosomal View/Gene Cluster View. (A) Gene concepts located near to each other on the chromosome are connected. Transcriptome data has been integrated and glyphs are scaled accordingly, i.e., the circle size represents the expression values of the corresponding transcripts. Regions with several strongly differentially expressed genes are possible indications for secondary metabolite gene clusters. The figure depicts chromosome II of *A. nidulans* with the transcriptomic data of the asexual developmental growth stage after 48 h (A48). **(B)** Detailed view of the neighborhood of two differentially expressed gene clusters: the orsellinic gene cluster (AN7909–AN7914) and a putative, uncharacterized gene cluster (AN11582–AN7879).

information network includes the data of the chromosomal position of all genes. If two genes are neighbors, an edge is drawn between them. We applied Ondex' *genomic view* layout to immediately check the transcriptome data for the regulation of fungal gene clusters, because it lays out each chromosome separately and keeps the spatial information (see **Figure 4**). Differentially expressed gene clusters are recognized by the regions of the chromosome where several neighboring genes are depicted with larger glyphs (representing high fold-changes). This way, differentially expressed gene clusters could be identified in our analysis of the transcriptome data from Nahlik et al. (2010) (see **Supplementary File S2**).

For *A. nidulans*, there were many genes with little or no information in the network. One way to enhance the completeness of the functional annotation is to consider gene annotations from orthologous genes for related species. Orthologous genes are known to likely have a similar function. The orthology information was integrated for the relatively well-characterized fungal species *A. fumigatus* and *S. cerevisiae*. Another approach is the retrieval of additional data from other databases or the prediction of gene functions by applying bioinformatics methods. Context-sensitive workflow items beneficial to the exploratory analysis of experimental data of the resulting network were developed (see **Figure 5**). The implemented methods use web services ensuring that the processed data is up to date while outsourcing computations to the service providers.

- **InterProScan.** The protein sequences of selected genes are retrieved from the BioMart web service (Kinsella et al., 2011). The sequence is used to predict protein domains by invoking the web service for InterProScan (Zdobnov and Apweiler, 2001). The retrieved protein domains and their corresponding information are added to the network.
- **Homolog search.** The protein sequences of selected genes are retrieved from BioMart web service (Kinsella et al., 2011). The web service for NCBI-BLAST (Altschul et al., 1990) is invoked with these sequences in order to search for similar sequences in UniProtKB (Magrane and UniProt Consortium, 2011). Significant results and their corresponding details are integrated into the network. Additionally, a bidirectional BLAST was implemented to check whether similar sequences were really homologous.
- **Full-text literature search.** Selected genes and their corresponding synonyms are used to search all available full-texts at PubMed Central. The metadata of publications is retrieved from the web service and subsequently integrated into the information network. It is used to download the full text, which itself is scanned for occurrences of any gene name and synonym which is present in the information network. The text bodies are pre-computed using suffix trees (Ukkonen, 1995), in order to allow high-speed text-search using many keywords in large texts. To connect the publication to the network, edges are drawn between the publication and any identified gene.

The application of these interactive menu items facilitated the on-the-fly retrieval of additional data as part of our analysis workflow. The resulting networks were laid out adequately with the *genomic view* layout, which is already part of the Ondex suite. All resulting networks have been manually checked if previously unobserved relations between the data concept lead to new hypotheses.

3. RESULTS

In this study, we present new extensions of the Ondex system (Köhler et al., 2006) and demonstrate how they can be effectively applied to extract integrated information networks for new insights. We have enhanced the features within Ondex by implementing customizable context-sensitive menus which allow interactive integration of additional data while exploring the integrated information network in the Ondex front-end. The application of these context-sensitive menus enables interactive extensions of the network to be made by the user. This process is illustrated in **Figures 3–6**. The new functionality facilitates the gathering of additional information, which helps to retrieve existing annotations from web services and supports making hypotheses about possible gene functions. With the help of the interactive menus, we can overcome the strict separation between data integration and visualization. In this section, our approaches for the exploration of the specific data are presented. The precise workflow depends on intermediate results and the research focus of the experiment. To our knowledge, this is the first application of an integrative network analysis approach to *A. nidulans*.

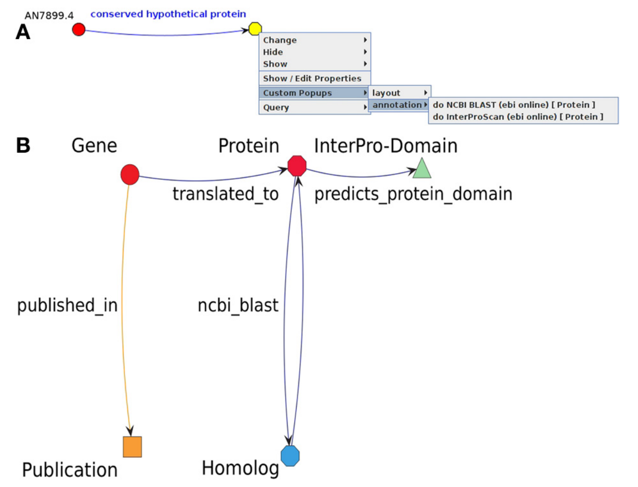


FIGURE 5 | Example application of the context-sensitive menu and meta-legend from the resulting network. (A) Screenshot of context-sensitive menu available for the concept class *protein*. The protein concept of an associated gene may be either used to invoke an NCBI-BLAST or InterProScan protein domain prediction. **(B)** Meta-legend of resulting network which has been integrated using the customized context-sensitive menu. The gene names can be used to retrieve publications from PubMed Central or to download protein sequences from the BioMart web service. The protein sequence can be subsequently used to predict the protein domains with the help of InterProScan. Another possibility is to search for similar genes with the help of NCBI-BLAST. A backward BLAST is available in order to provide a bidirectional BLAST to make the prediction of homologs more reliable. All operations are only performed for selected nodes and use web services to integrate up-to-date data.

3.1. INFORMATION NETWORK FOR *Aspergillus nidulans*

Data integration is especially important for less studied organisms, where often no reference genome data repositories such as Ensembl (Flicek et al., 2013) are available. Information networks function as the basis of the validation, prioritization, and selection of candidates from candidate lists resulting from modeling techniques. Usually, the gene annotation for less-studied organisms is highly fragmented and therefore it is necessary to call upon a large selection of less comprehensive resources in order to construct a representative annotation set. In order to expand the number of predicted functional annotations, it is common to integrate data from orthologous genes from closely-related species.

We applied Ondex to integrate publicly available data from *A. nidulans* (see **Table 1**). The resulting Ondex information network facilitates exploration of the existing data (see **Figure 1**). The network contains information for 10,527 genes, which are connected according to their chromosomal position. This allows for the detection and analysis of fungal chromosomal gene clusters. Furthermore, these genes were annotated with the three Gene Ontology (GO) domains: *biological process*, *cellular component*, and *molecular function* (Ashburner et al., 2000). The hierarchical structure of the Gene Ontology is preserved and 23,431, 3003, and 9397 different GO terms are integrated for each domain, respectively. Another set of functional annotations for *A. nidulans* is available from the Functional Catalog (FunCat) (Ruepp et al., 2004). This resource has more than 29,500 mappings between genes and 656 functional categories for *A. nidulans*. A large fraction of genes (2917) can also be mapped to pathways from the Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa et al., 2012) and its corresponding hierarchical structure which contains 438 unique entities. Additionally, a genome-wide metabolic network model published by David et al. (2008) was integrated. This model incorporates 904 instances of enzymatic reactions being regulated by particular genes in *A. nidulans*. The whole metabolic network is comprised of 794 reactions and 738 metabolites. Pre-computed orthologous genes in *Aspergillus fumigatus* and *Saccharomyces cerevisiae* were also integrated from the Aspergillus Genome Database (AspGD) (Arnaud et al., 2010). The annotations of 2261 *S. cerevisiae* and 7054 *A. fumigatus* genes have contributed to filling in the gaps in the *A. nidulans* annotation in the instances where orthologous genes between these organisms and *A. nidulans* were identified. AspGD also provides manually curated occurrences of *A. nidulans* in 2123 publications, which were also imported and made available in the Ondex information network.

3.2. INTEGRATION OF EXPERIMENTAL DATA

Publicly available data for *A. nidulans* was integrated into an information network which was subsequently used to compare a wild-type and a Δ*csnE*-mutant at different developmental stages by re-analyzing published microarray data (Nahlik et al., 2010). After the integration of experimental data taken from Nahlik et al. (2010), it was possible to filter the network to display only nodes representing genes where the regulation was affected by the mutation. This processing facilitated the identification of general trends in the datasets.

A total number of 1252 genes were found to be differentially expressed due to the Δ*csnE* mutation, when only non-redundant gene identifiers were counted at all measurement points (see **Table 2**). The distribution of differentially expressed genes (DEGs) between different contrasts shows that, despite the fact that *csnE* is only expressed during the first vegetative growth phase, most changes in gene expression occur at later stages of sexual development after 48 h (see **Table 2**). (The term *contrast* refers to the comparison of transcript abundances between different conditions at certain time points, i.e., the Δ*csnE* mutant versus the wild-type at different developmental stages.) This implies that most changes caused by the mutant take place before cell differentiation. Specifically, 1161 genes (95.1% of all 1252 DEGs) have a |fold-change| ≥4 during sexual or asexual development in contrast to only 157 DEGs (12.5%) during vegetative growth. A similar proportion holds for higher |fold-changes| ≥8 (see **Table 2**). In fact, the effect of a gradually increased number of DEGs caused by the Δ*csnE*-mutation can already be observed by comparing the two different time points of the vegetative growth phase.

3.3. VISUAL EXPLORATION OF FUNCTIONAL ANNOTATIONS

The network approach automatically considers that genes are mapped to different hierarchical levels of the functional annotation. As an example, in **Figure 3B** the gene AN3256.4 is associated with two different levels in the annotation hierarchy, i.e., the highest level *secondary metabolism* and a lower level *metabolism of phenylpropanoids*. In the pie chart visualization in **Figure 3A**, only the highest level is displayed and the gene is part of the section *secondary metabolism*. In order to show more detailed information about the lower hierarchies, new diagrams need to be drawn. The annotation-orientated network of each contrast forms a basis for further exploration, i.e., other functional annotation schemes such as GO, can be simultaneously shown and genes of interest can be displayed within the full context of all their annotated functional categories.

We performed a visual assessment of the functional annotation for all strongly differentially expressed transcripts at all four contrasts. Unlike Nahlik et al. (2010), we integrated publicly available functional annotations for *A. nidulans*, namely Functional Catalog and Gene Ontology. We tried to estimate whether our results (using automatically created data sources for the annotation) are comparable to the original publication (using manually assigned functional categories). The most prominent functional categories are *secondary metabolism*, *stress and defence related genes*, *cell wall* and genes associated with *transport* processes. In addition to the results published by Nahlik et al. (2010), a large proportion of differentially expressed genes is associated with *primary metabolism*. Due to the difference in the underlying functional annotation, the details of the results from the study of Nahlik et al. (2010) were not

Table 2 | Number of differentially expressed genes for each growth stage.

Developmental stage	|Fold-change| ≥ 4	|Fold-change| ≥ 8
V14	50	21
V20	134	45
V14 and V20	157	49
A48	980	438
S48	577	236
A48 and S48	1161	499
Total non-redundant DEGs	1252	530

In each analyzed growth stage (V14—vegetative growth after 14 h, V20—vegetative growth after 20 h, S48—sexual development after 48 h, A48—asexual development after 48 h), the wildtype is compared with a ΔcsnE mutant. The number of differentially expressed genes and strongly differentially expressed genes between both genotypes are shown.

completely comparable. Nevertheless, our analysis could reproduce the main findings of the original publication, i.e., the set of mainly regulated functional categories and the observation that the largest transcriptomic changes occur after 48 h. This endorses the manual classification of the authors, as well as the one offered by publicly-available annotation resources based on ontologies.

An exploration of the distribution of functional annotations within the network provides a quick, intuitive overview of affected processes and forms the basis for further in-depth analyses of gene functions. It is an alternative to commonly used visualizations of functional annotations with the help of bar or pie charts (see **Figure 3A**) and provides a starting point for a more detailed data interpretation.

3.4. EXPLORATION OF FUNGAL GENE CLUSTERS

The genomic view provided by the integrated Ondex network allows gene clusters, which are *strongly* differentially expressed, to be easily identified (see **Figure 4** and **Supplementary File S2**). Clusters of several strongly differentially expressed genes are possible indications for secondary metabolite gene clusters induced in the respective developmental stage.

Using the genomic view, the orsellinic acid synthesis cluster (AN7909–AN7914) is immediately identified as being strongly expressed in the vegetative growth phase after 20 h and in both developmental stages at 48 h (see **Figure 4**). These genes have been subject to the newly added interactive function—carrying out a full-text literature at PubMed Central. The genes that are part of the orsellinic acid cluster are linked to publications that have investigated their gene function. Although the precise function of products of this gene cluster is unknown, it has been shown that it is expressed if *A. nidulans* is co-cultured with the actinobacteria *Streptomyces rapamycinicus*, which is found in the same biological habitat (Schroeckh et al., 2009). The bacteria induces the expression of the orsellinic acid by histone modifications; in particular through the main histone acetyltransferase complex Saga/Ada (Nützmann et al., 2011). This gene cluster is therefore proposed to be part of a signalling pathway, which is involved in the communication between microbes of different species. The published data suggests that the interplay between fungi and microbes might be connected to the fungal development via the signalosome complex of *A. nidulans*. The exploration of the information network with Ondex linked this experiment, investigating the fungal signalosome, to publications focusing on fungal-bacterial interaction.

A second differentially expressed gene cluster of high interest was the characterized sterigmatocystin biosynthesis pathway which is composed of 25 genes (AN7805–AN7825) and located on chromosome IV (Brown et al., 1996) (see **Supplementary File S2**). This gene cluster is only expressed during the asexual growth stage, where large amounts of intermediate metabolites of this chemical structure have been verified by Nahlik et al. (2010). They possibly result from an inhibited secretion of the metabolite into the medium. The regulation of this gene cluster is very important since sterigmatocystin contributes to the defence of the cell against other microorganisms in the same habitat during this developmental phase. A more detailed functional annotation of this gene cluster was undertaken using our newly developed context-sensitive workflows for the Ondex system.

3.5. GENE ANNOTATION USING THE CONTEXT-SENSITIVE MENUS

Gene clusters found in the previous step were explored in greater detail in order to confirm their relevance. At this stage, several context-sensitive menus were used to obtain additional annotations (see **Figure 5**). As gene clusters encode for a single functional unit, genes encoding for particular enzymes, pathway regulators, and related transporters should be near each other on the chromosome in fungi. A hypothesis about the function of genes surrounding the sterigmatocystin gene cluster (AN7805–AN7825) was formed using Ondex. The InterProScan web service was invoked for neighboring genes surrounding the cluster. The observation that the neighboring gene AN7797.4 is highly down-regulated (fold-change $= -3.25$ and adjusted P-value $<$ 10^{-7}) and the prediction of a transmembrane protein domain by InterProScan led us to conjecture that this is a potential sterigmatocystin transporter, which needs to be validated experimentally (data not shown).

Another example of our exploratory data analysis was the application of the InterProScan for gene AN7899.4, in the region of the predicted NRPS AN7884.4 and the already described NRPS-PKS AN7909.4 (see **Figure 6A**). The gene is strongly differentially expressed in the mutant during vegetative growth after 20 h (fold-change $= 20.25$ and adjusted P-value $< 10^{-5}$). The InterProScan predicts that this gene contains a serine hydrolase domain and is therefore catalytically active. Additionally, the protein sequence was used to invoke a BLAST web service in order to search for potential similar genes. We found that only the CPSG_01145 gene from the pathogenic fungus *Coccidioides posadasii* had a high sequence similarity of more than 60%. It is annotated as citrinin biosynthesis oxidoreductase CtnB and therefore is likely to be involved in the secondary metabolism of this fungus. These findings and the chromosomal location of AN7899.4 in the proximity of two secondary metabolite gene clusters has not been reported before and make this gene interesting for further experimental research. The example shows that the added Ondex functionality helps to make new hypotheses which otherwise would not have been recognized.

Literature data is the most reliable and abundant source of information, which is regularly updated. The full text of a large fraction of publications can be mined with the help of the PubMed Central database web service. We were able to take advantage of this functionality using the context-sensitive menu system developed to support this application case. The full-text search was executed for all papers relating to a gene cluster (AN11582–AN7879) which was strongly differentially expressed during the asexual growth stage (see **Figure 6B**). This cluster was of great interest due to its close location to the orsellinic acid gene cluster. For three genes, an associated publication was found. A more detailed inspection revealed that in the paper by Shimizu et al. (2010) AN7876.4 and AN7878.4 were predicted to encode the transaminase B genes, whereas in the paper by Schroeckh et al. (2009) it was found that this gene cluster is co-expressed with the orsellinic acid gene cluster during co-cultivation with

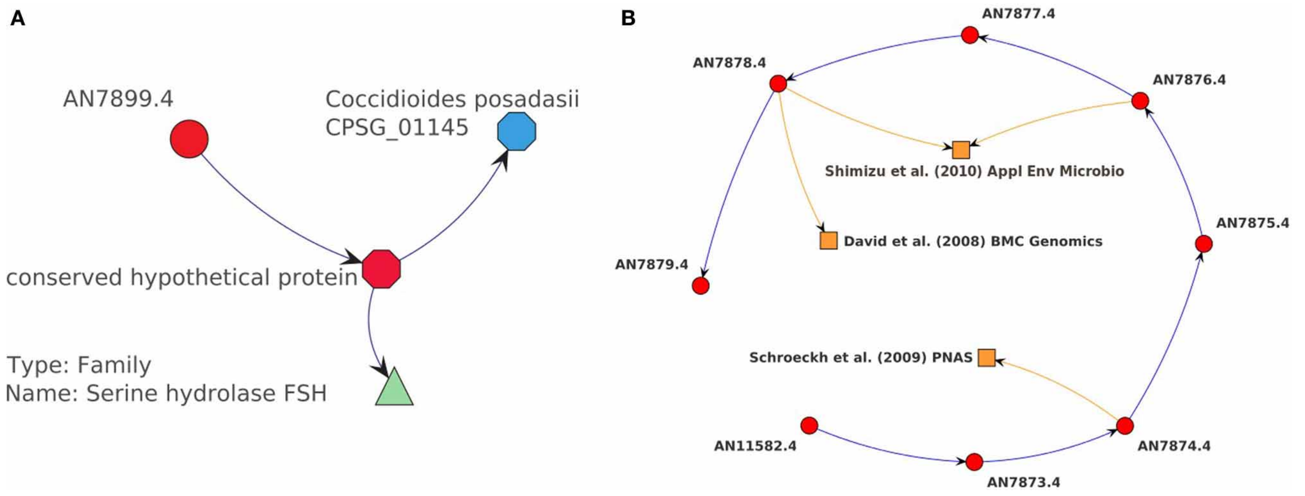

A
AN7899.4
Coccidioides posadasii
CPSG_01145
conserved hypothetical protein
Type: Family
Name: Serine hydrolase FSH
B
AN7877.4
AN7878.4
AN7876.4
Shimizu et al. (2010) Appl Env Microbio
AN7875.4
David et al. (2008) BMC Genomics
AN7879.4
Schroeckh et al. (2009) PNAS
AN11582.4
AN7874.4
AN7873.4

FIGURE 6 | Example annotation resulting from the usage of context-sensitive menus. (A) A gene of interest (AN7899.4) is explored further by retrieving the protein sequence (CADANIAG00003918) from BioMart web service. This sequence is used to invoke the InterProScan which predicts a serine hydrolase domain. The protein sequence is also used to execute a bidirectional BLAST-search finding a potential homologous gene (CPSG_01145) which is involved in the secondary metabolism of *Coccidioides posadasii*. **(B)** A sequence of neighbored strongly differentially expressed genes (AN11582.4–AN7879.4) is selected to retrieve publications from the full-text search from PubMed Central. Three different publications are returned (orange squared glyphs).

S. rapamycinicus. This approach shows that our extension easily reveals and visualizes connections between different studies which supports data interpretation. In contrast to a sole analysis in a web browser, Ondex instantly integrates newly found citations within the information network. The new concepts can form the basis for further additional data integration. That way, it is much easier to reproduce the same chain of reasoning. Additionally, the new network can benefit from other basic functionalities of Ondex, i.e., interactive or automatic visualization of the information network, the creation of additional labels, and the usage of filters. Overall, the integration of new knowledge into the networks ensures that it is possible to keep track of different data sources and the connection between them.

In summary, the custom workflows for *A. nidulans* are a proof-of-concept for our extensions to the Ondex framework. The user-defined context-sensitive menus provide new functionality that makes better use of existing features of Ondex. As the import of additional data is controlled by the user and can be limited to particular sub-networks, this approach helps to address the scalability problem when working with large datasets. Eventually, this procedure leads to a lower overall memory usage of Ondex. The implementation of the new functionality within Ondex emphasizes high cohesion, low coupling, and encapsulation, thus ensuring the re-usability of the code. Individual menu items can be seen as add-on elements, easily allowing the Ondex functionality to be extended in a modular way. An integrated editor allows easy implementation of new menus, which can be adapted for the specific data and analysis requirements.

Our extensions to the Ondex data integration and visualization framework improves its applicability for exploratory data analysis. The presented context-sensitive workflows extend the functionality of Ondex and helped to propose new interpretations of experimental data. Although the presented data integration scheme is tailored for the interpretation of the gene expression data in the context of secondary metabolite analysis, this framework and the presented workflows will benefit the analysis of other datasets. Data can be interactively visualized and additional data can be integrated on demand. Thus, the user is not limited to a pre-defined analysis workflow and to previously integrated data. At the same time, the advantages of computer-assisted data integration and visualizations are retained.

4. DISCUSSION

The quality of network inference models does not only need to be assessed with the help of quantitative models but the resulting network topology also needs to be evaluated qualitatively. Currently, the qualitative assessment requires in-depth expert knowledge about the components in the network model and about its dependence upon the experimental setting. Online resources providing static, pre-integrated knowledge, such as BiologicalNetworks (Kozhenkov et al., 2011) and GeneMania (Mostafavi et al., 2008), focus on widely-studied model organisms or require the upload of experimental in-house data. With the increasing number of sequenced organisms, we predict a further diversification of studied organisms and an increased need to create custom integration networks. Thus, the application and improvement of data integration and visualization software providing the possibility to compose integrated datasets using custom workflows according to user specifications is essential. Currently, the utilization of such tools is hampered by the challenge of proper data integration and visualization for large datasets. Here, we describe the extension of the data integration and visualization framework Ondex allowing the user to build context-sensitive workflows. The workflows described here are examples of an exploration of experimental results followed by a more detailed analysis, which have led to new hypotheses about the functions of currently unannotated genes. The strength of our approach is that it captures the essential information from a complex network of integrated publicly available data while the analysis can be individually tailored for each network region of interest in order to reduce the overall computational effort. In the manner of an exploratory data analysis, the workflow can be easily adjusted by the scientist to develop innovative research questions and identify patterns within the data that emerge from a combination of analysis and expert judgement. The provision of data integration functionalities with the help of pop-up menus is convenient and intuitive. This way, the graphical interface does not need additional separate windows and the researcher does not have to become acquainted with a specialized user interface, e.g., via a scripting language.

We used Ondex to integrate publicly-available key datasets for *A. nidulans*, which would otherwise be spread over different resources. The mapping of the information allows the data to be easily explored and visualized in an intuitive manner. This network can then serve as a scaffold for further integration of additional experimental data. The information connected to a gene locus helps the user to confirm predictions and generate hypotheses. In the case where comprehensive data about a gene set of interest is missing, context-sensitive menus can be applied for the prediction of gene functions. Therefore, Ondex can help to steer the selection of new experiments and define new directions for further investigation. This is the first integrative approach based on networks applied to *A. nidulans*. This study demonstrates the exploration of co-expressed gene clusters for secondary metabolite biosynthesis pathways. The exploratory analysis helped to link the data to other publications covering a fungal-bacteria interaction. It also enabled the identification and annotation of differentially expressed genes in the proximity of gene clusters. The uncharacterized gene AN7797.4 may be a transporter involved in the sterigmatocystin pathway, whereas the uncharacterized gene AN7899.4 may be part of the metabolic pathway of the orsellinic acid. It needs to be explored in further experiments whether and how these genes are directly involved in the regulation of these clusters.

Our proposed procedure to analyze gene expression data with the focus on fungal secondary metabolite gene clusters could have been performed without the assistance of the Ondex framework. In a traditional approach, we would have filtered the interesting differentially expressed genes in spreadsheets resulting from statistical analysis. Afterwards, additional information would be gathered using a web browser. Different resources such as genome browsers, the online InterProScan tool, the online BLAST tool, and the PubMed Central search interface need to be consulted. These research steps need to be performed in succession for each gene of interest separately while keeping in mind that each gene may have different gene identifiers (which is especially important for performing a full-text literature search). By providing these data integration functionalities through context-sensitive menus in the Ondex framework, the data interpretation procedure is sped up, it is more reproducible, and it helps to direct the researcher's focus on the data interpretation rather than the methodology of retrieving it. Another advantage of the approach to data integration supported by the Ondex framework is that it facilitates tracking of different sources of data and the path of reasoning and exploration. This would not be possible using web resources and their interfaces alone, which work mostly in a sequential, linear manner. All publicly available information about a gene can be consolidated within one graph, making the navigation easier and ensuring the best possible quality of data, as all relevant data can be efficiently collected. If the information about a gene locus is missing, the utility of Ondex to draw conclusions is limited and additional experimental data or bioinformatic methods are necessary to fill the gap. Thus, the completeness of the underlying functional annotation is of particular importance as it has a major impact on the subsequent interpretation of the dataset. In our example, it became apparent that most conclusions about the functional categorization of *A. nidulans* genes can be drawn from the Functional Catalog, which has already been successfully applied to fungal genes and proteins in other studies (Priebe et al., 2011). Ondex visualization ensured that additional information provided from GO was not disregarded at any point during analyses.

The standard procedure in Ondex is to integrate available data from different data sources prior to the visualization and data analysis. If a large amount of data is integrated, it results in large datasets which need to be handled and visualized by the Ondex software framework. Currently, the complexity of layout algorithms and the computational limitations, i.e., memory or CPU, make it challenging to manage the vast amount of data in a user-friendly and responsive manner. Additionally, if access to the most recent information is very important and the underlying data changes frequently, the time-consuming step of data integration has to be repeated regularly. If the data originates from user-made, computationally demanding calculations, a frequent data integration becomes computationally infeasible. Our extension of the Ondex framework overcomes these limitations by offering the option to apply these steps to a selected part of the network via the context-sensitive menus. Thus, the required amount of data is reduced, current data can be instantly downloaded from web resources, and intensive calculations need only be performed for subsets of the available data relevant to the current focus of investigation. Hence, the memory and computational load is reduced and access to the most recent data is guaranteed.

In this way, the context-sensitive menus make the interactive data analysis more efficient and user-friendly by providing data integration and filtering on-the-fly. The precise workflow of data analysis does not have to be established for the whole data integration process beforehand and the integration can be repeatedly applied and adjusted during the interactive analysis. This extension to the Ondex framework now combines the advantages of two data integration paradigms, i.e., of data warehousing and federated data integration, into one easy-to-use single system. The extensions to Ondex reported here have significantly improved its suitability for its usage for the qualitative assessment of inferred network models.

AUTHOR CONTRIBUTIONS

Fabian Horn initiated and led the work presented here, contributed as the main author to this manuscript and carried out the analysis. Martin Rittweger implemented the context-sensitive menus in Ondex and carried out the analysis. Jan Taubert supported the general implementation in Ondex and contributed to conceptualization and writing of the manuscript. Artem Lysenko supported the implementation with respect to Ondex scripting capabilities and provided feedback on the manuscript. Christopher Rawlings and Reinhard Guthke supervised the work at Rothamsted and the HKI, respectively, and provided feedback on the manuscript. All authors have read and acknowledged the manuscript.

FUNDING

This study was supported by the International Leibniz Research School for Microbial and Molecular Interactions (ILRS), as part of the excellence graduate school Jena School for Microbial Communication (JSMC), supported by the Deutsche Forschungsgemeinschaft. The work on Ondex has been funded by BBSRC Grants BBS/B/13640 and BB/F006039/1.

ACKNOWLEDGMENTS

Thanks to the curators of AspGD for providing the fully manually curated publication list. We would also like to thank the reviewers for their constructive comments that helped to improve the structure and readability of the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://www.frontiersin.org/journal/10.3389/fgene.2014.00021/abstract

Supplementary File S1 | Functional Annotation of Differentially Expressed Genes. A spreadsheet which contains the number of differentially expressed genes for each Functional Catalog (FunCat) category. In order to make the results more comparable to the original publication by Nahlik et al. (2010), the categories are merged. The merging rules and the resulting numbers are given.

Supplementary File S2 | List of Strongly Differentially Expressed Gene Clusters. For each growth stage, the clusters and their associated gene identifiers are listed (|fold-change| ≥8 and adjusted *P*-value ≤0.05). An annotation is given if the cluster is known or predicted to be part of fungal secondary metabolism.

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

Altwasser, R., Linde, J., Buyko, E., Hahn, U., and Guthke, R. (2012). Genome-wide scale-free network inference for *Candida albicans*. *Front. Microbiol.* 3:51. doi: 10.3389/fmicb.2012.00051

Arnaud, M. B., Chibucos, M. C., Costanzo, M. C., Crabtree, J., Inglis, D. O., Lotia, A., et al. (2010). The Aspergillus genome database, a curated comparative genomics resource for gene, protein and sequence information for the Aspergillus research community. *Nucleic Acids Res.* 38, D420–D427. doi: 10.1093/nar/gkp751

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., et al. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.* 25, 25–29. doi: 10.1038/75556

Bebek, G., Koyutürk, M., Price, N. D., and Chance, M. R. (2012). Network biology methods integrating biological data for translational science. *Brief Bioinform.* 13, 446–459. doi: 10.1093/bib/bbr075

Benjamini, Y., and Yekutieli, D. (2001). The control of the false discovery rate in multiple testing under dependency. *Ann. Stat.* 29, 1165–1188. doi: 10.1214/aos/1013699998

Brakhage, A. A. (2013). Regulation of fungal secondary metabolism. *Nat. Rev. Microbiol.* 11, 21–32. doi: 10.1038/nrmicro2916

Brakhage, A. A., and Schroeckh, V. (2011). Fungal secondary metabolites - strategies to activate silent gene clusters. *Fungal Genet. Biol.* 48, 15–22. doi: 10.1016/j.fgb.2010.04.004

Brown, D. W., Yu, J. H., Kelkar, H. S., Fernandes, M., Nesbitt, T. C., Keller, N. P., et al. (1996). Twenty-five coregulated transcripts define a sterigmatocystin gene cluster in *Aspergillus nidulans*. *Proc. Natl. Acad. Sci. U.S.A.* 93, 1418–1422. doi: 10.1073/pnas.93.4.1418

Cantone, I., Marucci, L., Iorio, F., Ricci, M. A., Belcastro, V., Bansal, M., et al. (2009). A yeast synthetic network for *in vivo* assessment of reverse-engineering and modeling approaches. *Cell* 137, 172–181. doi: 10.1016/j.cell.2009.01.055

Cockell, S. J., Weile, J., Lord, P., Wipat, C., Andriychenko, D., Pocock, M., et al. (2010). An integrated dataset for in silico drug discovery. *J. Integr. Bioinform.* 7, 116. doi: 10.2390/biecoll-jib-2010-116

David, H., Ozçelik, I. S., Hofmann, G., and Nielsen, J. (2008). Analysis of *Aspergillus nidulans* metabolism at the genome-scale. *BMC Genomics* 9:163. doi: 10.1186/1471-2164-9-163

Fazius, E., Shelest, V., and Shelest, E. (2011). SiTaR: a novel tool for transcription factor binding site prediction. *Bioinformatics* 27, 2806–2811. doi: 10.1093/bioinformatics/btr492

Flicek, P., Ahmed, I., Amode, M. R., Barrell, D., Beal, K., Brent, S., et al. (2013). Ensembl 2013. *Nucleic Acids Res.* 41, D48–D55. doi: 10.1093/nar/gks1236

Guthke, R., Möller, U., Hoffmann, M., Thies, F., and Töpfer, S. (2005). Dynamic network reconstruction from gene expression data applied to immune response during bacterial infection. *Bioinformatics* 21, 1626–1634. doi: 10.1093/bioinformatics/bti226

Hassani-Pak, K., Legaie, R., Canevet, C., van den Berg, H. A., Moore, J. D., and Rawlings, C. J. (2010). Enhancing data integration with text analysis to find proteins implicated in plant stress response. *J. Integr. Bioinform.* 7, 121. doi: 10.2390/biecoll-jib-2010-121

Hecker, M., Lambeck, S., Toepfer, S., van Someren, E., and Guthke, R. (2009). Gene regulatory network inference: data integration in dynamic models-a review. *Biosystems* 96, 86–103. doi: 10.1016/j.biosystems.2008.12.004

Horn, F., Heinekamp, T., Kniemeyer, O., Pollmächer, J., Valiante, V., and Brakhage, A. A. (2012). Systems biology of fungal infection. *Front. Microbiol.* 3:108. doi: 10.3389/fmicb.2012.00108

Horn, F., Nützmann, H.-W., Schroeckh, V., Guthke, R., and Hummert, C. (2011). "Optimization of a microarray probe design focusing on the minimization of cross-hybridization," in *Proceedings of the International Conference on Bioinformatics and Computational Biology (BIOCOMP'11)*, Vol. 1, eds H. R. Arabnia and Q.-N. Tran (Las Vegas, NV: CSREA Press), 3–9. ISBN: 1-60132-172-4.

Hull, D., Wolstencroft, K., Stevens, R., Goble, C., Pocock, M. R., Li, P., et al. (2006). Taverna: a tool for building and running workflows of services. *Nucleic Acids Res.* 34, W729–W732. doi: 10.1093/nar/gkl320

Huttenhower, C., and Hofmann, O. (2010). A quick guide to large-scale genomic data mining. *PLoS Comput. Biol.* 6:e1000779. doi: 10.1371/journal.pcbi.1000779

Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M. (2012). KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res.* 40, D109–D114. doi: 10.1093/nar/gkr988

Kelder, T., Conklin, B. R., Evelo, C. T., and Pico, A. R. (2010). Finding the right questions: exploratory pathway analysis to enhance biological discovery in large datasets. *PLoS Biol.* 8:e1000472. doi: 10.1371/journal.pbio.1000472

Köhler, J., Baumbach, J., Taubert, J., Specht, M., Skusa, A., Rüegg, A., et al. (2006). Graph-based analysis and visualization of experimental results with Ondex. *Bioinformatics* 22, 1383–1390. doi: 10.1093/bioinformatics/btl081

Kinsella, R. J., Kähäri, A., Haider, S., Zamora, J., Proctor, G., Spudich, G., et al. (2011). Ensembl BioMarts: a hub for data retrieval across taxonomic space. *Database (Oxford)* 2011, bar030. doi: 10.1093/database/bar030

Kozhenkov, S., Sedova, M., Dubinina, Y., Gupta, A., Ray, A., Ponomarenko, J., et al. (2011). Biologicalnetworks–tools enabling the integration of multi-scale data for the host-pathogen studies. *BMC Syst. Biol.* 5:7. doi: 10.1186/1752-0509-5-7

Linde, J., Hortschansky, P., Fazius, E., Brakhage, A. A., Guthke, R., and Haas, H. (2012). Regulatory interactions for iron homeostasis in *Aspergillus fumigatus* inferred by a systems biology approach. *BMC Syst. Biol.* 6:6. doi: 10.1186/1752-0509-6-6

Linde, J., Wilson, D., Hube, B., and Guthke, R. (2010). Regulatory network modelling of iron acquisition by a fungal pathogen in contact with epithelial cells. *BMC Syst. Biol.* 4:148. doi: 10.1186/1752-0509-4-148

Lysenko, A., Hindle, M. M., Taubert, J., Saqi, M., and Rawlings, C. J. (2009). Data integration for plant genomics–exemplars from the integration of *Arabidopsis thaliana* databases. *Brief Bioinform.* 10, 676–693. doi: 10.1093/bib/bbp047

Magrane, M., and UniProt Consortium (2011). UniProt Knowledgebase: a hub of integrated protein data. *Database (Oxford)* 2011, bar009. doi: 10.1093/database/bar009

Mostafavi, S., Ray, D., Warde-Farley, D., Grouios, C., and Morris, Q. (2008). Genemania: a real-time multiple association network integration algorithm for predicting gene function. *Genome Biol.* 9, S4. doi: 10.1186/gb-2008-9-s1-s4

Nahlik, K., Dumkow, M., Bayram, O., Helmstaedt, K., Busch, S., Valerius, O., et al. (2010). The COP9 signalosome mediates transcriptional and metabolic response to hormones, oxidative stress protection and cell wall rearrangement during fungal development. *Mol. Microbiol.* 78, 964–979. doi: 10.1111/j.1365-2958.2010.07384.x

Newman, D. J., and Cragg, G. M. (2012). Natural products as sources of new drugs over the 30 years from 1981 to 2010. *J. Nat. Prod.* 75, 311–335. doi: 10.1021/np200906s

Nützmann, H.-W., Reyes-Dominguez, Y., Scherlach, K., Schroeckh, V., Horn, F., Gacek, A., et al. (2011). Bacteria-induced natural product formation in the fungus *Aspergillus nidulans* requires Saga/Ada-mediated histone acetylation. *Proc. Natl. Acad. Sci. U.S.A.* 108, 14282–14287. doi: 10.1073/pnas.1103523108

Pavlopoulos, G. A., Wegener, A.-L., and Schneider, R. (2008). A survey of visualization tools for biological network analysis. *Biodata Min.* 1, 12. doi: 10.1186/1756-0381-1-12

Priebe, S., Linde, J., Albrecht, D., Guthke, R., and Brakhage, A. A. (2011). FungiFun: a web-based application for functional categorization of fungal genes and proteins. *Fungal Genet. Biol.* 48, 353–358. doi: 10.1016/j.fgb.2010.11.001

Rao, C., Toutenburg, H., and Schomaker, M. (2008). *Linear models and generalizations: least squares and alternatives*. Springer series in statistics. Berlin: Springer.

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., et al. (2004). The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Res.* 32, 5539–5545. doi: 10.1093/nar/gkh894

Sanchez, J. F., Somoza, A. D., Keller, N. P., and Wang, C. C. C. (2012). Advances in Aspergillus secondary metabolite research in the post-genomic era. *Nat. Prod. Rep.* 29, 351–371. doi: 10.1039/c2np00084a

Scharf, D. H., Heinekamp, T., Remme, N., Hortschansky, P., Brakhage, A. A., and Hertweck, C. (2012). Biosynthesis and function of gliotoxin in *Aspergillus fumigatus*. *Appl. Microbiol. Biotechnol.* 93, 467–472. doi: 10.1007/s00253-011-3689-1

Schroeckh, V., Scherlach, K., Nützmann, H.-W., Shelest, E., Schmidt-Heck, W., Schuemann, J., et al. (2009). Intimate bacterial-fungal interaction triggers biosynthesis of archetypal polyketides in *Aspergillus nidulans*. *Proc. Natl. Acad. Sci. U.S.A.* 106, 14558–14563. doi: 10.1073/pnas.0901870106

Shimizu, M., Fujii, T., Masuo, S., and Takaya, N. (2010). Mechanism of de novo branched-chain amino acid synthesis as an alternative electron sink in hypoxic *Aspergillus nidulans* cells. *Appl. Environ. Microbiol.* 76, 1507–1515. doi: 10.1128/AEM.02135-09

Smyth, G. K., and Speed, T. P. (2003). Normalization of cDNA microarray data. *Methods* 31, 265–273. doi: 10.1016/S1046-2023(03)00155-5

Taubert, J., Sieren, K. P., Hindle, M., Hoekman, B., Winnenburg, R., Philippi, S., et al. (2007). The OXL format for the exchange of integrated datasets. *J. Integr. Bioinform.* 4, 62. doi: 10.2390/biecoll-jib-2007-62

Töpfer, S., Guthke, R., Driesch, D., Woetzel, D., and Pfaff, M. (2006). "The NetGenerator algorithm: reconstruction of gene regulatory networks," in *KDECB*, Lecture notes in computer science, Vol. 4366, eds K. Tuyls, R. L. Westra, Y. Saeys, and A. Nowé (Berlin: Springer), 119–130.

Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley Series in Behavioral Science. Reading, MA: Addison-Wesley Publishing Company.

Tukey, J. W. (1980). We need both exploratory and confirmatory. *Am. Stat.* 34, 23–25. doi: 10.1080/00031305.1980.10482706

Ukkonen, E. (1995). On-line construction of suffix-trees. *Algorithmica* 14, 249–260. doi: 10.1007/BF01206331

Walton, J. D. (2000). Horizontal gene transfer and the evolution of secondary metabolite gene clusters in fungi: an hypothesis. *Fungal Genet. Biol.* 30, 167–171. doi: 10.1006/fgbi.2000.1224

Weber, M., Henkel, S. G., Vlaic, S., Guthke, R., van Zoelen, E. J., and Driesch, D. (2013). Inference of dynamical gene-regulatory networks based on time-resolved multi-stimuli multi-experiment data applying NetGenerator v2.0. *BMC Syst. Biol.* 7:1. doi: 10.1186/1752-0509-7-1

Whitlock, M. C. (2005). Combining probability from independent tests: the weighted Z-method is superior to Fisher's approach. *J. Evol. Biol.* 18, 1368–1373. doi: 10.1111/j.1420-9101.2005.00917.x

Zdobnov, E. M., and Apweiler, R. (2001). InterProScan–an integration platform for the signature-recognition methods in InterPro. *Bioinformatics* 17, 847–848. doi: 10.1093/bioinformatics/17.9.847

Conflict of Interest Statement: 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.

Received: 26 July 2013; paper pending published: 15 August 2013; accepted: 21 January 2014; published online: 20 February 2014.

Citation: Horn F, Rittweger M, Taubert J, Lysenko A, Rawlings C and Guthke R (2014) Interactive exploration of integrated biological datasets using context-sensitive workflows. Front. Genet. ***5****:21. doi: 10.3389/fgene.2014.00021*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

METHODS ARTICLE
published: 26 February 2014
doi: 10.3389/fgene.2014.00040

Modular network construction using eQTL data: an analysis of computational costs and benefits

Yen-Yi Ho[1]*, *Leslie M. Cope*[2] and *Giovanni Parmigiani*[3]

[1] Division of Biostatistics, School of Public Health, University of Minnesota, Minneapolis, MN, USA
[2] The Sidney Kimmel Cancer Center, Johns Hopkins School of Medicine, Baltimore, MD, USA
[3] Dana-Farber Cancer Institute and Harvard School of Public Health, Boston, MA, USA

Edited by:
Benjamin Haibe-Kains, Princess Margaret Cancer Centre, Canada

Reviewed by:
Mikhail P. Ponomarenko, Institute of Cytology and Genetics of Siberian Branch of Russian Academy of Sciences, Russia
Erick Antezana, Norwegian University of Science and Technology, Norway
Jaume Bacardit, Newcastle University, UK

***Correspondence:**
Yen-Yi Ho, Division of Biostatistics, School of Public Health, University of Minnesota, 420 Delaware Street, S.E., Mayo Building, MMC 303, Minneapolis, MN 55403, USA
e-mail: yho@umn.edu

Background: In this paper, we consider analytic methods for the integrated analysis of genomic DNA variation and mRNA expression (also named as eQTL data), to discover genetic networks that are associated with a complex trait of interest. Our focus is the systematic evaluation of the trade-off between network size and network search efficiency in the construction of these networks.

Results: We developed a modular approach to network construction, building from smaller networks to larger ones, thereby reducing the search space while including more variables in the analysis. The goal is achieving a lower computational cost while maintaining high confidence in the resulting networks. As demonstrated in our simulation results, networks built in this way have low node/edge false discovery rate (FDR) and high edge sensitivity comparing to greedy search. We further demonstrate our method in a data set of cellular responses to two chemotherapeutic agents: docetaxel and 5-fluorouracil (5-FU), and identify biologically plausible networks that might describe resistances to these drugs.

Conclusion: In this study, we suggest that guided comprehensive searches for parsimonious networks should be considered as an alternative to greedy network searches.

Keywords: Bayesian networks, search algorithm, network variable selection, eQTL, chemotherapy resistance

1. INTRODUCTION

Beginning with work by Schadt et al. (2005), a number of recent studies combine SNP datasets with transcriptional, metabolomic or other data to develop network models for common diseases that link response to treatment (Chen et al., 2008; Schadt, 2009; Chang and McGeachie, 2011). Schadt describes the principle behind this approach: "In the context of common human diseases, the disease states can be considered emergent properties of molecular networks, as opposed to the core biological processes associated with a disease being driven by responses to changes in a small number of genes" (Schadt, 2009). These methods have proved effective in several practical settings (Pe'er et al., 2001; Mehrabian et al., 2005; Zhu et al., 2007; Chen et al., 2008; Yang et al., 2009) but there are open problems and overcoming the computational difficulties associated with high dimensional data analysis is of particular interest. Approaches commonly used to manage the computational burden include reducing the number of genes by pre-filtering based on gene function or the results of univariate analysis, (Imoto et al., 2003; Li et al., 2005; Chang and McGeachie, 2011), and improving the efficiency of the search for solutions, for instance by using greedy algorithms (Friedman et al., 2000; Yu et al., 2002; Teyssier, 2005).

Most recently, hybrid approaches like the H2PC algorithm (Gasse et al., 2012) combine the greedy hill-climbing step with a constraint-based optimization, although these have not yet been adapted for use on a mixture of continuous and discrete variables, limiting their applicability to networks incorporating several types of genomic data. Others have incorporated transcription-factor, or protein–protein binding information from biological knowledge bases to improve gene network inference. The GRAM algorithm (Bar-Joseph et al., 2003), as well as the approaches by Xu et al. (2004), and Tu et al. (2006) are representative of this strategy. Alternatively, other approaches for studying genetic networks consider only pairwise relationships such as correlation or partial correlations (Zhang and Horvath, 2005; Lasserre et al., 2013). These approaches investigate the association between pairs of genes, and hence do not consider the directionality of an edge.

In this study, we plot a unique course suggested to us by Schadt's use of SNP-transcript-phenotype trios in causal analysis (Schadt et al., 2005), wherein we build the causal network up modularly from smaller, data-driven network components. Here *network* is used in the sense of Bayesian networks, our tool of choice for describing the dependence structure between variables. At the most basic level, this can be thought of as a strategy for selecting the most informative genomic and transcriptomic sites for use in network models. Although they did not incorporate the philosophy into variable selection, Pe'er et al. (2001) also emphasized the value of basing network inferences on small but high-confidence subnetworks: "We hypothesize that if we can find a subnetwork ... with a relatively high confidence, then our estimate of edges and other features in this region will be more reliable. While a full-scale network is currently of insufficient quality, statistically significant sub-networks can be reconstructed. Indeed, such subnetworks often correspond to biologically meaningful relations between genes" (Pe'er et al., 2001). The goal is to strike a balance between the high computational costs of large scale network analysis, on the one hand, and the loss of information contained in the data necessitated by aggressive pre-filtering steps and greedy approaches to network development on the other. We are looking for an equilibrium point where component networks are small enough that searching through them is computationally feasible but large enough to capture important network substructures.

We propose a network-driven feature selection strategy, whereby sets of variables are chosen on the basis of their role in small subnetworks, and then iteratively assembled into larger structures. To investigate the utility of this approach, referred to as nPARS for **network Partition and Reassembly Search**, we evaluate it in an extensive set of biologically plausible simulations, comparing it to a gold standard exhaustive search for a best fitting network as well as the commonly-used greedy hill-climbing algorithm. We also demonstrate our proposed approach in a data set of cellular responses to two chemotherapeutic agents: docetaxel and 5-fluorouracil (5-FU) and discuss possible extensions.

2. METHODS

2.1. BAYESIAN NETWORKS FOR GENETIC NETWORK DISCOVERY

We chose Bayesian networks to represent the widely used class of network models that aim to capture the dependence structure in a dataset. A particularly attractive feature of Bayesian networks is their ability to accommodate genomic data of various types by using continuous or discrete nodes to represent variables under consideration, for example: continuous nodes to represent continuous measurements such as gene expression, and discrete nodes to represent discrete variable such as genotype.

Given a Bayesian network structure, the approach to calculate likelihood and network score has been well-established in the literature. The novelty of this paper is to introduce the nPARS search algorithm, described in section 2.2, to guide the search process and to visit parts of the network space that reflect parts of the true underlying network structure in a given data, since the search space is oftentimes enormous. Formally, a Bayesian network is a graphical representation of the joint distribution of a set of variables (Pearl, 1988) consisting of two components: (1) a directed acyclic graph in which nodes correspond to random variables, and directed edges to dependencies between variables; for example, L → E indicates that the status at node L is associated with the alteration of status of node E. And (2) the joint distribution of the random variables decomposed according to the graphical model, under an assumption of Markov conditional independence.

Thus the dependence structure can be described as $P\left(X_1, X_2, ..., X_p|G\right) = \prod_i^p P\left(X_i|Pa(X_i), G\right)$, where $Pa(X_i)$ represents the parents of nodes X_i in graph G. The conditional distributions in the described equation were specified according to the types (discrete or continuous) of X_i. For discrete nodes, we assume they follow multinomial distribution with parameter θ_d and the prior distribution of θ_d follows Dirichlet. For continuous nodes, we assume linear Gaussian conditional densities given the value of its parents and apply Gaussian-inverse gamma priors. For example, assuming a continuous node, X_i, has both continuous parents (Pa_c)and discrete parents (Pa_d), we apply the following distribution model:

$$P\big(X_i|Pa_c(X_i) = u, Pa_d(X_i) = j\big) = N(m_j + \beta_j.u,\ \sigma^2),$$

$$\big(m_j,\ \beta_j|\sigma_j\big) \sim N(\mu_j,\ \sigma_j\tau_j^{-1}),$$

$$\sigma_j \sim I\Gamma\left(\frac{\rho_j}{2}, \frac{\phi_j}{2}\right).$$

Given a network structure, the likelihood function and network score can be found in Bøttcher and Dethlefsen (2003, pp. 3–6, 11–12). We follow Bøttcher and Dethlefsen's implementation of Bayesian networks and also refer the reader to these publications (Friedman et al., 2000; Bøttcher and Dethlefsen, 2003; Bøttcher, 2004) for a complete discussion of Bayesian networks and the software (Bøttcher and Dethlefsen, 2003) we used to fit and analyze the data.

2.1.1. Ranking network structures

All else equal, the best fitting network model can be identified by maximizing the log posterior probability of the network G given the data d, herein called the *network score* and denoted

$$S(G) = \log P(G|d) \propto \log P(d|G) + \log P(G). \quad (1)$$

In the simple example shown in **Figure 1**, nodes corresponding to SNP markers are denoted by L, expression by E and the disease, or phenotypic outcome by D. The SNPs, being discrete variables, are shown with a black background in the graphical network representation while the continuous nodes are shown in white. Assuming that gene expression level or phenotypic status could not change SNP genotypes, we restrict the possible network structures so that no edges come from the expression and phenotypic nodes to the SNP node at locus L, leaving a total of 12 possible DAGs that can be generated from the triplet $\{L, E, D\}$. In this example, the best fitting network structure for the $\{L, E, D\}$ triplet will turn out to be G_{10} with $S = -5558.75$.

We have made a few adaptions to the likelihood-based network score $S(G)$ to address certain practical concerns. When comparing network structures with different sets of nodes, and especially different numbers of nodes, the network scores of Equation (1) may be on different scales. And, all else equal, we prefer a network in which molecular variables are strongly associated with the phenotype D over one with very tight molecular associations but weak correlation with outcome. To achieve these goals, we define the average network improvement score (φ):

$$\varphi = \frac{\lambda(S - S_0) + (1 - \lambda)S}{\alpha}. \quad (2)$$

where S is the network score of the structure under consideration, and S_0 is the network score of its corresponding *null* network, obtained by removing the edge(s) to "D". For example, for network structure G_{10} in **Figure 1**, the null network is G_2. In addition, λ is a tuning parameter between 0 and 1, and α is the number of nodes considered in the network.

The quantity (S-S0) measures the improvement in the network score resulting from adding an edge to "*D*". The numerator of φ is a weighted average of these two parts: the difference $(S - S_0)$ and the network score *S*. In addition, the tuning parameter, λ is used to adjust the weight of the two parts. To weight the two parts equally, we set $\lambda = 0.5$ in the following analysis.

To adjust for the number of nodes in network scores, we divide the numerator of φ by the number of nodes. This is used as a simple approximation of the effect of the number of nodes in the marginal likelihood. From Equation (1), when considering networks with no edges and assuming the nodes have the same distribution, the log marginal likelihood decreases linearly when adding nodes in the model, providing a heuristic justification for our specification.

The φ score so defined favors network structures that have both high posterior support and strong association with the phenotypic outcome. Using the previous example, the network score of G_{10} is −5558.75, and the score of its corresponding null graph, G_2 is −5757.38. Hence, $\varphi = \frac{0.5[-5558.75-(-5757.38)]+(0.5\times -5558.75)}{3} = -893.35$.

2.2. BUILDING NETWORKS

Our motivating hypothesis is that a network built on genomic sites and transcripts shown to be important in smaller network structures will be both accurate and computationally efficient. Accordingly we took a triplet—a SNP genotype taken together with an expression measure, and the phenotypic outcome—to be the basic building "module" in nPARS with larger networks formed by merging candidate triplets. The process can be divided into three main steps: (1) construct and score all triplets, (2) select the most informative of the resulting subnetworks, and (3) assemble these into larger networks. We will describe each of these in a little more detail in the next paragraph.

2.2.1. Constructing three-node subnetworks

To form the basic building blocks, we decompose the whole network space into all possible (*L*, *E*, *D*) triplets, calculating a network score and best fitting structure for each. For the data set described in section 4, there are a total of $2330 \times 3554 = 8,280,820$ (*L*, *E*, *D*) triplets, and for each we find the network structures with the best network scores, as described above.

2.2.2. Selection

Triplets are selected on the basis of the biological relevance of their best fitting network structures as well as the network scores. We exclude any (*L*, *E*, *D*) subnetworks containing a node of degree zero (having no connections with other nodes), so that only adequately connected networks are admitted for further analysis. Thus we select the subnetworks with structures shown as G_6, G_7, ..., G_{12} in **Figure 1**. Next, we apply the average network improvement score (φ) to select the subnetworks that have both large support from the posterior and significant relevance to the outcome of interest. Subnetworks are ranked according to the φ scores. We then choose the top k_1 subnetworks for further analysis. It is possible that after this step, there is only one (*L*, *E*, *D*) left. In this case, the algorithm reports this single three-node network. Our search in this step is exhaustive, which we find to be a significant strength of our approach.

2.2.3. Reassembly

The final step is to build larger network structures from the chosen triplets. In doing this we considered first that the larger

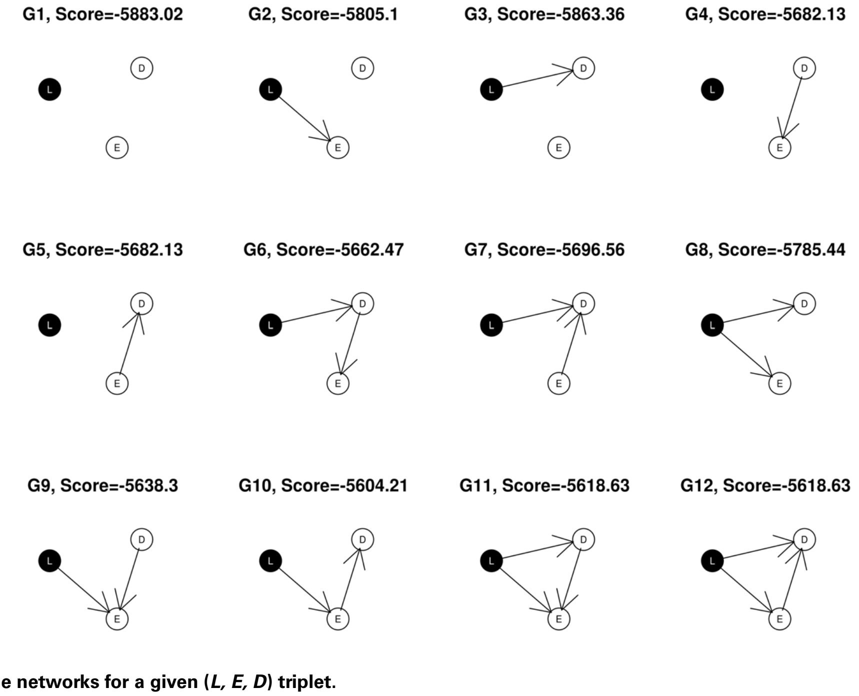


FIGURE 1 | All 12 possible networks for a given (*L, E, D*) triplet.

networks should contain two or more complete triplets, rather than mix and match individual nodes from different triplets, in order to preserve information that may be held jointly in those variables. Secondly, it should be permissible to reconstruct edges within triplets in addition to adding connections between triplets. These two considerations thus defined the assembly process, wherein a new Bayesian network is built *from scratch* using the nodes from a set of triplets. In our tests we assembled every pair of high scoring (L, E, D) triplets into four to five node networks, and used an exhaustive search to find the top scoring structure for each. We then build larger networks sequentially, adding additional triplets to the best scoring five node networks. At some point, as the networks get larger the exhaustive search option becomes computationally infeasible. This in fact happens fairly early, but we anticipate that the improved variable selection afforded by the modular approach would continue to pay dividends even if a greedy algorithm were used to construct edges at later stages in the assembly.

We summarize the three steps in the nPARS algorithm: construction of subnetworks, selection, and assembly as follows:

1. **Construction of subnetworks**:
   - (a) Partition the whole network space into (L, E, D) subsets and
   - (b) construct subnetworks.
2. **Selection**: Select the subnetworks with
   - (a) network structures that are among G_6, ..., G_{12} in **Figure 1** and
   - (b) top k_1 subnetworks with largest φ scores.
3. **Reassembly**:
   - (a) Assemble two or more subnetworks into the union of their nodes;
   - (b) Re-construct the assembled networks by scoring all possible network structures with the set of nodes.
   - (c) Report the top k_2 networks with largest φ scores.

The diagram shown in **Figure 2** is a simple example for the nPARS algorithm described above. In **Figure 2**, eight triplet combinations are generated from four SNP loci (L_1, L_2, L_3, and L_4), and two expression measurements (E_1 and E_2). These three-node subnetworks are considered the basic building blocks (modules) of the nPARS algorithm. In the second selection step, three subnetworks are selected and a larger six-node network is re-constructed *from scratch* using the nodes from the selected subnetworks. In this example, L_4 does not enter into the final reassembly step, since in the first step, the subnetworks associated with L_4 do not connect with any expression (E) and phenotype (D).

3. TESTING

To rigorously evaluate performance of our network partition and assembly approach (nPARS), we simulated a set of plausible gene networks, comparing our partition and assembly approach to an exhaustive (Exh) search on the one hand and a greedy search with random restarts on the other (Greedy) . These algorithms are evaluated by comparing the reported final network structures to the assumed true network structure, to determine how frequently the correct nodes and edges are recovered. In these simulations we intentionally evaluate small networks, concentrating on the four- and five-node structures that result from joining two triplets. There are two reasons for this: (1) The exhaustive search for a best fitting network structure, which represents the gold standard of performance in these simulations, quickly becomes computationally prohibitive as a network gets larger. (2) We hope to model biologically plausible gene systems and to understand how features of those systems affect performance, and are not confident that human intuition is scalable in these regards.

3.1. SIMULATION SETTINGS

To examine performance, we investigate seven simulated network structures, shown in **Figure 3**. Some of these scenarios are observed during the experimental data analysis presented in section 4 and others are developed from biological theory. For example, scenarios 1 and 2 are constructed based on the fundamental dogma of gene expression: DNA → RNA → phenotype. In scenario 3, 4, and 7, we add direct edges from L to D in keeping with structures identified in the course of analyzing experimental data. In addition, in scenario 5 and 6, we examine network structures with long connections ($L \rightarrow E_1 \rightarrow E_2 \rightarrow D$). Scenario 7 could be considered as the worse case scenario because SNP loci contribute directly to D without alteration gene expression levels.

When simulating data, in order to mimic real world situations, we add unrelated SNP markers and expression measures as noise. The simulated data sets contain five SNP markers, five expression measures, and one continuous disease outcome. We simulate the data in the following four sample sizes: 100, 200, 500, 1000. SNP markers are simulated to have genotypes aa, Aa, and AA, with probability 0.25, 0.5, 0.25, respectively. Gene expression values from independent transcripts are simulated as $N(10, \sqrt{3.6})$. Expression values (E_i) with edge effect β, for example from L_i are generated using the linear regression model: $E_i = 8 + \beta \cdot L_i + \epsilon_i$, $\epsilon_i \sim N(0, \sqrt{3.6})$. Phenotypic outcomes (D) are then generated based on genotype, and expression values through another linear regression model.

Specifically, we generate the simulated data using the following models: in scenario 1 and 2, $D = \frac{\beta}{2} \cdot E_1 + \frac{\beta}{2} \cdot E_2 + \epsilon_i$; in scenario 3, $D = \beta \cdot I(L_1 = 1) + \beta I(L_1 = 2) + \beta \cdot E_1 + \beta \cdot E_2 + \epsilon_i$; in scenario 4, $D = \beta \cdot I(L_1 = 1) + \beta I(L_1 = 2) + \beta \cdot E_1 + \beta \cdot E_2 + \epsilon_i$; in scenario 5, $D = \frac{\beta}{2} \cdot E_2 + \epsilon_i$; in scenario 6, $D = 3\beta \cdot E_2 + \epsilon_i$, and in scenario 7, $D = \beta \cdot I(L_1 = 1) + \beta I(L_1 = 2) + \beta \cdot I(L_2 = 1) + \beta I(L_2 = 2) + \beta \cdot E_1 + \beta \cdot E_2 + \epsilon_i$, where I is the indicator function. In the above equations, all ϵ_i are generated from $N(0, \sqrt{3.6})$. We evaluate the performance of each of the three algorithms for various β values.

3.2. COMPARISON OF NODE RECOVERY

3.2.1. Algorithms

Three algorithms are implemented in this simulation study: nPARS, Exh, and Greedy. We apply nPARS as described previously. Specifically, in the selection step we keep all the subnetworks with more than one edge. In the final assembly step, we report the top 1 scoring network structure.

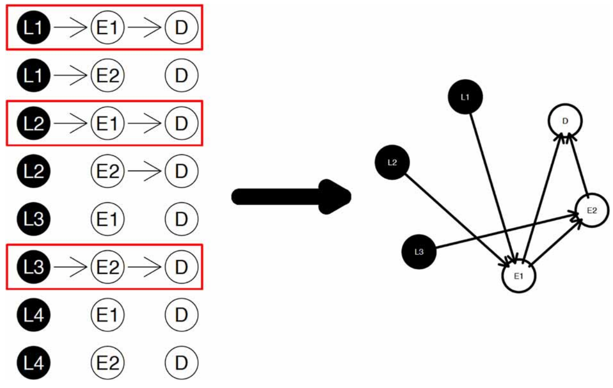


FIGURE 2 | The three steps in the nPARS algorithm: (1) constructing three-node subnetworks, (2) select subnetworks, and (3) reassemble into larger networks.

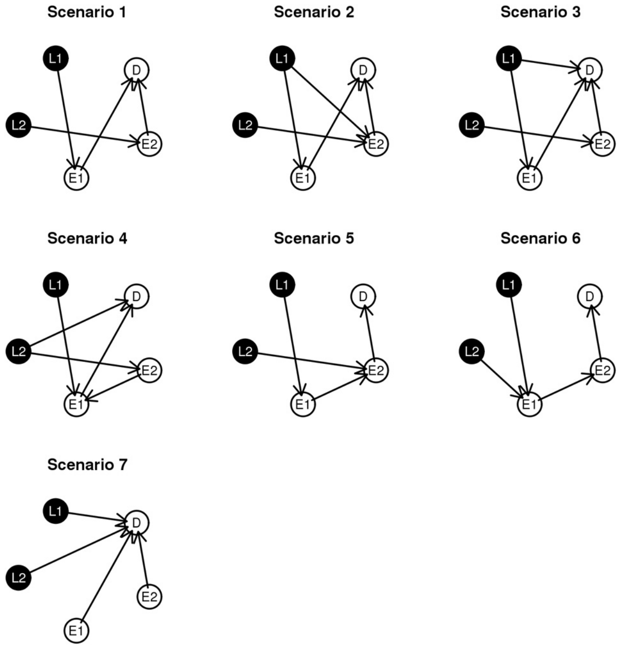


FIGURE 3 | Graphical representation of the simulation scenarios.

For comparison, in Exh, we define the network space to be all network structures that can be generated by all possible $\{L_1, L_2, E_1, E_2, D\}$ five-node combinations, and exhaustively score all of them reporting the network with the largest φ score. In the simulation, we perform greedy search with 10 random restarts, stopping when the network score converges or when the algorithm reaches 100 iterations.

3.2.2. Evaluation

Our first aim in the simulation analysis is to investigate whether nPARS recovers the correct nodes. For each true five-node network structure, we categorize the "nodes" in the final reported network structure as true positive (tp), false positive (fp) or false negative (fn) and evaluate the recovery of nodes using Sensitivity $= \frac{\text{tp}}{\text{tp+fn}}$, and FDR $= \frac{\text{fp}}{\text{tp+fp}}$.

3.2.3. Results

The comparisons of node recovery are shown in **Figures 4–6**. In all simulation scenarios, nPARS (black line) exhibits slightly lower node sensitivity than Exh (red line) when sample size is the same. In addition, nPARS demonstrates lower node false discovery rate (FDR) than Exh and Greedy (green line) in all seven scenarios.

In addition, Greedy demonstrates highest sensitivity but also relatively large FDR in all simulation scenarios. In all simulation scenarios, Greedy reports networks with edge connections between almost all the nodes in the data. There are 6 out of 11 (54.5%) false nodes in the simulation dataset, and the average node FDR of Greedy search is 54.4%($\frac{0.544}{0.545} \approx$ 99.8%) when sample size (n) is less than 1000. In other words, Greedy falsely recovered 99.8% of the false nodes in the simulation dataset when n is less than 1000. This number decreases to 51.7%($\frac{0.517}{0.545} \approx$ 94.9%) when $n = 1000$.

3.3. COMPARISON OF EDGE RECOVERY

3.3.1. Algorithms

The nPARS and Exh algorithms are implemented as described in section 3.2. However, as our findings from node recovery indicate, Greedy search often reports networks with too many nodes, and thus achieves high edge sensitivity at the price of a high number of false positive nodes. Hence, for edge the recovery comparison, it is desirable to control the number of nodes. In this analysis, we restrict the search space of the Greedy algorithm to network structures with no more than five nodes by adding an additional stopping rule requiring that when the network reaches five-nodes it stops. We call the revised version, GreedyE. As above, we categorize edges into tp, fp, fn, and calculate edge sensitivity and edge FDR based on the assumed true network.

3.3.2. Results

In most of the studied scenarios, nPARS has better performance than GreedyE in terms of edge sensitivity, as shown in **Figures 7–9**, given the same sample size. The exceptions occur in a few instances in scenario 1, 3, and 7, when the edge effect β is small. When β is increased in scenarios 3 and 7, nPARS tends to have better edge sensitivity compared to GreedyE. In scenario 1, nPARS appears to have similar edge sensitivity compared to GreedyE. Exh has the best edge sensitivity recovery in almost all the scenarios.

In terms of edge FDR, GreedyE demonstrates the highest edge FDR in all simulation scenarios. nPARS shows similar edge FDR compare to Exh except in scenarios 1 and 2, when β is relatively small. In general, when considering both edge sensitivity and FDR, nPARS often demonstrates better edge sensitivity with the benefit of lower edge FDR compare to GreedyE. Exh has the best performance, however, in practice it is not feasible to implement Exh.

Overall, in the comparison with Greedy search, nPARS demonstrates lower FDR in both node and edge recoveries. In the comparison with Exh, nPARS demonstrates similar FDR in both

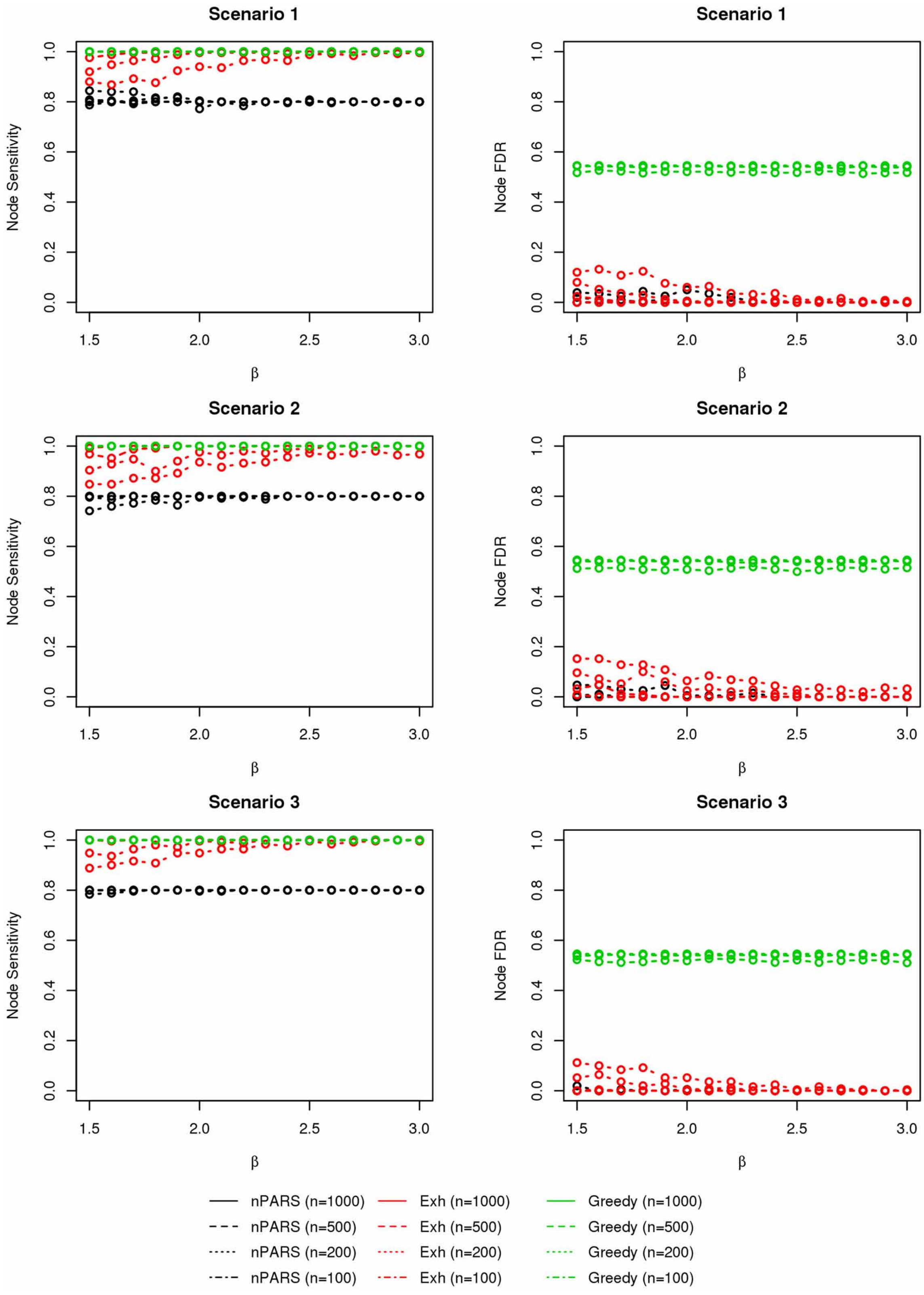


FIGURE 4 | The sensitivity and FDR of node comparisons in simulation scenario 1–3. nPARS shows lowest node FDR with slightly lower node sensitivity compare to exhaustive search. Although greedy search exhibits highest node sensitivity but it reports relatively high node FDR.

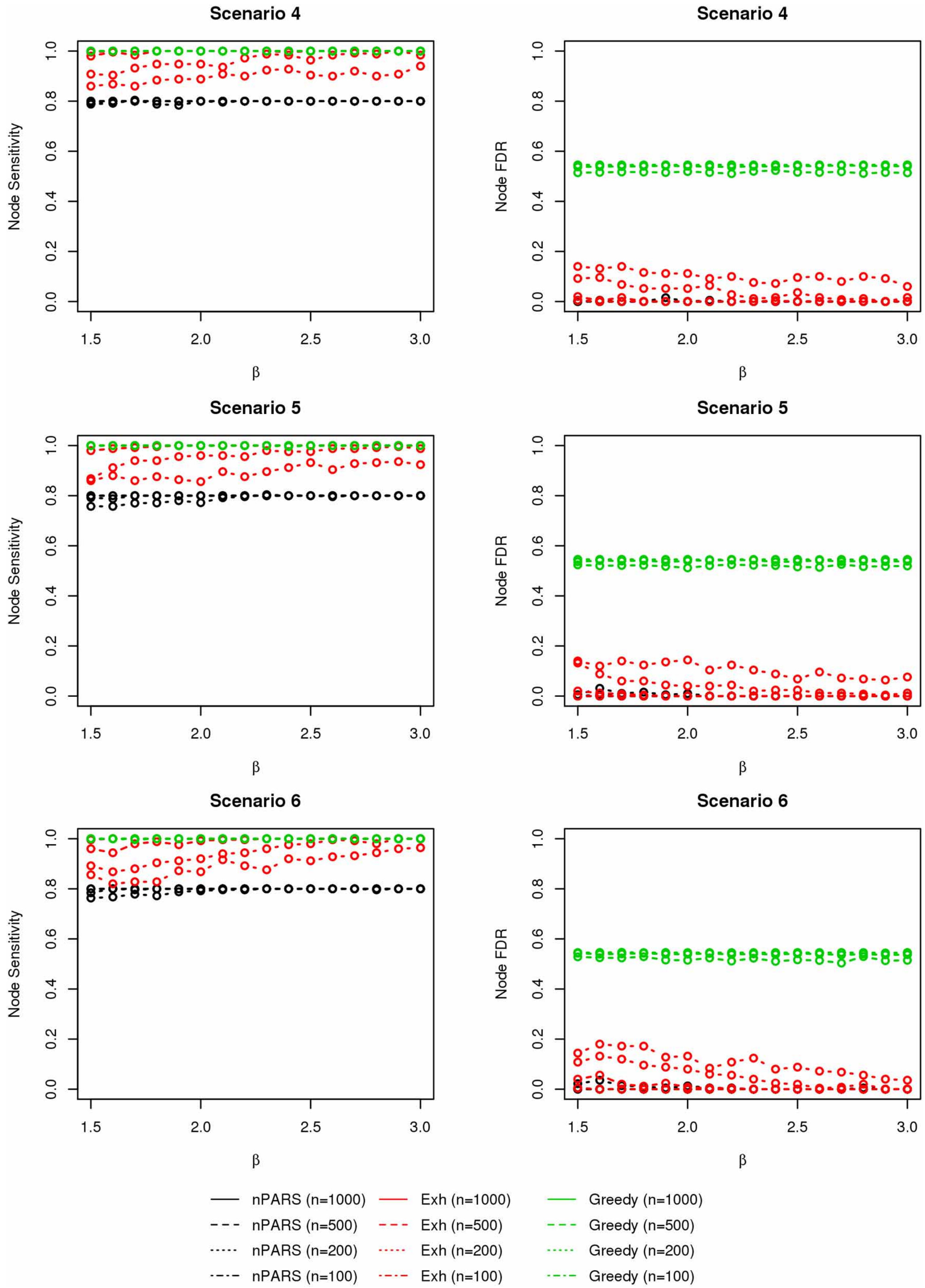


FIGURE 5 | The sensitivity and FDR of node comparisons in simulation scenario 4–6. The same legend is used as in **Figure 3**. nPARS shows lowest node FDR with slightly lower node sensitivity compare to exhaustive search. Although greedy search exhibits highest node sensitivity but it reports relatively high node FDR.

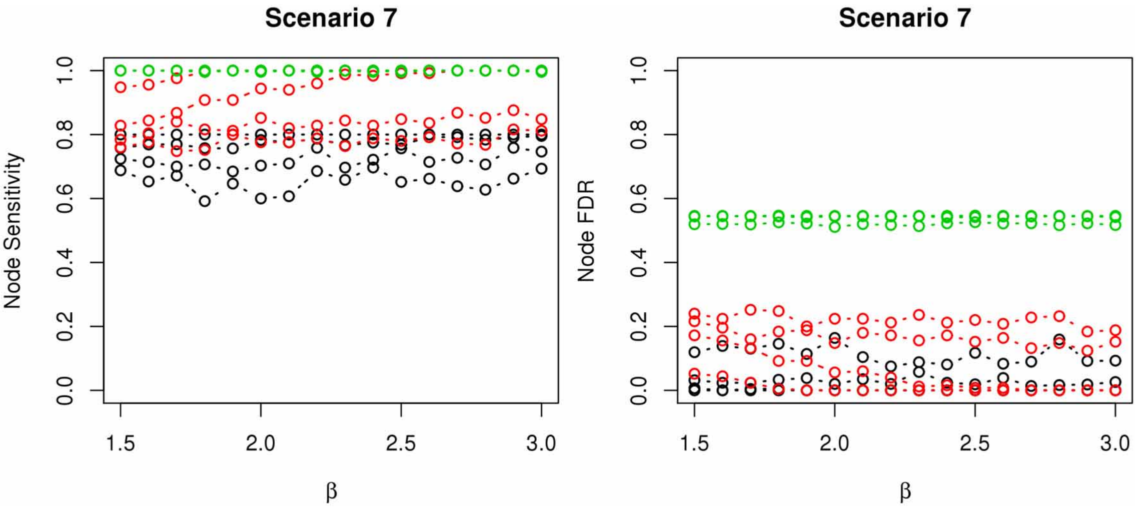


FIGURE 6 | The sensitivity and FDR of node comparisons in simulation scenario 7. The same legend is used as in **Figure 3**. nPARS shows lowest node FDR with slightly lower node sensitivity compare to exhaustive search. Although greedy search exhibits highest node sensitivity but it reports relatively high node FDR.

node and edges recoveries but lower sensitivity. It is also notable, however, that nPARS achieved strikingly low, node FDRs in our tests, suggesting that the stepwise approach to network development may offer protection against over-fitting. For example, the stage 1 of nPARS requires that each expression node demonstrate a clear and simple link between some locus L and disease D, which makes it difficult for false nodes to make it to a full, five-node network in stage 2. In comparison, it could be relatively easy for the exhaustive procedure to complete a strong, four-node network with a noisy false fifth variable.

With regard to computational efficiency, under simulation scenario 1 with $\beta = 0.8$, nPARS takes about 32 s to complete 1 iteration, Greedy search takes about 52 s and Exh takes 2387 s (39 min and 47 s) with a single 2.3 GHz CPU core on a 64-bit AMD Opteron-based server. The run times are similar in magnitude under other scenarios. The time complexity of the nPARS algorithm depends on the parameters k_1 and k_2. The time complexity of the first step of nPARS grows linearly with increasing number of genes. If k_1 and k_2 are fixed regardless of the number of genes considered in the study, then the time complexity of nPARS algorithm grows linearly with increasing number of genes. The R source code and documentation of the nPARS algorithm are available at http://www.biostat.umn.edu/~yho/research.html.

4. IMPLEMENTATION

4.1. CELLULAR RESPONSE TO ANTICANCER DRUGS DATA

In this example, we investigate differential responses to two chemotherapeutic agents: docetaxel and 5-FU. Both are widely used for a broad spectrum of cancers including colorectal, gastric, and head and neck cancer (Herbst and Khuri, 2003; Wang et al., 2004). Inter-individual variations in response to these anti-neoplastic drugs are commonly observed in cancer patients. Although several studies have shown that the resistance to docetaxel and 5-FU in human cancer cell are significantly inheritable (Watters et al., 2004), little is known about the underlying genetic mechanisms for this resistance.

This dataset includes 140 participants from 12 three-generation CEPH Utah families provided by the Genetic Analysis Workshop 15 (GAW15) (Cheung et al., 2005) and PharmGKB (Klein et al., 2001). Each family has approximately eight sibships in the third-generation. For each individual in the study, data from multiple sources was combined, including genotype, mRNA abundance, and cellular cytotoxicity levels in lymphoblastoid cells.

Genotypes of 2882 autosomal and X-linked SNPS, from across the whole genome, were generated by the SNP Consortium (http://snp.cshl.org/linkagemaps/) and provided through GAW15. We remove 552 SNP markers that have a high proportion of missing values (>0.3) or which are insufficiently polymorphic (minor allele frequency <0.1). We also examine the Mendelian consistency of the SNP genotypes and corrected them using Pedcheck and Merlin algorithms (O'Connell and Weeks, 1998; Abecasis et al., 2002).

Lymphoblastoid cells were isolated from each patient and 8793 mRNA transcripts were measured using Affymetrix Human Focus Arrays in previous studies (Cheung et al., 2003, 2005; Morley et al., 2004). We obtained the Affymetrix CEL files for all array hybridizations through GAW15. We then preprocessed the expression measures using RMA (Irizarry et al., 2003) and used mean expression intensities for replicates. For 3554 of the 8793 genes tested, Morley et al. (2004) found greater variation among individuals than between replicate determinations on the same individual. Hence, we choose these 3554 expression measures for further analyses.

The docetaxel and 5-FU cytotoxicity measures were obtained using lymphoblastoid cell lines derived from each participants and are available from the PharmGKB website http://www.pharmgkb.org/index.jsp. The percentages of LCL cell viability at

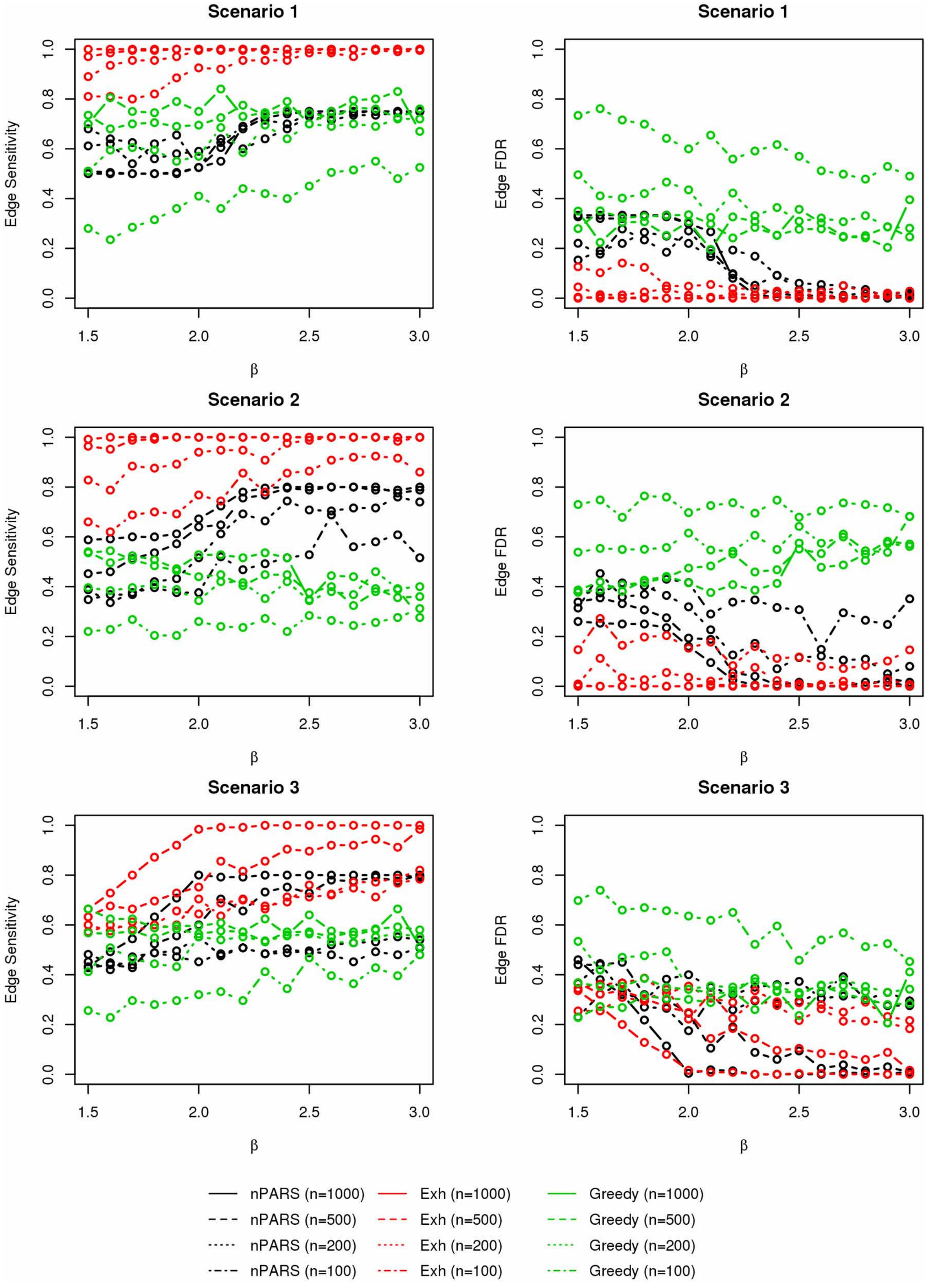


FIGURE 7 | The sensitivity and FDR of edge comparisons in simulation scenario 1–3. nPARS has better performance than GreedyE in terms of edge sensitivity except in scenario 1 and 3. In scenario 1, nPARS has comparable edge sensitivity compare to GreedyE. nPARS has lower edge FDR compare to GreedyE.

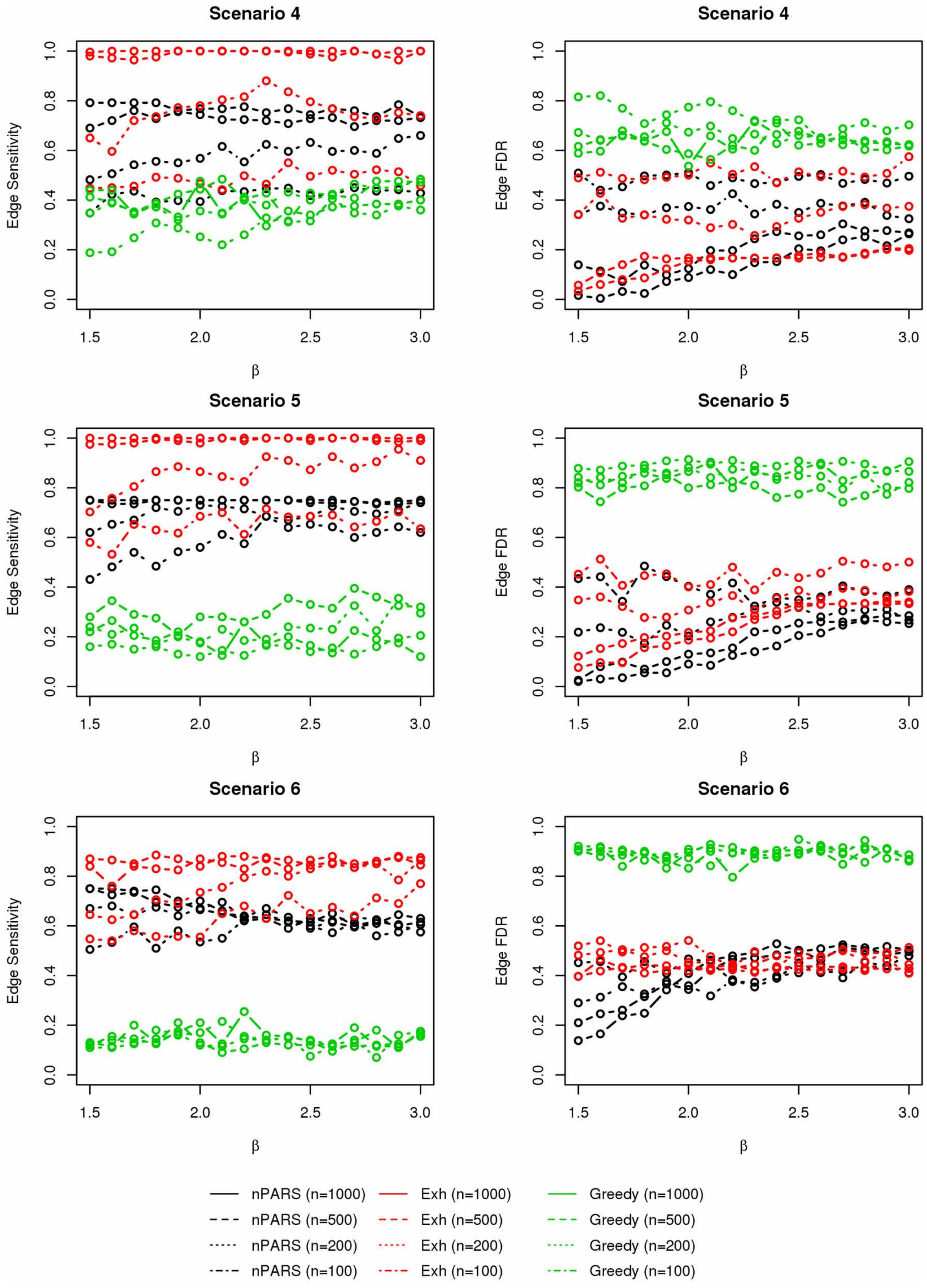


FIGURE 8 | The sensitivity and FDR of edge comparisons in simulation scenario 4–6. The same legend is used as in **Figure 6**. nPARS has higher edge sensitivity and lower edge FDR compare to GreedyE in scenario 4–6.

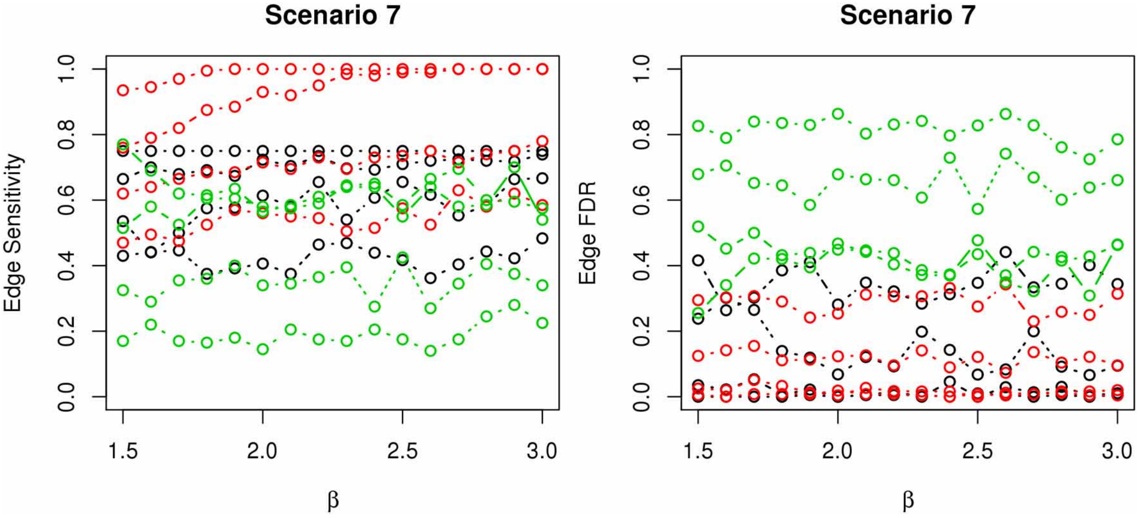


FIGURE 9 | The sensitivity and FDR of edge comparisons in simulation scenario 7. The same legend is used as in **Figure 6**. nPARS has higher edge sensitivity and lower edge FDR than GreedyE in these scenarios.

0.1, 0.5, 1, 5, 10, 50, 100 nM for docetaxel and at 0.76, 1.92, 3.84, 5.77, 7.68, 19.2, 38.4, 76.8 mM for 5-FU were measured and recorded for each individual. More detail about the cytotoxicity experiment procedures can be found in Watters et al. (2004).

4.2. FAMILIAL AGGREGATION OF RESPONSES TO CHEMOTHERAPEUTIC AGENTS

In **Figure 10**, we plot the percentages of cell viability against the $\log_e$ dose of docetaxel and 5-FU for each individuals. A large area under the log dose response curve indicates strong chemoresistance. In the following analysis, for each individual, we use the area under the log-dose response curve as a summary representing the chemo-resistance outcome. There is one missing observation at 0.1 nM for docetaxel, there are four missing observations at 0.76 mM for 5-FU and there are no missing observation at the end dose for either agents. Since missing the first dose will underestimate the area under the curve, we apply linear regression models to predict the missing cytotoxicity values from non-missing observations at other does using data from the same individual.

Familial aggregation of the responses to chemotherapeutic agents can be observed. For example, individuals in the Utah 1346 pedigree (blue) show generally higher level of resistance than individuals in Utah 1424 (orange), Utah 1416 (green), and Utah 1362 (light blue) families in both graphs.

4.3. RESULTS USING nPARS ALGORITHM

We apply the nPARS algorithm to this data, with 2330 SNP loci (L) and 3554 gene expression measures (E). We use the area under the log dose response curve as the phenotypic outcome (D), and analyze docetaxel and 5-FU separately. For each phenotypic outcome, we exhaustively score all possible $2330 \times 3554 = 8,280,820$ triplets combinations in the partition step. The subnetwork for each triplet is determined by the highest network score. Among these triplets, there are 825,637 ($\approx$10.0%) triplets whose best fitted subnetworks are among $G_6, \ldots, G_{12}$ for docetaxel and 635,390 ($\approx$7.7%) for 5-FU.

Among these, we select the top 100 scoring triplets for reassembly. We list the top 10 scoring triplets in **Tables 1**, **2** for docetaxel and 5-FU, respectively. Particularly, our results suggest four important SNP markers: rs1333798, rs695937, rs2056737, and rs1485768 because they appear many times in the top ranking networks for both docetaxel and 5-FU. In the subsequent reassembly step, we combine every two triplets into $(100, 2) = 4950$ sets of four or five nodes. After calculating the φ score for all resulting 4950 networks, we select the top 20. We present the five-node networks, if they have two gene expression as intermediate variables, in **Tables 3**, **4**, for docetaxel and 5-FU, respectively. The corresponding network structures are plotted in **Figures 11**, **12**.

To estimate the variance explained by the top scoring networks, we perform linear regression adjusting for family. In the regression model, we used the area under the log-dose response curve as response variable and the nodes reported in the final networks (shown in **Tables 3**, **4**) as predictors, while adjusting for family. The results are shown in the final column in **Tables 3**, **4** for docetaxel and 5-FU, respectively. After adjusting for family structure, we observe that the top scoring networks reported by nPARS explain a significant amount of variation in drug resistance outcomes. The mean adjusted R^2 are 48.63% and 33.01% for docetaxel and 5-FU, respectively. In addition, we obtain p-values using an F test based on linear regression models. All top scoring networks show p-value smaller than 0.00001 for docetaxel and smaller than 0.01 for 5-FU. Even after Bonferroni correction for multiple comparisons, all remain statistically significant except subnetwork #7 and #9 for 5-FU.

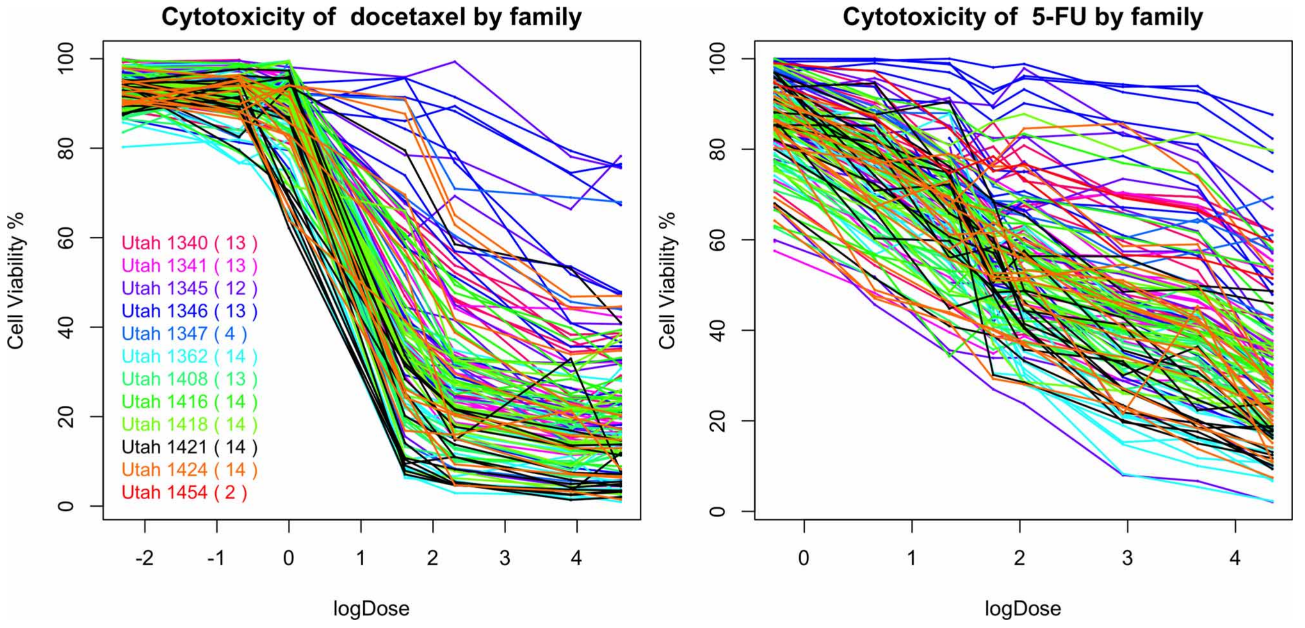


FIGURE 10 | The cytotoxicity responses of docetaxel and 5-FU.

Table 1 | Top 10 scoring triplets for docetaxel, and associated φ scores.

	L	Location of *L* (Chr: Mb)	*E*	Location of *E* (Chr: Mb)	φ
1	rs1333798	13:88.8	CCL20	02:228.7	−103.04
2	rs695937	03:64.2	CCL20	02:228.7	−106.51
3	rs2056737	02:156.8	CCL20	02:228.7	−110.01
4	rs1333798	13:88.8	PSTPIP2	18:43.6	−113.34
5	rs1333798	13:88.8	SPARC	05:151.1	−113.64
6	rs1333798	13:88.8	PON2	7:95.1	−114.70
7	rs1333798	13:88.8	BUD31	7:99.0	−114.71
8	rs1485768	04:177.6	EGFL6	X:13.6	−114.81
9	rs1333798	13:88.8	VCAM1	01:101.2	−114.93
10	rs1333798	13:88.8	USP39	02:85.9	−115.56

Table 2 | Top 10 scoring triplets for 5-FU, and associated φ scores.

	L	Location of *L* (Chr:Mb)	*E*	Location of *E* (Chr:Mb)	φ
1	rs695937	03:64.2	CCL20	02:228.7	−105.80
2	rs1333798	13:88.8	CCL20	02:228.7	−106.64
3	rs2056737	02:156.8	CCL20	02:228.7	−111.19
4	rs1333798	13:88.8	PON2	7:95.1	−114.17
5	rs1333798	13:88.8	FFAR2	19:35.9	−114.56
6	rs1485768	04:177.6	EGFL6	X:13.6	−114.71
7	rs1333798	13:88.8	UPB1	22:24.9	−115.30
8	rs2056737	02:156.8	FKBP5	6:35.6	−115.94
9	rs1015453	X:14.0	C5AR1	19:47.8	−115.98
10	rs2056737	02:156.8	TPM2	9:35.7	−116.62

Through this experimental data analysis, we intend to demonstrate the implementation of nPARS in a large-scale genomic data set. The analysis results suggest that rs1333798, rs1485768, rs2056737, and rs695937 and CCL20 combinations might explain the cytotoxicity responses observed in the lymphoblastoid cell lines for both docetaxel and 5-FU. rs1485768 is within the VEGFC gene which is involved in multiple cancer related pathways. In addition, rs695937 locates within the PRICKLE2 gene coding

Table 3 | Top scoring five-node subnetworks for docetaxel*.

	L_1	L_2	E_1	E_2	φ	Adjusted R^2 (%)
1	rs1333798	rs1485768	CCL20	EGFL6	−71.26	48.75
2	rs2056737	rs1333798	CCL20	ADARB1	−71.28	50.15
3	rs2056737	rs1333798	CCL20	PRKCA	−71.85	49.36
4	rs2056737	rs1333798	CCL20	BUD31	−71.96	47.96
5	rs1485768	rs1333798	EGFL6	CD93	−71.97	41.47
6	rs2056737	rs1333798	CCL20	CCNA1	−72.24	47.97
7	rs2056737	rs1333798	CCL20	UPB1	−73.23	50.09
8	rs2056737	rs1333798	CCL20	RAI14	−73.33	50.18
9	rs2056737	rs1333798	CCL20	VCAM1	−73.37	48.96
10	rs2056737	rs1333798	CCL20	PSTPIP2	−73.4	51.36

All p values <0.00001.

Table 4 | Top scoring five-node subnetworks for 5-FU.

	L_1	L_2	E_1	E_2	φ	*p* value	Adjusted R^2 (%)
1	rs2056737	rs695937	CCL20	UPB1	−71.46	3.81×10^{-5}	40.51
2	rs695937	rs2056737	CCL20	CRIP1	−74.24	1.03×10^{-4}	37.85
3	rs2056737	rs1333798	CCL20	ADARB1	−74.62	3.38×10^{-3}	25.72
4	rs695937	rs2056737	CCL20	IL18R1	−74.69	1.07×10^{-5}	43.69
5	rs695937	rs2056737	CCL20	BLMH	−75.09	3.40×10^{-5}	40.81
6	rs2056737	rs1333798	CCL20	UPB1	−75.41	1.13×10^{-3}	29.41
7	rs2056737	rs1333798	CCL20	PRKCA	−75.46	6.80×10^{-3}	23.17
8	rs695937	rs2056737	CCL20	TPM2	−75.52	3.57×10^{-4}	34.24
9	rs1333798	rs2056737	CCL20	BUD31	−75.59	0.01	21.68

region. PRICKLE2 belongs to the Wnt signalling pathway which regulates many downstream genes through its interaction with the T-cell factor family of transcription factors. The wnt signaling pathway also leads to remodeling of the cytoskeleton which is the main drug action of docetaxel, though the exact connection between these genetic variants and CCL20 expression is not yet clear.

CCL20 is a chemokine and it provokes proliferation and adhesion to collagen for several types of cancer cells (Beider et al., 2009). It is also believed that CCL20 is relevant to chemoresistance for various kind of cancers (Chang et al., 2008). For docetaxel resistance in lymphoblastoid cells, it is possible that CCL20 may influence resistance through regulation of actin cytoskeleton via the chemokine singling pathway, since cytoskeleton function is the main drug target of docetaxel. Genes' expressions that are likely to co-regulate with CCL20 and contribute to docetaxel resistance include EGFL6, ADARB1, PRKCA, BUD31, CD93, CCNA1, UPB1, RAI14, VCAM1, and PSTPIP2. Some of these genes are likely to be relevant to chemo-resistance response through cell cycle regulation, adhesion, or carcinogenesis pathways, EGFL6, PRKCA, VCAM1. ADARB1 and BUD31 are involved in mRNA precursor editing and modification. CD93, RAI14, and PSTPIP2 are part of cytoskeleton or interact with cytoskeleton function.

In addition, as indicated in the reported top fifth scoring network, the genetic variations in two SNP markers: rs1485768 and rs1333798 might contribute to the variation in gene expression of EGFL6, CD93. EGFL5 and CD93 playing important roles in regulating cell cycle, and remodeling cytoskeleton.

As for resistance to 5-FU, the CCL20 chemokine is also found to be crucial. CCL20 might play an important role through mediating DNA degradation or GPCR pathways. Other genes that could potentially co-regulate 5-FU resistance together with CCL20 include UPB1, CRIP1, ADARB1, IL18R1, BLMH, PRKCA, TPM2, BUD31, ITGAM, and RAB8B. Specifically, UPB1 participates in the 5-FU drug metabolic pathway by converting fluoro-beta-ureidopropionate to fluoro-beta-alanine (FBAL). FBAL is the major secretable form of 5-FU found in patients' urine sample. Although feasible biological hypotheses could be suggested based on our analysis results, further experiments are needed to validate the roles of these genetic factors in chemotherapy response.

5. CONCLUSION

To meet the growing need for efficient data analysis at the level of biological systems, we have developed and evaluated a modular approach to the construction of genetic networks. Our goal was to strike an appropriate balance between two potential sources of error. There is the error introduced when a necessarily less-than-exhaustive search through high-dimensional network space misses important regions of that space. This risk can be reduced by judicious variable selection to reduce the size of the search, but "judicious" is a loaded term and ideally the variable selection

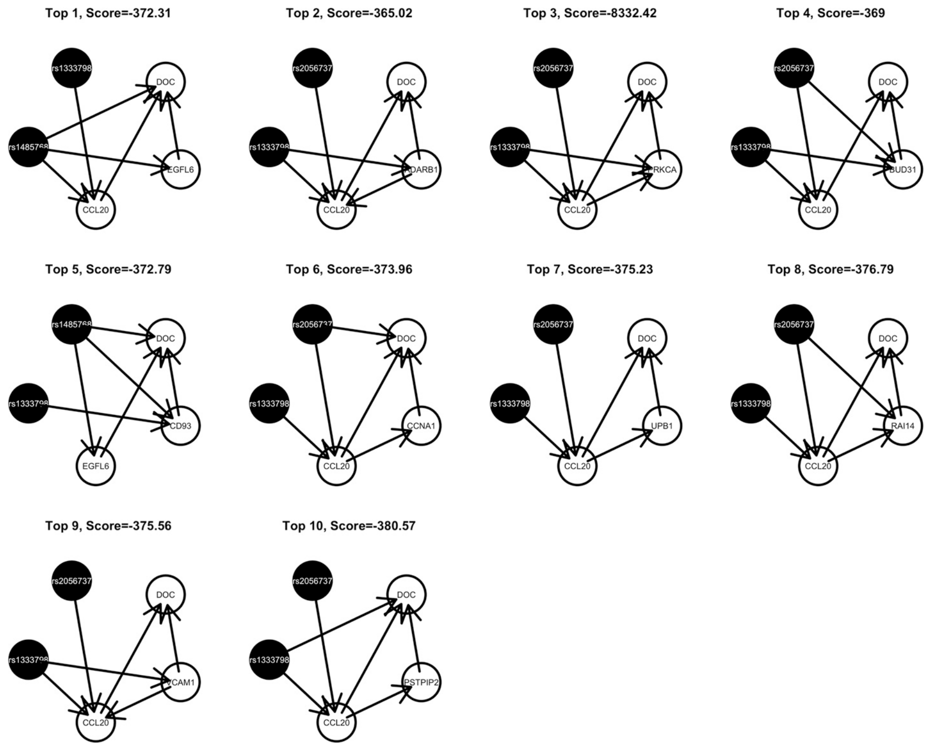


FIGURE 11 | Top scoring five-node network structures for docetaxel reported from nPARS algorithm.

step would capture some of the information that is distributed jointly across network components. By building a network from small components identified in an exhaustive search we hope to improve variable selection while controlling the computational burden.

The main focus of the paper is to assess the advantage of network-driven feature selection strategy. Based on our study findings, this network construction strategy provides ways to focus on small subnetworks that present with higher signal and allow more reliable estimation of network structure. In a set of extensive simulations, we compared the performance of the modular nPARS approach to that of both the greedy and exhaustive searches, evaluating the performance of each across a variety of scenarios. In these analyses, nPARS outperformed the greedy search which tended to have high FDRs for both nodes and edges, and proved competitive with the exhaustive search.

The fact that nPARS achieves better performance in terms of false discovery than exhaustive search in some simulation scenarios is beyond our expectation, and we suggest two possible factors: (1) Although we have attempted to represent a range of biologically realistic networks, there may be some bias in the system whereby the variable selection criteria implicit in nPARS is particularly appropriate to the network structures modeled in some of those scenarios. (2) One of the goals driving this method was to improve the effectiveness of the search through network space by including only those variables that made a significant contribution to smaller network structures. By requiring clear links between locus L, transcript E and phenotype D in the first stage of the algorithm, we make it less likely that a noisy false node is available for inclusion in the larger network later on. Without such a filtering step, it is relatively easy for the exhaustive procedure to complete a strong four-node network with a noisy, false fifth node. By either cause, we would anticipate that in larger, more complex networks, that nPARS' advantage over the exhaustive procedure would diminish. Unfortunately it is not yet practical to scale the exhaustive approach to test this.

We did not explicitly model family structure when constructing the Bayesian networks on our chemo-resistance application, assuming that any similarity of phenotypic values between relatives could be fully explained by the genetic variables considered in a network. However, since pedigree data was available for the samples in the drug response study, we used it in evaluating the top scoring networks we reported. Specifically, we performed a linear regression analysis that included family structure, to see how well the genetic variables explained drug response after adjusting for pedigree structure. We obtained small p-values and large adjusted R^2, suggesting that the reported networks play significant roles in drug resistance responses.

Other limitation of the proposed nPARS algorithm is that the algorithm in its current specification focuses on identifying structures related to (L ,E, D). As demonstrate by simulation scenario 7, nPARS has considerable power to detect cases where L contribute to D directly ($L \rightarrow D$). However, in scenario 7, if we replace E_1 and E_2 by L_3 and L_4, then

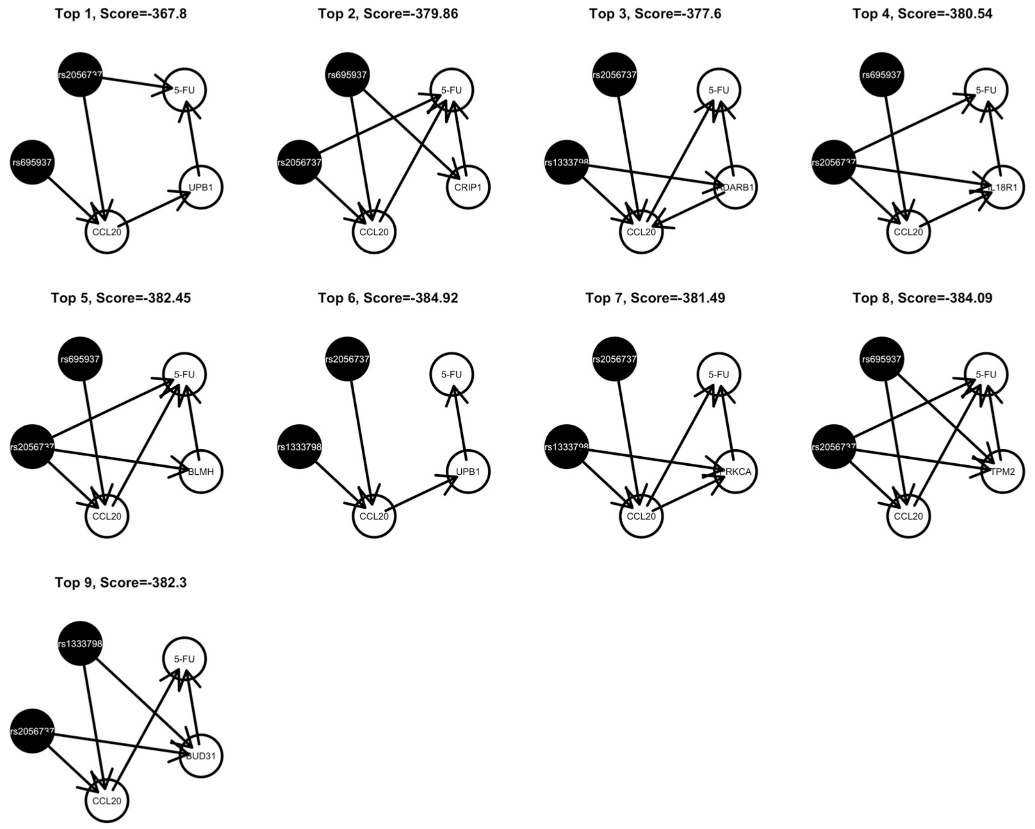


FIGURE 12 | Top scoring five-node network structures for 5-FU reported from nPARS algorithm.

nPARS would have a diminished power to detect such case. The algorithm can be easily modified to consider this modified scenario but increased amount of computational intensity will be expected.

Furthermore, our implementation of nPARS is tailored to the SNP—expression—phenotype setting in which it was tested, but could be readily modified to accommodate other genetic or epigenetic data in place of SNPs, including copy number and DNA methylation, though it may be necessary to modify the scoring functions or re-weight the prior distribution on network structures to reflect the unique biological characteristics of each data type. Potential direction for future research is to accommodate pedigree structure into the marginal likelihood score of Bayesian networks. But this approach would require considerable amount of samples to have enough power for detecting effects. We anticipate to have demonstrated that a practical compromise between exhaustive and greedy searches can improve on both and that our method can be the basis for future expansions.

ACKNOWLEDGMENTS

We acknowledge NIH R01 grant GM031575 and HG002386 for the collection and distribution of CEPH data. Yen-Yi Ho was partially supported by 2P30CA077598, P50CA101955, UL1TR000114 and U54-MD008620. Giovanni Parmigiani and Leslie M. Cope are partially supported by NSF-DMS-1041698.

REFERENCES

Abecasis, G. R., Cherny, S. S., Cookson, W. O., and Cardon, L. R. (2002). Merlin–rapid analysis of dense genetic maps using sparse gene flow trees. *Nat. Genet.* 30, 97–101. doi: 10.1038/ng786

Bar-Joseph, Z., Gerber, G. K., Lee, T. I., Rinaldi, N. J., Yoo, J. Y., Robert, F., et al. (2003). Computational discovery of gene modules and regulatory networks. *Nat. Biotechnol.* 21, 1337–1342. doi: 10.1038/nbt890

Beider, K., Abraham, M., Begin, M., Wald, H., Weiss, I. D., Wald, O., et al. (2009). Interaction between CXCR4 and CCL20 pathways regulates tumor growth. *PLoS ONE* 4:e5125. doi: 10.1371/journal.pone.0005125

Bøttcher, S. G. (2004). *Learning Bayesian Networks with Mixed Variables*. Ph.D. thesis, Aalborg University.

Bøttcher, S. G., and Dethlefsen, C. (2003). Deal: a package for learning bayesian networks. *J. Stat. Softw.* 8, 200–203.

Chang, H.-H., and McGeachie, M. (2011). Phenotype prediction by integrative network analysis of SNP and gene expression microarrays. *Conf. Proc. IEEE Eng. Med. Biol. Soc.* 2011, 6849–6852.

Chang, K.-P., Hao, S.-P., Chang, J.-H., Wu, C.-C., Tsang, N.-M., Lee, Y.-S., et al. (2008). Macrophage inflammatory protein-3α is a novel serum marker for nasopharyngeal carcinoma detection and prediction of treatment outcomes. *Clin. Cancer Res.* 14, 6979–6987. doi: 10.1158/1078-0432.CCR-08-0090

Chen, Y., Zhu, J., Lum, P. Y., Yang, X., Pinto, S., MacNeil, D. J., et al. (2008). Variations in DNA elucidate molecular networks that cause disease. *Nature* 452, 429–435. doi: 10.1038/nature06757

Cheung, V. G., Conlin, L. K., Weber, T. M., Arcaro, M., Jen, K.-Y., Morley, M., et al. (2003). Natural variation in human gene expression assessed in lymphoblastoid cells. *Nat. Genet.* 33, 422–425. doi: 10.1038/ng1094

Cheung, V. G., Spielman, R. S., Ewens, K. G., Weber, T. M., Morley, M., and Burdick, J. T. (2005). Mapping determinants of human gene expression by regional and genome-wide association. *Nature* 437, 1365–1369. doi: 10.1038/nature04244

Friedman, N., Linial, M., Nachman, I., and Pe'er, D. (2000). Using Bayesian networks to analyze expression data. *J. Comput. Biol.* 7, 601–620. doi: 10.1089/106652700750050961

Gasse, M., Aussem, A., and Elghazel, H. (2012). "An experimental comparison of hybrid algorithms for Bayesian network structure learning," in *ECML-PKDD 2012* (Bristol), 58–73.

Herbst, R. S., and Khuri, F. R. (2003). Mode of action of docetaxel – a basis for combination with novel anticancer agents. *Cancer Treat. Rev.* 29, 407–415. doi: 10.1016/S0305-7372(03)00097-5

Imoto, S., Kim, S., Goto, T., Miyano, S., Aburatani, S., Tashiro, K., et al. (2003). Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network. *J. Bioinform. Comput. Biol.* 1, 231–252. doi: 10.1142/S0219720003000071

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., et al. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics* 4, 249–264. doi: 10.1093/biostatistics/4.2.249

Klein, T. E., Chang, J. T., Cho, M. K., Easton, K. L., Fergerson, R., Hewett, M., et al. (2001). Integrating genotype and phenotype information: an overview of the PharmGKB project. pharmacogenetics research network and knowledge base. *Pharmacogenomics J.* 1, 167–170. doi: 10.1038/sj.tpj.6500035

Lasserre, J., Chung, H.-R., and Vingron, M. (2013). Finding associations among histone modifications using sparse partial correlation networks. *PLoS Comput. Biol.* 9:e1003168. doi: 10.1371/journal.pcbi.1003168

Li, H., Lu, L., Manly, K. F., Chesler, E. J., Bao, L., Wang, J., et al. (2005). Inferring gene transcriptional modulatory relations: a genetical genomics approach. *Hum. Mol. Genet.* 14, 1119–1125. doi: 10.1093/hmg/ddi124

Mehrabian, M., Allayee, H., Stockton, J., Lum, P. Y., Drake, T. A., Castellani, L. W., et al. (2005). Integrating genotypic and expression data in a segregating mouse population to identify 5-lipoxygenase as a susceptibility gene for obesity and bone traits. *Nat. Genet.* 37, 1224–1233. doi: 10.1038/ng1619

Morley, M., Molony, C. M., Weber, T. M., Devlin, J. L., Ewens, K. G., Spielman, R. S., et al. (2004). Genetic analysis of genome-wide variation in human gene expression. *Nature* 430, 743–747. doi: 10.1038/nature02797

O'Connell, J. R., and Weeks, D. E. (1998). PedCheck: a program for identification of genotype incompatibilities in linkage analysis. *Am. J. Hum. Genet.* 63, 259–266. doi: 10.1086/301904

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann Publishers Inc.

Pe'er, D., Regev, A., Elidan, G., and Friedman, N. (2001). Inferring subnetworks from perturbed expression profiles. *Bioinformatics* 17, S215–S224. doi: 10.1093/bioinformatics/17.suppl_1.S215

Schadt, E. E. (2009). Molecular networks as sensors and drivers of common human diseases. *Nature* 461, 218–223. doi: 10.1038/nature08454

Schadt, E. E., Lamb, J., Yang, X., Zhu, J., Edwards, S., Guhathakurta, D., et al. (2005). An integrative genomics approach to infer causal associations between gene expression and disease. *Nat. Genet.* 37, 710–717. doi: 10.1038/ng1589

Teyssier, M. (2005). "Ordering-based search: a simple and effective algorithm for learning bayesian networks," in *In UAI*, 584–590.

Tu, Z., Wang, L., Arbeitman, M. N., Chen, T., and Sun, F. (2006). An integrative approach for causal gene identification and gene regulatory pathway inference. *Bioinformatics* 22, e489–e496. doi: 10.1093/bioinformatics/btl234

Wang, W., Cassidy, J., O'Brien, V., Ryan, K. M., and Collie-Duguid, E. (2004). Mechanistic and predictive profiling of 5-fluorouracil resistance in human cancer cells. *Cancer Res.* 64, 8167–8176. doi: 10.1158/0008-5472.CAN-04-0970

Watters, J. W., Kraja, A., Meucci, M. A., Province, M. A., and McLeod, H. L. (2004). Genome-wide discovery of loci influencing chemotherapy cytotoxicity. *Proc. Natl. Acad. Sci. U.S.A.* 101, 11809–11814. doi: 10.1073/pnas.0404580101

Xu, X., Wang, L., and Ding, D. (2004). Learning module networks from genome-wide location and expression data. *FEBS Lett.* 578, 297–304. doi: 10.1016/j.febslet.2004.11.019

Yang, X., Deignan, J. L., Qi, H., Zhu, J., Qian, S., Zhong, J., et al. (2009). Validation of candidate causal genes for obesity that affect shared metabolic pathways and networks. *Nat. Genet.* 41, 415–423. doi: 10.1038/ng.325

Yu, J., Smith, V. A., Wang, P. P., Hartemink, E. J., and Jarvis, E. D. (2002). "Using Bayesian network inference algorithms to recover molecular genetic regulatory networks," in *International Conference on Systems Biology (ICSB02)*.

Zhang, B., and Horvath, S. (2005). A general framework for weighted gene co-expression network analysis. *Stat. Appl. Genet. Mol. Biol.* 4:17. doi: 10.2202/1544-6115.1128

Zhu, J., Wiener, M. C., Zhang, C., Fridman, A., Minch, E., Lum, P. Y., et al. (2007). Increasing the power to detect causal associations by combining genotypic and expression data in segregating populations. *PLoS Comput. Biol.* 3:e69. doi: 10.1371/journal.pcbi.0030069

Conflict of Interest Statement: 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.

Received: 07 May 2013; accepted: 01 February 2014; published online: 26 February 2014.

Citation: Ho Y-Y, Cope LM and Parmigiani G (2014) Modular network construction using eQTL data: an analysis of computational costs and benefits. Front. Genet. ***5****:40. doi: 10.3389/fgene.2014.00040*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
GENETICS

METHODS ARTICLE
published: 16 May 2014
doi: 10.3389/fgene.2014.00123

Network Assessor: an automated method for quantitative assessment of a network's potential for gene function prediction

Jason Montojo*, Khalid Zuberi, Quentin Shao, Gary D. Bader and Quaid Morris

Donnelly Centre for Cellular and Biomolecular Research, University of Toronto, Toronto, ON, Canada

Edited by:
Benjamin Haibe-Kains, Princess Margaret Cancer Centre, Canada

Reviewed by:
Zoran Nikoloski, Max-Planck Institute of Molecular Plant Physiology, Germany
Gaurav Sablok, Istituto Agrario San Michele, Italy
Jian Ren, Sun Yat-sen University, China

***Correspondence:**
Jason Montojo, The Donnelly Centre, University of Toronto, 160 College Street, Toronto, Ontario M5S 3E1, Canada
e-mail: j.montojo@utoronto.ca

Significant effort has been invested in network-based gene function prediction algorithms based on the guilt by association (GBA) principle. Existing approaches for assessing prediction performance typically compute evaluation metrics, either averaged across all functions being considered, or strictly from properties of the network. Since the success of GBA algorithms depends on the specific function being predicted, evaluation metrics should instead be computed for each function. We describe a novel method for computing the usefulness of a network by measuring its impact on gene function cross validation prediction performance across all gene functions. We have implemented this in software called Network Assessor, and describe its use in the GeneMANIA (GM) quality control system. Network Assessor is part of the GM command line tools.

Keywords: network inference, function prediction, cross validation, network biology, machine learning

INTRODUCTION

Networks of gene-gene functional interactions (or more generally, associations) have proven useful to predict gene function (Zhang et al., 2004; Mostafavi et al., 2008; Peña-Castillo et al., 2008). In this model, the nodes of the network are genes and the edges represent specific types of associations between them. For example, a gene can be connected to other genes that inhibit or promote it, that encode similar protein domains, that share similar expression profiles, that are located close together on the same chromosome, or whose products physically interact with its products.

Various methods exist for predicting function from gene-gene networks. The most common approach uses some variation of the guilt by association (GBA) principle (Schwikowski et al., 2000; Hishigaki et al., 2001; Wu et al., 2002; Vazquez et al., 2003; Deng et al., 2004; Ye et al., 2005; Sharan et al., 2007; Franceschini et al., 2013; Zuberi et al., 2013). This assumes the function of a gene can be inferred from its neighbors in the network by following edges. Guilt-free approaches also exist, such as using the node degree of a gene without considering any properties of its neighbors (Gillis and Pavlidis, 2011). These algorithms use binary classification to perform predictions for one function at a time. Multi-label classifiers also exist that can make predictions for multiple functions simultaneously (Wang et al., 2013; Yu et al., 2013).

Network-based gene function algorithms have demonstrated strong performance for multifunctional genes (Gillis and Pavlidis, 2011). Algorithms that use binary classification are typically evaluated by using cross validation against a gold standard, such as gene annotations from Gene Ontology (GO) (The Gene Ontology Consortium, 2000) or FunCat (Ruepp et al., 2004). This process involves passing the association networks and a subset of the genes in a specific GO term as input to a binary classifier, which implements a particular prediction algorithm. The classifier then attempts to recover the withheld genes by ranking them based on the likelihood that they are members of the GO term. From this ranking, the area under the receiver operating characteristic curve (AUROC) and the area under the precision-recall curve (AUPR) metrics are computed (Fawcett, 2006). The AUROC and AUPR values are typically aggregated across GO terms to produce a mean AUROC and mean AUPR. The input association networks may need to be integrated into a single graph prior to binary classification, depending on the prediction algorithm used. A similar process can be used with multi-label classifiers when evaluating label-based performance (Tsoumakas et al., 2010).

When the input association networks and gold standard are held constant, we can use this process to compare the performance of different prediction algorithms. However, if we instead fix the algorithm and gold standard, we can assess the usefulness of the input association networks for particular tasks, such as assigning gene membership to GO terms.

Quantifying the usefulness of specific gene-gene networks for function prediction is difficult in general. The topology of a network may impact prediction performance differently depending on the function in question. Sometimes a small fraction of edges may account for most of the cross validation performance for a large number of GO terms (Gillis and Pavlidis, 2012). Larger networks are more likely to include more of these informative edges, but it's also possible for a large network to have only a few of them. Similarly, a small network can be constructed to contain a large proportion of such critical or exceptional edges.

The software we present, Network Assessor, was designed for gene function-specific quantification of the usefulness of association networks for prediction tasks. In particular, the software quantifies the predictive potential of one or more networks by reporting the differences in cross validation performance for each GO term, with and without the network(s) in question. Although the software provides built-in support for using the latest GO annotations, any annotation set can be used as the gold standard. Network Assessor has already been used to demonstrate that genetic interaction networks obtained under different experimental conditions provide complementary information that improves gene function prediction performance (Michaut and Bader, 2012).

Network Assessor currently uses the GeneMANIA (GM) algorithm (Mostafavi et al., 2008), which is a fast, real-time network integrator and binary classifier that uses GBA to infer gene function. Recent studies have indicated that cross validation performance of GBA-based algorithms depends on the GO term being tested (Gillis and Pavlidis, 2012). Although Network Assessor uses a GBA-based predictor, it can be readily extended to use non-GBA algorithms and even multi-label classifiers. Network Assessor permits term-by-term analysis by providing AUROC and AUPR metrics for each GO term rather than averaging over all GO terms.

Network Assessor was originally used to analyze changes in prediction performance between different releases of the GM web server (Warde-Farley et al., 2010; Zuberi et al., 2013). The results of this analysis help identify issues with GM network data, make parameter decisions and are used to evaluate new networks for inclusion in the system.

EXPERIMENTAL OBJECTIVES

We designed Network Assessor to quantify the usefulness of an association network for predicting gene function. However, directly measuring the predictive potential of an arbitrary network in isolation by simply assessing the degree to which the association network connects nodes with similar labels is not necessarily informative because of the synergistic nature of network data. For example, suppose you have two non-overlapping networks, A and B, and another network C that overlaps with both A and B. Predictions that use only A, B, or C in isolation would be very different from those made using the integration of all three because the set of reachable neighbors in the latter is much larger. This difference is significant for predictors based on label propagation (Kato et al., 2009) that utilize indirect connections between genes. Our approach allows us to measure the impact of adding (or withholding) network C.

Since prediction performance may vary by GO term, Network Assessor computes the relative predictive potential of an association network by measuring the impact of adding it to (or removing it from) a network with known predictive potential for each GO term. This allows researchers to examine differences in performance by GO term size and position in the GO hierarchy.

LIMITATIONS OF CURRENT TECHNIQUES

Alternative techniques for quantifying the usefulness of networks in gene function prediction exist. For instance, identifying which gene functions follow the GBA principle in a given network can be accomplished by applying statistics originally developed for testing spatial clustering in proximity networks (Kleessen et al., 2013). The degree of global spatial autocorrelation (such as Moran's I statistic) indicates whether gene expression correlates well with gene function. This is useful for investigating which gene functions follow the GBA principle. In contrast to these methods, Network Assessor measures the effect on prediction performance of an arbitrary network for all known gene functions one at a time.

Furthermore, Network Assessor provides a network integrator to allow the evaluation of sets of networks from different sources. This is critical for organisms with poor annotation coverage. Intra-species transfer of annotations and the integration of functional interaction networks derived through orthology would likely improve prediction performance (Klie et al., 2012).

In particular, Network Assessor also permits the analysis of different combinations of networks. It uses GM's various network integration algorithms to combine multiple weighted undirected networks into a single weighted undirected network. For example, the default behavior uses the Simultaneous Weights and Unregularized algorithms (Mostafavi and Morris, 2010) to assign weights to each network indicating its information content for the GO term prediction task at hand. This weight is multiplied with each of that network's edges during network integration. Weights can also be assigned equally by network.

Network Assessor makes analysis of association networks more accessible to both computational and non-computational scientists who otherwise must script their own analysis or do not have access to automated tools, respectively.

NETWORK ASSESSOR

Network Assessor measures the predictive potential of an association network by following a five step process (**Figure 1**). First, the set of baseline networks are combined into a weighted, undirected graph using GM's "automatic" network integration algorithm (Zuberi et al., 2013). Specifically, we use the GO Biological Process (BP)-based Simultaneous Weights algorithm for queries with less than five genes, and the Unregularized algorithm for five or more genes (Mostafavi and Morris, 2010). This follows the default behavior of GM's network integrator.

Second, this composite network is used during K-fold cross validation to recover the annotations in the user-provided gold standard, such as a set of GO terms and the lists of genes they annotate. Annotated nodes are treated as positive examples and all others are treated as negative. The GM algorithm is used as the binary classifier during cross validation and an AUROC/AUPR statistic is computed for each annotation, for each fold. A perfect classifier produces AUROC and AUPR values equal to 1. A random classifier achieves AUROC equal to 0.5 and AUPR close to P/(P+N) where P and N are the number of actual positive and negative examples, respectively (Schrynemackers et al., 2013).

Next, Steps 1 and 2 are repeated using the subject (i.e., non-baseline) networks. Typically, this comprises the baseline networks with the association network(s) of interest added (or removed, if the association network is part of the baseline).

Finally, the percentage differences of the AUROC and AUPR values are computed for each annotation. The results are sorted by these differences to highlight which annotations perform better or worse when predictions are made with or without the association network of interest. This method is a generalization of the leave-one-out analysis that we described in (Costanzo et al., 2010)

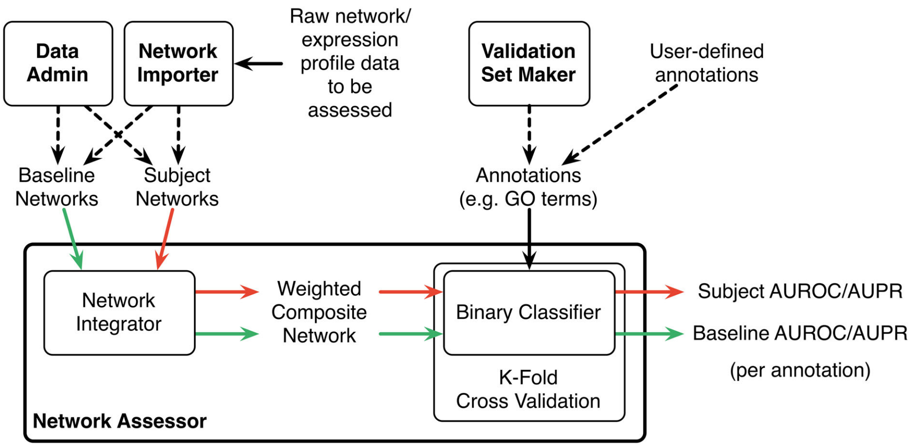


FIGURE 1 | Schematic diagram of the Network Assessor workflow. The green arrows indicate the first round of cross validation. Red arrows are second round. AUROC/AUPR statistics are computed for each node label (e.g., GO term) for each round of cross validation. Dashed lines indicate alternative options.

to measure the contribution of individual genetic interaction datasets to our understanding of functional relationships in yeast.

The results produced by Network Assessor are reported as a table in tab-delimited text format. Although we describe in this protocol how to use Network Assessor with gene-gene associations to validate against GO annotations, Network Assessor is generic enough to be used directly with any type of network data and gold standard.

Network Assessor is bundled with over 1800 GM networks for eight different model organisms. However, it is possible to use any organism and networks by using the Id Importer tool documented at http://pages.genemania.org/tools/.

MATERIALS

Prerequisites

- Familiarity with the Windows, Mac OS X, or Linux command line.

Reagents

- An association network in tab-delimited format. The first two columns are the identifiers of the interactors. These can be a mix of gene symbols, UniProt accessions/IDs, Ensembl Gene/Protein IDs, RefSeq mRNA/Protein IDs, TAIR IDs or Entrez Gene IDs. Optionally, a third column can be added to indicate the weight of the interaction. Here is an example weighted network with two interactions. "<TAB>" denotes a tab character, which must not be surrounded by spaces in the network file:

```
BRCA1<TAB>RAD50<TAB>0.25
BRCA1<TAB>MRE11A<TAB>0.34
```

Equipment

- A computer with at least 8 GB of RAM and an internet connection.

Equipment setup

The following software is required:

- Windows XP 64-bit, Mac OS 10.6 or Ubuntu Linux 8.04 64-bit (or equivalent) or later.
- Java 1.6 64-bit or later.
- GM command line tools version 3.3 or later (http://pages.genemania.org/tools/).

Procedure

Steps 1–3: Set up the command line environment.

1. Create a new directory for storing the results of your work.
2. Copy the GM command line tools JAR file into this directory. In later steps, we will assume this file is called genemania.jar. Also copy your association data into this directory.
3. In your shell, set the current directory to the working directory you just created.

Steps 4–10: Install baseline association data for your organism of choice.

4. Run the following command in your shell to list the available baseline data sets (The full documentation for this command and the ones below is available at http://pages.genemania.org/tools):

```
java -jar genemania.jar DataAdmin list
```

The following shows the output of this command:

```
Data Set ID Total Size Database Version
2013-10-15 9351.08 MB 15 October 2013
2013-10-15-core 2059.38 MB 15 October 2013
2013-10-15-open_license 9324.49 MB 15
  October 2013
2012-08-02 5994.14 MB 19 July 2012
2012-08-02-core 1764.09 MB 19 July 2012
2012-08-02-open_license 5963.38 MB 19
  July 2012
...
```

5. Note the “Data Set ID” (first column) of the data you wish to install. The latest release as of this writing is “2013-10-15” which corresponds to database version 15-Oct-2013.
6. Run the following command to download the base data set. This should take no more than a few seconds using a 3 Mbit/s connection.

```
java -jar genemania.jar DataAdmin \\
install 2013-10-15
```

7. Run the following command to list the available organisms for the selected data set:

```
java -jar genemania.jar DataAdmin \\
list-data
```

Here is an example of the output:

```
Data ID Description Status
1 A. thaliana Arabidopsis (2494 MB)
2 C. elegans Worm (282 MB)
3 D. melanogaster Fly (792 MB)
4 H. sapiens Human (2361 MB)
5 M. musculus Mouse (2137 MB)
6 S. cerevisiae Baker’s yeast (458 MB)
7 R. norvegicus Rat (576 MB)
8 D. rerio Zebrafish (248 MB)
```

8. Note the “Data ID” (first column)” of the organism you wish to install. For example, human data is “4.”
9. Run the following command to install the organism-specific baseline data set. Human data takes approximately 60 min to download and install using a 3 Mbit/s connection. In general it takes 10–70 min depending on the organism selected.

```
java -jar genemania.jar DataAdmin \\
install-data gmdata-2013-10-15 4
```

Note: `\\` indicates a line continuation and should not be included in the command.

Step 10: Download a gold standard for cross validation.

10. Run the following command to download the latest GO annotations and save it to a file. You can choose a particular GO branch, such as “bp” for biological process, “cc” for cellular component, or “mf” for molecular function, or “all” for everything. In the following example, the GO terms for taxonomy ID 9606 from the “bp” branch will be saved in the file “go-terms.txt.” By default, this command will download the annotations from the European Bioinformatics Institute GO MySQL server. It takes about 8 min on a 3 Mbit/s connection.

```
java -jar genemania.jar \\
ValidationSetMaker \\
--organism 9606 --branch bp \\
--query go-terms.txt
```

Here are the taxonomy IDs of the organisms currently available in GM:

Organism	Taxonomy ID
A. thaliana	3702
C. elegans	6239
D. melanogaster	7227
H. sapiens	9606
M. musculus	10090
S. cerevisiae	4932
R. norvegicus	10116

Step 11: Import the association data you want to analyze into your data set.

11. Run the following command to install your association data. This assumes your association data is stored in a file called “network.txt,” is for the organism with taxonomy ID 9606 (human), and will be saved with the name “network1” and categorized into group “group1.”

```
java --Xmx6G --jar genemania.jar \\
NetworkImporter \\
--data gmdata-2013-10-15 \\
--organism 9606 \\
--name "network1" --group "group1" \\
--filename network.txt
```

Step 12: Use Network Assessor to analyze the association data you imported.

12. To specify all GM networks as a baseline, use “`coexp, coloc,gi,path,pi,predict,spd`.” To measure the impact of your network in isolation not including the baseline networks, use “network1” for the “–network” parameter. To measure the impact of your network added to the baseline, use “coexp,coloc,gi,path,pi,predict,spd,network1” instead. The following example will assess your network in isolation, using 5-fold cross validation on four simultaneous processing threads with GO terms containing between 3 and 10 annotations, inclusive, and store the results in “go-terms.result.txt”:

```
java --Xmx6G -jar genemania.jar \\
NetworkAssessor \\
--data gmdata-2013-10-15 \\
--auto-negatives \\
--baseline "coexp,coloc,gi,path, \\
pi, predict,spd" --seed 1 \\
--threads 4 --networks "network1" \\
--organism 9606 --folds 5 --min 3 \\
--max 10 --query go-terms.txt \\
--outfile go-terms. result.txt
```

Cross validation is a highly-parallelizable process since each annotation in the validation set is assessed independently. Network Assessor can automatically distribute the work across all cores of a multi-core system by specifying the number of threads to use. You can also partition "go-terms.txt" into multiple files and process each file on a different cluster node. Since Network Assessor is a memory- and computation-intensive program, ensure that at least 6 GB of RAM are free prior to starting the assessment. It takes an eight core 2.53 GHz Intel Xeon E5540 system approximately 24 s per line in "go-terms.txt" on average for the 15-Oct-2013 full human data set using eight processing threads. Since human is our largest dataset, using another organism or a subset of the networks will allow faster cross validation times. On a 10-node cluster of similar nodes, assessing the network against 1000 GO terms would take around 40 min of real time (6.6 h of CPU time).

The "–min 3" and "–max 10" parameters instruct Network Assessor to only consider GO terms with at least 3 and no more than 10 annotations. This is important because binary classification algorithms generally perform worse with small GO terms. Using ranges 3–10 and 11–300 will give similarly sized partitions when used with the current GO database (see below for further explanation).

The "–threads" parameter should be set to the number of physical cores on your computer. For example, use "4" for a single quad core processor. For a dual-processor system with eight cores each, use "16."

Using the same non-zero "–seed" ensures the results of different runs of Network Assessor are reproducible, as long as all parameters and inputs are the same. Otherwise Network Assessor will have slightly different results due to how the folds are randomized. Specifying a seed will guarantee the K-folds of the baseline and subject data sets are partitioned the same way.

Here is a sample of the first four columns of Network Assessor's output:

QUERY	BASELINE-AUC-ROC	SUBJECT-AUC-ROC	%ERR-AUC-ROC
GO:0000046	0.498133458	0.548483434	0.101077
GO:0000117	0.471654812	0.516121807	0.094279
GO:0000114	0.461791463	0.503638908	0.09062

The first column indicates the GO term used in the assessment. The second column is the mean AUROC of the baseline networks across the K-folds. The third is the mean AUROC of the subject networks, which in this example is the new network in isolation. Finally, the fourth column shows the % improvement in subject AUROC compared to the baseline, computed as follows:

$$\%ERR_{AUROC} = \frac{SUBJECT_{AUROC}}{BASELINE_{AUROC}} - 1$$

In addition to these, the actual output file has similar columns for the AUPR and precision-at-10% recall statistics. If you run into any issues or have any questions, you can get in touch with the GM team at http://pages.genemania.org/contact/.

NETWORK ASSESSOR AND GeneMANIA QUALITY CONTROL

The dataset used by the GM gene function prediction server is updated on a regular basis. It performs real-time predictions for eight model organisms using over 530 million gene-gene functional associations organized into over 1800 networks. These associations are the edges of networks, which are weighted, undirected graphs, and come from numerous independent third-party sources. For example, co-expression networks are derived from gene expression profiles from GEO (Barrett et al., 2013); protein and genetic interactions from BioGRID (Chatr-Aryamontri et al., 2013); protein interactions inferred through orthology from I2D (Brown and Jurisica, 2007); pathway interactions from Pathway Commons (Cerami et al., 2011); and protein interactions from iRefIndex (Razick et al., 2008). Shared protein domain associations are derived from InterPro (Hunter et al., 2012) and PFAM (Punta et al., 2012). Identifiers and their metadata are sourced from Ensembl (Flicek et al., 2013).

Data imported from third parties can change without notice so each GM release reflects the state of those sources at a fixed point in time. For example, in an older data update, R6 (19-July-2012), cross validation results indicated a general drop in performance relative to the previous release, R5 (21-Dec-2011). This prompted further investigation, through which we discovered GM no longer recognized 10% (2344) of the human gene symbols that R5 supported. This was due to changes within Ensembl between R5

Table 1 | Median AUROC and AUPR for all networks in R6 and R8, as well as the default networks of each, respectively (bold indicates higher number per comparison).

	R6	R8	R6 (default)	R8 (default)
GO TERM SIZE = 3–10				
Median AUROC	0.650	**0.694**	0.627	**0.684**
95% CI	±0.316	±0.311	±0.334	±0.329
versus R6 (*p*-value)		*4.74 × 10^{-82}	*1.41 × 10^{-27}	
versus R8 (*p*-value)				*8.11 × 10^{-10}
GO TERM SIZE = 11–300				
Median AUROC	0.871	**0.890**	0.857	**0.882**
95% CI	±0.217	±0.195	±0.246	±0.212
versus R6 (*p*-value)		*9.96 × 10^{-258}	*5.68 × 10^{-129}	
versus R8 (*p*-value)				*4.53 × 10^{-35}
GO TERM SIZE = 3–10				
Median AUPR	0.012	**0.019**	0.019	**0.026**
95% CI	±0.349	±0.409	±0.343	±0.408
versus R6 (*p*-value)		*1.35 × 10^{-28}	*1.34 × 10^{-18}	
versus R8 (*p*-value)				*5.19 × 10^{-19}
GO TERM SIZE = 11–300				
Median AUPR	0.185	**0.220**	0.181	**0.215**
95% CI	±0.412	±0.528	±0.415	±0.529
versus R6 (*p*-value)		*8.45 × 10^{-256}	4.62 × 10^{-1}	
versus R8 (*p*-value)				3.56 × 10^{-2}

The Wilcoxon rank sum test was performed on the following pairs conditions: R6 versus R8, R6 versus R6 (default), and R8 versus R8 (default). The p-values for these tests are listed with statistically significant values (p < 0.01) marked with an asterisk.

and R6 beyond our control as well as the conservative nature of GM's identifier mapping process. For instance, if a gene symbol is found to map to multiple distinct genes, that symbol is dropped to avoid ambiguity. When considering all identifiers that GM recognizes (>273,000), including Uniprot IDs and synonyms, the net loss was 3% (8196).

The set of recognized identifiers determines which associations are imported from the third-party sources. If at least one interactor in an association is not recognized, that association is not imported, so the loss of gene symbols led to a loss in interactions, including those that might indirectly connect two genes with retained identifiers. These changes affected prediction performance. We corrected this issue in the latest release, R8, which addresses the identifier mapping issues introduced in R6 and now outperforms both that and R5.

Default networks

The GM dataset is represented as a collection of weighted, undirected graphs. The human dataset contains 164 million edges organized into 395 networks. Of these edges, 156 million are co-expression. To ensure responsiveness and high availability for the GM web server, it is not practical to always use all association networks for each prediction. Instead, GM uses a semi-manually curated subset of networks by default. This includes all the networks described above except predicted interactions that are not inferred through orthology, and select co-expression networks. To determine which co-expression networks to include, all the default networks and all co-expression networks are combined using the GO BP-based Simultaneous Weights algorithm, which assigns each network a weight. The top 20 co-expression networks with the highest weights are selected for membership in the default set. This results in only 6.8 million co-expression edges retained. The number of edges across all default networks is 13.7 million, which is about 8% of the total.

Assessment of default networks

Network Assessor was used to assess the predictive potential of default networks of R6 in isolation vs. all networks in R6 using human data. The same was done for R8. Five fold cross validation was used in each case against GO BP annotations that were downloaded on 18-Jul-2013 from the European Bioinformatics Institute GO MySQL database mirror. Following the work of Mostafavi and Morris (2010), GO terms were grouped based on the number of genes annotated by each term since GO terms

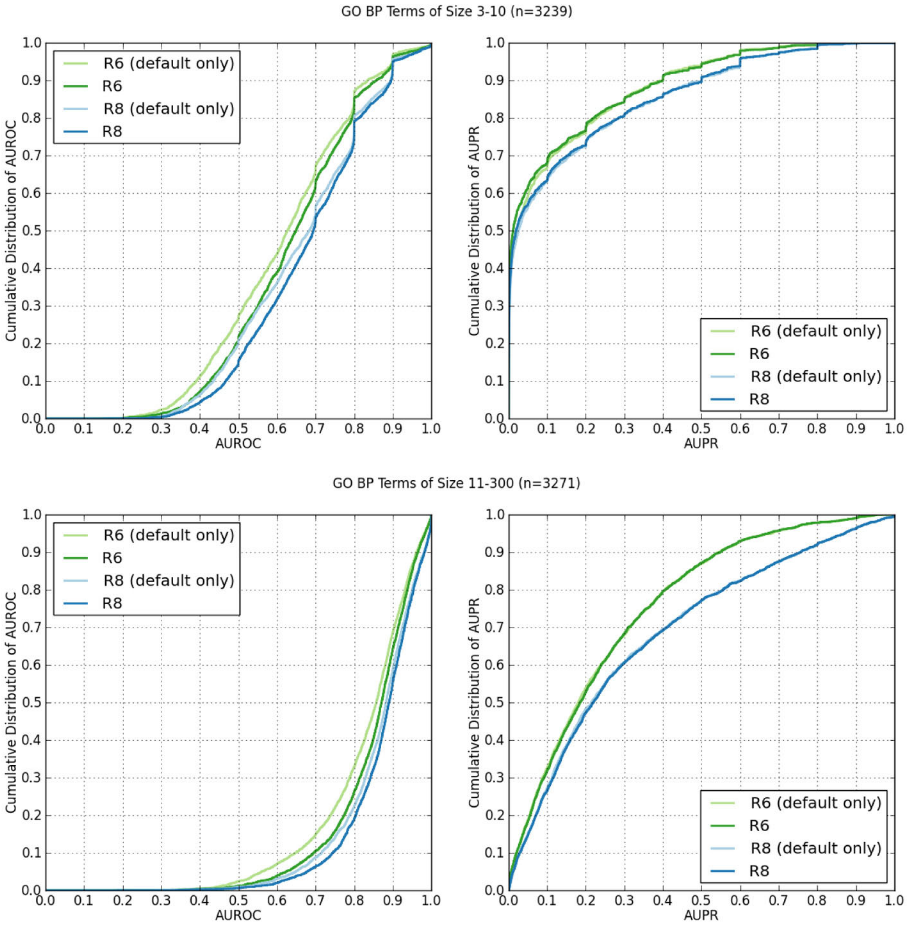


FIGURE 2 | Cumulative distributions of AUROC and AUPR of GO BP terms containing 3–10 annotations, and 11–300 from human network data from R6 and R8. The "(default)" suffix indicates only the networks selected by default on the web server were used from the data set. The lack of the suffix indicates all available networks were used.

with fewer annotations tend to exhibit worse prediction performance. This resulted in two partitions with similar sizes: 3–10 annotations ($n = 3239$) and 11–300 ($n = 3271$). These results are summarized in **Table 1**. In general, the full set of networks consistently performs better than the default set; except for AUPR on GO terms containing 3–10 annotations, where both the R6 and R8 defaults have higher AUPR than the full. This is likely due to overfitting since each network is assigned a weight by the integration algorithm, and the full set of networks contains more than twice as many networks as the default set. Performance also increased for all measures in R8 compared to R6.

Figure 2 shows the cumulative distributions of the AUROC and AUPR of GO BP terms containing 3–10 annotations ($n = 3239$), and 11–300 ($n = 3271$).

Network Assessor was also used to analyze the relative predictive potential of default networks, as well as other key types of networks by measuring the degree to which prediction performance decreases when they are removed. **Table 2** shows the AUROC and AUPR of GO BP terms containing 3–10 annotations, and 11–300 for R8 with default, co-expression, co-localization, genetic interaction, pathway, physical (protein) interaction, predicted, and shared protein domain networks removed, respectively. Median AUROC dropped by at least 4% when default or co-expression networks were removed. Median AUPR dropped by at least 14% when default, physical interaction, or shared protein domain networks were removed for terms containing 3–10 annotations. Median AUPR increased by almost 30% when co-expression networks were left out for the same terms. This is likely due to overfitting, which has been observed for smaller GO terms (Mostafavi et al., 2008). Median AUPR also increased by 3.4% when genetic interaction networks were left out for the same terms. In general, AUPR dropped by at least 15% when default networks were removed. AUPR also dropped substantially when predicted networks, most of which are derived through orthology from yeast, worm, fly, mouse, rat, were removed. This agrees with Klie et al. (2012) about the importance of intra-species transfer of annotations.

In **Figure 3**, the AUROC of most of the GO BP terms dropped when default networks were removed. The drop in AUPR was even more pronounced, regardless of the number of annotations in the GO term. This indicates an overall loss of precision and sensitivity when predictions were made without the default networks. The same analysis was performed for GO molecular function terms with similar results, which agrees with the findings of Mostafavi and Morris (2010).

CONCLUSION

Network Assessor has proven useful for measuring the impact of the changes that occur in the third-party sources from which the GM prediction web server derives its training data and can be used by others for similar analysis with custom data.

Table 2 | Median AUROC and AUPR for R8 when all networks are used (All) compared to when default (-default), co-expression (-coexp), co-localization (-coloc), genetic interaction (-gi), pathway (-path), physical (protein) interaction (-pi), predicted (-predict), and shared protein domain (-spd) networks are removed, respectively.

	All	-default	-coexp	-coloc	-gi	-path	-pi	-predict	-spd
R8									
Total edges	1.64×10^{8}	1.50×10^{8}	6.94×10^{6}	1.63×10^{8}	1.59×10^{8}	1.64×10^{8}	1.63×10^{8}	1.63×10^{8}	1.63×10^{8}
Edges removed from all	0	1.37×10^{7}	1.57×10^{8}	4.87×10^{5}	4.85×10^{6}	1.16×10^{5}	2.75×10^{5}	1.99×10^{5}	1.02×10^{6}
R8: GO TERM SIZE = 3–10									
Median AUROC	0.694	0.685	0.675	0.694	0.695	0.692	0.694	0.694	0.688
95% CI	±0.311	±0.309	±0.332	±0.311	±0.310	±0.311	±0.309	±0.310	±0.313
% difference from all		−1.3%	−2.8%	−0.1%	0.1%	−0.4%	−0.1%	−0.1%	−0.9%
versus all(*p*-value)		*2.17×10^{-3}	*1.43×10^{-23}	*5.04×10^{-67}	*4.23×10^{-109}	*1.46×10^{-4}	*5.86×10^{-13}	*4.05×10^{-17}	7.05×10^{-1}
R8: GO TERM SIZE = 11–300									
Median AUROC	0.890	0.866	0.864	0.889	0.890	0.887	0.887	0.890	0.880
95% CI	±0.195	±0.206	±0.224	±0.196	±0.195	±0.197	±0.199	±0.196	±0.199
% difference from all		−2.8%	−2.9%	−0.1%	0.0%	−0.3%	−0.4%	0.0%	−1.1%
versus all(*p*-value)		*0	*6.91×10^{-183}	1.91×10^{-1}	*8.24×10^{-20}	*1.54×10^{-22}	*1.01×10^{-27}	8.83×10^{-1}	*1.88×10^{-219}
R8: GO TERM SIZE = 3–10									
Median AUPR	0.019	0.011	0.025	0.019	0.020	0.019	0.016	0.018	0.017
95% CI	±0.409	±0.377	±0.406	±0.409	±0.408	±0.407	±0.392	±0.411	±0.404
% difference from all		−44.8%	29.8%	0.0%	3.4%	−4.0%	−16.4%	−4.8%	−14.0%
versus all(*p*-value)		*2.43×10^{-9}	*5.94×10^{-15}	*6.98×10^{-61}	*5.10×10^{-83}	*8.71×10^{-13}	*1.97×10^{-11}	*3.40×10^{-24}	*1.82×10^{-4}
R8: GO TERM SIZE = 11–300									
Median AUPR	0.220	0.186	0.209	0.219	0.220	0.218	0.206	0.218	0.215
95% CI	±0.528	±0.527	±0.530	±0.528	±0.528	±0.525	±0.528	±0.528	±0.527
% difference from all		−15.6%	−5.0%	−0.4%	0.1%	−0.6%	−6.2%	−0.9%	−2.2%
versus all(*p*-value)		*1.14×10^{-181}	*4.89×10^{-9}	3.20×10^{-1}	*3.63×10^{-38}	*3.92×10^{-16}	*1.20×10^{-43}	*3.40×10^{-24}	*1.82×10^{-4}

The number of edges removed for each analysis is also listed, as well as total edges used during assessment. The Wilcoxon rank sum test was used to compute the listed p-values, where significant values ($p < 0.01$) are marked with an asterisk.

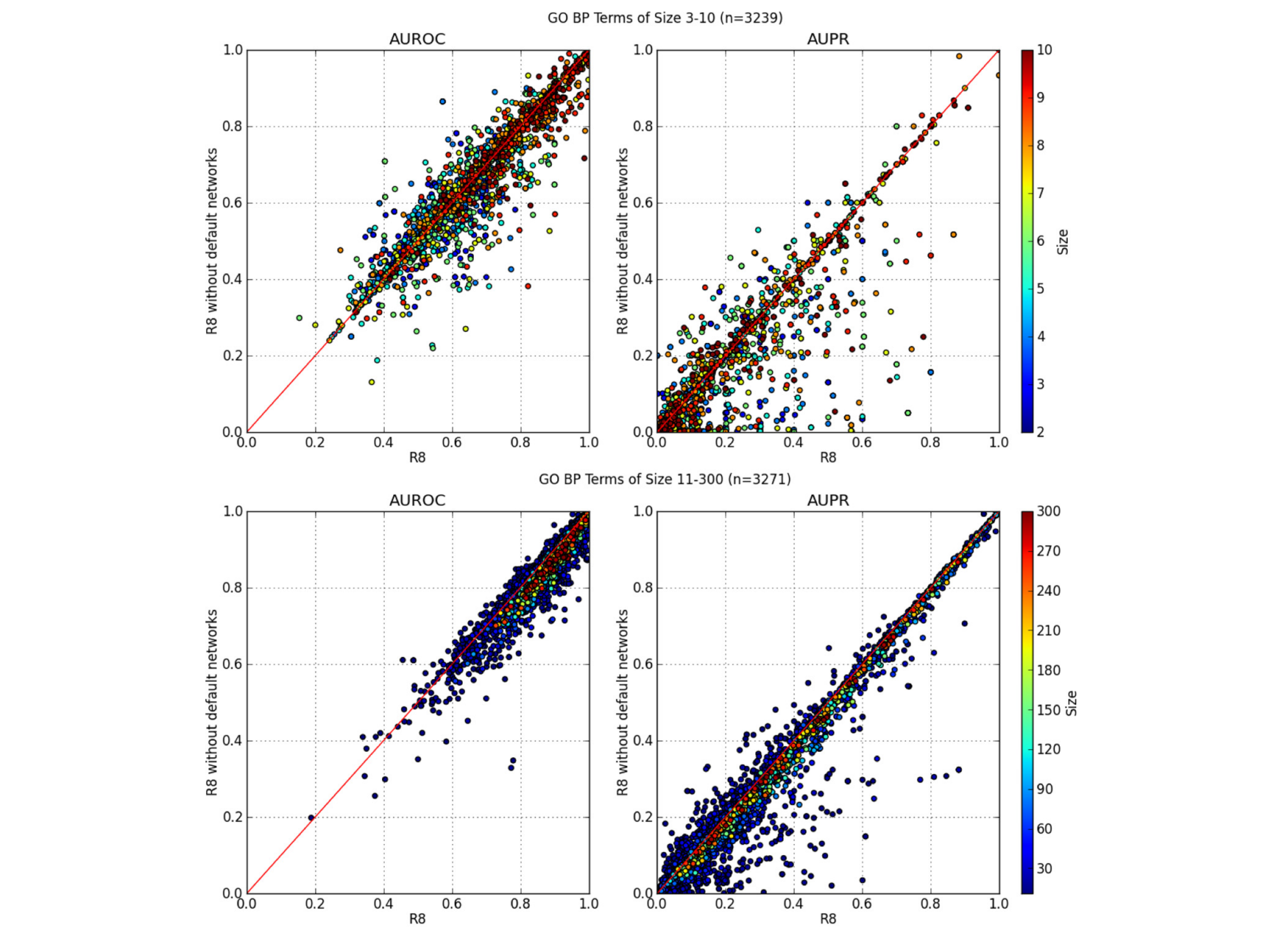


FIGURE 3 | AUROC and AUPR performance of each GO term. X-axis denotes performance using all networks from R8 while the Y-axis is R8 without default networks. GO terms containing 90 genes or more consistently performed better using all networks from R8.

Network Assessor is open-source and is part of the GM project. Code is available on request, although migration to GitHub is planned.

ACKNOWLEDGMENTS

We would like to thank the members of Morris and Bader labs, especially Magali Michaut and Ruth Isserlin, for their valuable feedback. This work was supported by the Ontario Government's Ministry of Research and Innovation via the Global Leadership Round in Genomics & Life Sciences (GL2) program. This work was also supported by a grant from the Biomedical Technology, Bioinformatics and Computational Biology (BBCB) at National Institute of General Medical Sciences (NIGMS), National Institutes of Health (NIH) [P41 GM103504].

REFERENCES

Barrett, T., Wilhite, S. E., Ledoux, P., Evangelista, C., Kim, I. F., Tomashevsky, M., et al. (2013). NCBI GEO: archive for functional genomics data sets–update. *Nucleic Acids Res.* 41, D991–5. doi: 10.1093/nar/gks1193

Brown, K. R., and Jurisica, I., (2007). Unequal evolutionary conservation of human protein interactions in interologous networks. *Genome Biol.* 8, R95. doi: 10.1186/gb-2007-8-5-r95

Cerami, E. G., Gross, B. E., Demir, E., Rodchenkov, I., Babur, Ö., Anwar, N., et al. (2011). Pathway Commons, a web resource for biological pathway data. *Nucl. Acids Res.* 39(Suppl. 1), D685–D690. doi: 10.1093/nar/gkq1039

Chatr-Aryamontri, A., Breitkreutz, B. J., Heinicke, S., Boucher, L., Winter, A., Stark, C., et al. (2013). The BioGRID interaction database: 2013 update. *Nucleic Acids Res.* 41, D816–D823. doi: 10.1093/nar/gks1158

Costanzo, M., Baryshnikova, A., Bellay, J., Kim, Y., Spear, E. D., Sevier, C. S., et al. (2010). The genetic landscape of a cell. *Science* 327, 425–431. doi: 10.1126/science.1180823

Deng, M., Tu, Z., Sun, F., and Chen, T. (2004). Mapping Gene Ontology to proteins based on protein-protein interaction data. *Bioinformatics* 20, 895–902. doi: 10.1093/bioinformatics/btg500

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recogn. Lett.* 27, 861–874. doi: 10.1016/j.patrec.2005.10.010

Flicek, P., Ahmed, I., Amode, M. R., Barrell, D., Beal, K., Brent, S., et al. (2013). Ensembl 2013. *Nucl. Acids Res.* 41, D48–D55. doi: 10.1093/nar/gks1236

Franceschini, A., Szklarczyk, D., Frankild, S., Kuhn, M., Simonovic, M., Roth, A., et al. (2013). STRING v9.1: protein-protein interaction networks, with increased coverage and integration. *Nucleic Acids Res.* 41, D808–15. doi: 10.1093/nar/gks1094

Gillis, J., and Pavlidis, P. (2011). The impact of multifunctional genes on "guilt by association" analysis. *PLoS ONE* 6:e17258. doi: 10.1371/journal.pone.0017258

Gillis, J., and Pavlidis, P. (2012). "Guilt by association" is the exception rather than the rule in gene networks. *PLoS Comput. Biol.* 8:e1002444. doi: 10.1371/journal.pcbi.1002444

Hishigaki, H., Nakai, K., Ono, T., Tanigami, A., and Takagi, T. (2001). Assessment of prediction accuracy of protein function from protein-protein interaction data. *Yeast* 18, 523–531. doi: 10.1002/yea.706

Hunter, S., Jones, P., Mitchell, A., Apweiler, R., Attwood, T. K., Bateman, A., et al. (2012). InterPro in 2011: new developments in the family and domain prediction database. *Nucleic Acids Res.* 40, D306–D312. doi: 10.1093/nar/gkr948

Kato, T., Kashima, H., and Sugiyama, M. (2009). Robust label propagation on multiple networks. *IEEE Trans. Neural Netw.* 20, 35–44. doi: 10.1109/TNN.2008.2003354

Kleessen, S., Klie, S., and Nikoloski, Z. (2013). Data integration through proximity-based networks provides biological principles of organization across scales.*Plant Cell* 25,1917–1927. doi: 10.1105/tpc.113.111039

Klie, S., Mutwil, M., Persson, S., and Nikoloski, Z. (2012). Inferring gene functions through dissection of relevance networks: interleaving the intra-and inter-species views. *Mol. Biosyst.* 8, 2233–2241. doi: 10.1039/c2mb25089f

Michaut, M., and Bader, G. D. (2012). Multiple genetic interaction experiments provide complementary information useful for gene function prediction. *PLoS Comput. Biol.* 8:e1002559. doi: 10.1371/journal.pcbi.1002559

Mostafavi, S., and Morris, Q. (2010). Fast integration of heterogeneous data sources for predicting gene function with limited annotation. *Bioinformatics* 26, 1759–1765. doi: 10.1093/bioinformatics/btq262

Mostafavi, S., Ray, D., Warde-Farley, D., Grouios, C., and Morris, Q. (2008). GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome Biol.* 9(Suppl. 1), S4. doi: 10.1186/gb-2008-9-s1-s4

Peña-Castillo, L., Tasan, M., Meyers, C. L., Lee, H., Joshi, T., Zhang, C., et al. (2008). A critical assessment of Mus musculus gene function prediction using integrated genomic evidence. *Genome Biol.* 9(Suppl. 1), S2. doi: 10.1186/gb-2008-9-s1-s2

Punta, M., Coggill, P. C., Eberhardt, R. Y., Mistry, J., Tate, J., Boursnell, C., et al. (2012). The Pfam protein families database. *Nucleic Acids Res.* 40, D290–D301. doi: 10.1093/nar/gkr1065

Razick, S., Magklaras, G., and Donaldson, I. M. (2008). iRefIndex: a consolidated protein interaction database with provenance. *BMC Bioinformatics* 9:405 doi: 10.1186/1471-2105-9-405

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., et al. (2004). The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Res.* 32, 5539–5545. doi: 10.1093/nar/gkh894

Schrynemackers, M., Küffner, R., and Geurts, P. (2013). On protocols and measures for the validation of supervised methods for the inference of biological networks. *Front. Genet.* 4:262. doi: 10.3389/fgene.2013.00262

Schwikowski, B., Uetz, P., and Fields, S. (2000). A network of protein-protein interactions in yeast. *Nat. Biotechnol.* 18, 1257–1261. doi: 10.1038/82360

Sharan, R., Ulitsky, I., and Shamir R. (2007). Network-based prediction of protein function. *Mol Sys Biol* 3, 88. doi: 10.1038/msb4100129

The Gene Ontology Consortium. (2000). Gene ontology: tool for the unification of biology. *Nat. Genet.* 25, 25–29. doi: 10.1038/75556

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). "Mining multi-label data," in *Data Mining and Knowledge Discovery Handbook,* eds O. Maimon and L. Rokach (New York, NY: Springer), 667–685.

Vazquez, A., Flammini, A., Maritan, A., and Vespignani, A. (2003). Global protein function prediction from protein-protein interaction networks. *Nat. Biotechnol.* 21, 697–700. doi: 10.1038/nbt825

Wang, H., Huang, H., and Ding, C. (2013). Function–function correlated multi-label protein function prediction over interaction networks. *J. Comput. Biol.* 20, 322–343. doi: 10.1089/cmb.2012.0272

Warde-Farley, D., Donaldson, S. L., Comes, O., Zuberi, K., Badrawi, R., Chao, P., et al. (2010). The GeneMANIA prediction server: biological network integration for gene prioritization and predicting gene function. *Nucl. Acids Res.* 38(Suppl. 2), W214–W220. doi: 10.1093/nar/gkq537

Wu, L. F., Hughes, T. R., Davierwala, A. P., Robinson, M. D., Stoughton, R., and Altschuler, S. J. (2002). Large-scale prediction of *Saccharomyces cerevisiae* gene function using overlapping transcriptional clusters. *Nat. Genet.* 31, 255–265. doi: 10.1038/ng906

Ye, P., Peyser, B. D., Pan, X., Boeke, J. D., Spencer, F. A., and Bader, J. S. (2005). Gene function prediction from congruent synthetic lethal interactions in yeast. *Mol. Syst. Biol.* 1, 2005.0026. doi: 10.1038/msb4100034

Yu, G., Rangwala, H., Domeniconi, C., Zhang, G., and Yu, Z. (2013). Protein function prediction using multi-label ensemble classification. *EEE/ACM Trans. Comput. Biol. Bioinform.* doi: 10.1109/TCBB.2013.111. [Epub ahead of print].

Zhang, W., Morris, Q. D., Chang, R., Shai, O., Bakowski, M. A., Mitsakakis, N., et al. (2004). The functional landscape of mouse gene expression. *J. Biol.* 3, 21. doi: 10.1186/jbiol16

Zuberi, K., Franz, M., Rodriguez, H., Montojo, J., Lopes, C. T., Bader, G. D., et al. (2013). GeneMANIA prediction server 2013 update. *Nucl. Acids Res.* 41, W115–W122. doi: 10.1093/nar/gkt533

Conflict of Interest Statement: 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.

Received: 06 January 2014; paper pending published: 31 January 2014; accepted: 21 April 2014; published online: 16 May 2014.

Citation: Montojo J, Zuberi K, Shao Q, Bader GD and Morris Q (2014) Network Assessor: an automated method for quantitative assessment of a network's potential for gene function prediction. Front. Genet. ***5****:123. doi: 10.3389/fgene.2014.00123*

This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.

frontiers in
BIOENGINEERING AND BIOTECHNOLOGY

HYPOTHESIS AND THEORY ARTICLE
published: 20 May 2014
doi: 10.3389/fbioe.2014.00013

A Bayesian framework that integrates heterogeneous data for inferring gene regulatory networks

Tapesh Santra*

Systems Biology Ireland, University College Dublin, Dublin, Ireland

Edited by:
Frank Emmert-Streib, Queen's University Belfast, UK

Reviewed by:
Sampsa Hautaniemi, University of Helsinki, Finland
Gianluca Bontempi, Université Libre de Bruxelles, Belgium

Correspondence:
Tapesh Santra, Systems Biology Ireland, University College Dublin, Belfield, Dublin 4, Republic of Ireland
e-mail: tapesh.santra@ucd.ie

Reconstruction of gene regulatory networks (GRNs) from experimental data is a fundamental challenge in systems biology. A number of computational approaches have been developed to infer GRNs from mRNA expression profiles. However, expression profiles alone are proving to be insufficient for inferring GRN topologies with reasonable accuracy. Recently, it has been shown that integration of external data sources (such as gene and protein sequence information, gene ontology data, protein–protein interactions) with mRNA expression profiles may increase the reliability of the inference process. Here, I propose a new approach that incorporates transcription factor binding sites (TFBS) and physical protein interactions (PPI) among transcription factors (TFs) in a Bayesian variable selection (BVS) algorithm which can infer GRNs from mRNA expression profiles subjected to genetic perturbations. Using real experimental data, I show that the integration of TFBS and PPI data with mRNA expression profiles leads to significantly more accurate networks than those inferred from expression profiles alone. Additionally, the performance of the proposed algorithm is compared with a series of least absolute shrinkage and selection operator (LASSO) regression-based network inference methods that can also incorporate prior knowledge in the inference framework. The results of this comparison suggest that BVS can outperform LASSO regression-based method in some circumstances.

Keywords: network inference, Bayesian statistics, data interpretation, statistical, variable selection, gene regulatory networks

INTRODUCTION

Understanding how genes regulate each other to orchestrate cellular phenotypes is a fundamental challenge of Biology. A straightforward way of uncovering gene regulatory networks (GRNs) is to perturb each gene of the network, e.g. by means of siRNAs and chemical inhibitors, and measure the effects of these perturbations on the expression of other genes in the network (Kholodenko et al., 2002; Wagner, 2002). However, the effects of such perturbations rapidly propagate through the entire network, causing widespread, global changes in the gene expressions, making it challenging to differentiate the direct interactions from the indirect ones. Several computational approaches were proposed to unmask the direct gene regulatory interactions by systematically analyzing perturbation responses (Kholodenko et al., 2002; Repsilber et al., 2002; Wagner, 2002; Gardner et al., 2003; Hartemink, 2005; Rogers and Girolami, 2005; de la Fuente and Makhecha, 2006; Margolin et al., 2006; Bansal et al., 2007). Many of these studies found that the steady-state perturbation responses of a gene are linearly dependent on the same of its direct regulators (Kholodenko et al., 2002; Gardner et al., 2003; Rogers and Girolami, 2005; de la Fuente and Makhecha, 2006; Bansal et al., 2007). These findings presented a unique opportunity of identifying direct genetic interactions by simply solving a set of linear equations. Although this approach seems simple in theory, implementing it in practice is not straightforward. First, biological measurements are noisy and contain experimental errors. The noise in biological datasets may cause significant errors while reconstructing GRNs by solving linear equations. Second, and perhaps most importantly, in order to solve these linear equations, one needs to perturb a GRN at least as many times as the number of genes in the network and measure the responses of all its genes after each perturbation (Kholodenko et al., 2002; Gardner et al., 2003; Rogers and Girolami, 2005; de la Fuente and Makhecha, 2006; Bansal et al., 2007). Therefore, reconstructing genome scale GRNs using the above method requires thousands (for simple organisms, e.g. bacteria, fungus, etc.) and often tens of thousands (for complex organisms such as mammals) of perturbation experiments that are time consuming and expensive. Most perturbation experiments, except those performed in some simple model organisms such as *Escherichia coli* (Baba et al., 2008) or yeast (Hughes et al., 2000), involve far fewer perturbations than the number of genes in the GRN. As a result, the datasets produced by these experiments do not have enough information for a full reconstruction (by solving linear equations) of the corresponding GRNs. Several statistical algorithms have been proposed to resolve this issue. For instance, some authors used singular value decomposition and linear regression (Yeung et al., 2002; Guthke et al., 2005; Zhang et al., 2010) to reconstruct GRNs using datasets obtained from a small number of perturbation experiments. Huang et al. (2010) used regulator filtering, forward selection, and linear regression for GRN reconstruction; and Imoto et al. (2003) used non-parametric regression embedded within a Bayesian network for the same purpose. Several other regression techniques such as the elastic net (Zou and Trevor, 2005; Friedman et al., 2010) and least absolute shrinkage and selection operator (LASSO; van Someren et al., 2003; Li and Yang, 2004; van Someren et al., 2006; Shimamura et al., 2007; Hecker et al., 2009, 2012; Lee et al., 2009; Charbonnier et al., 2010; Gustafsson and Hornquist, 2010; James et al., 2010; Pan et al., 2010; Peng et al., 2010; Wang et al., 2013) have also been widely used to reconstruct GRNs from noisy and insufficient perturbation response data.

Although many of these algorithms perform reasonably well, it is being increasingly clear that the accuracy of these algorithms can be significantly increased by integrating external data sources, e.g. gene sequence, single nucleotide polymorphism (SNP), protein–protein interaction (PPI), etc., in the network reconstruction process (Yeung et al., 2011; Lo et al., 2012). Public data repositories provide a rich resource of biological data related to gene regulation. Integrating data from these external data sources into network inference algorithms has become a primary focus of the systems and computational biology community. Previously, James et al. (2010) incorporated documented transcription factor binding sites (TFBS) information to infer the GRN of *E. coli.* Djebbari and Quackenbush (2008) used preliminary GRN derived from PubMed indexed literature and PPI databases as prior knowledge for their Bayesian network reconstruction algorithm. Zhu et al. (2004) combined TFBS and PPI data to infer GRNs. Imoto et al. (2003) used PPI, documented TFBS, and well studied pathways as prior information for their network inference method. Lee et al. (2009) presented a systematic way to incorporate various types of biological knowledge, such as the gene ontology (GO) annotations, data from ChIP–ChIP experiments, and a comprehensive collection of information about sequence polymorphisms. Yeung et al. (2005), Yeung et al. (2011), and Lo et al. (2012) developed a Bayesian model averaging approach to systematically integrate publicly available TFBS data, ChIP–ChIP data, physical interactions, genetic interactions, additional expression data, and literature curation.

This study is an extension of our previous work (Santra et al., 2013) which used a Bayesian framework that was designed to reconstruct biochemical networks by analyzing steady-state perturbation response data. In our previous study, we used Bayesian variable selection (BVS) algorithm to account for model uncertainty under noisy and insufficient data. Only generic topological knowledge such as sparsity of biochemical networks was used as prior information in the network reconstruction process. No external knowledge regarding potential interactions between network components was used to guide the inference process. The contributions of this study are four folds. First, a simple and an intuitive technique is proposed to incorporate external knowledge into the BVS framework in the form of a prior distribution. Second, a new way of integrating protein interactions among transcription factors (TFs) into the network inference framework is proposed. Although, PPI data were used previously (Zhu et al., 2008) in the context of GRN inference, the approach used by previous researchers was very different from the approach used in this study. For instance, protein interactions among target genes were used by Zhu et al. (2008) to determine co-regulation of multiple genes. Here, we use protein interaction among TFs to determine combinatorial regulations by multiple TFs. Third, as a proof of concept, the proposed methodology is applied to a gene expression dataset obtained from a liver-enriched TF regulatory network, revealing that it significantly outperforms our previous work. Finally, the performance of the proposed method is compared with a LASSO regression-based network inference method using publicly available gene expression datasets.

The rest of this study is organized as follows. In the next Section "Linear Model of Gene Regulation", I briefly discuss linear models of gene regulation, followed by a detailed description of the proposed BVS algorithm in Sections "The Bayesian Variable Selection Algorithm" and "Sampling Scheme for the Proposed BVS Framework." In Section "Integrating External Data to Formulate $P(A^i)$," I present a new method of integrating external data sources in the BVS formulation. An implementation of this method to infer a liver-specific GRN is then discussed in Section "Inferring Liver-Specific Gene Regulatory Network from Perturbation Response Data." In this section, I also compared the performance of the proposed BVS algorithm with our previous work. The results of comparing the proposed method with other network inference techniques are presented in Section "Inferring GRN of Human Breast Epithelium and Comparison with LASSO." Finally, in the conclusion section, I discuss the advantages and disadvantages of our algorithm and future directions.

LINEAR MODEL OF GENE REGULATION

When a GRN is perturbed, the effect of the perturbation rapidly propagates through the entire network, causing widespread, global changes in the expression levels of its genes. It has been shown (Rogers and Girolami, 2005; Bansal et al., 2007; Lo et al., 2012) that the responses ($\boldsymbol{x^i} = \{\mathrm{x}_{ij}, j = 1, \ldots, n_\mathrm{p}\}$) of a gene ($g_i$), to a series of ($n_\mathrm{p}$) perturbations, are linearly dependent on the responses ($\boldsymbol{X^i} = \{x_{kj}, k = 1, \ldots, n_i, j = 1, \ldots, n_\mathrm{p}, k \neq i\}$) of its direct regulators ($\boldsymbol{g^i} = \{g_k, k = 1, \ldots, n_i, k \neq i\}$), i.e.,

$$\boldsymbol{x^i} = \boldsymbol{X^i}^T \boldsymbol{\beta^i} \tag{1}$$

where n_i is the number of regulators of the gene (g_i), and $\boldsymbol{\beta^i} = \{\beta_{ik}, k = 1, \ldots, n_i, k \neq i\}$ are the linear coefficients that represent the strengths and types of the interactions between the gene (g_i) and its direct regulators ($\boldsymbol{g^i}$).

The measurements of the expression levels usually contain experimental errors, and may not exactly fit into the above Eq. 1. The difference between the left and right hand side of Eq. 1 caused by such errors are called the "residuals." In order to compensate for errors, the residuals are added to Eq. 1 leading to,

$$\boldsymbol{x^i} = \boldsymbol{X^i}^T \boldsymbol{\beta^i} + \boldsymbol{\epsilon^i} \tag{2}$$

where $\boldsymbol{\epsilon^i} = \{\epsilon_{ij}, j = 1, \ldots, n_\mathrm{p}\}$ represents the residuals caused by measurement errors. It can be easily showed that the residual variables (ϵ_{ij}) are linear combinations of the individual measurement errors associated with the perturbation responses of the gene (g_i) and its regulators ($\boldsymbol{g^i}$) (Kariya and Kurata, 2004). Since, the measurement errors are random in nature, the residual variables are also random variables, and by central limit theorem, these variables have Gaussian distribution (Kariya and Kurata, 2004). It is further assumed that the residual variables ($\boldsymbol{\epsilon^i}$) are independent of each other and have 0 mean and variance σ^2 which depend on the extent of experimental/measurement error in the dataset (Rogers and Girolami, 2005; de la Fuente and Makhecha, 2006; Bansal et al., 2007; Vignes et al., 2011; Santra et al., 2013).

To identify the direct regulators ($\boldsymbol{g^i}$) of the gene (g_i), one needs to calculate $\boldsymbol{\beta^i}$ by solving Eq. 2 in a least-square sense. The elements (β_{ik}) of $\boldsymbol{\beta^i}$ whose absolute values are significantly >0 are then selected as direct interactions, and the corresponding genes (g_k) are considered to be the direct regulators of g_i. However, solving Eq. 2 requires at least as many perturbations as the number of genes (n) in the network (Kholodenko et al., 2002; Rogers and Girolami, 2005; de la Fuente and Makhecha, 2006; Santra et al., 2013). Under most circumstances, it is not possible to perform so many perturbation experiments, and therefore, in such cases, a full GRN reconstruction is not feasible by solving Eq. 2, either exactly or in a least-square sense. This issue is resolved by variable selection algorithms.

BAYESIAN VARIABLE SELECTION ALGORITHM

Variable selection algorithms find the most likely set of regulators ($\boldsymbol{g^i}$) for each gene (g_i) by iteratively solving Eq. 2. It should be noted that the inferred interactions between a gene (g_i) and its regulators ($\boldsymbol{g^i}$) may not always represent causal relationships. In many cases, these interactions represent "acausal" dependencies between gene expressions (Guyon and Elisseeff, 2003). Yet, it has been shown that variable selection algorithms can infer gene regulatory programs with reasonable accuracy (Yeung et al., 2005, 2011; Lo et al., 2012). The mechanism of a simple variable selection technique in the context of GRN reconstruction is described below.

(a) First, a random set of genes $\left(\boldsymbol{g_1^i}\right)$ are selected as the potential regulators of a gene (g_i), and the least-square estimates $\left(\boldsymbol{\beta_1^i}\right)$ of the corresponding interaction strengths and the resulting sum of square error $\left(\epsilon_{i1}^{\mathrm{SOS}} = ||\boldsymbol{\epsilon_1^i}||^2\right)$ are calculated.

(b) At the next iteration, a different set of genes $\left(\boldsymbol{g_2^i}\right)$ are selected as the potential direct regulators of gene g_i, and again, the least-square estimates $\left(\boldsymbol{\beta_2^i}\right)$ of corresponding interaction strengths and the resulting sum of square error $\left(\epsilon_{i2}^{\mathrm{SOS}} = ||\boldsymbol{\epsilon_2^i}||^2\right)$ are calculated.

(c) The newly calculated sum of square error $\left(\epsilon_{i2}^{\mathrm{SOS}}\right)$ is then compared with the one $\left(\epsilon_{i1}^{\mathrm{SOS}}\right)$ calculated in the previous iteration. If $\epsilon_{i2}^{\mathrm{SOS}} < \epsilon_{i1}^{\mathrm{SOS}}$, then the new set of potential regulators $\left(\boldsymbol{g_2^i}\right)$ is considered more likely to directly regulate g_i than the previous one $\left(\boldsymbol{g_1^i}\right)$, otherwise the old set is retained as the most likely potential regulators.

(d) For each gene (g_i), the above procedure is repeated for all possible combination of potential regulators until a set of regulators is found that has the minimum sum of squared error.

The above scheme is simple in theory, but there are some major obstacles in implementing it in practice. For instance, if we want to reconstruct a GRN involving 1000 genes, then, for each gene, we need to iterate through 2^{999} possible combinations of potential regulators to find its most likely direct regulators. Iterating through so many possible combinations is not feasible even for the most advanced computing systems. Therefore, we must adopt a smarter strategy to find the most likely set of regulators of each gene in a GRN. BVS algorithms (in general) implement efficient search strategies to identify the most likely regulators of a gene in a reasonable time. Here, I adopted a BVS framework which is similar to our previous work (Santra et al., 2013) with a few exceptions.

To formulate the BVS algorithm, it is convenient to represent the topology of a GRN using a binary "adjacency" matrix ($\boldsymbol{A}$). A non-zero entry ($A_{ik} = 1$, $k \neq i$) of this matrix represents direct regulation of one gene (g_i) by another (g_k, $k{\neq}i$), whereas the zero elements indicate no direct regulation. Consequently, the non-zero elements of the ith row ($\boldsymbol{A^i} = \{A_{ik}, k = 1, \ldots, n, k \neq i\}$) of this matrix represent interactions between the gene g_i and its direct regulators ($\boldsymbol{g^i}$). Note that the binary adjacency matrix ($\boldsymbol{A}$) and the matrix of interaction strengths ($\boldsymbol{\beta}$) are closely related, since absence of direct interaction ($A_{ik} = 0$, $i \neq k$) between two genes (g_i, g_k) implies zero interaction strength ($\beta_{ik} = 0$, $i \neq k$). In other words, the elements (β_{ik}, $i \neq k$) of the interaction strength matrix ($\boldsymbol{\beta}$) corresponding to the zero elements ($A_{ik} = 0$, $i \neq k$) of the binary adjacency matrix ($\boldsymbol{A}$) are also zero. Therefore, finding the most likely direct regulators of a gene (g_i) amounts to finding the most likely combination of 0s and 1s in the ith row ($\boldsymbol{A^i}$) of the binary matrix $\boldsymbol{A}$.

To avoid iterating through all possible combinations of $\boldsymbol{A^i}$, BVS algorithms adopt a Bayesian approach. Bayesian algorithms closely mimic the natural learning process of human brain that updates its knowledge about certain events when it receives new information about the event. In these algorithms, the prior knowledge about a certain event is represented by its prior distribution which assigns a prior probability to each possible outcome of the event. When new information becomes available, the prior probabilities are updated using Bayes' theorem. The updated probability distribution is known as the posterior distribution. The posterior distributions represent our up-to-date knowledge about a certain event based on the data that have been recently available.

In the context of GRN reconstruction, any prior knowledge about the possible regulators ($\boldsymbol{g^i}$) of each gene (g_i) is encoded in the prior distributions ($P(\boldsymbol{A^i})$) of the binary vectors $\boldsymbol{A^i}$. In our previous work (Santra et al., 2013), we formulated the prior distribution $P(\boldsymbol{A^i})$ to penalize gene regulation models with too many regulators and favored sparse models where each gene is regulated by a small number of regulators. No other external knowledge was used to formulate the prior distribution of $\boldsymbol{A^i}$. Here, we take a different approach and formulate a more informative prior distribution of $\boldsymbol{A^i}$ by integrating TFBS and PPI between TFs. The process of integrating PPI and TFBS data into the prior distribution of $\boldsymbol{A^i}$ is an important aspect of data integration and will be discussed in detail in the next section.

Prior information about the possible values of the interaction strengths ($\boldsymbol{\beta^i}$) is rarely available. In the absence of any specific prior knowledge of the possible values of $\boldsymbol{\beta^i}$, it is safe to assume that its non-zero elements can take a wide range of positive or negative values depending on whether the corresponding interaction is activating or repressing. The zero elements represent no direct interaction and correspond to the zero elements of $\boldsymbol{A^i}$. This assumption is formulated by assigning a multivariate Gaussian prior to the non-zero elements of $\boldsymbol{\beta^i}$. The prior distribution of $\boldsymbol{\beta^i}$ is assumed to have zero mean and covariance matrix $\boldsymbol{V}_{\boldsymbol{\beta^i}}$, which is a ($n_i \times n_{\mathrm{i}}$) matrix that represents our prior knowledge about the possible ranges of values of $\boldsymbol{\beta^i}$. A common approach is to assume that the prior covariance matrix ($\boldsymbol{V}_{\boldsymbol{\beta}^i}$) of $\boldsymbol{\beta}^{\boldsymbol{i}}$ is proportional to the scaled fisher information matrix (FIM) of $\boldsymbol{\beta}^{\boldsymbol{i}}$, i.e. $\boldsymbol{V}_{\boldsymbol{\beta}^i} = c\sigma^2 \left(\boldsymbol{X}^{\boldsymbol{i}^T} \boldsymbol{X}^{\boldsymbol{i}} \right)^{-1}$, where c is the proportionality constant (also known as Zellner's constant) which determines the span of the prior distribution of $\boldsymbol{\beta}^{\boldsymbol{i}}$ (Zellner, 1986; Ishwaran and Rao, 2005; Gupta and Ibrahim, 2009) and σ^2 is the scaling factor which is the same as the variances of the residual variables ϵ_{ij}. The above formulation of the covariance matrix assumes that the variances/covariances of the interaction strengths depend not only on the inherent variability of the perturbation responses, but also on the variance of the measurement errors. It was shown by other researchers that the choice of the proportionality constant c has a significant impact on the performance of BVS algorithms and several studies were conducted to find the most appropriate value of c (George and Foster, 2000; Fernández et al., 2001; Hansen and Yu, 2001; Liang et al., 2008). Fernández et al. (2001) demonstrated that among the commonly used values, $c = \max\left(n_p, n_i^2\right)$ performs the best in most scenarios. Therefore, this value was chosen for the BVS framework presented in this study.

The prior knowledge about the noise variance σ^2 is incorporated in its prior distribution. Previously, the noise variance σ^2 was assumed to have a gamma distribution with shape and scale parameters, α and β, respectively (Santra et al., 2013). The values of these parameters were set to 1 to ensure a flat prior, which represents our lack of prior knowledge about extent of noise in the dataset. Here, in order to avoid extra hyper parameters (α, β), we assumed that σ^2 has Jeffrey's prior (Fernández et al., 2001), i.e.$p(\sigma^2) \sim \frac{1}{\sigma^2}$, which is an uninformative "improper" prior that relies on the notion that noises in biological data are unlikely to cause very large residuals in the linear models.

These prior distributions can then be updated to posterior distributions based on the measured perturbation responses of the network using Bayes formula. Here, we are interested in the posterior distributions of binary vectors $\boldsymbol{A}^{\boldsymbol{i}}$, $i = 1, \ldots, n$, since these vectors represent the network topology. It is straightforward to show that the posterior distribution ($P(\boldsymbol{A}^{\boldsymbol{i}}|\boldsymbol{x}^{\boldsymbol{i}}, \boldsymbol{X}^{\boldsymbol{i}})$) of $\boldsymbol{A}^{\boldsymbol{i}}$ given the perturbation responses of gene g_i and its regulators is (Liang et al., 2008; Note 1 in Supplementary Material)

$$P\left(\boldsymbol{A}^{\boldsymbol{i}}|\boldsymbol{x}^{\boldsymbol{i}}, \boldsymbol{X}^{\boldsymbol{i}}\right) \propto \left[(1+c)^{-\left(\frac{n_i+1}{2}\right)} \left(1 - \frac{c}{1+c} R^2\right)^{\frac{-(n_p-1)}{2}} \right] P\left(\boldsymbol{A}^{\boldsymbol{i}}\right) \tag{3}$$

here $R^2 = 1 - \frac{\left(x^i - X^{i^T} \widehat{\beta^i}\right)^T \left(x^i - X^{i^T} \widehat{\beta^i}\right)}{\left(x^i - \overline{x^i}\right)^T \left(x^i - \overline{x^i}\right)}$ is the coefficient of determination of the linear model shown in Eq. 2, where $\widehat{\boldsymbol{\beta}^{\boldsymbol{i}}} = \left(\boldsymbol{X}^{\boldsymbol{i}^T} \boldsymbol{X}^{\boldsymbol{i}}\right)^{-1} \boldsymbol{X}^{\boldsymbol{i}^T} \boldsymbol{x}^{\boldsymbol{i}}$ is the least-square estimate of $\boldsymbol{\beta}^{\boldsymbol{i}}$, and $\overline{\boldsymbol{x}}^{\boldsymbol{i}}$ is the sample average of $\boldsymbol{x}^{\boldsymbol{i}}$.

Finding the most likely regulators of gene g_i is equivalent to finding the configuration of $\boldsymbol{A}^{\boldsymbol{i}}$ that maximizes the above posterior probability (Eq. 3). But, as discussed before, finding this configuration requires iterating through all possible configurations of $\boldsymbol{A}^{\boldsymbol{i}}$, which is hardly possible for large networks. An alternative approach is to estimate the "expected" configuration of $\boldsymbol{A}^{\boldsymbol{i}}$ using model averaging techniques that identify a number of "good enough" configurations instead of a single "best" configuration. The average of these good configurations is commonly used as an approximation of the "expected" configuration of $\boldsymbol{A}^{\boldsymbol{i}}$. The "good enough" configurations of $\boldsymbol{A}^{\boldsymbol{i}}$ can be determined in reasonable time by drawing samples from the above posterior distribution (Eq. 3) using a Markov Chain Monte Carlo (MCMC)-based sampling algorithm.

SAMPLING SCHEME FOR THE PROPOSED BVS FRAMEWORK

A typical MCMC-based sampling algorithm iteratively explores different configurations of $\boldsymbol{A}^{\boldsymbol{i}}$ in order to find those with relatively high posterior probability. In each iteration, it calculates the posterior probability of the current and a proposed new configuration of $\boldsymbol{A}^{\boldsymbol{i}}$. However, in some cases, it is not possible to calculate the posterior probability of certain configurations of $\boldsymbol{A}^{\boldsymbol{i}}$. For instance, when $n_i \gg n_p$, i.e. the number of 1s in $\boldsymbol{A}^{\boldsymbol{i}}$ is larger than the number of perturbations, then the corresponding data matrix $\boldsymbol{X}^{\boldsymbol{i}}$ has dimensions $n_\mathrm{p} \times n_\mathrm{i}$ and suffers from rank deficiency. Therefore, the Gram matrix $\boldsymbol{X}^{\boldsymbol{i}^T} \boldsymbol{X}^{\boldsymbol{i}}$ is non-invertible and the corresponding coefficient of determination ($\boldsymbol{R}$) and the posterior probability ($P(\boldsymbol{A}^{\boldsymbol{i}}|\boldsymbol{x}^{\boldsymbol{i}}, \boldsymbol{X}^{\boldsymbol{i}})$) do not exist. Previously (Santra et al., 2013), we addressed this issue by adding a diagonal loading ($\boldsymbol{X}^{\boldsymbol{i}^T} \boldsymbol{X}^{\boldsymbol{i}} + \boldsymbol{\delta I}$) to the Gram matrix, ensuring its invertibility. However, this approach requires the estimation of an optimal value for the loading constant (δ), which adds to the complexity of the sampling process. Additionally, the effects of diagonal loading on the overall outcome of BVS algorithms are not well understood. In this study, a different strategy is adopted to address the above issue. Here, in order to avoid rank deficiency, the search space (ζ) of the MCMC algorithm is constrained to only those configurations of $\boldsymbol{A}^{\boldsymbol{i}}$ which has less number of 1s than the number of perturbations, i.e. $n_i < n_\mathrm{p}$. The restricted search space is denoted by ζ_{n_p} $\left(\zeta_{n_\mathrm{p}} \subset \zeta\right)$, where the subscript n_p indicates the upper limit on the number of 1s in the configurations of $\boldsymbol{A}^{\boldsymbol{i}}$. The above approach has two major advantages over the previous method. First, it ensures the existence of the posterior probability without artificial diagonal loading of the Gram matrix. Second, it decreases the computational complexity of the MCMC algorithm by reducing the size of the data matrix $\boldsymbol{X}^{\boldsymbol{i}}$. This property makes this approach particularly attractive for inferring large GRNs where computational complexity is a major issue for MCMC-based variable selection algorithms. The computational cost of sampling can be significantly reduced by further restricting the search space to an even smaller subspace $\left(\zeta_k \subseteq \zeta_{n_\mathrm{p}}\right)$, which contains only those configurations of $\boldsymbol{A}^{\boldsymbol{i}}$ that have less than k (where $k \leq n_\mathrm{p}$) numbers of 1s. Restricting the search space to ζ_k implies that the MCMC algorithm will explore regulatory programs (configurations of $\boldsymbol{A}^{\boldsymbol{i}}$) consisting of at most $k \leq n_\mathrm{p}$ regulators for each gene (g_i). For accurate network inference, it is therefore desirable to assign the restriction parameter (k) a reasonable value that is not far from the ground truth. Although, there is no easy way of determining an optimal k, one can use prior information about the topology of the network to have a broad estimate of this parameter. This is discussed in the results section where the proposed algorithm is implemented on experimental data sets to infer GRNs. In the rest of this section, I continue with the discussion of the MCMC-based sampling algorithm, which is used in this study to explore the restricted search space (ζ_k) of potential gene regulatory programs.

A Metropolis–Hastings algorithm was implemented to systematically explore ζ_k and identify highly probable regulatory programs ($\boldsymbol{A^i}$). The sampling algorithm starts with a random configuration of $\boldsymbol{A^i} \in \zeta_k$. A new configuration $\boldsymbol{A^{i'}} \in \zeta_k$ is then proposed based on a proposal distribution Q. The proposal distribution (Q) is formulated as follows. Let $\eta(\boldsymbol{A^i}) \subseteq \zeta_k$ denote a set of binary vectors consisting of all possible configurations that can be obtained by changing one of the elements of $\boldsymbol{A^i}$ from 0 to 1 or vice versa. Define a proposal distribution Q as follows.

$$Q\left(\boldsymbol{A^i}, \boldsymbol{A^{i'}}\right) = \begin{cases} = \frac{1}{|\eta(\boldsymbol{A^i})|} & \mathit{if}\, \boldsymbol{A^{i'}} \in \eta\left(\boldsymbol{A^i}\right) \\ 0 & \mathit{if}\, \boldsymbol{A^{i'}} \notin \eta\left(\boldsymbol{A^i}\right) \end{cases} \tag{4}$$

Based on the above proposal distribution, an acceptance ratio $\alpha = \frac{P\left(\boldsymbol{A^{i'}}|\boldsymbol{X^{i'}},\boldsymbol{x^i}\right)Q\left(A^{i'},A^i\right)}{P\left(\boldsymbol{A^i}|\boldsymbol{X^i},\boldsymbol{x^i}\right)Q\left(A^i,A^{i'}\right)}$ is computed. The proposed new configuration $\boldsymbol{A^{i'}}$ is then accepted with probability $\min(1, \alpha)$. If accepted, $\boldsymbol{A^{i'}}$ is added to the sequence of drawn samples and becomes the current configuration. Else, $\boldsymbol{A^i}$ remains the current configuration. Repeating this procedure in an iterative manner gives rise to an irreducible Markov chain in the restricted search space (ζ_k). This Markov chain asymptotically converges (Geyer, 2011) to the desired posterior ($P(\boldsymbol{A^i}|\boldsymbol{x^i}, \boldsymbol{X^i})$). Upon convergence, the samples drawn by the chain resemble those drawn from the posterior ($P(\boldsymbol{A^i}|\boldsymbol{x^i}, \boldsymbol{X^i})$), and therefore, the most probable configurations of $\boldsymbol{A^i}$ appear more frequently in the drawn samples than the improbable ones. These samples are then used to determine an "average" or "expected" regulatory program for each gene (g_i). The expected probability that a gene (g_i) is regulated by another gene (g_k) is estimated by calculating the ratio of the number (n_{ij}) of samples whose jth element is 1 to the total number (n_s) of samples, i.e. $P\left(A_{ij} = 1|\boldsymbol{x_i}, \boldsymbol{X^i}\right) = \frac{n_{ij}}{n_s}$ (Mukherjee and Speed, 2008). Calculating this probability for each pair of genes results in a probabilistic representation of the network topology.

The above sampling algorithm draws samples from the posterior distribution of $\boldsymbol{A^i}$ (Eq. 3) which depends on its prior distribution. This can be exploited to incorporate prior knowledge from external data sources into the BVS algorithm. To do so, the prior distribution ($\mathrm{P}(\boldsymbol{A^i})$) needs to be formulated in such a way that it favors the likely interactions supported by external data sources. This will bias the posterior of $\boldsymbol{A^i}$ toward the interactions that are supported by external data. Below, I show a scheme that integrates TFBS with PPI information to formulate the prior distribution $P(\boldsymbol{A^i})$.

INTEGRATING EXTERNAL DATA TO FORMULATE $P(A^i)$

Genes regulate each other via several mechanisms, e.g. transcriptional regulation, methylation, histone acetylation, etc. Among the known mechanisms of gene regulation, transcriptional regulation via TFs is perhaps the most well-studied gene regulatory mechanism. In the case of transcriptional regulation, proteins produced by regulatory genes undergo post-translational modifications and then either directly bind to the promoter regions of target genes or form multi-protein transcription factor complexes (TFCs) that bind to the gene promoters and regulate the activity of the corresponding genes. The regulatory proteins and TFCs bind genes at specific locations containing specific nucleotide sequences, commonly referred to as TFBS. These binding sites are experimentally determined by ChIP–ChIP experiments (Hughes et al., 2000) and/or computationally predicted by statistical algorithms (Matys et al., 2006; Bryne et al., 2008; Bailey et al., 2009; Ernst et al., 2010). There are a number of databases that contains vast amount of information on binding specificities (TFBS) of several TFs and TFCs (Matys et al., 2006; Bryne et al., 2008; Bailey et al., 2009). However, there are some limitations of incorporating these informations as prior knowledge into a network inference algorithm. First, the binding specificities are known only for a fraction of all TFs and TFCs that are found in nature. For a large number of TFs and TFCs, such information is unavailable. It is challenging to interpret the unavailability of information in an unambiguous manner. For instance, it is difficult to determine whether the lack of information represents absence of interaction or simply lack of knowledge about the presence of interaction. Second, TFs may indirectly regulate genes by forming protein complexes (TFCs) with other TFs which directly bind to gene promoters. Many of these indirect regulations are not well characterized, further contributing to the incompleteness of prior knowledge regarding gene regulation.

To address the above issues, I propose a simple scheme of incorporating available knowledge into the prior distribution of $\boldsymbol{A^i}$. The proposed prior distribution favors potential regulatory interactions supported by TFBS data available in public databases. However, it does not exclude the possibility of potentially new interactions that are not supported by external sources. Furthermore, it uses information regarding protein interactions among the TFs to determine potential indirect gene regulations. These indirect regulatory interactions, along with the TFBS specificities, are then collectively used as potential regulatory interactions in the formulation of the prior distribution of $\boldsymbol{A^i}$. A step-by-step description of using external data sources to formulate the prior distribution of $\boldsymbol{A^i}$ is shown below.

Step 1: First, TFBS information are collected from multiple external sources, e.g. public databases such as HTRIDB (Bovolenta et al., 2012), ENCODE (Hughes et al., 2000), KEGG (Ogata et al., 1999), ConsensusPathDB (Kamburov et al., 2011), etc., published literature (Ernst et al., 2010), computational TFBS prediction services such as MEME (Bailey et al., 2009), TRANSFAC (Bryne et al., 2008), JASPER (Matys et al., 2006), TRED (Jiang et al., 2007), etc.

Step 2: Next, information regarding PPIs among known TFs are obtained from publicly available sources. Recently, Ravasi et al. (2010) determined a comprehensive map of physical interactions among mammalian TFs using mammalian two-hybrid system. They identified around 800 protein interactions among human and mouse TFs. Arguably, this dataset is the most reliable source of information regarding protein interactions among TFs and is used in the large-scale GRN inference study later in this study. However, Ravasi et al.'s study does not cover all mammalian TFs, in which case proteins interaction databases such as

STRING (Szklarczyk et al., 2011), HPRD (Keshava Prasad et al., 2009), IntAct (Kerrien et al., 2012), BIND (Bader et al., 2003), KEGG (Ogata et al., 1999) is used to determine PPI between TFs. It should be noted that many of these databases store functional and computationally predicted PPIs which may not always represent physical protein bindings. Since, we are interested in physical interactions among TFs, only physical PPIs are carefully selected from the above databases, functional and computationally predicted PPIs are excluded from the list of potential TF–TF protein interactions.

Step 3: The above information is then used to build a prior network that contains both direct and indirect regulations supported by external data. Potential direct regulations are identified using TFBS information as described above (see Step 1). Potential indirect regulations are identified based on the assumption that if a TF binds to another TF which targets a certain gene, then the former indirectly regulates the target of the later (**Figure 1**). Both direct and indirect regulations are incorporated in the prior network as potential transcriptional interactions. The prior network is represented by a weighted adjacency matrix ($\mathbf{\Gamma}$). The non-zero elements of this matrix represent potential transcriptional regulations supported by TFBS and PPI data. The value of a non-zero element ($\Gamma_{ij} \neq 0$) represents our confidence on the regulation of a gene (g_i) by another (g_j). In this study, equal confidence is placed on all potential transcriptional regulations that are supported by TFBS and PPI data, i.e., $\Gamma_{ij} = \alpha_c$ if gene g_i has a TFBS for g_j or any of its binding partners. Here, α_c is called the confidence parameter. The ith row ($\mathbf{\Gamma}^{\boldsymbol{i}}$) of the prior adjacency matrix ($\mathbf{\Gamma}$) represents our prior knowledge about the regulatory program of gene g_i and is used to formulate the prior distribution of the binary vector $\boldsymbol{A^i}$ in the following manner.

$$P(\boldsymbol{A^i}) \propto \exp(\mathbf{\Gamma}^{\boldsymbol{i}^T} \boldsymbol{A^i}) : \boldsymbol{A^i} \in \zeta_k \\ = 0 \text{ otherwise.} \qquad (5)$$

The above prior distribution ensures that the prior probability of $\boldsymbol{A^i} \in \zeta_k$ depends only on the number of interactions ($A_{ij} = 1$) which are supported by prior information ($\Gamma_{ij} = \alpha_c$). This implies that if two different configurations of $\boldsymbol{A^i}$ have different numbers of potentially new interactions ($A_{ij} = 1$, $\Gamma_{ij} = 0$) but the same number of previously known interactions ($A_{ij} = 1$, $\Gamma_{ij} = \alpha_c$), then these two configurations have the same prior probability. Therefore, the above prior distribution (Eq. 5) favors regulatory programs (configurations of $\boldsymbol{A^i}$) that have large number of known interactions ($\Gamma_{ij} = \alpha_c$) but does not penalize the presence of previously unknown interactions, allowing such interactions to be seamlessly inferred by the variable selection algorithm.

As a proof of concept, I implemented the above BVS algorithm to reconstruct a liver-specific transcription regulatory network by analyzing perturbation response data. To show the effectiveness of integrating TFBS and PPI data in the BVS framework, I used four different prior settings for $\boldsymbol{A^i}$. In the first setting, no external data source was used to formulate the prior distribution of $\boldsymbol{A^i}$ and all possible regulatory programs (configurations of $\boldsymbol{A^i}$) were considered equally likely *a priori*. In the second setting, no external data sources were used, but the prior distribution of $\boldsymbol{A^i}$ was designed to favor sparse regulatory programs, i.e., the configurations of $\boldsymbol{A^i}$ which has relatively fewer non-zero elements than zero elements. This approach is similar to that we adopted in our previous work (Santra et al., 2013). In the third setting, a prior network was constructed using only direct regulatory interactions that were predicted from publicly available TFBS information. This prior network was then used to formulate the prior distribution of $\boldsymbol{A^i}$ as shown in Eq. 5. In the final setting, I used both direct and indirect regulatory interactions that were predicted from both TFBS and PPI interaction data to construct the prior network. This prior network was then used to formulate the prior distribution of $\boldsymbol{A^i}$. The results of the above analysis are described in detail in the following section.

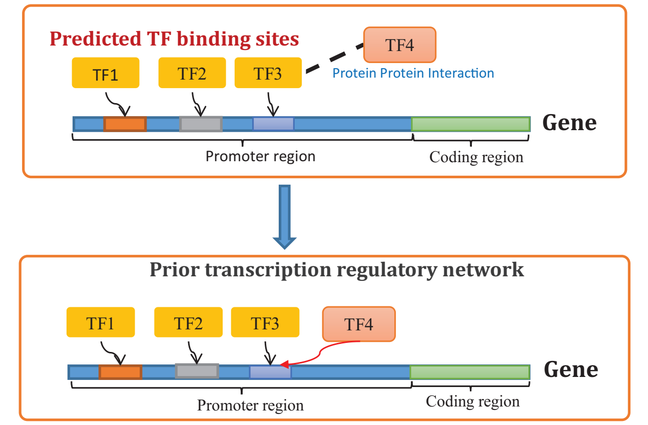

Predicted TF binding sites
TF1
TF2
TF3
TF4
Protein Protein Interaction
Gene
Promoter region
Coding region
Prior transcription regulatory network
TF1
TF2
TF3
TF4
Gene
Promoter region
Coding region

FIGURE 1 | Constructing prior transcription regulatory network using TFBS and PPI data.

INFERRING LIVER-SPECIFIC GENE REGULATORY NETWORK FROM PERTURBATION RESPONSE DATA

Genes that play key roles in liver development, physiology, and disease are found to be tightly regulated by a handful of TFs, such as hepatocyte nuclear factors (HNF1A, HNF1B, HNF3A, HNF3B, HNF3G, HNF4A, HNF4G, and ONECUT1), CCAAT/enhancer-binding proteins (CEBPA and CEBPB), peroxisome proliferator activated receptors (PPARA, PPARD, and PPARG), retinoic acid receptors (RARA, RARB, and RARG), retinoid receptors (RXRA, RXRB, and RXRG), and RAR-related orphan receptors (RORA and RORC) (Schrem et al., 2002, 2004; Odom et al., 2004, 2006; Tomaru et al., 2009). The genes that encode these TFs are known to transcriptionally regulate each other to maintain a particular sequence of events leading to the normal development of liver tissues (Schrem et al., 2002, 2004; Odom et al., 2004, 2006; Tomaru et al., 2009). Therefore, uncovering the GRN involving the above genes is a fundamental step in understanding the physiological processes of liver development. For this purpose, Tomaru et al. (2009) perturbed the above GRN, one gene at a time, using siRNAs and measured the steady-state expression levels of these genes after each perturbation. Here, these measurements were used to infer the topology of the above GRN.

As mentioned above, four different versions of the aforementioned BVS framework were used for network inference, each with a different prior distribution of $\boldsymbol{A^i}$. In the first case, all configurations of $\boldsymbol{A^i}$ were assumed to have equal prior probability, i.e. $P(\boldsymbol{A^i}) = \gamma$, where γ is a constant.

In the second case, the prior distribution of $\boldsymbol{A^i}$ was designed to assign higher probabilities to those configurations of $\boldsymbol{A^i}$ which have fewer ones than zeroes. For this purpose, $\boldsymbol{A^i}$ was assumed to have a beta binomial distribution,

$$P\left(\boldsymbol{A^i}\right) = \binom{n_\mathrm{r}}{n_i} \frac{B\left(n_i + \alpha\,,\, n_\mathrm{r} - n_i + \beta\right)}{B(\alpha, \beta)} \tag{6}$$

Here, n_r is the number of potential regulators in gene g_i. When all genes in the network are considered to be the potential regulators of g_i, $n_\mathrm{r} = n - 1$. The values of the shape parameters (α, β) were kept the same as those used in our previous work (Santra et al., 2013), i.e. $\alpha = 1$, $\beta = 2$.

In the third setting, only TFBS information was used to construct the prior network (**Figure 2A**). TFBS information were collected from HTRIDB (Bovolenta et al., 2012), MEME (Bailey et al., 2009), TRANSFAC (Bryne et al., 2008), JASPER (Matys et al., 2006), TRED (Jiang et al., 2007), and SABioscience (www.sabiosciences.com). Here, only those TFBS that were found within a 5000 bp region of the gene promoters were included in the analysis. This resulted in a total of 106 potential transcriptional regulations (excluding autoregulations, see Table S1 in Supplementary Material for details) among the 21 TFs mentioned above. These regulatory interactions were represented by a prior adjacency matrix ($\boldsymbol{\Gamma_{\mathrm{TFBS}}}$) whose non-zero elements represent potential gene regulations and are assigned a value of $\alpha_\mathrm{c} = 2$. The ith row $\left(\boldsymbol{\Gamma^i_{\mathrm{TFBS}}}\right)$ of this matrix ($\boldsymbol{\Gamma_{\mathrm{TFBS}}}$) represents our prior knowledge on the regulatory program of the ith gene g_i, based solely on TFBS information, and was used to formulate the prior distribution of $\boldsymbol{A^i}$.

In the fourth setting, both TFBS and PPI among TFs (**Figure 2B**; Table S2 in Supplementary Material) were used to determine potential gene regulations. The TFBS information was collected as described above. Information regarding PPIs among the above TFs was obtained from STRING (Szklarczyk et al., 2011) and HPRD (Keshava Prasad et al., 2009) databases (Table S2 in Supplementary Material). The TFBS and PPIs were used to determine potential direct and indirect regulatory interactions as described in the previous section (see **Figure 1**). These resulted in a total of 217 potential gene regulatory interactions (excluding autoregulations; see Table S3 in Supplementary Material for details) which were used to construct the prior network matrix ($\boldsymbol{\Gamma_{\mathrm{TFBS+PPI}}}$). The non-zero elements of this matrix ($\boldsymbol{\Gamma_{\mathrm{TFBS+PPI}}}$) were assigned a value of $\alpha_\mathrm{c} = 2$. The rows of the prior matrix ($\boldsymbol{\Gamma_{\mathrm{TFBS+PPI}}}$) were then used to formulate the prior distributions $P(\boldsymbol{A^i})$, $i = 1, \ldots, n$.

In all the above cases, the search space for the MCMC sampler was restricted to ζ_k, where the subscript k represents the upper limit on the number of regulators for each gene. The value of k was chosen to be the same as the average number of regulators per gene $\left(\frac{217}{21} \approx 10\right)$ in the prior network ($\boldsymbol{\Gamma_{\mathrm{TFBS+PPI}}}$) constructed from TFBS and PPI data.

The GRNs reconstructed using the above prior settings were then compared to a gold standard network (GSN) which was deduced by Tomaru et al. (2009) using matrix RNAi combined with rt-qPCR and Chromatin Immunoprecipitation (X-ChIP) experiments (see Figure S1 in Supplementary Material). To reconstruct the GSN, Tomura et al. knocked down 19 of the above genes, one at a time, and measured the responses of these genes to each knockdown. If a gene responded to the knockdown of another, then the former was considered to be potentially regulated by the later. Based on this assumption, a set of potential gene regulatory interactions ($\boldsymbol{G_{\mathrm{RNAi}}}$) were determined. This was followed by X-ChIP/qPCR analysis that determined the DNA binding preferences of six (TCF1, FOXA1, FOXA2, HNF4A, ONECUT1, and RXRA) of the above TFs. If a TF was found on the promoter of a target gene in the X-ChIP experiment, then the later was considered to be potentially regulated by the former. A second set of potential gene regulations ($\boldsymbol{G_{\mathrm{XChIP}}}$) were identified based on the X-ChIP measurements. The set of interactions ($\boldsymbol{G_{\mathrm{ref}}}$) that were common to both the above networks ($\boldsymbol{G_{\mathrm{RNAi}}}$ and $\boldsymbol{G_{\mathrm{XChIP}}}$) were then considered to represent the GSN ($\boldsymbol{G_{\mathrm{ref}}} = \boldsymbol{G_{\mathrm{RNAi}}} \cap \boldsymbol{G_{\mathrm{XChIP}}}$). The networks inferred by the proposed BVS frameworks with different prior setting were then compared with the above GSN. Since the GSN contains information regarding the regulatory activities of only six (out of 21) TFs, I compared only the interactions involving these TFs. The activities of the remaining 15 TFs were excluded from the comparison.

Recall that the proposed BVS algorithm uses MCMC sampling to estimate the posterior interaction probabilities. These posterior probabilities represent the *a posteriori* confidence on each interaction based on the perturbation response, TFBS and PPI data. If the posterior probability of an interaction is higher than a certain threshold ($p_{\tau\mathrm{h}}$), then the corresponding interaction is considered to be a true interaction. On the other hand, if a posterior probability is less than or equal to this threshold, then the corresponding interaction is thought to be absent in the GRN. This implies that the topology of the reconstructed GRN depends on the threshold probability ($p_{\tau\mathrm{h}}$) and therefore, any comparison between the reconstructed GRN and the true GRN also depends on the choice of this threshold. For a more objective assessment, multiple GRNs are constructed from the above posterior probabilities using multiple different thresholds. Each reconstructed GRN is then compared with the true GRN and the number of correctly and incorrectly inferred interactions are counted. These counts are used to calculate the true positive rates (TPRs), false positive rates (FPRs), and precisions (PREs) of the reconstructed GRNs. The TPR is the ratio of total number of the correctly identified interactions to the total number of interactions present in the GSN (Fawcett, 2004; Powers, 2011); the FPR is the ratio of the total number of incorrectly identified interactions and the total number of possible interactions that are absent in the GSN (Fawcett, 2004; Powers, 2011); PRE is the ratio of the total number of correctly identified interactions to the total number of interactions present in the inferred network. Then, the TPRs (Y-axis) are plotted against the FPRs (X-axis), and the PREs (Y-axis) are plotted against TPRs (X-axis) in two separate plots, commonly known as receiver operating characteristic (ROC) and precision recall (PR) curves, respectively (Fawcett, 2004; Powers, 2011). The areas under these curves, denoted by AUROC and AUPR, give an objective assessment of the accuracy of the GRNs reconstructed by the BVS algorithms (Fawcett, 2004; Powers, 2011). Both AUROC

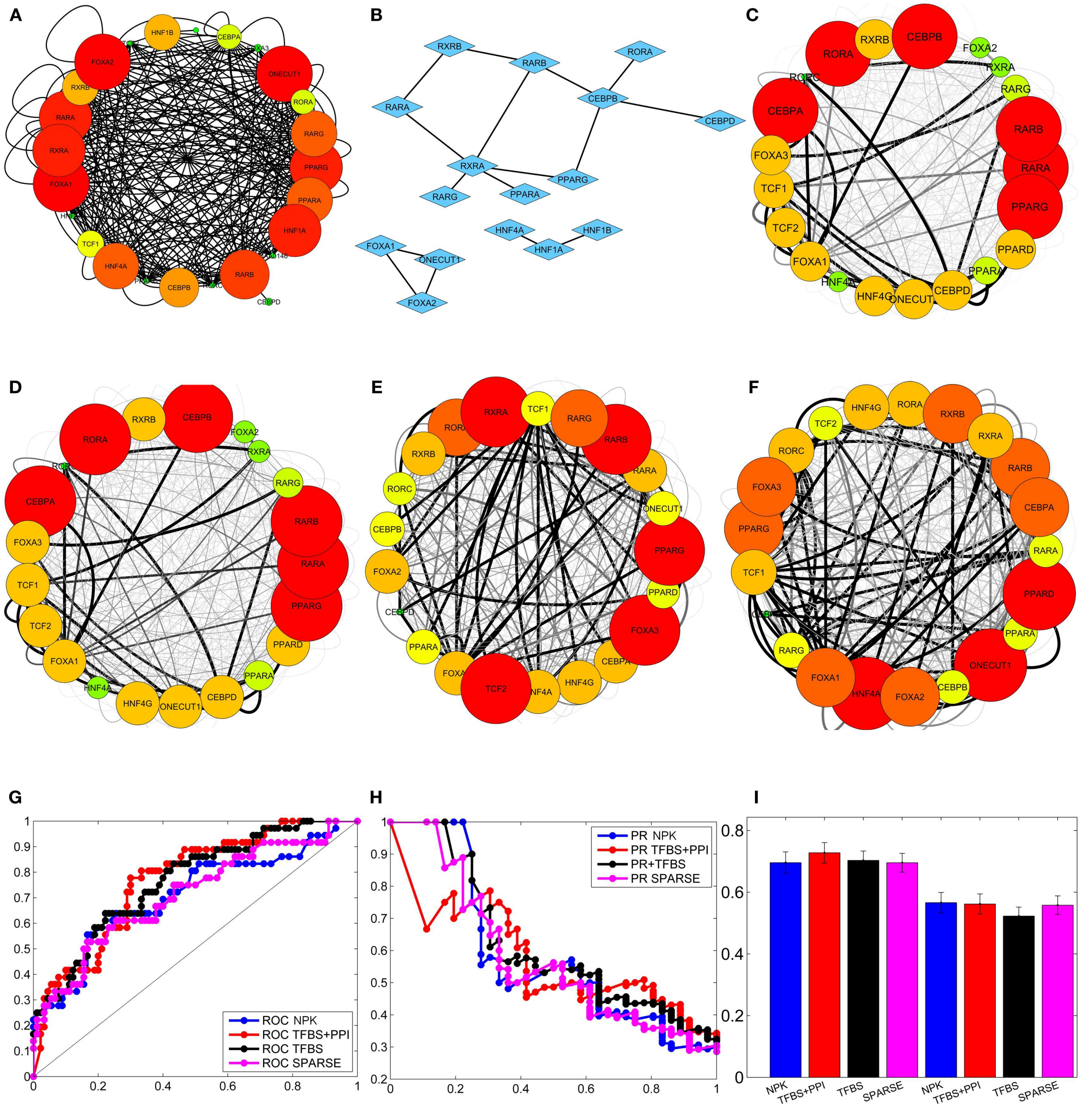


FIGURE 2 | Integrating external data to infer liver-specific transcription regulatory network (Tomaru et al., 2009). (A) Prior network constructed from TFBS information. **(B)** PPI among transcription factors. **(C)** Network inferred by flat prior ($P(\boldsymbol{A}^i) = \gamma$). **(D)** Network inferred using sparse prior. **(E)** Network inferred using prior network constructed from TFBS information only. **(F)** Network inferred using prior network constructed from TFBS and PPI information. The interactions that occur with high and low posterior probabilities are represented by darker/thicker and lighter/thinner edges, respectively, in **(C)**, **(D)**, **(E)**, and **(F)**. **(G)** Average ROC curves of the inferred networks. **(H)** Average PR curves of the inferred networks. **(I)** mean and standard deviation of the area under the ROC and PR curves of the inferred networks.

and AUPR can have values between 0 and 1, and the closer these values are to 1, the better is the accuracy of the inferred networks, with AUROC = 1 and AUPR = 1 being the ideal case. To perform a robust comparison, the proposed BVS algorithm was executed 50 times under each prior setting, producing 50 posterior networks for each prior network (see **Figures 2C–F** for sample posterior networks inferred from different priors). ROC, PR curves, and the areas under these curves (AUROC and AUPR, respectively) were calculated from each posterior network. The average ROC and PR curves of the networks that were inferred from the same network prior was then calculated for each prior setting (**Figures 2G,H**). The mean and standard deviations of the corresponding AUROC and AUPR values, calculated under different prior settings, are shown in **Figure 2I**. The AUROC values calculated under different prior settings were then compared using Mann–Whitney *U* test (Mann and Whitney, 1947) to assess the effects of different network priors on the accuracy of the proposed BVS algorithm. These results suggest that the BVS framework that incorporates both the TFBS and PPI data performed better than those which incorporate no prior information ($p = 0.99 \times 10^{-6}$), only TFBS information ($p = 2.05 \times 10^{-4}$) as prior knowledge, and the sparse prior ($p = 2.4 \times 10^{-6}$). These results support our hypothesis that TFBS and PPI data can be collectively more predictive of potential GRNs than TFBS data alone.

Finally, I assessed the sensitivity of the BVS framework to the confidence parameter (α_c) by looking at the agreement between results obtained under different values of this parameter. For this purpose, five different values ($\alpha_c = 1, 2, 3, 4, 5$) of the confidence parameters were used to formulate a total of 10 prior distributions, five of these use only TFBS information and the remaining five use both TFBS and PPI information. A GRN was reconstructed using each of these prior distributions, leading to 10 inferred networks. These networks were then compared with each other to determine whether different values of the confidence parameter (α_c) had significant effect on the network inference process. The inferred networks were then compared with the networks inferred from no prior knowledge (NPK) and sparse priors, the prior networks ($\mathbf{\Gamma_{TFBS}}$, $\mathbf{\Gamma_{TFBS+PPI}}$), and the reference network (REF). Pearson correlation coefficient was used for comparing these networks. The resulting correlation coefficients are shown in **Figure 3**. Values close to unity indicate high degree of similarities between networks. The networks inferred from the same type of prior distribution are in close agreement with each other, despite different values of the confidence parameter α_c. This suggests that the proposed BVS framework is relatively insensitive to different values of α_c. However, the networks inferred from different types of priors are mostly different from each other. Additionally, the inferred networks are also considerably different from the prior networks suggesting that the proposed Bayesian framework indeed strikes a balance between prior information and observed data.

Encouraged by the above results, I implemented the proposed BVS framework to infer the regulatory mechanisms of the human breast epithelium and compared its performance with a state-of-the-art network inference method, which relies on LASSO regression. The results of this comparison are described in detail in the next section.

INFERRING GRN OF HUMAN BREAST EPITHELIUM AND COMPARISON WITH LASSO

For large-scale GRN inference, I used a set of mRNA expression measurements obtained from human epithelium at different stages of cancer development (Graham et al., 2010). The dataset was produced by Graham et al. (2010) who performed gene expression analysis of breast epithelium tissue samples obtained from 42 patients (18 cancer free, 18 had prophylactic mammoplasty, and 6 had reduction mammoplasty) in order to understand the differences in expression profiles of histologically normal breast epithelium and usual-risk controls undergoing reduction mammoplasty. These expression profiles were used to infer the GRN that governs the regulatory mechanisms of human breast epithelium. The natural genetic variations caused by SNP, copy number variations, mutation, epigenetic regulation, etc., were considered to be genetic perturbations that led to different gene expression profiles among different patients. To save computational time, only top 2000 probe sets (1337 genes) with the highest between-sample variances were selected (Table S4 in Supplementary Material). Among the selected probes, there were 93 known TFs (Table S5 in Supplementary Material) which were used as potential regulators of the selected genes for network inference.

Four different prior settings were used for the BVS framework. The parameter settings for the flat and sparse priors were left the same as before. TFBS information were collected from ENCODE (Hughes et al., 2000; Ernst et al., 2010), MEME (Bailey et al., 2009), TRANSFAC (Bryne et al., 2008), and JASPER (Matys et al., 2006) to construct the prior network ($\mathbf{\Gamma_{TFBS}}$) that contains only direct gene regulations (**Figure 4A**). This network ($\mathbf{\Gamma_{TFBS}}$) contains 4963 number of potential gene regulations between 93 TFs and 1317 target genes (Table S6 in Supplementary Material). Information regarding PPI among TFs (**Figure 4B**) was collected from physical TF binding data published by Ravasi et al. (2010) (Table S7 in Supplementary Material). This information along with the TFBS data were used to construct a second prior network ($\mathbf{\Gamma_{TFBS+PPI}}$) which contains 16,372 potential regulatory interactions supported by both types of data (Table S8 in Supplementary Material). The confidence parameter (α_c) was set to 2 and the restriction parameter (k) were assigned a value of 12 $\left(k = \frac{16{,}372}{1317} \approx 12\right)$. The above prior settings, when used with the proposed BVS framework led to four different posterior networks that were then used for performance evaluation and comparison purposes.

For performance comparison, a LASSO regression-based GRN inference algorithm (Wang et al., 2013) was selected due to recent popularity of LASSO-based methods in the network inference community (van Someren et al., 2003; Li and Yang, 2004; van Someren et al., 2006; Shimamura et al., 2007; Hecker et al., 2009, 2012; Lee et al., 2009; Charbonnier et al., 2010; Gustafsson and Hornquist, 2010; James et al., 2010; Pan et al., 2010; Peng et al., 2010; Wang et al., 2013). LASSO is a regularized version of least-square regression which uses the constraint that $||\boldsymbol{\beta}||^1$, the L^1-norm of the regression coefficients, is no greater than a given value. This is equivalent to an unconstrained minimization of the least-squares penalty with an added penalty $\lambda||\boldsymbol{\beta}||^1$, where λ is a constant. As the penalty is increased, LASSO regression drives more and more of the regression coefficients ($\boldsymbol{\beta}$) to 0, leaving fewer and fewer non-zero coefficients. Both LASSO and BVS share some similarities in their core formulations but differ in some key aspects in their implementations. For instance, both these algorithms rely on linear regression models, but LASSO uses absolute shrinkage regularization to deal with curse of dimensionality where BVS uses MCMC sampling for the same purpose. Therefore, comparing the results obtained from LASSO- and BVS-based techniques may reveal the strengths and weaknesses of algorithms which rely on regularization and MCMC sampling. Similar to the BVS

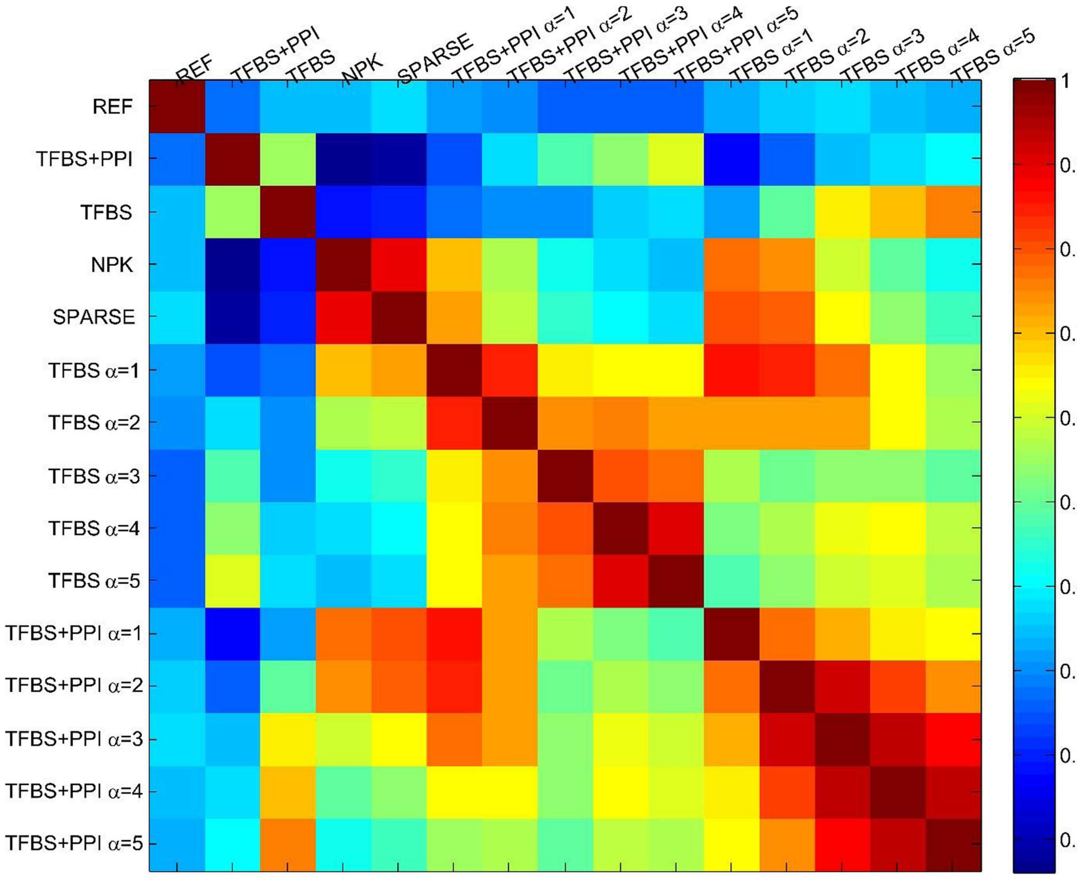


FIGURE 3 | The sensitivity of the BVS framework to the confidence parameter (α_c). Here, REF represents the reference/gold standard network. TFBS represents the prior network that uses only TFBS information. TFBS + PPI represents the prior network that uses both TFBS and PPI information. No prior knowledge (NPK) represents the network that was inferred using flat prior. SPARSE represents the network that was inferred using sparse prior. TFBS $\alpha = x$ represents the posterior network inferred from $\boldsymbol{\Gamma}_{\mathbf{TFBS}}$ with the confidence parameter set to $\alpha_c = x$. TFBS + PPI $\alpha = x$ represents the posterior network inferred from $\boldsymbol{\Gamma}_{\mathbf{TFBS+PPI}}$ with the confidence parameter set to $\alpha_c = x$. The above heatmap represents the similarities (in terms of Pearson correlation coefficients) among the reference, prior, and posterior networks. Values close to 1 (dark red) represent close agreement and values close to zero (dark blue) represent a lack of agreement between network topologies. This figure suggests that the prior networks (TFBS and TFBS + PPI) do not have significant overlap with the reference network (correlation coefficients 0.42, 0.31, respectively). This is due to the fact that only 19 and 16% of the interactions that are present in the prior networks (TFBS and TFBS + PPI) are also present in the reference network (REF). Additionally, posterior networks inferred from the same prior network have a high degree of topological similarity (correlation coefficients 0.6–0.95), regardless of the value of the confidence parameter (α_c).

framework, three different prior settings were used for the LASSO-based algorithm. In the first case, no prior information was used, and in the second and third cases, $\boldsymbol{\Gamma}_{\mathbf{TFBS}}$ and $\boldsymbol{\Gamma}_{\mathbf{TFBS+PPI}}$ were used, respectively, as prior networks. The values of the regularization parameters were kept at their default values ($\lambda_1 = 0.2$, $\lambda_2 = 0.8$). This led to three different networks that were inferred by the LASSO-based algorithm.

To evaluate the accuracy of the inferred networks, I compared these to a GSN which consists of a collection of 1726 known gene regulatory interactions obtained from the HTRIdb, ConsensusPathDB and KEGG databases (**Figure 4C**, see Table S9 in Supplementary Material for details). The GSN contains interactions between only 27 (out of 93) TFs and their target genes. Therefore, only the regulatory activities of these 27 TFs were compared and the activities of the remaining 66 TFs were excluded from the comparison. The comparison was done using ROC and PR curves as mentioned in the previous section. The resulting AUROC and AUPR values are shown in **Figures 4D,E**. These results suggest that the performance of the proposed BVS algorithm increased significantly when prior information was incorporated into the inference method. In particular, TFBS and PPI data collectively were more predictive of regulatory interactions than TFBS information alone. Moreover, BVS algorithm performed better than the LASSO-based method under all circumstances. As in the previous section, the performance of BVS algorithm was found not to be sensitive (**Figure 4F**) to different values of the confidence parameter (α_c).

A possible reason behind the poor performance of LASSO can be low precision of the prior networks. The prior networks used in this study have many more interactions (≈5000, 16,000) than

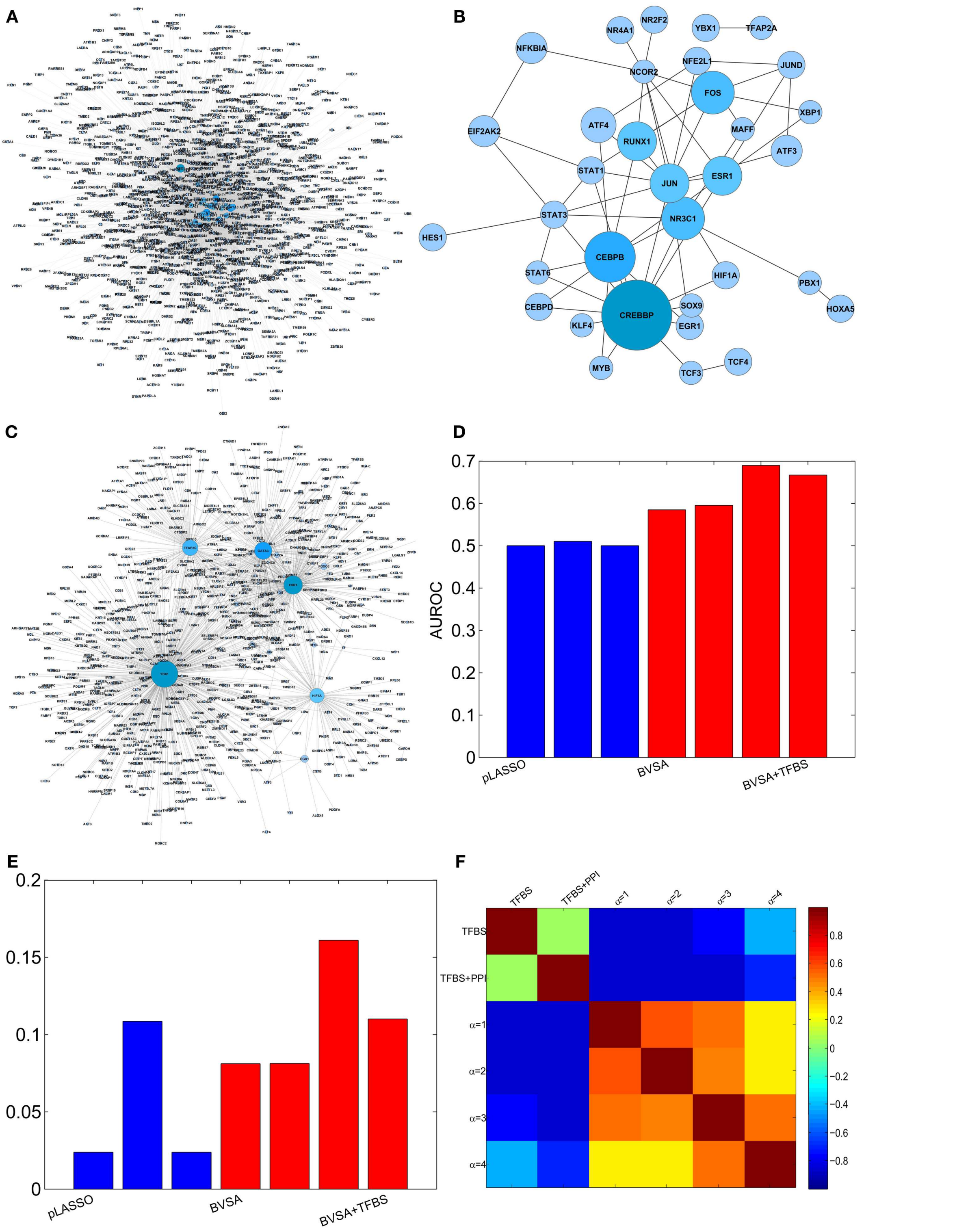


FIGURE 4 | Reconstructing GRN of human breast epithelium and comparison with LASSO. (A) Prior network based on TFBS information. **(B)** PPI among TFs. **(C)** The gold standard network. **(D)** AUROCs of LASSO and BVS algorithms under different prior settings. **(E)** AUPRs of LASSO and BVS algorithms under different prior settings. **(F)** Sensitivity of the BVS algorithm to the confidence parameter (α_c). Here, TFBS represents the prior network constructed from TFBS data, TFBS + PPI represents the prior network constructed from both TFBS and PPI information, $\alpha = 1, 2, 3, 4$ represents the networks inferred from $\boldsymbol{\Gamma}_{\mathbf{TFBS+PPI}}$ with confidence parameters $\alpha_c = 1, 2, 3, 4$, respectively.

the REF (≈1700 interactions) and therefore have very low precision. It was shown before that the performance of LASSO degrades rapidly as the precision of the prior information decreases (Wang et al., 2013). Additionally, the above results depend largely on the quality of the GSN which is a generic network consisting of the interactions involving the selected genes and TFs. This network does not necessarily reflect the tissue-specific behavior of the gene regulatory programs in breast cancer cells and therefore may not be ideal for performance evaluation purposes. However, this network was used as gold standard due to unavailability of information regarding tissue-specific GRNs.

DISCUSSION

In this study, I presented a new approach that incorporates TFBS data along with protein interactions among TFs in a BVS framework to infer GRNs. The main hypothesis behind this approach was that integrating protein interactions among TFs with TFBS data increases the predictive power of the inference process, especially in a variable selection setting. This was demonstrated by inferring a liver-specific transcription regulatory network and the gene regulation program of human breast epithelium, and evaluating the accuracy of the inferred networks based on known interactions. However, there are several shortcomings of the proposed data integration method. For instance, adding all indirect interactions, predicted from TF–TF PPIs, may result in a very large number of potential interactions, leading to a very low precision prior which may not contribute to the predictive power of the inference process. This issue can be mitigated by using information on protein complexes from relevant databases when these databases mature. The precision of the prior network can also be improved by removing unlikely edges that can be determined by other types of data, e.g. eQTL data.

Moreover, the proposed BVS framework relies on a linear regression model of gene regulation. Although linear regression models are extensively used by network inference community due to ease of implementation, it was recently shown that tree-based regression models may be better suitable than linear regression models in network reconstruction problems (Huynh-Thu et al., 2010). Therefore, a possible upgrade of the proposed Bayesian framework will be to replace the linear regression-based gene regulation models by tree-based regression models. Additionally, in this study, I focused mainly on two types of external data sources, consensus motif data, and PPI data. There are a plethora of other functional genomics data, e.g. GO, SNP, gene orthology, etc., which can also be predictive of potential gene regulatory interactions. Our next objective is to find a meaningful way of incorporating such information into the BVS framework.

ACKNOWLEDGMENTS

This work was supported by the Science Foundation Ireland under Grant No. 06/CE/B1129 and European Union Grant PRIMES No. FP7-HEALTH-2011-278568.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this study can be found online at http://www.frontiersin.org/Journal/10.3389/fbioe.2014.00013/abstract

REFERENCES

Baba, T., Huan, H. C., Datsenko, K., Wanner, B. L., and Mori, H. (2008). The applications of systematic in-frame, single-gene knockout mutant collection of *Escherichia coli* K-12. *Methods Mol. Biol.* 416, 183–194. doi:10.1007/978-1-59745-321-9_12

Bader, G. D., Betel, D., and Hogue, C. W. V. (2003). BIND: the biomolecular interaction network database. *Nucleic Acids Res.* 31, 248–250. doi:10.1093/nar/gkg056

Bailey, T. L., Boden, M., Buske, F. A., Frith, M., Grant, C. E., Clementi, L., et al. (2009). MEME suite: tools for motif discovery and searching. *Nucleic Acids Res.* 37, W202–W208. doi:10.1093/nar/gkp335

Bansal, M., Belcastro, V., Ambesi-Impiombato, A., and di Bernardo, D. (2007). How to infer gene networks from expression profiles. *Mol. Syst. Biol.* 3, 78. doi:10.1038/msb4100120

Bovolenta, L., Acencio, M., and Lemke, N. (2012). HTRIdb: an open-access database for experimentally verified human transcriptional regulation interactions. *BMC Genomics* 13:405. doi:10.1186/1471-2164-13-405

Bryne, J. C., Valen, E., Tang, M. H., Marstrand, T., Winther, O., da Piedade, I., et al. (2008). JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update. *Nucleic Acids Res.* 36, D102–D106. doi:10.1093/nar/gkm955

Charbonnier, C., Chiquet, J., and Ambroise, C. (2010). Weighted-LASSO for structured network inference from time course data. *Stat. Appl. Genet. Mol. Biol.* 9, 15. doi:10.2202/1544-6115.1519

de la Fuente, A., and Makhecha, D. P. (2006). Unravelling gene networks from noisy under-determined experimental perturbation data. *Syst. Biol. (Stevenage)* 153, 257–262. doi:10.1049/ip-syb:20050061

Djebbari, A., and Quackenbush, J. (2008). Seeded Bayesian networks: constructing genetic networks from microarray data. *BMC Syst. Biol.* 2:57. doi:10.1186/1752-0509-2-57

Ernst, J., Plasterer, H. L., Simon, I., and Bar-Joseph, Z. (2010). Integrating multiple evidence sources to predict transcription factor binding in the human genome. *Genome Res.* 20, 526–536. doi:10.1101/gr.096305.109

Fawcett, T. (2004). ROC graphs: notes and practical considerations for researchers. *Patt. Recognit. Lett.* 27, 882–891. doi:10.1016/j.patrec.2005.10.012

Fernández, C., Ley, E., and Steel, M. F. J. (2001). Benchmark priors for Bayesian model averaging. *J. Econom.* 100, 381–427. doi:10.1016/j.etap.2012.05.002

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* 33, 1–22.

Gardner, T. S., di Bernardo, D., Lorenz, D., and Collins, J. J. (2003). Inferring genetic networks and identifying compound mode of action via expression profiling. *Science* 301, 102–105. doi:10.1126/science.1081900

George, E., and Foster, D. (2000). Calibration and empirical Bayes variable selection. *Biometrika* 87, 731–747. doi:10.1186/1753-6561-5-S9-S5

Geyer, C. J. (2011). *Handbook of Markov Chain Monte Carlo*, eds S. Brooks, A. Gelman, G. Jones, and X. L., Meng (Taylor & Francis).

Graham, K., de las Morenas, A., Tripathi, A., King, C., Kavanah, M., Mendez, J., et al. (2010). Gene expression in histologically normal epithelium from breast cancer patients and from cancer-free prophylactic mastectomy patients shares a similar profile. *Br. J. Cancer* 102, 1284–1293. doi:10.1038/sj.bjc.6605576

Gupta, M., and Ibrahim, J. (2009). An information matrix prior for Bayesian analysis in generalized linear models with high dimensional data. *Stat. Sin.* 19, 1641–1663.

Gustafsson, M., and Hornquist, M. (2010). Gene expression prediction by soft integration and the elastic net-best performance of the DREAM3 gene expression challenge. *PLoS ONE* 5:e9134. doi:10.1371/journal.pone.0009134

Guthke, R., Moller, U., Hoffmann, M., Thies, F., and Topfer, S. (2005). Dynamic network reconstruction from gene expression data applied to immune response during bacterial infection. *Bioinformatics* 21, 1626–1634. doi:10.1093/bioinformatics/bti226

Guyon, I., and Elisseeff, A. (2003). An introduction to variable and feature selection. *J. Mach. Learn. Res.* 3, 1157–1182.

Hansen, M., and Yu, B. (2001). Model selection and the principle of minimum description length. *J. Am. Stat. Assoc.* 96, 746–774. doi:10.1198/016214501753168398

Hartemink, A. J. (2005). Reverse engineering gene regulatory networks. *Nat. Biotechnol.* 23, 554–555. doi:10.1038/nbt0505-554

Hecker, M., Goertsches, R., Engelmann, R., Thiesen, H., and Guthke, R. (2009). Integrative modeling of transcriptional regulation in response to antirheumatic therapy. *BMC Bioinformatics* 10:262. doi:10.1186/1471-2105-10-262

Hecker, M., Goertsches, R. H., Fatum, C., Koczan, D., Thiesen, H. J., Guthke, R., et al. (2012). Network analysis of transcriptional regulation in response to intramuscular interferon-beta-1a multiple sclerosis treatment. *Pharmacogenomics J.* 12, 134–146. doi:10.1038/tpj.2010.77

Huang, T., Liu, L., Qian, Z., Tu, K., Li, Y., and Xie, L. (2010). Using GeneReg to construct time delay gene regulatory networks. *BMC Res. Notes* 3:142. doi:10.1186/1756-0500-3-142

Hughes, T. R., Marton, M. J., Jones, A. R., Roberts, C. J., Stoughton, R., Armour, C. D., et al. (2000). Functional discovery via a compendium of expression profiles. *Cell* 102, 109–126. doi:10.1016/S0092-8674(00)00015-5

Huynh-Thu, V. A., Irrthum, A., Wehenkel, L., and Geurts, P. (2010). Inferring regulatory networks from expression data using tree-based methods. *PLoS ONE* 5:e12776. doi:10.1371/journal.pone.0012776

Imoto, S., Kim, S., Goto, T., Miyano, S., Aburatani, S., Tashiro, K., et al. (2003). Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network. *J. Bioinform. Comput. Biol.* 1, 231–252. doi:10.1142/S0219720003000071

Ishwaran, H., and Rao, J. (2005). Spike and slab variable selection: frequentist and Bayesian strategies. *Ann. Stat.* 33, 730–773. doi:10.1214/009053604000001147

James, G., Sabatti, C., Zhou, N., and Zhu, J. (2010). Sparse regulatory networks. *Ann. Appl. Stat.* 4, 663–686. doi:10.1214/10-AOAS350

Jiang, C., Xuan, Z., Zhao, F., and Zhang, M. Q. (2007). TRED: a transcriptional regulatory element database, new entries and other development. *Nucleic Acids Res.* 35, D137–D140. doi:10.1093/nar/gkl1041

Kamburov, A., Pentchev, K., Galicka, H., Wierling, C., Lehrach, H., and Herwig, R. (2011). ConsensuspathDB: toward a more complete picture of cell biology. *Nucleic Acids Res.* 39, D712–D717. doi:10.1093/nar/gkq1156

Kariya, T., and Kurata, H. (2004). "Generalized least squares estimators," in *Generalized Least Squares*, eds D. J. Balding, N. A. C. Cressie, G. Fitzmaurice, H. Goldstein, I. M. Johnstone, G. Molenberghs, et al. (Chichester: Wiley), 25–66.

Kerrien, S., Aranda, B., Breuza, L., Bridge, A., Broackes-Carter, F., Chen, C., et al. (2012). The IntAct molecular interaction database in 2012. *Nucleic Acids Res.* 40, D841–D846. doi:10.1093/nar/gkr1088

Keshava Prasad, T. S., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S., et al. (2009). Human protein reference database–2009 update. *Nucleic Acids Res.* 37, D767–D772. doi:10.1093/nar/gkn892

Kholodenko, B. N., Kiyatkin, A., Bruggeman, F. J., Sontag, E., Westerhoff, H. V., and Hoek, J. B. (2002). Untangling the wires: a strategy to trace functional interactions in signaling and gene networks. *Proc Natl Acad Sci U.S.A.* 99, 12841–12846. doi:10.1073/pnas.192442699

Lee, S. I., Dudley, A. M., Drubin, D., Silver, P. A., Krogan, N. J., Pe'er, D., et al. (2009). Learning a prior on regulatory potential from eQTL data. *PLoS Genet.* 5:e1000358. doi:10.1371/journal.pgen.1000358

Li, F., and Yang, Y. (2004). Recovering genetic regulatory networks from micro-array data and location analysis data. *Genome Inform.* 15, 131–140.

Liang, F., Paulo, R., and Molina, G. (2008). Mixtures of g priors for Bayesian variable selection. *J. Am. Stat. Assoc.* 103, 410–423. doi:10.1198/016214507000001337

Lo, K., Raftery, A. E., Dombek, K. M., Zhu, J., Schadt, E. E., Bumgarner, R. E., et al. (2012). Integrating external biological knowledge in the construction of regulatory networks from time-series expression data. *BMC Syst. Biol.* 6:101. doi:10.1186/1752-0509-6-101

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

Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Dalla Favera, R., et al. (2006). ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* 7(Suppl. 1):S7. doi:10.1186/1471-2105-7-S1-S7

Matys, V., Kel-Margoulis, O. V., Fricke, E., Liebich, I., Land, S., Barre-Dirrie, A., et al. (2006). TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Res.* 34, D108–D110. doi:10.1093/nar/gkj143

Mukherjee, S., and Speed, T. (2008). Network inference using informative priors. *Proc. Natl. Acad. Sci. U.S.A.* 105, 14313–14318. doi:10.1073/pnas.0802272105

Odom, D. T., Dowell, R. D., Jacobsen, E. S., Nekludova, L., Rolfe, P. A., Danford, T. W., et al. (2006). Core transcriptional regulatory circuitry in human hepatocytes. *Mol. Syst. Biol.* 2, doi:10.1038/msb4100059

Odom, D. T., Zizlsperger, N., Gordon, D. B., Bell, G. W., Rinaldi, N. J., Murray, H. L., et al. (2004). Control of pancreas and liver gene expression by HNF transcription factors. *Science* 303, 1378–1381. doi:10.1126/science.1089769

Ogata, H., Goto, S., Sato, K., Fujibuchi, W., Bono, H., and Kanehisa, M. (1999). KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.* 27, 29–34. doi:10.1093/nar/27.1.29

Pan, W., Xie, B., and Shen, X. (2010). Incorporating predictor network in penalized regression with application to microarray data. *Biometrics* 66, 474–484. doi:10.1111/j.1541-0420.2009.01296.x

Peng, J., Zhu, J., Bergamaschi, A., Han, W., Noh, D. Y., Pollack, J. R., et al. (2010). Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer. *Ann. Appl. Stat.* 4, 53–77. doi:10.1214/09-AOAS271

Powers, D. (2011). Evaluation: from precesion, recall and F-measure, informedness, markedness and correlation. *J. Mach. Learn. Technol.* 2, 37–63.

Ravasi, T., Suzuki, H., Cannistraci, C. V., Katayama, S., Bajic, V. B., Tan, K., et al. (2010). An atlas of combinatorial transcriptional regulation in mouse and man. *Cell* 140, 744–752. doi:10.1016/j.cell.2010.01.044

Repsilber, D., Liljenstrom, H., and Andersson, S. G. (2002). Reverse engineering of regulatory networks: simulation studies on a genetic algorithm approach for ranking hypotheses. *BioSystems* 66, 31–41. doi:10.1016/S0303-2647(02)00019-9

Rogers, S., and Girolami, M. (2005). A Bayesian regression approach to the inference of regulatory networks from gene expression data. *Bioinformatics* 21, 3131–3137. doi:10.1093/bioinformatics/bti487

Santra, T., Kolch, W., and Kholodenko, B. (2013). Integrating Bayesian variable selection with modular response analysis to infer biochemical network topology. *BMC Syst. Biol.* 7:57. doi:10.1186/1752-0509-7-57

Schrem, H., Klempnauer, J., and Borlak, J. (2002). Liver-enriched transcription factors in liver function and development. Part I: the hepatocyte nuclear factor network and liver-specific gene expression. *Pharmacol. Rev.* 54, 129–158. doi:10.1124/pr.54.1.129

Schrem, H., Klempnauer, J., and Borlak, J. (2004). Liver-enriched transcription factors in liver function and development. Part II: the C/EBPs and D site-binding protein in cell cycle control, carcinogenesis, circadian gene regulation, liver regeneration, apoptosis, and liver-specific gene regulation. *Pharmacol. Rev.* 56, 291–330. doi:10.1124/pr.56.2.5

Shimamura, T., Imoto, S., Yamaguchi, R., and Miyano, S. (2007). Weighted lasso in graphical Gaussian modeling for large gene network estimation based on microarray data. *Genome Inform.* 19, 142–153. doi:10.1142/9781860949852_0013

Szklarczyk, D., Franceschini, A., Kuhn, M., Simonovic, M., Roth, A., Minguez, P., et al. (2011). The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Res.* 39, D561–D568. doi:10.1093/nar/gkq973

Tomaru, Y., Nakanishi, M., Miura, H., Kimura, Y., Ohkawa, H., Ohta, Y., et al. (2009). Identification of an inter-transcription factor regulatory network in human hepatoma cells by matrix RNAi. *Nucleic Acids Res.* 37, 1049–1060. doi:10.1093/nar/gkn1028

van Someren, E. P., Vaes, B. L., Steegenga, W. T., Sijbers, A. M., Dechering, K. J., and Reinders, M. J. (2006). Least absolute regression network analysis of the murine osteoblast differentiation network. *Bioinformatics* 22, 477–484. doi:10.1093/bioinformatics/bti816

van Someren, E. P., Wessels, L., Backer, E., and Reinders, M. (2003). Multicriterion optimization for genetic network modeling. *Signal Process* 83, 763–775. doi:10.1016/S0165-1684(02)00473-5

Vignes, M., Vandel, J., Allouche, D., Ramadan-Alban, N., Cierco-Ayrolles, C., Schiex, T., et al. (2011). Gene regulatory network reconstruction using Bayesian networks, the Dantzig selector, the Lasso and their meta-analysis. *PLoS ONE* 6:e29165. doi:10.1371/journal.pone.0029165

Wagner, A. (2002). Estimating coarse gene network structure from large-scale gene perturbation data. *Genome Res.* 12, 309–315. doi:10.1101/gr.193902

Wang, Z., Xu, W., San Lucas, F. A., and Liu, Y. (2013). Incorporating prior knowledge into gene network study. *Bioinformatics* 29, 2633–2640. doi:10.1093/bioinformatics/btt443

Yeung, K., Bumgarner, R., and Raftery, A. (2005). Bayesian model averaging: development of an improved multi-class, gene selection and classification tool for microarray data. *Bioinformatics* 21, 2394–2402. doi:10.1093/bioinformatics/bti319

Yeung, K. Y., Dombek, K. M., Lo, K., Mittler, J. E., Zhu, J., Schadt, E. E., et al. (2011). Construction of regulatory networks using expression time-series data of a genotyped population. *Proc. Natl. Acad. Sci. U.S.A.* 108, 19436–19441. doi:10.1073/pnas.1116442108

Yeung, M., Tegner, J., and Collins, J. (2002). Reverse engineering gene networks using singular value decomposition and robust regression. *Proc Natl Acad Sci U.S.A* 99, 6163–6168. doi:10.1073/pnas.092576199

Zellner, A. (1986). "On assessing prior distributions and Bayesian regression analysis with g-prior distributions," in *Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti*, eds P. K. Goel and A. Zellner (Amsterdam: Elsevier), p. 233.

Zhang, S., Ching, W., Tsing, N., Leung, H., and Guo, D. (2010). A new multiple regression approach for the construction of genetic regulatory networks. *Artif. Intell. Med.* 48, 153–160. doi:10.1016/j.artmed.2009.11.001

Zhu, J., Lum, P., Lamb, J., GuhaThakurta, D., Edwards, S. W., Thieringer, R., et al. (2004). An integrative genomics approach to the reconstruction of gene networks in segregating populations. *Cytogenet. Genome Res.* 105, 363–374. doi:10.1159/000078209

Zhu, J., Zhang, B., Smith, E. N., Drees, B., Brem, R. B., Kruglyak, L., et al. (2008). Integrating large-scale functional genomic data to dissect the complexity of yeast regulatory networks. *Nat. Genet.* 40, 854–861. doi:10.1038/ng.167

Zou, H., and Trevor, H. (2005). Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Series B* 67, 301–320. doi:10.1111/j.1467-9868.2005.00527.x

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

Received: 14 January 2014; accepted: 28 April 2014; published online: 20 May 2014.
Citation: Santra T (2014) A Bayesian framework that integrates heterogeneous data for inferring gene regulatory networks. Front. Bioeng. Biotechnol. ***2****:13. doi: 10.3389/fbioe.2014.00013*
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Bioengineering and Biotechnology.

frontiers in
GENETICS

ORIGINAL RESEARCH ARTICLE
published: 24 June 2014
doi: 10.3389/fgene.2014.00177

Relevance of different prior knowledge sources for inferring gene interaction networks

Catharina Olsen[1,2]***, Gianluca Bontempi***[1,2]***, Frank Emmert-Streib***[3]***, John Quackenbush***[4,5] ***and Benjamin Haibe-Kains***[6,7]*****

[1] *Machine Learning Group (MLG), Université Libre de Bruxelles (ULB), Brussels, Belgium*
[2] *Interuniversity Institute of Bioinformatics Brussels ULB-VUB, Brussels, Belgium*
[3] *Computational Biology and Machine Learning Laboratory, Center for Cancer Research and Cell Biology, Queen's University Belfast, Belfast, UK*
[4] *Department of Biostatistics and Computational Biology, Dana-Farber Cancer Institute, Harvard School of Public Health, Boston, MA, USA*
[5] *Department of Cancer Biology, Dana-Farber Cancer Institute, Boston, MA, USA*
[6] *Bioinformatics and Computational Genomics, Princess Margaret Cancer Centre, University Health Network, Toronto, ON, Canada*
[7] *Medical Biophysics Department, University of Toronto, Toronto, ON, Canada*

Edited by:
Mathieu Blanchette, McGill University, Canada

Reviewed by:
Bartek Wilczynski, University of Warsaw, Poland
Veronica Vinciotti, Brunel University, UK

****Correspondence:***
Benjamin Haibe-Kains, Princess Margaret Cancer Centre, University Health Network, Toronto Medical Discovery Tower, 11th floor, Room 310, 101 College Street, Toronto, ON M5G 1L7, Canada
e-mail: bhaibeka@uhnresearch.ca

When inferring networks from high-throughput genomic data, one of the main challenges is the subsequent validation of these networks. In the best case scenario, the true network is partially known from previous research results published in structured databases or research articles. Traditionally, inferred networks are validated against these known interactions. Whenever the recovery rate is gauged to be high enough, subsequent high scoring but unknown inferred interactions are deemed good candidates for further experimental validation. Therefore such validation framework strongly depends on the quantity and quality of published interactions and presents serious pitfalls: (1) availability of these known interactions for the studied problem might be sparse; (2) quantitatively comparing different inference algorithms is not trivial; and (3) the use of these known interactions for validation prevents their integration in the inference procedure. The latter is particularly relevant as it has recently been showed that integration of priors during network inference significantly improves the quality of inferred networks. To overcome these problems when validating inferred networks, we recently proposed a data-driven validation framework based on single gene knock-down experiments. Using this framework, we were able to demonstrate the benefits of integrating prior knowledge and expression data. In this paper we used this framework to assess the quality of different sources of prior knowledge on their own and in combination with different genomic data sets in colorectal cancer. We observed that most prior sources lead to significant *F*-scores. Furthermore, their integration with genomic data leads to a significant increase in *F*-scores, especially for priors extracted from full text PubMed articles, known co-expression modules and genetic interactions. Lastly, we observed that the results are consistent for three different data sets: experimental knock-down data and two human tumor data sets.

Keywords: prior knowledge, validation, colon cancer, knockdown, network inference

1. INTRODUCTION

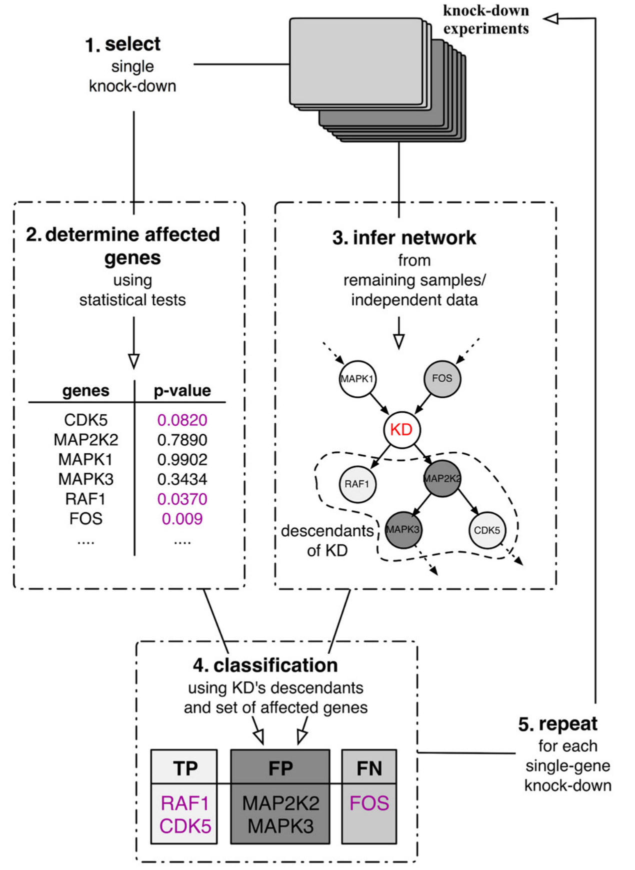
Whilst it is now widely accepted that cellular processes are in general not only governed by single genes but instead also by networks of interacting genes (Barabási and Oltvai, 2004), there is no gold-standard for validating these biological networks (Yngvadottir et al., 2009; Fernald et al., 2011). However, as network inference is increasingly used in biomedical research such as drug discovery or disease classification (Barabási et al., 2011), also the subsequent validation needs to be revisited. The most commonly used approach consists in comparing the inferred network to known interactions stored in biological databases and research articles (Altay et al., 2013). However, this approach has three major drawbacks: Firstly, these interactions are rarely complete, secondly they might not be appropriate for the studied problem and lastly, their quality has not yet been evaluated.

An alternative use for this prior knowledge is its integration into the network inference algorithms in order to improve the quality of inferred networks. Indeed, we and others showed that the combination of data and prior knowledge significantly improves the quality of networks compared to networks inferred from data only (Djebbari and Quackenbush, 2008; Mukherjee and Speed, 2008; Olsen et al., 2014). However, if prior knowledge is used to improve the inference process its subsequent use in the quality assessment would dramatically increase the risk of overfitting.

Recently, we proposed a purely data-driven approach relying on experimental perturbation data to identify the set of relevant genes for a given problem (Olsen et al., 2014). This validation framework not only provides the possibility to compare different inference algorithms but furthermore allows us to independently assess different sources of prior knowledge by themselves and in combination with expression data.

In this follow-up paper to Olsen et al. (2014), we use the proposed validation framework to evaluate the quality of a variety of prior sources, both in combination with different publicly available tumor data sets and by themselves. We retrieved the prior knowledge using the two web applications *Predictive Networks* (Haibe-Kains et al., 2012b) and *GeneMANIA* (Mostafavi et al., 2008), for a total of eight different sources. After the assessment of the different prior sources' quality, we infer networks using three different microarray data sets: experimental knock-down data from cell line experiments and two publicly available human tumor data sets. We quantitatively assess their quality through the estimation of *F*-scores, a well established quality metrics in network inference.

We observe that most prior sources lead to significant *F*-scores. Their integration with genomic data leads to a significant increase in *F*-scores, especially for priors extracted from full text PubMed articles, known co-expression modules and genetic interactions. We also observe that the results are consistent for three different data sets: experimental knock-down data and two human tumor data sets. Furthermore, we observe that combining different sources can be beneficial compared to using a single prior source.


knock-down experiments
1. select single knock-down
2. determine affected genes using statistical tests
3. infer network from remaining samples/ independent data
MAPK1
FOS
KD
RAF1
MAP2K2
MAPK3
CDK5
descendants of KD
4. classification using KD's descendants and set of affected genes
TP
RAF1
CDK5
FP
MAP2K2
MAPK3
FN
FOS
5. repeat for each single-gene knock-down

genes	p-value
CDK5	0.0820
MAP2K2	0.7890
MAPK1	0.9902
MAPK3	0.3434
RAF1	0.0370
FOS	0.009
....	....

FIGURE 1 | Quantitative validation framework for network inference. The framework relies on a set of single-gene knock-down experiments in a leave-one-out cross-validation scheme.

2. MATERIALS AND METHODS

2.1. METHOD—VALIDATION OF INFERRED NETWORKS

The best case scenario in most real-world application is partial knowledge of the true, data-generating network. Therefore, the assessment of any inferred network cannot depend on this knowledge alone. As an alternative, we proposed a purely data-driven validation framework proposed in Olsen et al. (2014). This validation framework depends on the availability of experimental intervention data such as knock-down experiments. This type of data allows us, for each knock-down experiment separately, to statistically evaluate whether or not a gene in the data set was significantly affected by the experiment. In this case, this relation should be reflected in any inferred network in the sense that the affected gene can be found downstream of the knocked down gene. This in turn then allows us to quantitatively assess the quality of inferred gene interaction networks by computing quality measures such as precision, recall or *F*-score (Sokolova et al., 2006). The outline of the framework is depicted in **Figure 1**. Suppose that a number of single gene knock-down experiments were carried out. Then one can use these experiments in a five step procedure:

1. Select a single knock-down and all corresponding replicates from the collection.
2. Use these samples to determine the set of genes that were significantly affected by the perturbation experiments by means of statistical tests.
3. Use the remaining independent samples to infer a directed network.
4. Classify the knock-down's descendants (in the inferred network) into true positives, false positives and false negatives with respect to the affected genes identified in step 2. The descendants of a node in the network are defined to be the set of its children and grandchildren.
5. Repeat steps 1–4 until all perturbations have been used to assess the network's local predictive power.

In Olsen et al. (2014), a network was inferred from the samples not related to the single knock-down experiment (step 3). However, in the same article it was shown that these knock-down samples from cell line experiments can be used for validation not only in such a cross-validation scheme but also for networks inferred from independent tumor samples, which demonstrates the generalizability of our validation approach.

The classification of the nodes in the network (step 4) follows the rationale that statistically significantly affected genes should be found in a directed network downstream of the perturbed gene, its descendants (**Figure 1**). Therefore all genes in the set of descendants which are significantly affected by the perturbation can be classified as true positives (TP) and all significantly affected genes that are inferred outside of the set of descendants as false negatives (FN). Genes that are part of the descendants in the inferred network but are not significantly affected by the perturbation are then false positives (FP).

This classification then allows us to compute the F-score, the harmonic average of precision and recall

$$F = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \in [0, 1], \tag{1}$$

where $F = 0$ corresponds to no correctly identified affected genes and $F = 1$ corresponds to perfect classification.

To control for the density of the network and thus guaranteeing that the F-scores are meaningful, we generated 1000 random networks. Each random network is obtained from the inferred network by shuffling the genes in this network.

2.2. MATERIAL—DATA

Throughout this study, we use the perturbation data described in Olsen et al. (2014), which are publicly available in the NCBI Gene Expression Omnibus (GEO) repository (Barrett et al., 2005), under accession number GSE53091. The samples of this data set consist of eight single gene knock-downs, namely CDK5, HRAS, MAP2K1, MAP2K2, MAPK1, MAPK3, NGFR, and RAF1. These genes belong to the RAS signaling pathway which has been showed to play a key role in colorectal cancer (Zenonos and Kyprianou, 2013). The knockdown experiments were performed in two colon cancer cell lines, SW480 and SW620 (NCBI Gene Expression Omnibus (GEO) repository (Barrett et al., 2005) accession number GSE53091). For each knock-down, six biological replicates were obtained together with controls in both cell lines, in total 125 samples. The data set furthermore consists of the 339 variables over expressing RAS as identified in Bild et al. (2005) and used in Olsen et al. (2014).

For each of the knocked down genes we identify the significantly affected genes by comparing the expression of genes in control versus those of the knock-down experiments with a Wilcoxon Rank Sum test, using a false discovery rate (FDR, Benjamini and Hochberg, 1995) <10% as a threshold for statistical significance. In **Table 1** we present the number of affected genes for each of the knock-down experiments.

We will use two publicly available tumor cancer data sets (expO, 2009; Jorissen et al., 2010) to infer the networks. The first data set (*expO*) contains 292 human tumor samples and is accessible from GEO under accession number GSE2109. The second (*jorissen*) data contains 290 samples and is accessible from GEO under accession number GSE14333.

Table 1 | Number of genes significantly affected by KD (out of 339 genes) based on gene expression data with FDR <10%.

KD	CDK5	HRAS	MAP2K1	MAP2K2
Number of affected genes	73	122	33	38
	MAPK1	**MAPK3**	**NGFR**	**RAF1**
	117	59	99	61

2.3. MATERIAL—SOURCES OF PRIOR KNOWLEDGE

Possible sources of prior knowledge are manifold and include published articles, interactions stored in biological databases or similarity of gene expression values, also referred to as gene co-expression, from published data sets. To efficiently access this information a number of different tools have been implemented including *GeneMANIA* (Mostafavi et al., 2008) and Predictive Networks (Haibe-Kains et al., 2012a). The former allows to upload a set of genes and returns a network of the known interactions distinguishable by source (**Table 2**) whereas the latter uses text mining to retrieve known interactions from PubMed abstracts and furthermore queries structured biological databases. Both tools allow to download the interactions as flat text files, which enables further use of these priors into advanced genomic analyses such as gene interaction network inference.

Here we will use the complete prior set retrieved by *Predictive Networks* (PN) and priors separated by source from *GeneMANIA*. The different number of known interactions identified by each tool and source are presented in **Table 2**. These can be roughly grouped into three categories: (1) Co-expression and genetic with >1000 interactions; (2) PN and co-local, pathway and shared with 100 to ~400 interactions; and (3) physical and predicted with <50 interactions.

3. RESULTS

In this section we use the proposed validation framework (**Figure 1**) to independently assess the quality of the different priors retrieved with *Predictive Networks* and *GeneMANIA* (**Table 2**) in isolation and in combination with three different genomic data sets.

We use the inference procedure introduced in Haibe-Kains et al. (2012a,b) which is a two-step procedure implemented in the R/Bioconductor package *predictionet*. The first step is a feature selection step based on the minimum redundancy, maximum relevance (mRMR, Ding and Peng, 2005; Meyer et al., 2007) criterion whose robustness is improved by the integration of prior knowledge. The subsequent step is an arc orientation procedure using a criterion based on interaction information (McGill, 1954) in which prior integration is used to help orient the edges which could not be oriented from the genomic data. Given the central role of priors in predictionet, we implemented a hyperparameter, referred to as prior weight (w), enabling users to tune their confidence in the prior knowledge incorporated into the network inference procedure. Prior weight w can take value from 0 to 1; low w stands for low confidence in prior data. Note that $w = 0$ forces *predictionet* to ignore priors (only genomic data are taken into account), while predictionet with $w = 1$ will infer networks solely based on prior information, therefore ignoring genomic data.

We use each of the three different data sets (kd, *expO* and *jorissen*) to build networks integrating the different prior knowledge sources with different prior weights $w \in$

Table 2 | Specifications of prior knowledge retrieval tools: *GeneMANIA* (GM) and *Predictive Networks* (PN).

Tool	Source		# interactions
PN	PubMed and databases	(PN)	419
	Co-expr	(GM2)	2760
	Co-local	(GM3)	292
	Genetic	(GM4)	1546
GM	Pathway	(GM5)	100
	Physical	(GM6)	38
	Predicted	(GM7)	29
	Shared	(GM8)	199

{0, 0.25, 0.5, 0.75, 0.95, 1}. The validation is then carried out for each of the eight knocked down genes. We thus obtain eight *F*-scores, one for the descendants of each KD. These *F*-scores are then further assessed by comparing them to *F*-scores of 1000 random networks.

3.1. PRIOR INFORMATION ONLY

The first step in the assessment of the different prior sources' quality is the evaluation of the networks inferred using only these sources (prior weight $w = 1$). In **Figure 2**, we present the results in terms of *F*-scores and significance compared to random networks. When assessing this figure with respect to the number of significant results obtained by each prior source, we can observe that PN performs best with seven out of eight significant results. The next best prior sources are GM6 and GM5 with six significant KDs. With the exception of GM3, all prior sources have at least two significant results. Furthermore, the *F*-scores obtained using prior source PN are amongst the highest values for all KDs except NGFR. On the contrary, GM6 obtains six significant KDs but the *F*-scores are all below those obtained by PN.

Assessing the prior sources' performance with respect to the eight knock-downs, it can be observed that some KDs are in general better predicted than others. Whilst most prior sources are able to obtain significant results for HRAS, MAP2K1, MAPK1, and RAF1, significant results for half the prior sources for CDK5 and MAPK3 they struggle to provide meaningful information for inference of gene interactions in the context of colorectal cancer with the remaining two knock-downs (MAP2K2 and NGFR).

3.2. COMBINATION OF DATA AND PRIOR INFORMATION

In this section we assess the networks inferred from genomic data (KD data in cross-validation; **Figure 1**) and prior knowledge with equal weight ($w = 0.5$). In a first analysis, we compare these *F*-scores to those obtained when inferring networks from data only ($w = 0$) and from prior knowledge only ($w = 1$). A statistical test (Wilcoxon rank test) shows that the combination of data and prior significantly improves the networks (p-values <0.05) compared to data only (Supplementary Table 1) and prior only (Supplementary Table 1, with exception of GM2).

In **Figure 3**, we present these *F*-scores for each knock-down and for each of the eight prior sources. For each knock-down, the results are ordered by *F*-score values, starting with the best result and color-coded by prior source. The best prior source for four out of the eight knock-downs in PN: MAP2K2, MAPK1, MAPK3, and RAF1. The second highest number of best knock-downs is reached by GM2: CDK5, HRAS, and MAP2K1. The best prior source for NGFR is GM4. On the contrary, the performance of GM3, GM6, and GM7 prior sources is amongst the lowest.

3.3. MOST CONSISTENT PRIOR SOURCE ACROSS THREE DIFFERENT DATA SETS

In this section, we will show that the results presented in the previous section for the KD data also hold true when the networks are inferred in combination with the two human tumor data sets. In **Table 3**, we present the prior source that yielded the highest *F*-score for each of the eight knock-downs (prior weight $w = 0.5$). This table summarized the results in Supplementary Figures 9 and 10.

The main observation is that the best prior source is consistent for all three data sets for four of the eight knock-downs: MAP2K1, MAPK1, MAPK3, and NGFR. For the remaining four

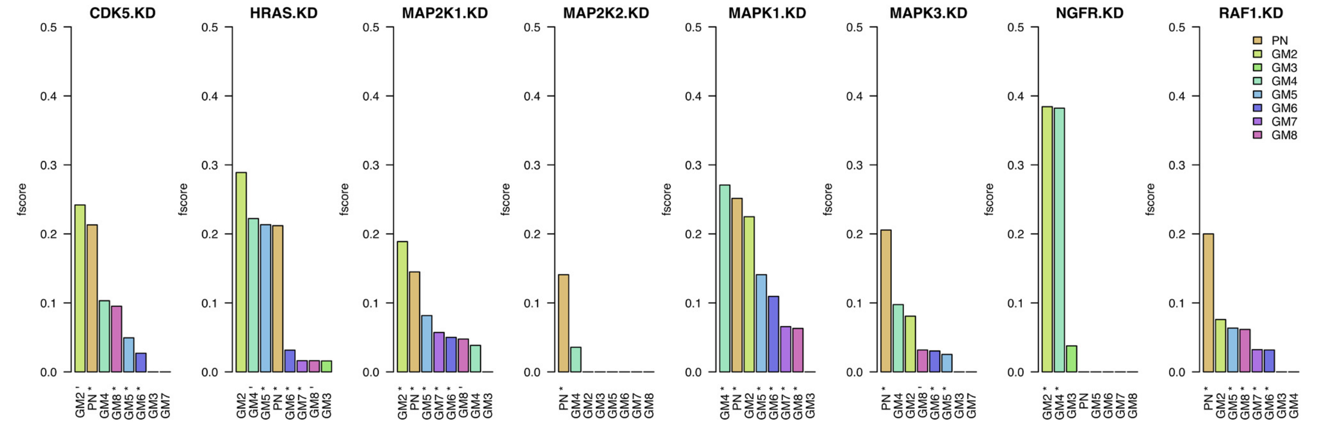


FIGURE 2 | Results when inferring networks with *predictionet* using only prior knowledge ($w = 1$). The height of each bar corresponds to the obtained *F*-score, colored by prior source. The *x*-axis specifies the prior source and includes * if the *F*-score is significant with *p*-value <0.05 and ⁻ for *p*-values < 0.1.

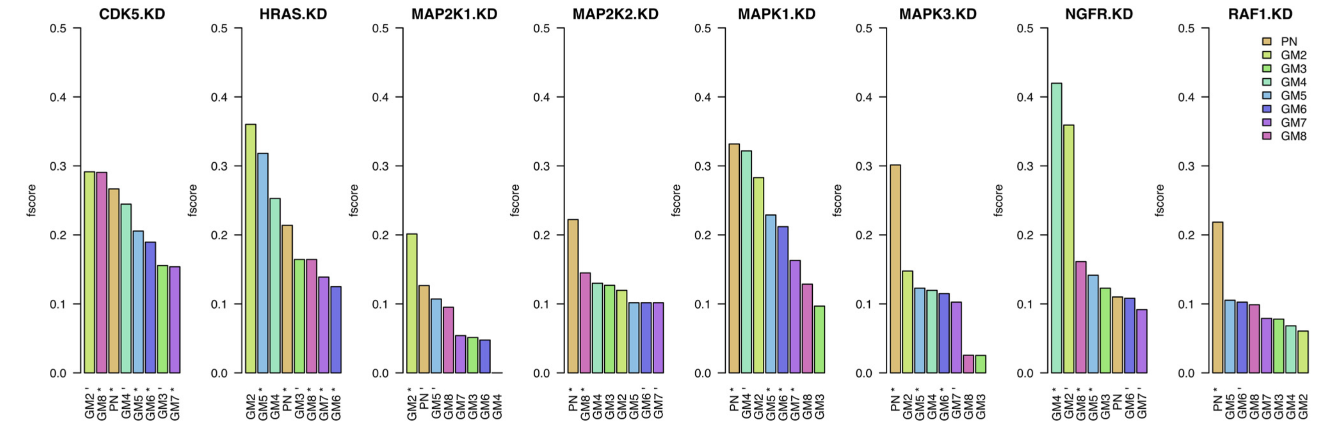


FIGURE 3 | Results when inferring networks with *predictionet* using data and prior knowledge (*w* = 0.5). The height of each bar corresponds to the obtained *F*-score, colored by prior source. The *x*-axis specifies the prior source and includes * if the *F*-score is significant with *p*-value <0.05 and ⁻ for *p*-values < 0.1.

Table 3 | Best single prior source across three large colorectal cancer data sets (kd for knock-down experiments in colorectal cancer cell lines, *expO* and *jorissen* for large human colon tumor data) when combined with microarray gene expression data (prior weight *w* = 0.5).

KD	KD data	expO	Jorissen
CDK5	GM2	PN	PN
HRAS	GM2	GM4	GM2
MAP2K1	GM2	GM2	GM2
MAP2K2	PN	PN	GM7
MAPK1	PN	PN	PN
MAPK3	PN	PN	PN
NGFR	GM4	GM4	GM4
RAF1	PN	GM8	PN

knock-downs, the best prior source is consistent for two out of the three data sets: PN for CDK5, MAP2K2, and RAF1 and GM2 for HRAS.

3.4. COMBINING DIFFERENT PRIOR SOURCES

In this section we investigate whether the combination of prior sources (from a single prior source upto all eight sources) is beneficial to the quality of the inferred networks. For each knock-down, we infer a network using the best prior source, then we add the second best, etc. (**Figure 3**). We test this procedure on the two independent human tumor data sets *expO* and *jorissen*, the corresponding results are presented in **Figure 4** and Supplementary Figure 11, respectively.

When combining *expO* data and with an increasing number of prior sources, the results are better than those obtained using only one source for six out of the eight KDs. For the other two, namely MAP2K1 and NGFR, we have already observed in section 3.1 that most prior sources are not informative. The number of prior sources that need to be combined to obtain the highest significant *F*-scores depends on the knock-down and range between three and eight. It is therefore not only important to determine whether prior sources are relevant by themselves but also which combination of sources will lead to the best results. Similar observations can be made for the *jorissen* data set (Supplementary Figure 11).

4. DISCUSSION

Using the quantitative validation framework we recently introduced in Olsen et al. (2014), we assessed the relevance of different sources of prior information for the inference of large gene interaction networks from high-throughput gene expression data sets. Our results suggest that most prior sources, which include known interactions extracted from research articles, genetic and physical interactions, co-expression and pathway databases yield significant networks in colorectal cancer when used in isolation. Furthermore, concurring with our previous results, we demonstrated that the vast majority of prior sources significantly improves the inference of gene interaction networks when combined with microarray gene expression data.

In our case study we showed that priors extracted from the *Predictive Networks* web application and the co-expressions reported in *GeneMANIA* are the most relevant prior sources in colorectal cancer as they yield the best networks in our validation study. We also showed that these results are consistent across three data sets, composed of a set of knock-down experiments in colorectal cancer cell lines and large collections of human colon tumor samples.

As expected, the quality of inferred gene interaction networks is not uniform over the network topology. For the eight genes we knocked down to investigate their effects in colorectal cancer cell lines, we were able to infer statistically significant subnetworks for most, but not all of them. For instance, we observed that the effects of NGFR, and MAP2K2 knock-downs are particularly difficult to model. Interestingly, genetic interactions and co-expression prior data enabled to build high quality networks for NGFR, which suggests that priors extracted from diverse sources are highly complementary.

Our study supports the use of prior information into network inference and we are now working on improving methods

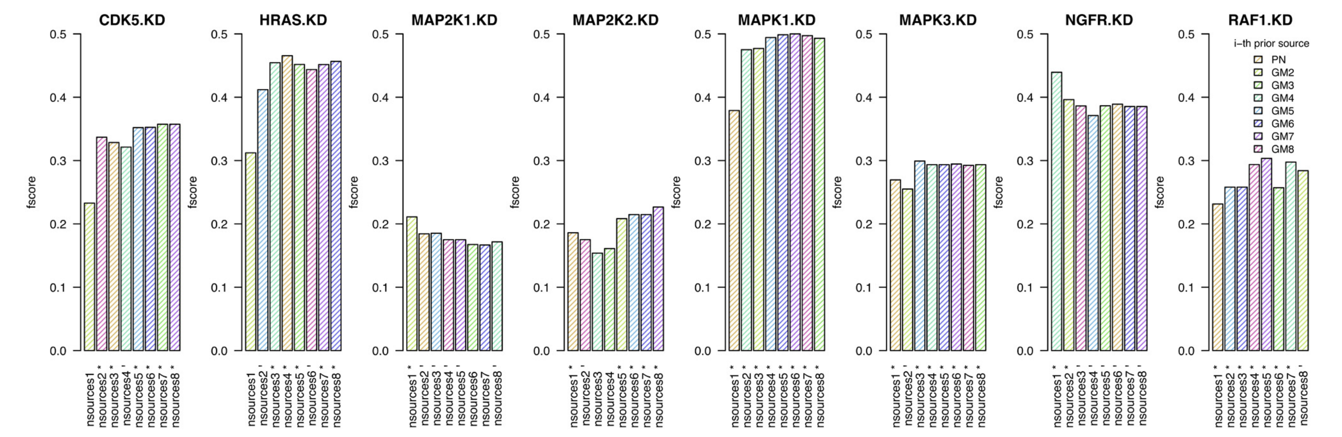


FIGURE 4 | Results when inferring networks with *predictionet* using *expO* data and prior knowledge ($w = 0.5$). The height of each bar corresponds to the obtained *F*-score, colored by which prior source was added. The *x*-axis specifies the prior source and includes * if the *F*-score is significant with *p*-value $<$ 0.05 and ' for *p*-values $<$ 0.1.

to extract high-quality, context-specific prior information, as well as developing novel approaches to integrate these priors to generate better large-scale gene interaction networks. A second aspect that requires further development is the implementation of tools to better combine different prior sources with the hope to significantly improve the local quality of large biological networks.

ACKNOWLEDGMENT

FUNDING

Gianluca Bontempi and Catharina Olsen were supported by the Belgian French Community ARC (Action de Recherche Concertée) funding.

DATA

The authors thank the expO and Jorissen teams for making their data publicly available.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://www.frontiersin.org/journal/10.3389/fgene.2014.00177/abstract

REFERENCES

Altay, G., Altay, N., and Neal, D. (2013). Global assessment of network inference algorithms based on available literature of gene/protein interactions. *Turk. J. Biol.* 37, 547–555. doi: 10.3906/biy-1210-8

Barabási, A.-L., and Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nat. Rev. Genet.* 5, 101–113. doi: 10.1038/nrg1272

Barabási, A.-L. L., Gulbahce, N., and Loscalzo, J. (2011). Network medicine: a network-based approach to human disease. *Nat. Rev. Genet.* 12, 56–68. doi: 10.1038/nrg2918

Barrett, T., Suzek, T. O., Troup, D. B., Wilhite, S. E., Ngau, W.-C., Ledoux, P., et al. (2005). NCBI GEO: mining millions of expression profiles–database and tools. *Nucleic Acids Res.* 33, D562–D566. doi: 10.1093/nar/gki022

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* 57, 289–300.

Bild, A. H., Yao, G., Chang, J. T., Wang, Q., Potti, A., Chasse, D., et al. (2005). Oncogenic pathway signatures in human cancers as a guide to targeted therapies. *Nature* 439, 353–357. doi: 10.1038/nature04296

Ding, C., and Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *J. Bioinform. Comput. Biol.* 3, 185–205. doi: 10.1142/S0219720005001004

Djebbari, A., and Quackenbush, J. (2008). Seeded bayesian networks: constructing genetic networks from microarray data. *BMC Syst. Biol.* 2:57. doi: 10.1186/1752-0509-2-57

expO. (2009). *expO Expression Data*. Available online at: expo.intgen.org/geo

Fernald, G. H., Capriotti, E., Daneshjou, R., Karczewski, K. J., and Altman, R. B. (2011). Bioinformatics challenges for personalized medicine. *Bioinformatics* 27, 1741–1748. doi: 10.1093/bioinformatics/btr408

Haibe-Kains, B., Olsen, C., Bontempi, G., and Quackenbush, J. (2012a). *Predictionet: Inference for Predictive Networks Designed for (But Not Limited to) Genomic Data*. R package version 1.1.5.

Haibe-Kains, B., Olsen, C., Djebbari, A., Bontempi, G., Correll, M., Bouton, C., et al. (2012b). Predictive networks: a flexible, open source, web application for integration and analysis of human gene networks. *Nucleic Acids Res.* 40, D866–D875. doi: 10.1093/nar/gkr1050

Jorissen, R. N., Gibbs, P., Christie, M., Prakash, S., Lipton, L., Desai, J., et al. (2010). Metastasis-associated gene expression changes predict poor outcomes in patients with dukes stage b and c colorectal cancer. *Clin. Cancer Res.* 15, 7642–7651. doi: 10.1158/1078-0432.CCR-09-1431

McGill, W. (1954). Multivariate information transmission. *Psychometrika* 19, 97–116. doi: 10.1007/BF02289159

Meyer, P., Kontos, K., Lafitte, F., and Bontempi, G. (2007). Information-theoretic inference of large transcriptional regulatory networks. *EURASIP J. Bioinform. Syst. Biol.* 2007:79879. doi: 10.1155/2007/79879

Mostafavi, S., Ray, D., Farley, D. W., Grouios, C., and Morris, Q. (2008). GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome Biol.* 9:S4+. doi: 10.1186/gb-2008-9-s1-s4

Mukherjee, S., and Speed, T. P. (2008). Network inference using informative priors. *Proc. Natl. Acad. Sci. U.S.A.* 105, 14313–14318. doi: 10.1073/pnas.0802272105

Olsen, C., Fleming, K., Prendergast, N., Rubio, R., Emmert-Streib, F., Bontempi, G., et al. (2014). Inference and validation of predictive gene networks from biomedical literature and gene expression data. *Genomics*. doi: 10.1016/j.ygeno.2014.03.004. [Epub ahead of print].

Sokolova, M., Japkowicz, N., and Szpakowicz., S. (2006). "Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation," in *AI 2006: Advances in Artificial Intelligence, Lecture Notes in Computer Science*, Vol. 4304, eds A. Sattar and

B.-H. Kang (Berlin; Heidelberg: Springer), 1015–1021. doi: 10.1007/11941439_114

Yngvadottir, B., MacArthur, D., Jin, H., and Tyler-Smith, C. (2009). The promise and reality of personal genomics. *Genome Biol.* 10:237. doi: 10.1186/gb-2009-10-9-237

Zenonos, K., and Kyprianou, K. (2013). RAS signaling pathways, mutations and their role in colorectal cancer. *World J. Gastrointest. Oncol.* 5, 97–101. doi: 10.4251/wjgo.v5.i5.97

Conflict of Interest Statement: 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.

Received: 10 January 2014; accepted: 26 May 2014; published online: 24 June 2014.
Citation: Olsen C, Bontempi G, Emmert-Streib F, Quackenbush J and Haibe-Kains B (2014) Relevance of different prior knowledge sources for inferring gene interaction networks. Front. Genet. ***5****:177. doi: 10.3389/fgene.2014.00177*
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics.