frontiers in Research Metrics and Analytics

PERSPECTIVE
published: 13 February 2017
doi: 10.3389/frma.2017.00001

Accessibility and Update Status of Published Software: Benefits and Missed Opportunities

Xunhe Huang[1], Jieru Xie[1], Newton O. Otecko[2,3] and Minsheng Peng[2,3]*

[1] School of Life Sciences, Jiaying University, Meizhou, China, [2] State Key Laboratory of Genetic Resources and Evolution, Yunnan Laboratory of Molecular Biology of Domestic Animals, and Germplasm Bank of Wild Species, Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming, China, [3] Kunming College of Life Science, University of Chinese Academy of Sciences, Kunming, China

Online accessibility and continuous updating of software packages is important for their application and resource-sharing. We evaluated the accessibility and update status of 214 software packages published in *Molecular Ecology Resources* in the past 15 years (2001–2015) and reported the associated benefits and missed opportunities. Original publications describing these software packages have been cited 31,886 times in total, with 19.6% of them being cited more than 100 times. However, 41.6% (89/214) of the source websites provided in the original papers were inaccessible, of which only 29.2% (26/89) had alternative websites. Up to 63.1% (135/214) of the software packages had not been updated since their publication or were inaccessible. A huge disparity in accessibility, 100% (41/41) vs. 63.6% (110/173), was observed between websites deposited in long-term servers vs. academic institution websites, personal homepages, or e-mails, respectively. In addition, software packages that were accessible and updated were significantly more likely to be cited, compared to those that were inaccessible and un-updated. By guaranteeing the accessibility of their published software packages, for instance, by depositing them in long-term servers and keeping the software packages updated, software developers will further resource-sharing and promote reproduction and citation of their software.

Keywords: updates, software, accessibility, citation, benefits, reproduction

OPEN ACCESS

Edited by:
Xianwen Wang,
Dalian University of Technology, China

Reviewed by:
Tim Clark,
Massachusetts General Hospital, USA
Jiang Li,
Zhejiang University, China

****Correspondence:***
Xunhe Huang
hxh826@jyu.edu.cn

Received: *10 October 2016*
Accepted: *24 January 2017*
Published: *13 February 2017*

Citation:
Huang X, Xie J, Otecko NO and Peng M (2017) Accessibility and Update Status of Published Software: Benefits and Missed Opportunities. Front. Res. Metr. Anal. 2:1. doi: 10.3389/frma.2017.00001

There is a large number of bioinformatics software packages released every year, facilitating various scientific studies. However, problems with accessing the websites in which software packages are deposited, as well as lack of appropriate updates to enhance software compatibility with changes in technology landscapes and user experiences, may significantly undermine developments in the associated research fields. In the present study, we evaluated how citation frequency, an index of utility, is related to the accessibility and update status of software packages published in *Molecular Ecology Resource* (MER). MER, formerly *Molecular Ecology Notes*, is an important platform for publication of bioinformatics tools for ecological, evolutionary, and conservation biology (Narum, 2016).

In the past 15 years (2001–2015), a total of 214 software packages inclusive of programs, R packages, codes, and scripts, were published in MER as COMPUTER PROGRAMS or PROGRAM

NOTES (Table S1 in Supplementary Material; Huang et al., 2016). Of the 214 software packages, 41 (19.2%) were deposited into long-term repositories such as GitHub, Google, Sourceforge, or CRAN, 169 (79%) into personal homepages or academic institute (Colleges or Universities) websites, and 4 (1.9%) linked to e-mail or other formats without websites. The first deposition into the long-term repositories was reported in the year 2007 (**Figure 1**), which coincides with the launching of GitHub in 2007, and CRAN, which although was launched earlier, became well known around 2008. The same period marks a turning point after which there is a general decline in deposition into personal homepages or academic institute websites, but a rise in depositions in long-term repositories (**Figure 1**). The papers related to the 214 software packages were cited 31,886 times in total, ranging from 0 to 4,498, with an average of 149 times (ISI web of knowledge, June 19, 2016).

Progress in research is heavily reliant on high quality and accessible software packages (Katz et al., 2014; Howison and Bullard, 2016). As discussed in previous publications, the basic rationale for mentioning software packages in journal articles is to allow other researchers to identify and locate them for onward utility (Howison and Bullard, 2016). We evaluated the accessibility status of the 214 software packages published in MER (Table S1 in Supplementary Material; Huang et al., 2016). Briefly, 125/214 (58.4%) of the software packages whose websites were indicated in the original papers were accessible, while up to 41.6% (89/214) were inaccessible. However, 26/89 (29.2%) of the inaccessible software packages had new websites or alternative sources provided, all of which were accessible, hence the final inaccessibility rate was reduced to 29.4% (63/214). Still, this rate is higher than that reported in a recent software accessibility survey (Howison and Bullard, 2016). It is noteworthy that 100% (41/41) of the software packages deposited in the long-term hosting services were accessible. On the contrary, only 63.6% (110/173) of the remaining software packages deposited in other types of repositories were accessible. In general, accessibility is an important prerequisite for resource-sharing. **Figure 2** highlights a steady upward trend of accessibility of software packages from the year 2008, possibly related to the introduction of GitHub and CRAN, which have become popular among researchers for depositing their software, and reflects the role of long-term servers in resource-sharing. Interestingly, the average citation for software packages with online accessibility was significantly higher than those which were inaccessible (186.1 vs. 60.1; $P = 0.017$, T test for two independent samples). Whereas many factor like standardization of citation principles, user friendliness, and non-substitutionability of software packages influence their "citability" (Howison and Bullard, 2016; Smith et al., 2016), accessibility also seems to promote reproduction and citation of software packages.

It is also typical for software packages to require updates to fix bugs and/or improve their performance (Howison

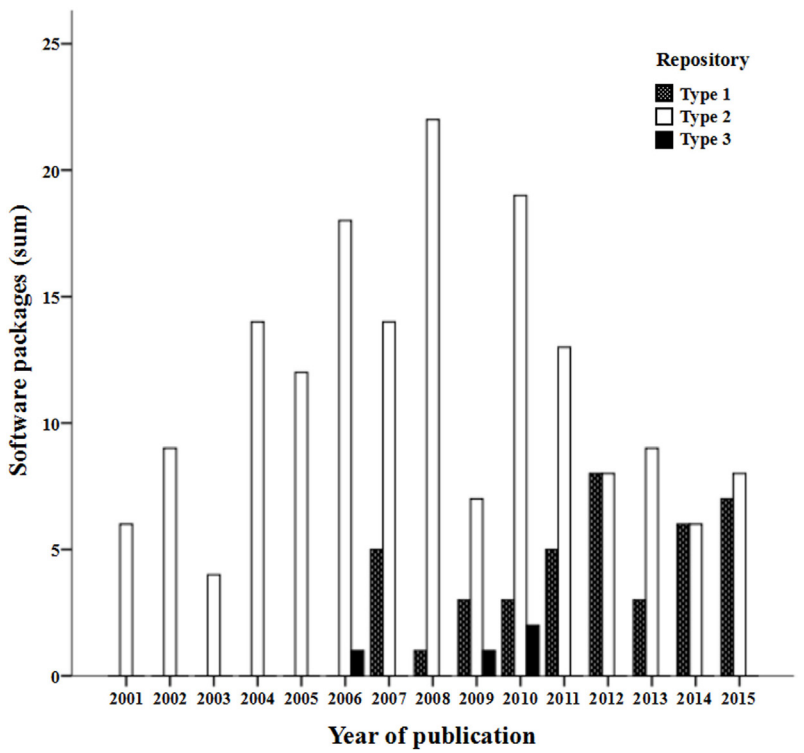

FIGURE 1 | Statistical summary of 214 software packages published in Molecular Ecology Resource (2001–2015). Type 1, long-term servers such as GitHub, Google, Sourceforge, or CRAN; type 2, personal homepage or academic institute websites; type 3, e-mail or no website.

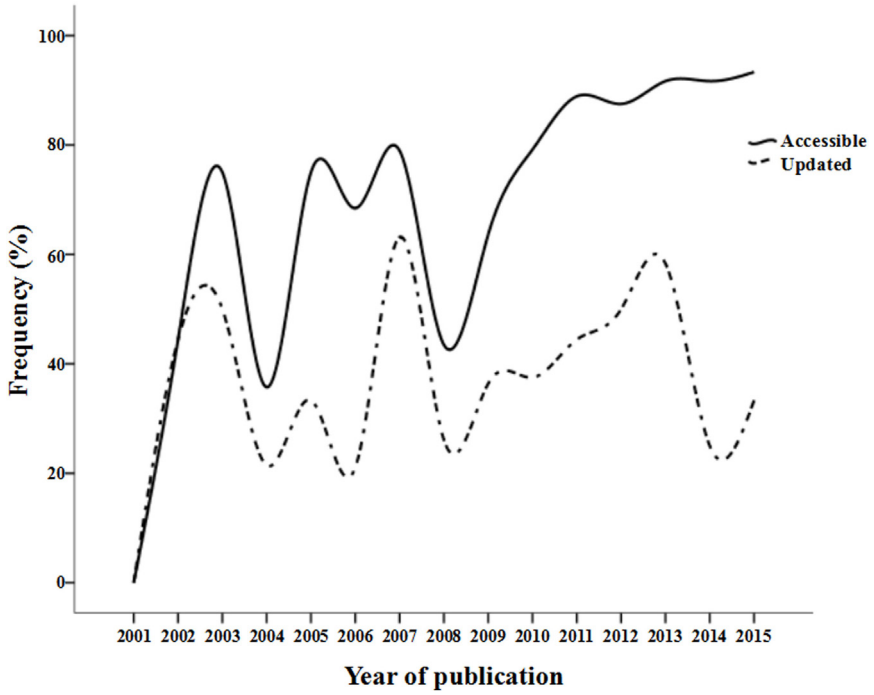

FIGURE 2 | Analysis of the accessibility and update status of 214 software packages (2001–2015) published in Molecular Ecology Resource.

and Bullard, 2016). Assessing the update status of the 214 published software packages (Table S1 in Supplementary Material; Huang et al., 2016), we observed that 79/214 (36.9%) had been updated since publication, 72/214 (33.6%) had not had any updates, and 63/214 (29.4%) were inaccessible; hence their update status was indeterminable. Overall, 63.1% (135/214) software packages had not been updated or were inaccessible. Although it could be likely that more recently published software may have not yet needed updates, 93/141 (66.0%) of the software packages published before 2011 had not had any updates or were inaccessible (**Figure 2**). On further scrutiny, it became evident that software packages with online accessibility and updated status had more literature citations (**Table 1**), and all the software packages that had been updated (79/214) were accessible. Overall, with the indeterminable status considered as "not updated," updated software packages were more likely to have higher citation frequency than the un-updated ones (mean 244.8 vs. 92.9; $P = 0.040$, T test for two independent samples), with a correlation between software update status and accessibility observed (Pearson coefficient = 0.494; $P = 0.001$).

In order to validate these findings, we additionally assessed a total of 49 software packages published in *BMC Bioinformatics* in the year 2004, volume 5 (Table S2 in Supplementary Material; Huang et al., 2016). The results of this validation are in agreement with our initial findings. The accessibility rate was only 34.7% (17/49), with all the software packages deposited in long-term repositories (4/4) being accessible. Software packages with online accessibility had higher mean citation frequency in published literature compared with the inaccessible ones, 195.1 vs. 36.7, respectively ($P = 0.163$, T test for two independent samples). In addition, 28.6% (14/49) of the software packages had been updated since their publication, and the mean citation frequency of the accessible and updated software packages (14/17) was higher compared with that of accessible but un-updated software packages (3/17) (236.5 vs.1.7, respectively; $P = 0.576$). This analysis also indicated a strong correlation between software update status and accessibility (Pearson coefficient = 0.868; $P = 0.001$), with all the 14 updated software packages being accessible.

Publication of software under open-source policy, including the links to their source websites, is a sound approach for building up a rich resource that could be utilized by researchers all over the world for non-commercial purposes. Accessibility of the software packages/codes is a fundamental element that facilitates their independent reproduction, validation, and/or utility. However, as illustrated in these analyses, inaccessibility of published software packages remains a major weak-link that needs to be addressed. Going forward, software developers and publishers should not only strongly commit to following the author guidelines of journals but also include the software alongside the journal article itself, possibly as part of the workflow (Howison and Bullard, 2016) or as a separate software paper (Smith et al., 2016), and/or deposit them into long-term servers like GitHub (GitHub, 2014) rather than only personal homepages. At the same time, technologies and user preferences keep evolving, necessitating incremental improvement of software packages. This may sometimes be

TABLE 1 | Analysis of software citation according to the accessibility and update status of software packages.

	Accessible and updated (a) (n = 79)	Accessible and not updated (b) (n = 72)	Inaccessible and indeterminate (c) (n = 63)	P-value†		
				(a,b)	(a,c)	(b,c)
No. citations (% of N)	19,342 (60.7)	8,758 (27.5)	3,786 (11.9)	0.221	0.033	0.378
Mean citations (SD)	244.8 (674.9)	121.6 (542.8)	60.1 (110.6)			

†*T test P-values for two independent samples shown.*

subject to availability of funding beyond the lapsing of the initial projects. However, where new updates are available, they should be centrally archived to ensure continuous availability. Additionally, where new software versions or alternative websites are available, timely notices should be released to guide users accordingly.

This article highlights the benefits, to both the software developers and the general scientific community, of keeping software updated and in continuous access, failure of which would result into missed opportunities for reproduction and further improvements. Achieving this requires concerted efforts of not only software developers and authors but also of publishing journals and software repositories.

AUTHOR CONTRIBUTIONS

XH designed the research; JX collected the data; XH and NO analyzed the data; and XH, NO, and MP wrote the paper.

ACKNOWLEDGMENTS

The authors thank Zhijia Wen, Wenli Liu, Shihao Zhang, Yujing He, and Jiaxuan Li of Jiaying University for technique assistance. This work was supported by Guangdong Natural Science Foundation (2014A030307018), Innovation and Strong School Project of Jiaying University (CQX019), and Science and Technology Planning Project of Guangdong Province (2016A030303068). NO thanks the support of CAS-TWAS President's Fellowship Program for Doctoral Candidates. MP thanks the support from the Youth Innovation Promotion Association, Chinese Academy of Sciences.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://journal.frontiersin.org/article/10.3389/frma.2017.00001/full#supplementary-material.

REFERENCES

GitHub. (2014). *Making Your Code Citable with GitHub & Zenodo*. Available at: https://guides.github.com/activities/citable-code/

Howison, J., and Bullard, J. (2016). Software in the scientific literature: problems with seeing, finding, and using software mentioned in the biology literature. *J. Assn. Inf. Sci. Technol.* 67, 2137–2155. doi:10.1002/asi.23538

Huang, X., Xie, J., Otecko, N., and Peng, M. (2016). [dataset] Accessibility and update status of published software: benefits and missed opportunities. doi:10.6084/m9.figshare.4257596

Katz, D. S., Choi, S.-C. T., Lapp, H., Maheshwari, K., Löffler, F., Turk, M., et al. (2014). Summary of the first workshop on sustainable software for science: practice and experiences (WSSSPE1). *JORS* arXiv:1404.7414. doi:10.5334/jors.an

Narum, S. (2016). Editorial 2016. *Mol. Ecol. Resour.* 16, 1–6. doi:10.1111/1755-0998.12494

Smith, A. M., Katz, D. S., Niemeyer, K. E., and FORCE11 Software Citation Working Group. (2016). Software citation principles. *PeerJ. Comput. Sci.* 2, e86. doi:10.7717/peerj-cs.86

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.