Enabling the Verification of Computational Results

An Empirical Evaluation of Computational Reproducibility

Victoria Stodden
University of Illinois at
Urbana-Champaign
Champaign, IL, USA
vcs@stodden.net

Matthew S. Krafczyk
University of Illinois at
Urbana-Champaign
Champaign, IL, USA
mkrafcz2@illinois.edu

Adhithya Bhaskar
University of Illinois at
Urbana-Champaign
Champaign, IL, USA
bhaskar7@illinois.edu

ABSTRACT

The ability to independently regenerate published computational claims is widely recognized as a key component of scientific reproducibility. In this article we take a narrow interpretation of this goal, and attempt to regenerate published claims from author-supplied information, including data, code, inputs, and other provided specifications, on a different computational system than that used by the original authors. We are motivated by Claerbout and Donoho's exhortation of the importance of providing complete information for reproducibility of the published claim. We chose the Elsevier journal, the Journal of Computational Physics, which has stated author guidelines that encourage the availability of computational digital artifacts that support scholarly findings. In an IRB approved study at the University of Illinois at Urbana-Champaign (IRB #17329) we gathered artifacts from a sample of authors who published in this journal in 2016 and 2017. We then used the ICERM criteria generated at the 2012 ICERM workshop "Reproducibility in Computational and Experimental Mathematics" to evaluate the sufficiency of the information provided in the publications and the ease with which the digital artifacts afforded computational reproducibility. We find that, for the articles for which we obtained computational artifacts, we could not easily regenerate the findings for 67% of them, and we were unable to easily regenerate all the findings for any of the articles. We then evaluated the artifacts we did obtain (55 of 306 articles) and find that the main barriers to computational reproducibility are inadequate documentation of code, data, and workflow information (70.9%), missing code function and setting information, and missing licensing information (75%). We recommend improvements based on these findings, including the deposit of supporting digital artifacts for reproducibility as a condition of publication, and verification of computational findings via re-execution of the code when possible.

CCS CONCEPTS

• **Information systems → Database design and models**; **Digital libraries and archives**; **Computing platforms**; • **Software and its engineering → Empirical software validation**; • **Theory of computation →** *Computability*;

KEYWORDS

reproducible research, data access, code access, workflows, provenance, reproducibility policy

ACM Reference Format:
Victoria Stodden, Matthew S. Krafczyk, and Adhithya Bhaskar. 2018. Enabling the Verification of Computational Results: An Empirical Evaluation of Computational Reproducibility. In *P-RECS'18: First International Workshop on Practical Reproducible Evaluation of Computer Systems, June 11, 2018, Tempe, AZ, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3214239.3214242

1 INTRODUCTION

In this article we identify barriers and outline solutions to the dissemination of "really reproducible research." We follow Claerbout and Donoho's exhortation of the importance of including complete author-provided information that enables transparency and reproducibility for computational and data-enabled claims [Buckheit and Donoho 1995; Claerbout 1994; Donoho et al. 2009; Schwab et al. 2000]. For the purposes of this study, we adopt the Claerbout definition of reproducibility: "computational reproducibility" [Stodden et al. 2013b] which refers to the verification of the computational steps, including input data, parameters, and other information, that generated the computational claims presented in the associated article [Barba 2018]. Note that this definition does not verify *scientific* correctness. Therefore, our study evaluates the sufficiency of information regarding availability of digital artifacts such as data and code, and seeks to procure artifacts from authors if they are not accessible via the article alone. Finally we evaluate the ease at which we can then regenerate the associated published claims using the author artifacts. The aim of this work is to better understand author and community needs regarding artifact availability to enable computational reproducibility. We build on the Transparency and Openness Promotion Guidelines that promote artifact availability [Nosek et al. 2015] and the Reproducibility Enhancement Principles (REP) that provide community steps toward reproducibility [Stodden et al. 2016]. It is important to note that this work is focused on computational reproducibility and we are not making statements about the scientific correctness of the claims made in the articles studied.

2 EXPERIMENTAL DESIGN FOR TESTING COMPUTATIONAL REPRODUCIBILITY

We designed an experiment to assess the reproducibility of computational publications as follows. We chose to analyze articles published in the Journal of Computational Physics, both because it is a leading journal that exclusively publishes computational

P-RECS'18, June 11, 2018, Tempe, AZ, USA

ACM ISBN 978-1-4503-5861-3/18/06.
https://doi.org/10.1145/3214239.3214242

scientific findings and physics is a domain that to the best of our knowledge has not yet been studied with regard to computational reproducibility [Gentleman and Lang 2007; Ioannidis et al. 2008; King 1995]. Additionally the Journal of Computational Physics (JCP) is in the Elsevier family of journals and at the time of publication of the articles we studied, authors guidelines encouraged both data and code availability as follows:

> Research data should be made available free of charge to all researchers wherever possible and with minimal reuse restrictions. ... Research data can include but are not limited to: raw data, processed data, software, algorithms, protocols, methods, materials. ... [Elsevier will e]ncourage and support researchers and research institutions to share data where appropriate and at the earliest opportunity.[1]

These author guidelines have been updated and the following text now appears directly on the JCP Author Guidelines webpage under "Research Data"[2], in addition to the [unchanged] above text on the Elsevier Research Data Policy page:[3]

> This journal encourages and enables you to share data that supports your research publication where appropriate, and enables you to interlink the data with your published articles. Research data refers to the results of observations or experimentation that validate research findings. To facilitate reproducibility and data reuse, this journal also encourages you to share your software, code, models, algorithms, protocols, methods and other useful materials related to the project.[4]

We chose a sample size of 300 based on power calculations. To get 300 articles we started with Issue 322 of the Journal of Computational Physics and collected articles through Issue 331 which gave us 307. We eliminated one article which was a comment on a previously article. We then applied evaluation criteria to the 306 articles to assess the following three categories: the level of information in the article enabling computational reproducibility; the level of information provided on computational artifacts such as data and code that support the claims made in the article; and the facility at which that information and artifacts enabled the regeneration of the computational results in the article. We choose the evaluation criteria published as in Appendix D "Best Practices for Publishing Research" [Bailey et al. 2013; Stodden et al. 2013a]. The ICERM workshop brought together a broad cross-section of computational scientists and mathematicians with other stakeholders such as publishers and software developers to discuss these issues and brainstorm ways to improve current practices, and produced the evaluation criteria.

We scoured the articles in our study to find associated data, code, and other artifacts. Six of the 306 articles provided sufficient information allowing us to discover the digital artifacts without contacting the authors, and we emailed the corresponding author of the remaining articles with a detailed request for supporting data and/or code (we emailed 298 authors as there were two articles in our sample with the same author). The emails were sent from one of two undergraduate students to minimize bias through potential name recognition and to test the ability of junior scholars to obtain responses. A follow-up email was sent after two weeks if there was no response.

Table 1: Artifact Access via Information in the Article (N=306)

No discussion in the article, and no artifacts made available	58.8%
Some discussion of artifacts, none made available	35.6%
Some artifacts made available	5.6%

This work builds on and extends the 2016 work of Collberg and Proebsting that found 38% of computer science authors released their source code and of those they found 32% of the results those codes support to be "weakly repeatable," which they defined as buildable within the first 30 minutes of attempting [Collberg and Proebsting 2016]. Their "weakly repeatable" metric is the closest to the replication attempts made in this study. We limited time to reproduce to 4 hours of human attempts to build and run the code, including runtime. This decision was based on the dual considerations of straightforwardness of implementation and researcher time. For those articles with extended runtimes we distinguished between those that failed to build and those that ran but exceeded the 4 hours limit. This research also extends work that attempts replication for computational claims published in *Science*, with the findings that they obtained artifacts such as data and code from 44% of articles and they were able to reproduce the claims for 26% of the articles [Stodden et al. 2018].

3 RESULTS FROM ARTIFACT AND REPLICATION EVALUATION

3.1 Evaluation 1: Artifact Information Provided in the Article

We first evaluated the presentation of information in the body of the article itself. As shown in Table 1 we found that only about 6% (17 articles) of articles gave information making some artifacts available, and about 36% discussed the artifacts, e.g. a mention of code, in the article.

Since we wished to obtain associated artifacts and test the ability to regenerate the computational results in this paper, we emailed the corresponding author for 298 articles with a detailed request. We did not receive a reply from 37% of the authors, we received a reply but did not receive any artifacts from 48% of authors, and roughly 15% supplied some artifacts to us.

[1] Obtained from the data policy guidelines linked to from the JCP Author Guidelines https://www.elsevier.com/about/our-business/policies/research-data Accessed April 9 2017.

[2] See https://www.elsevier.com/journals/journal-of-computational-physics/0021-9991/guide-for-authors Accessed April 8 2018

[3] https://www.elsevier.com/about/our-business/policies/research-data Accessed April 8 2018.

[4] See https://www.elsevier.com/journals/journal-of-computational-physics/0021-9991/guide-for-authors Accessed April 8 2018.

Table 2: ICERM Article Information Evaluation Criteria Implementation (n=55)

A precise statement of assertions to be made in the paper	100%
Full statement (or valid summary) of experimental results	100%
Salient details of data reduction & statistical analysis methods	73%
Necessary run parameters were given	86%
A statement of the computational approach, and why it rigorously tests the hypothesized assertions	100%
Complete statements of, or references to, algorithms and salient software details	63%
Discussion of the adequacy of parameters such as precision level and grid resolution	76%
Proper citation of all code and data used, including that generated by the authors	4%
Availability of computer code, input and output data, with some reasonable level of documentation	4%
Avenues of exploration examined throughout development, including negative findings	0%
Instructions for repeating computational experiments described in the article	79%
Precise functions were given, with settings	11%
Salient details of the test environment e.g. hardware, system software, and number of processors used	24%

Table 3: Evaluation of Artifacts and Archiving (n=55)

Data documented to clearly explain what each part represents	40%
Data archived with significant longevity expected	27%
Data location provided in the acknowledgements	13%
Authors have documented use and licensing rights	29%
Software documented well enough to run it and what it ought to do	71%
The code is publicly available with no download requirements	27%
There was some method to track software changes, and some persistence of archiving	20%

Table 4: Computational Reproducibility Evaluation (n=55)

Straightforward to reproduce with minimal effort	0%
Minor difficulty in reproducing	0%
Reproducible after some tweaking	9.1%
Could reproduce with fairly substantial skill and knowledge	16.4%
Reproducible with substantial intellectual effort	12.7%
Reproducible with substantial tedious effort	3.6%
Difficult to reproduce because of unavoidable inherent complexity	3.6%
Nearly impossible to reproduce	3.6%
Impossible to reproduce	50.9%

3.2 Evaluation 2: Archiving and Replication Efforts

After emailing authors we had artifacts for 55 articles (including the 6 articles we deemed possessed sufficient artifacts and were not emailed). We implemented the ICERM Evaluation Criteria discussed previously and present the results in Table 2. There is notable reporting weakness on avenues explored that did not directly contribute to the published output, for example model parameter tuning attempts, exploration of hypotheses that did not end up being supported, or various grid sizes tried. Clarity regarding precise function specifications, with precise inputs and settings, was a reporting weakness along with details of the experimental environment such as machine state information. Articles excelled at clarity in exposition of their assertions and results, and justification of their computational approach. We evaluated the artifacts we received, reported in Table 3, and then we report on our attempts to use the artifacts to regenerate the computational claims in the associated article in Table 4. We acknowledge a subjective aspect to our evaluation and follow related efforts in [Stodden et al. 2018]. Table 3 shows apparent weakness on all aspects of documentation and archiving.

As noted, we allocated 4 hours per article to attempt replication of the published claims given the artifacts provided. We used one of two systems with identical software environments: the first is a laptop with Arch linux OS with Intel core i7-4910MQ CPU @ 2.90 GHz (4 core 8 thread) and Nvidia GeForce GTX 980M and the second is a desktop with Arch linux OS with Intel core i7-6900K CPU @ 3.20 GHz (8 core 16 thread) and two Nvidia GeForce GTX 1080 GPUs (with one exception: we used a Windows 10 Virtual Machine for one article whose artifacts required MSVC to compile and Windows to run). All reproduction attempts were carried out in a linux environment with the exception of the windows-based article. Replications were carried out by people with extensive programming experience (from physics and from mechanical engineering). An attempt was made to build any software in the project and once built, test cases (if they existed) were run to verify that the software worked correctly. The article was then inspected for tables and figures, and the code searched for sections either explicitly mentioning these figures or producing data that was close to that in the article. In these cases, parameters were checked in the code to be sure they matched those specified in the article. If not all parameters could be found, the code was inspected for equations matching those in the article. If no resemblance was found, the article was marked as impossible to reproduce. If it became apparent that essential input data such as a model, grid resolution, or initial state was missing, the article was marked as impossible to reproduce. This is due to the fact that we could not proceed with the replication without this information. As indicated in 4, we expect an domain expert to be able to regenerate findings for half of the articles that provided artifacts (27/306 or 8.8% of the total number of articles in our study). None of the articles with artifacts were straightforward to reproduce, and about 9% could be considered fairly easy. With some skill and work another 40% can be replicated. When the articles did not replicate the failure was typically due to: libraries released with the article's method implemented but lacking test cases and input data such as initial conditions; parameter specifications were missing; code had evolved to a new version; missing function definitions; visualization code was missing (or proprietary); artifacts were provided for some but not all claims in the article. For the 55 articles with artifacts, we fully replicated none; partially replicated 32.7% (18); ran 54.5% (30); were able to build 3.6% (2); and had no progress on 9.1% (5). The GitHub repository associated with this article contains details on how classifications into these categories were made (see Conclusions section).

4 RECOMMENDATIONS

As a result of this work, several recommendations come to light.

- The development of community standards regarding the documentation of computational research. For example, providing information on what major functions do, input parameters and upstream function calls, function invocation sequences, library versions and dependencies, version controls including hashing, and workflow information.
- Journals can improve the communication of standards for publishing, including documentation standards. For example, the Elsevier guidelines for the Journal of Computational Physics could include examples of appropriate sharing. Some conferences and journals, e.g. PPoPP and the Journal of the American Statistical Association Applications and Case Studies, now require the submission of a structured two page Artifact Appendix with every manuscript detailing the associated code and data.
- Appropriate use of open licensing is essential for computational reproducibility since it enables legal access and use of author created work that falls under copyright, such as software [Stodden 2009]. All artifacts should carry appropriate open licenses such as the MIT License or the Creative Commons Public Domain Certification CC0.
- Cultural change and improved cyberinfrastructure and tools to enable the reporting of negative results and research avenues explored that inform but not part of the scientific findings directly. Exploratory work should be reported when it affects the interpretation of the published results.
- Greater investments in reproducibility research and cyberinfrasturcture and tools that support computational reproducibility.
- Researchers and others involved in the research and publication process need improved training in reproducibility practices. As noted in 2018, "training in best practices for digital scholarship and reproducibility should be integrated into research-methodology curricula" [Berman et al. 2018].

5 CONCLUSIONS

This work aims to provide greater clarity and guidance regarding the dissemination of computational claims. It highlights several areas for improvement: providing appropriate test cases; providing input models and parameters along with the solver engines and analysis/visualization code; and using version control to provide the precise code used to generate the published results. For the majority of articles in this study, no artifacts that enable computational reproducibility were available, which is a straightforward first step for the community to take. Our own artifacts for this study, including details on the generation of the tables we present, are available at see https://github.com/ReproducibilityInPublishing/P-RECS-2018-Enabling-Verification.

ACKNOWLEDGMENTS

The authors would like to thank David Wong, Yantong Zhang, and Alex Dickinson for outstanding research assistance. We also acknowledge support from NSF Award ACI-1659702 and the NCSA SPIN program.

REFERENCES

David H Bailey, Jonathan Borwein, and Victoria Stodden. 2013. Set the Default to 'Open'. *Notices of the AMS* (2013).

Lorena A. Barba. 2018. Terminologies for Reproducible Research. *CoRR* abs/1802.03311 (2018). arXiv:1802.03311 http://arxiv.org/abs/1802.03311

Francine Berman, Rob Rutenbar, Brent Hailpern, Henrik Christensen, Susan Davidson, Deborah Estrin, Michael Franklin, Margaret Martonosi, Padma Raghavan, Victoria Stodden, and Alexander S. Szalay. 2018. Realizing the Potential of Data Science. *Commun. ACM* 61, 4 (March 2018), 67–72. https://doi.org/10.1145/3188721

J. Buckheit and D. Donoho. 1995. *WaveLab Architecture.* Technical Report. Stanford University, http://www-stat.stanford.edu/ wavelab.

J. Claerbout. 1994. *Hypertext Documents about Reproducible Research.* Technical Report. Stanford University, http://sepwww.stanford.edu.

Christian Collberg and Todd A. Proebsting. 2016. Repeatability in Computer Systems Research. *Commun. ACM* 59, 3 (Feb. 2016), 62–69. https://doi.org/10.1145/2812803

David L Donoho, Arian Maleki, Inam Ur Rahman, Morteza Shahram, and Victoria Stodden. 2009. Reproducible research in computational harmonic analysis. *Computing in Science & Engineering* 11, 1 (2009), 8–18.

Robert Gentleman and Duncan Temple Lang. 2007. Statistical Analyses and Reproducible Research. *Journal of Computational and Graphical Statistics* 16, 1 (2007), 1–23. https://doi.org/10.1198/106186007X178663 arXiv:http://dx.doi.org/10.1198/106186007X178663

John P A Ioannidis, David B Allison, Catherine A Ball, Issa Coulibaly, Xiangqin Cui, Aedín C Culhane, Mario Falchi, Cesare Furlanello, Laurence Game, Giuseppe Jurman, Jon Mangion, Tapan Mehta, Michael Nitzberg, Grier P Page, Enrico Petretto, and Vera van Noort. 2008. Repeatability of published microarray gene expression analyses. *Nature Genetics* 41 (28 01 2008). http://dx.doi.org/10.1038/ng.295

Gary King. 1995. Replication, replication. *PS: Political Science and Politics* 28, 3 (1995), 444–452.

B. A. Nosek, G. Alter, G. C. Banks, D. Borsboom, S. D. Bowman, S. J. Breckler, S. Buck, C. D. Chambers, G. Chin, G. Christensen, M. Contestabile, A. Dafoe, E. Eich, J. Freese, R. Glennerster, D. Goroff, D. P. Green, B. Hesse, M. Humphreys, J. Ishiyama, D. Karlan, A. Kraut, A. Lupia, P. Mabry, T. Madon, N. Malhotra, E. Mayo-Wilson, M. McNutt, E. Miguel, E. Levy Paluck, U. Simonsohn, C. Soderberg, B. A. Spellman, J. Turitto, G. VandenBos, S. Vazire, E. J. Wagenmakers, R. Wilson, and T. Yarkoni. 2015. Promoting an open research culture. *Science* 348, 6242 (2015), 1422–1425. https://doi.org/10.1126/science.aab2374 arXiv:http://science.sciencemag.org/content/348/6242/1422.full.pdf

M. Schwab, N. Karrenbach, and J. Claerbout. 2000. Making scientific computations reproducible. *Computing in Science & Engineering* 2, 6 (2000), 61–67.

Victoria Stodden. 2009. The Legal Framework for Reproducible Scientific Research: Licensing and Copyright. *Computing in Science Engineering* 11, 1 (2009), 35–40. https://doi.org/10.1109/MCSE.2009.19

Victoria Stodden, Jonathan Borwein, and David H Bailey. 2013a. 'Setting the Default to Reproducible' in Computational Science Research. *SIAM News* (2013).

Victoria Stodden, Peixuan Guo, and Zhaokun Ma. 2013b. Toward Reproducible Computational Research: An Empirical Analysis of Data and Code Policy Adoption by Journals. *PloS one* 8, 6 (2013), e67111.

Victoria Stodden, Marcia McNutt, David H. Bailey, Ewa Deelman, Yolanda Gil, Brooks Hanson, Michael A. Heroux, John P.A. Ioannidis, and Michela Taufer. 2016. Enhancing reproducibility for computational methods. *Science* 354, 6317 (2016), 1240–1241. https://doi.org/10.1126/science.aah6168 arXiv:http://science.sciencemag.org/content/354/6317/1240.full.pdf

Victoria Stodden, Jennifer Seiler, and Zhaokun Ma. 2018. An empirical analysis of journal policy effectiveness for computational reproducibility. *Proceedings of the National Academy of Sciences* 115, 11 (2018), 2584–2589. https://doi.org/10.1073/pnas.1708290115 arXiv:http://www.pnas.org/content/115/11/2584.full.pdf