



Journal of Statistical Software

October 2014, Volume 61, Book Review 2.

<http://www.jstatsoft.org/>

Reviewer: Brigid Wilson
Department of Veterans Affairs

Implementing Reproducible Research

Victoria Stodden, Friedrich Leisch, Roger D. Peng (Eds.)

CRC Press, Boca Raton, 2014.

ISBN 978-1-4665-6159-5. 428 pp. USD 79.95 (P).

<http://www.crcpress.com/product/isbn/9781466561595>

The self-correcting nature of science has been questioned repeatedly in the decade since Ioannidis (2005), “Why Most Published Research Findings Are False”. The title of the paper, if not the entirety of its content, was widely publicized and called into question whether scientific research was truly contributing to human knowledge and whether public faith in science was misplaced. The existential crisis that ensued within the scientific community identified a lack of replication and reproducibility as a major problem that needed to be addressed.

Implementing Reproducible Research aims to provide researchers with tools that facilitate reproducibility and with practices that might push research towards reproducibility as a widely-held standard. The editors define and distinguish between replicability and reproducibility in the preface. Under their definitions, replication involves implementing experiments and collecting new data for analysis; reproducibility involves “the calculation of quantitative scientific results by independent scientists using the original datasets and methods”. By these definitions, then, reproducibility is necessary but not sufficient for replication.

Several chapters of the book are aimed at researchers performing computational experiments for which experimental data-gathering is not relevant. In such cases, the hurdles to reproducibility are computational rather than the physical, chemical, or biological hurdles of wet lab replication or the environmental and human hurdles of social science replication. These hurdles are not trivial and the tools and techniques described range from **Sumatra**, a software tool that aims to wrap version control, dependency tracking, platform releases, and input/output changes around the standard practices of researchers; to packaging code, data, and environment (using a tool called **CDE**); to hosting a virtual machine image of the experiment-running machine that can be stored and accessed through cloud computing. The last is presented in Bill Howe’s chapter, “Reproducibility, Virtual Appliances, and Cloud Computing”. Howe suggests that, among the options for documenting and disseminating a reproducible computational experiment, a trade-off exists between the effort required by the researcher and the effort required by others who wish to modify and build on the results. The reader is left to ponder: What are the responsibilities of the researcher? Do they end at reproducibility or are

there obligations to future researchers? These are questions for the community to consider as it continues to formulate research etiquette and good practices around disseminating results.

Other chapters are aimed at specific disciplines or settings. “Reproducible Bioinformatics Research for Biologists” lists the many dangers of inexpert bioinformatics performed by biologists and aims to “help those biologists who have zero or little background in computation to get started with good practices and tools for computational science.” With an assumption of no computational background, this chapter starts with a few basics (e.g., work in the command line and save your commands) that go a long way towards tracking steps and reproducing experiments. The chapter concludes with compiled languages that will help the biologist better understand what underlies the scripting languages they use. The need for computing skills among biologists, who are increasingly generating more data and more complex data than they were trained to analyze, is undeniable. But much of the list laid out in this chapter seems unrealistic. A caveat to biologist readers might be appropriate: if you are uninclined to pursue this level of computational training, initiate collaborations with statisticians and bioinformaticians.

“Reproducible Physical Science and the **Declaratron**” is written for a chemistry, crystallography, and material science audience focusing on “semantics”, composed of code and dictionaries, which the authors claim are lacking in the proprietary software currently in use. Unspecified units, unstandardized measures, and other ambiguities add researcher hours to any experiment and, despite the researchers’ best efforts, can yield results that are not reproducible. The authors introduce the **Declaratron** as public software developed to remedy this. “Open Science in Machine Learning” encourages researchers to publish their software and for the machine learning community to adapt standard protocol towards interoperability.

One chapter, “The Reproducibility Project: A Model of Large-Scale Collaboration for Empirical Research on Reproducibility” departs from the rest of the book in several ways. It is the only chapter in which the experiments discussed are in psychology, and direct replications – rerunning experiments, collecting and analyzing new data – are performed. It also presents a different attitude towards reproducibility. In the chapters discussing computational experiments, there is a recurring theme of a meritocracy among codes, algorithms, and (presumably) the researchers behind them. The authors of these chapters might not consider a computation experiment valid if it could not be reproduced. But when considering psychology experiments, there are several plausible reasons why a direct replication of an experiment may not confirm an original finding and a false result is just one in a list. The chapter authors make this explicit and present a framework for performing direct replication that includes open communication with the original authors. This inclusion of the original authors is essential if replication is to be considered an effort to advance the science on which the community of researchers builds by identifying false positives or providing further support to the original finding.

The “Practicing Open Science” chapter lists the requirements of open science as: open data, open source, open access, and open standards. The authors cite “release early, release often” as an open-source tenet. The authors make explicit that the merits of code and the reputation of its developer are inextricably linked, suggesting that a developer following this tenet is accepting reputation risks. To reward these risks of open science practitioners, they must be valued by the community.

The issue of disseminating code while retaining intellectual property protections is addressed in several chapters, with the most detail provided in Victoria Stodden’s chapter, “What Com-

putational Scientists Need to Know about Intellectual Property Law: A Primer”. Tools for literate programming like the **knitr** package allow programmers to integrate code and documentation. One chapter, “Developing Open-Source Scientific Practice”, acknowledges the social hurdles to reproducibility. Developing code with a community of researchers “is necessary for robust software ecosystems where we can share and verify our work”, but the incentive structures in publishing and academia can work against large, open collaborations.

All of the chapters can be read independently of the others and can be downloaded for free at <https://osf.io/s9tya/>. The book as a whole has something for everybody and provides an interesting snapshot of the available tools, platforms, and good practices for researchers as the scientific community aims to be more self-correcting.

References

Ioannidis JPA (2005). “Why Most Published Research Findings Are False.” *PLoS Medicine*, **2**, e124. doi:10.1371/journal.pmed.0020124.

Reviewer:

Brigid Wilson

Geriatric Research Education and Clinical Center

Department of Veterans Affairs

Louis Stokes Cleveland VAMC

10701 East Boulevard

Cleveland, OH 44106, United States of America

E-mail: brigid.wilson@va.gov

URL: http://www.va.gov/GRECC/Cleveland_GRECC.asp