Transparent and Reproducible Research with R

Daniel Anderson[1] & Joshua Rosenberg[2]

[1] University of Oregon
[2] University of Tennessee, Knoxville

Correspondence concerning this article should be addressed to Daniel Anderson, 5262 University of Oregon. E-mail: daniela@uoregon.edu

Proposal narrative: AERA Professional Development Proposal

The purpose of this proposal is to provide participants with an introduction to tools for open, transparent, and reproducible analysis workflows. Working in a reproducible framework represents a compromise between publishing a journal article alone, and full-scale replication (i.e., the "gold standard"). Some (e.g., R. D. Peng, 2011) have argued that reproducibility should be a basic, *minimal* standard as a means combating the so-called replicability crisis (see Hedges, 2018). We discuss R Markdown for integrating text with code and *GitHub* for collaboration and documentation of the project history. Through the use of these tools, quantitative analyses are more open and transparent, and the likelihood of the analysis being reproducible is increased.

Prerequisite skills or knowledge needed for course participation

All participants should have an interest in conducting open and transparent analyses. This training will be most useful to those with experience planning, carrying out, and writing up the results of a research project. We will specifically discuss moving from existing frameworks to reproducible frameworks using R, R Markdown, and *git*/*GitHub*.

All participants should have at least a basic familiarity with R and be comfortable with the idea of working in a scripting environment. Note that the presenters have partnered with DataCamp (https://www.datacamp.com) to help provide a platform for less experienced users to get "up to speed" prior to the training. DataCamp is an online learning platform for R (and related data science technologies) that includes direct instruction and opportunities to practice and apply the learned skills, all within the online platform. All participants will have access to the full suite of DataCamp modules for one month prior to, and one month following, our in-person training. Users with less experience with R will be asked to complete the Introduction to R, Working with the RStudio IDE: Part 1, and Introduction to the *tidyverse* modules prior to the training. Following the training, we recommend all participants re-visit the skills they have learned by completing the Reporting with R Markdown module.

Target course participants

Our target audience includes graduate students, emerging or early-career researchers, and continuing researchers interested in their own or their students'/trainees' work becoming more open, transparent, and reproducible. Because both an overview of ideas related to reproducibility as well as a number of quantitative and computational tools and approaches will be described, participants with less experience can benefit from developing a more conceptual understanding of open science and can turn to the tools later. This training is aimed at beginners who have little to no experience with R Markdown or version control, but who feel comfortable learning code and have at least a basic understanding of R.

Rationale

The basic premise of reproducible research is that quantitative analyses should be conducted and documented with sufficient clarity that independent researchers could reproduce all the results, exactly. Initially, this may sound relatively straightforward—of course

research findings should be reproducible—and we may like to think that most research in education adheres to these principles. Unfortunately, this is generally not the case. For example, in a large-scale review of growth models published from 2007-2012 across 47 education and psychology journals, Stevens J. J. and Tindal (2013) found that documenting even relatively routine procedures, such as how much missing data were involved and how the missing data were handled, was extraordinarily difficult. Indeed, in the vast majority of cases, this information was simply missing. These findings were likely the result of, at least in part, researchers attempting to summarize their study without getting bogged down in the details. The findings, however, are congruent with Buckheit and Donoho (1995), who argue "an article about computational science in a scientific publication is **not** the scholarship itself, it is merely the **advertising** of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures" (p. 5, emphasis in the original). This may seem a somewhat extreme view, but it clearly articulates that a journal article on its own is generally insufficient for the accumulation of scientific evidence. That is, it is difficult to build off the work of others, or verify study results, if you only have access to the published findings. It is important to note that reproducibility does not imply "correctness", but rather transparency in process. Indeed, part of the reason reproducible research is so important is that it allows other researchers a means of verifying the process, analysis, and ultimately, the validity of study findings.

Conducting reproducible research. Considerable recent attention has been paid to open and reproducible research in science generally (Bartling & Friesike, 2014; National Academies of Sciences & Medicine, 2018), but also in educational research (Cook, Lloyd, Mellor, Nosek, & Therrien, 2018; McBee, Makel, Peters, & Matthews, 2017; Zee & Reich, 2018). What is often lacking, however, is a clear accounting on how to actually begin engaging in open and reproducible research. This training seeks to fill that gap by providing an
initial introduction to tools to help make the process more tractable. In particular, we advocate for *conducting science publicly* through the publication of living code that is modified and updated as the project matures, along with *literate programming* a concept introduced by Knuth (1984) that weaves substantive text from the manuscript with analysis code. Although literate programming has its own learning curve, leading to an initial dip in productivity, it can eventually lead to massive gains in efficiency by all tables, figures, and in-text references to statistics (e.g., sample means) being updated produced through code and updated automatically each time the document is rendered. There is therefore no hand entering of data/statistics into tables, which can be error prone, and any tweaks to the model or data (including new data being added to the research) results in the entire document being updated automatically. The manuscript is therefore *dynamic* relative to the analysis (Xie, 2016). This process may sound complicated to implement, and it was even a few short years ago. Yet, the toolkit for producing dynamic, reproducible documents is rapidly expanding and is now far more accessible for the applied researcher. When this process is paired with a version control system such as *git*, and made publicly available through platforms such as *GitHub*, the project and process is far more open and transparent. Importantly, however, our training also discusses methods to ensure specific parts of the project (e.g., the raw data) are *not* available publicly.

Learning objectives

Participants in this training will: (a) understand why reproducibility is an increasingly important consideration for educational researchers; (b) be introduced to tools, specifically R Markdown and *git/GitHub*, for carrying out open and reproducible research; and (c) understand some of the remaining challenges that remain for open and reproducible research.

Course content

Our course introduces participants to the fundamental tenants of open and reproducible analysis workflows, including literate programming, documenting the project history, and working from a public platform (*GitHub*). Given that the training is four hours, we aim only to introduce participants to these concepts. However, despite the session serving as a primer, we prioritize hands-on applied practice. In our experience, the initial step in getting started can often be the greatest hurdle. A preliminary schedule follows:

Hour 1: Introduction to open and reproducible research.. The first hour will focus primarily on the substantive side of reproducible research–i.e., why are we all here? We will discuss the importance of reproducible research, covering infamous case-studies such as the Duke crisis (R. Peng, 2015) and others, but also introduce the ideas of literate programming and conducting science publicly. During the first hour, no specific code or tools will be covered and the focus will be on high-level conceptual understandings. The format will be primarily lecture with slides, with brief breakout sessions (< 5 minutes) in groups to discuss the covered topics.

Hour 2: R Markdown I. The second hour will be more hands-on and applied, asking participants to follow-along with one instructor, while the other roams the room and helps participants who are having trouble. We will introduce the very basics of R Markdown, including delineating code chunks from standard text, creating headers at different levels, creating bulleted lists, and bolding/italicizing text. We will also cover different code chunk options, including hiding/showing the code evaluating/not evaluating the code chunk. By the end of Hour 1, all participants should have at least a basic R Markdown document rendered to HTML with each of the aforementioned features.

Hour 3. In the third hour we will discuss moving R Markdown documents to different formats, including basic PDF documents and Microsoft Word. We then discuss the *papaja* (Aust & Barth, 2018) R package for preparing APA formatted manuscripts using the same basic R Markdown features. We also discuss including references within an R markdown document. Again, one instructor will lead a guided walk-through while the other circulates the room to assist participants. One possible complication with *papaja* is that it requires a *tex* distribution. We plan to address this by providing instructions prior to the training on installing the *tinytex* (Xie, 2018) package, which has all the required functionality but is a much smaller installation. By the end of Hour 3, all participants should have a basic APA formatted manuscript with at least one in-text citation and accompanying bibliography.

Hour 4

In the final hour of the training, we introduce participants to *GitHub*. The first 20 minutes of this section will be devoted to lecture, introducing participants to the basic commands (i.e., what a *repository* or *repo* is, as well as what it means to *stage*, *commit*, *pull*,

push, and *clone*). In the last 40 minutes we guide participants through creating a new repo and pushing their existing project to that repo. We then walk them through the process of making changes, committing those changes, and pushing them to the repo. Finally, we walk participants through the basics of *GitHub* platform to view the history of a project and conduct work openly. We also discuss the use of a *.gitignore* file to ensure specific files do *not* get pushed to the repo. By the end of Hour 4, all participants should have created their first publicly viewable GitHub repo that documents the history of their reproducible project from the workshop from their initial commit.

References

Aust, F., & Barth, M. (2018). *papaja: Create APA manuscripts with R Markdown*. Retrieved from https://github.com/crsh/papaja

Bartling, S., & Friesike, S. (2014). *Opening science: The evolving guide on how the internet is changing research, collaboration and scholarly publishing*. Springer.

Buckheit, J. B., & Donoho, D. L. (1995). Wavelab and reproducible research. In *Wavelets and statistics* (pp. 55–81). Springer.

Cook, B. G., Lloyd, J. W., Mellor, D., Nosek, B. A., & Therrien, W. (2018). Promoting open science to increase the trustworthiness of evidence in special education.

Hedges, L. V. (2018). Challenges in building usable knowledge in education. *Journal of Research on Educational Effectiveness*, *11*(1), 1–21.

Knuth, D. E. (1984). Literate programming. *The Computer Journal*, *27*(2), 97–111.

McBee, M., Makel, M., Peters, S. J., & Matthews, M. S. (2017). A manifesto for open science in giftedness research.

National Academies of Sciences, Engineering, & Medicine. (2018). *Open science by design: Realizing a vision for 21st century research*. Washington, DC: The National Academies Press. doi:10.17226/25116

Peng, R. (2015). The reproducibility crisis in science: A statistical counterattack. *Significance*, *12*(3), 30–32.

Peng, R. D. (2011). Reproducible research in computational science. *Science*, *334*(6060), 1226–1227.

Stevens J. J., Nese J. F. T., & Tindal, G. (2013). *Using longitudinal models to track student achievement: A literature synthesis.* Unpublished manuscript.

Xie, Y. (2016). *Dynamic documents with r and knitr*. Chapman; Hall/CRC.

Xie, Y. (2018). *Tinytex: Helper functions to install and maintain 'tex live', and compile 'latex' documents*. Retrieved from https://CRAN.R-project.org/package=tinytex

Zee, T. van der, & Reich, J. (2018). Open education science. *AERA Open*, *4*(3), 2332858418787466.