

reproduce.work: A framework to facilitate cross-platform computational reproducibility in scientific publishing

Alex P. Miller
USC Marshall School of Business
alex.miller@marshall.usc.edu

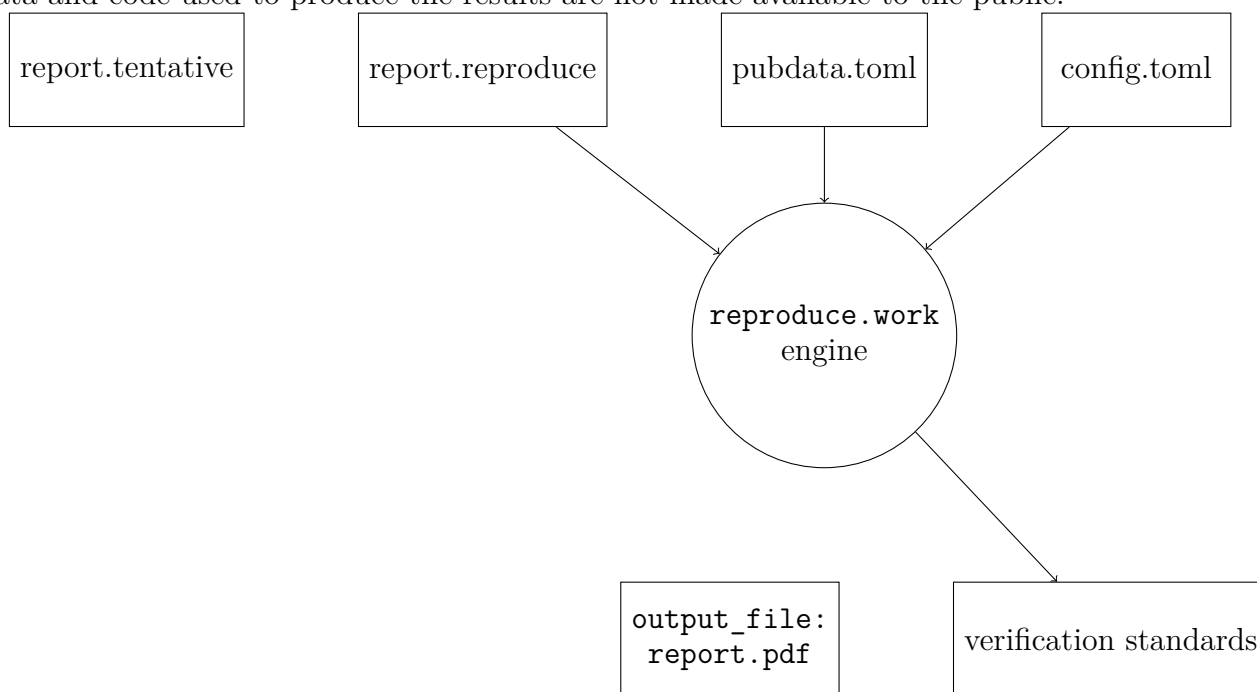
Last updated: October 5, 2023

Abstract

In metascience, computational reproduction is the process of reproducing the results of a scientific paper using the data and code provided by the authors of the paper. This subject sits within the broader context of "reproducibility" in scientific research, which has been core to the philosophy of science for decades. However, the practice of science has fallen woefully short of meeting even basic standards toward true and widespread reproducibility. In this project, we focus primarily on addressing the narrow problem of computational reproducibility. We propose a framework for facilitating computational reproducibility in scientific publishing, which we call reproduce.work. The reproducibility standards are designed to be cross-platform and to work with any programming language. We highlight the distinction between open and reproducible practices and showcase how our software framework encourages both simultaneously. The results of this very paper can be reproduced on any machine that can execute a containerized image using the reproduce.work workflow. We conclude by discussing the potential of the framework for improving rigor and fidelity of computational science for both producers and consumers of published work.

1 Introduction.

In metascience, computational reproduction is the process of reproducing the results of a scientific paper using the data and code provided by the authors of the paper. This subject sits within the broader context of “reproducibility” in scientific research, which is the idea that scientific results should be reproducible by other scientists (or anyone interested, for that matter). Concepts around reproducibility have been core to the philosophy of science for decades, but several aspects of the scientific method have been challenged by recent developments. First, recent conversations around p -hacking and fraud in scientific publishing have led to a “crisis” at the core of social science centered around the role of human error, incentives, and guile . Additionally , the increasing computational complexity of scientific research has led to a new set of challenges. Chief among these is the fact that the results of many scientific papers are not reproducible to even the lowest degree. This is because the data and code used to produce the results are not made available to the public.



1.1 Barriers to computational reproduction. Lack of documentation. Very rarely is it ever written down explicitly which code was run to generate which results in

a scientific report. Even if the code is open and published by the authors, it may be complex and difficult to parse. Scientific code is often written with the intention of generating specific set of results, rather than with the intention of being read and reproduced independently. As such, the code is often not documented in a way that is easy to understand.

Platform dependence. The reality is that is not enough to merely point to the code that generated a given piece of data. For even if a human were to track down the relevant commands, setting up the requisite computing environment to reproduce the results is often a non-trivial task. This is because the code that produced the results may depend on a specific version of a programming language, a specific version of a software package, or even a specific operating system.

Code complexity. Even if the computing environment is set up correctly, the code may be too complex to understand. This is especially true for code that is written by a researcher who is not a professional software developer. One script may “generate” a resulting variable for publication, but this script alone may depend on several other scripts and files in the process. Very rarely are complex scientific analyses possible to reproduce by running a simple function or script with a single command.

2 Background

advent of computational science has brought new challenges to the field.

At the heart of the concern with research by ousted Stanford president Mark is the fact that it is impossible to verify whether the images produced in the paper are fabricated or actually generated by the scientific equipment they claimed to use in the published manuscript.

The problem of reproducibility is not limited to the field of psychology.

Reproducibility across the disciplines: In 2012, the pharmaceutical company Amgen attempted to reproduce the results of 53 landmark cancer studies. They were only able to reproduce the results of 6 of the 53 studies.

In 2015, the journal Science published a study that attempted to reproduce the results of 100 psychology studies. They were only able to reproduce the results of 36 of the 100

studies.

In 2016, the journal Nature published a study that attempted to reproduce the results of 5 landmark cancer studies. They were only able to reproduce the results of 2 of the 5 studies.

In 2018, the journal Science published a study that attempted to reproduce the results of 21 landmark economics studies. They were only able to reproduce the results of 1 of the 21 studies.

this is known as the [replication crisis](#).

for real?

2.1 Related projects. There have been a number of projects with similar aims to this one.

Some work has been done to develop reproducible workflows within specific computing environments. See [example 1](#) in R.

While some initiatives have emerged to facilitate computational reproduction, there is no widely accepted standard for computational reproduction.

There do exist efforts around the ideas of open science, which are closely related but distinct from reproducible science. See [OSF](#), for example.

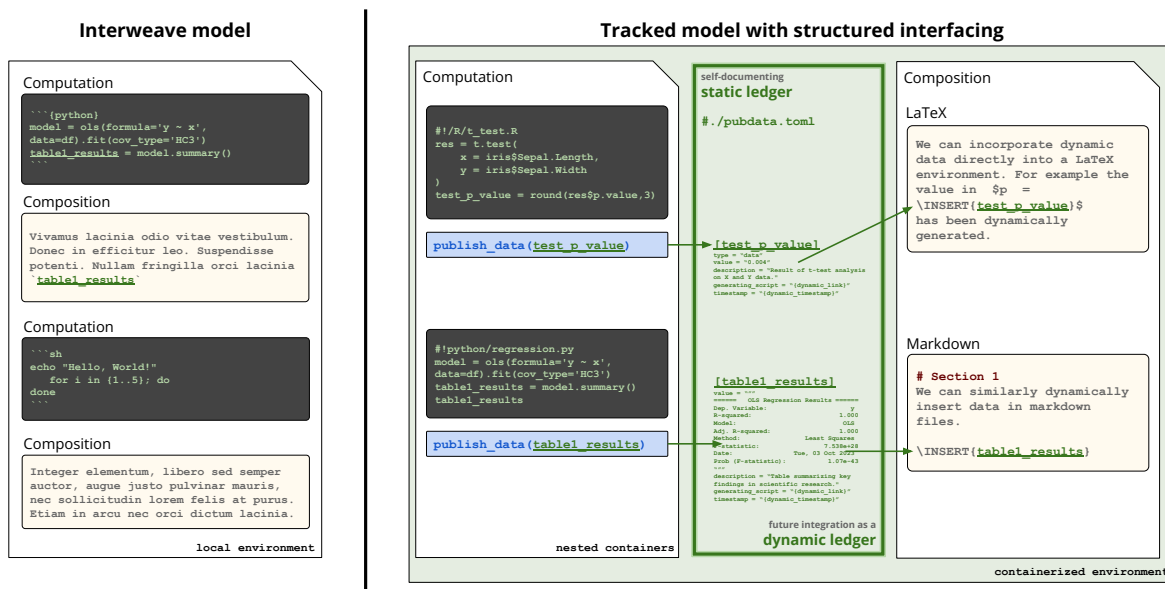
2.2 An ontology of computation and composition. Lay readers may be surprised to learn that the results of many scientific papers are not reproducible to even the lowest degree. (We put forward the following standard as the first of computational reproducibility: “More than one person verified that the results in the published paper match the results of the code and data provided by the authors.”)

This is because the data and code used to produce the results are not made available to the public.

such as the [ReScience journal](#) and [MyST Markdown](#)

While many advocate for the value of peer review in scientific publishing, as of 2023, there no scientific standard exists for true computational reproducibility.

Figure 1: Different models of ontology of computation and composition.



Note: The **interweave** model is one in which computation and composition happen linearly within the flow of a single document. Of course, many paradigms including RMarkdown and MyST are quite flexible and allow for a variety of workflows; however, we believe most existing software follows this model. This leads to our proposal of the **tracked model with structured interfacing**. In this model, the data and code are tracked in a way that allows for easy verification and traceability of reported scientific results with structured metadata.

This is a problem because it means that the quality of peer review varies from journal to journal, and even from paper to paper.

Do i include 0.068?

the most promising avenue might be emphasizing the importance of collaboration and transparency. By fostering a culture where researchers work together, openly share data and findings, and commit to thorough documentation, the reliability and replicability of work can be increased.

In the coming era, science must not be buoyed by computational power alone, but also by a culture of rigor, collaboration, and transparency. , the challenge lies not just in advancing science but in ensuring its very foundation remains robust, trustworthy, and verifiable.

3 Software

3.1 Seamless reproducible scientific computing via containerization. Scientific computing is very much like other computing, so it should come as no surprise that tools that have been developed to facilitate the reproduction of arbitrary computing tasks can be adapted to the ends of scientists. Containerization is a technology that allows for the creation of self-contained computing “containers” that can be run on any computer with the containerization software installed. While this does necessitate the installation of the containerization software, this is a one-time task that can be done by anyone with a computer. Once the containerization software is installed, the promise of containerization is that a container can be run on any computer, regardless of the operating system or other software installed on the computer. Many containerization standards exist; for present purposes this project uses the [Docker](#) containerization standard.

3.1.1 init.

3.1.2 develop.

3.1.3 build.

3.1.4 verify.

3.2 The reproduce.work workflow. The workflow put forward in this project is designed to be as simple as possible while accommodating a wide variety of scientific output.

1. Clone reproduce-work/template repository
2. Build and run the containerized interface; within the container:
 - Use specialized software, add data, and conduct analysis
 - reproduce.work software facilitates publishing of standardized computational report data with structured metadata
3. Verify compliance to open and reproducibility standards and publish project to the web

In the context of this version, “verify” will mean only to verify the computation repro-

duction of a given document. In future versions, we hope to expand this scope to include verification of the data collection and other analysis steps. On a more practical level, what this means is that for each piece of data or statistical output reported in a published document, we will verify two things:

1. That the data and code provided by the authors of the document produce the same output as the published document.
2. That each piece of data and statistical output reported in the published document is documented with structured metadata to trace the origin of the result and provide a link to the code that produced the result.

4 Future directions

In this section we mention some of the ambitious and yet under-developed ideas for future directions for this project.

4.1 Computational reproducibility with pre-registration.

4.1.1 *Pre-registered code with verifiable order-of-computation.*

pre-registered code \rightarrow timestamped data collection \rightarrow code execution \rightarrow verified results

4.1.2 *Adversarial collaborations.* A key requisite for successful adversarial collaborations is the ability to agree on code and data structure and standards. Being able to easily verify that results correspond to code that was written before but executed after the collection of data.

4.2 Computational reproducibility with specification curves or ML-driven analysis.

One may ask whether the framework here prevents questionable research practices including undesirable behavior like p -hacking or seed hacking. Unfortunately, the framework put forward here is not a panacea. The bar we set out to clear in this version of the project was to take steps toward a world where computational reproduction is possible. Developing soft-

ware specifically designed to discourage bad or encourage good aspects of research behavior is a task for future work. Nonetheless, it is valuable to discuss some ideas in which we may use our computing paradigm to encourage certain behaviors and practices of quantitative research.

One such method for exploration in future versions of this work is the potential for pre-registering exactly the model specification with which one wishes to estimate. If we were forced to pre-register our exact model specifications, we would be discouraged from pursuing models that depend on researcher degrees of freedom that can be exploited between the collection of data and reporting of results. As such, this may encourage scientists to explore the use of machine learning for analysis of heterogeneous treatment effects or specification curve analyses that are designed to be robust to specification searching.

5 Conclusion

The project put forward in the body of this project is a mere first step in the direction of computational reproduction. The software lacks support for many important features and the concept is heretofore untested. While there are many aspects to reproducibility in the epistemology of science, we believe that computational reproducibility is a necessary first step toward a world where scientific results are verifiable and trustworthy. However, given the low bars required to improve the existing practices around computational reproduction, we believe this project and the ideas put forward here have potential as the seed of a growing culture of scientific rigor and transparency.