Journal of eScience Librarianship

Volume 10
Issue 3 *Data Curation in Practice*

Article 7

2021-08-11

Computational Reproducibility: A Practical Framework for Data Curators

Sandra L. Sawchuk
Mount Saint Vincent University Library & Archives

Et al.

ISSN 2161-3974

JeSLIB 2021; 10(3): e1206
https://doi.org/10.7191/jeslib.2021.1206

Journal of eScience Librarianship
putting the pieces together: theory and practice

Full-Length Paper

Computational Reproducibility: A Practical Framework for Data Curators

Sandra L. Sawchuk[1] and Shahira Khair[2]

[1] Mount Saint Vincent University, Halifax, NS, Canada
[2] University of Victoria, Victoria, BC, Canada

Abstract

Introduction: This paper presents concrete and actionable steps to guide researchers, data curators, and data managers in improving their understanding and practice of computational reproducibility.

Objectives: Focusing on incremental progress rather than prescriptive rules, researchers and curators can build their knowledge and skills as the need arises. This paper presents a framework of incremental curation for reproducibility to support open science objectives.

Methods: A computational reproducibility framework developed for the Canadian Data Curation Forum serves as the model for this approach. This framework combines learning about reproducibility with recommended steps to improving reproducibility.

Conclusion: Computational reproducibility leads to more transparent and accurate research. The authors warn that fear of a crisis and focus on perfection should not prevent curation that may be 'good enough.'

Correspondence: Sandra Sawchuk: sandra.sawchuk@msvu.ca

Received: April 8, 2021 **Accepted**: June 4, 2021 **Published**: August 11, 2021

Disclosures: The authors report no conflict of interest.

Introduction

The practice of research data management (RDM) is becoming increasingly curatorial (Chassanoff et al. 2018; Peer and Wykstra 2016). Library professionals, who have traditionally played a large role in supporting researchers' data management efforts, are now providing an increasing amount of support for data curation (Steeves 2017). Data curation is the active management of research data as it is created, maintained, used or reused, and archived for long-term storage (Clary et al. 2020), and it ensures research data are FAIR, or Findable, Accessible, Interoperable, and Reusable (Johnston et al. 2017; Wilkinson et al. 2016). While repositories and data management plans help to make data findable and accessible, data sharing does not guarantee ease of use. Ensuring the interoperability and reusability of research data remains a challenge because most data are produced or analyzed computationally. Computational data are produced programmatically (Benureau and Rougier 2018), and while these data may have some value on their own, the ability to reproduce the dataset and study results depends on a number of varying factors, including well-annotated source code [or scripts for data processing and analysis], detailed methods and study parameters, specifications of the computational environment, and a list of required hardware, software and its dependencies, among other requirements (National Academies of Sciences 2019).

While computational reproducibility is not a new concept (see Claerbout n.d.), it is a complex one, especially for librarians and other data curators without extensive subject matter expertise in computing and software development. Additionally, many researchers are not formally trained as programmers, therefore they might be hesitant to share their code for fear that it is incomplete or incomprehensible (Barnes 2010). Yet, both in Canada and internationally, research funding is increasingly contingent on proof of responsible data stewardship. For instance, the policy for federal grant-funded research in Canada requires all digital research data, metadata and code be deposited in a recognized digital repository (Government of Canada n.d.).

Often, the work of librarians and data curators occurs downstream, after the research protocols and analyses are completed. It is also often the case that the data curator has limited contact with the research team, meaning questions about the data and the code remain unanswerable. The lack of access to the original researchers may be compounded by a lack of domain-specific knowledge on the part of the curator, as few of them have the skills or capacity to review and improve code (Kouper et al. 2017). This discrepancy can lead to the publication of datasets that omit key information, making published results non-reproducible. These considerations lead to the question: Are these datasets still valuable from a research re-use perspective? Can curated data and source code still be useful if the results they support are only partially reproducible?

This article proposes an approach of incremental curation according to abilities of the curator, arguing that partial reproducibility is better than nothing at all

(Broman n.d.). Librarians and data curators, through their acquired degrees and everyday work, are knowledgeable in documenting and organizing research materials. The addition of computational skills to these core competencies can be developed incrementally, allowing for the continued deposit of complex datasets. As Rasmussen says, “things are complex until we have solved how to deal with them; after that they are only complicated” (2018, 1).

Literature Review

In theory, reproducibility is simple in that experiments should be designed, executed, and documented in such a way that others can repeat them (Hatton and van Genuchten 2019). The more often a result can be independently verified, the more trust researchers and the wider community have in the results (Harvey and Oliver 2016; Johnston et al. 2017). There are multiple benefits to sharing reproducible data. The ability to reproduce research from existing code and data can: 1) prevent unnecessary data collection, which reduces harm, especially in marginalized communities (McCoach et al. 2020; Varcoe et al. 2009); 2) increase citation counts (Piwowar, Day, and Fridsma 2007); 3) improve confidence in the accuracy of study results (Krier and Strasser 2014; Wilson et al. 2014); and 4) rapidly advance knowledge (Freire, Bonnet, and Shasha 2012). In contrast, false or exaggerated research findings waste time and money (Ioannidis 2014), which hinders scientific progress and erodes trust in academic communities and beyond (Wilson et al. 2014).

Reproducibility in practice is more complex, as the term itself is used inconsistently (Steeves 2017; CURE Consortium 2017) and its meaning changes across the domains (Piccolo and Frampton 2016). In this paper, our focus is on computational reproducibility, and more specifically, the practices that enable software or code to be adequately curated for short-term reuse and long-term preservation.

Reproducibility differs from replicability (Stodden, Leisch, and Peng 2014). Where reproducibility involves using the same data and the existing code to generate the same results found in the original research, replicability involves generating the same results using existing code, but with new data (Benureau and Rougier 2018; Stodden, Leisch, and Peng 2014). Replicability is more difficult to achieve, as the new data must fit seamlessly into the existing structure of the project.

Computational reproducibility is possible when the source code or software from a project can be re-run to obtain the same results (Benureau and Rougier 2018). Computational assets may include input data, source code, software, and detailed information about the computing environment (Hinsen 2018; Piccolo and Frampton 2016; Wilson et al. 2014). These assets must be accompanied by a detailed description, preferably a narrative one, that details the research process step-by-step (Hinsen 2018).

Aspects of computational reproducibility

There is a spectrum of stages in which an experiment can be considered computationally reproducible (Peng 2011; Tatman, VanderPlas, and Dane 2018). On the low end of the spectrum, the algorithms and results are described, but only the finished paper is available. In the middle of the spectrum, the code and data are available alongside a fulsome description of the procedures required to reproduce the experiment. At the high end of the spectrum, the code, software, data, and the computing environment are all accessible, as well as a comprehensive narrative description of the process (Tatman, VanderPlas, and Dane 2018). Software and code are not used interchangeably here. Rather, software consists of the executable code that can manipulate or analyze the data with very little hands-on intervention (Singh, Bansal, and Jha 2015). In computing, the environment consists of the conditions under which the experiment was performed. Specifically, computational environments might include the operating system, software packages, the plugins or libraries used by the software, and their associated dependencies (Beaulieu-Jones and Greene 2017; Piccolo and Frampton 2016).

While it is helpful to have a written account of these details, it is quite cumbersome to recreate the environment based on information alone. Computing environments can be packaged into virtual containers for reuse (see Beaulieu-Jones and Greene 2017; Boettiger 2015; Dat Project 2018; Hale 2019; Steeves, Rampin, and Chirigati 2018), though this is predicated on the software being open enough for the environment to be executable.

Lengthy projects involving many stakeholders might be subjected to 'data friction' (Edwards et al. 2011). Data friction occurs at the places where data is moved. This situation happens across labs or research teams, and between formats or devices. Every time the data moves, there is a risk that it will be garbled, misinterpreted, or lost (Edwards et al. 2011). Over time, individual files can also deteriorate due to bit rot or be rendered unreadable because of format obsolescence (Johnston et al. 2017). Detailed documentation and active management throughout the lifespan of a project increase the chances that the data and code can be reused (Harvey and Oliver 2016).

Barriers to reproducibility for researchers

Reproducibility is impossible at any stage of the spectrum unless researchers willingly share their datasets. Reluctance to do so may stem from a fear of their research being scooped (Borgman 2012). Researchers may be less willing to share their code than their data, based on their assumption that a description of the code suffices (Boettiger 2015), or a concern that the code is too 'raw' or inelegant (Barnes 2010). Many researchers are not formally trained to build and maintain software (Wilson et al. 2014), they may not have the time or budget for documentation (Stodden 2010), they may not have enough knowledge of RDM to support the students and assistants working in their lab (Akmon et al. 2011), or they may not even view their research output as data (Borgman 2012).

The proliferation of general-purpose data repositories, while in itself is interpreted as a positive initiative, may lead to issues with the discovery of data shared for reuse. When datasets are shared, but not ‘FAIR’ (findable, accessible, interoperable, and reusable) (Wilkinson et al. 2016), they are essentially caught in an ‘information bottleneck,’ a closed network where only those closest to the research can understand and use the data to its full potential (Witt 2008). The most significant barrier to reproducibility for researchers, according to Borgman (2012), is the lack of demand for reusable datasets. In Canada, the federal government and the research community are actively responding to Borgman’s (2012) “Conundrum of Sharing Research Data” with interventions both at the level of policy and research infrastructure (Government of Canada 2016; Turp et al. 2020). The Federated Research Data Repository (FRDR) is one project that enhances the discoverability of research data in Canada by harvesting and standardizing metadata from individual repositories (Turp et al. 2020). This repository, in concert with the newly released RDM policy from the Tri-Agencies, can dramatically increase the demand for shared datasets (Turp et al. 2020). With increased demand comes the challenge of curation. Curators must be able to identify high quality and reusable datasets for deposit and, at the same time, work with researchers to implement best practices to enhance the reproducibility of new and existing research (Palmer et al. 2013).

Roles and issues for curators

While the existing literature on reproducibility is extensive, it has focused particularly on proposing solutions for researchers, rather than curators (Akmon 2017; Fear 2015; Goodman et al 2014; Gray et al. 2005; Hatton and van Genuchten 2019; Ioannidis 2014; King 2011; Kitzes et al. 2017; Macneil 2018; McCoach et al. 2020; Peng 2011; Sandve et al. 2013; Stodden 2010; Stodden 2012; Stodden 2013). Yet, the data curator’s role is not new—historians and archivists have been debating the value of saving raw data in the sciences for some time (Elliott 1974). Over the past 15 years, changes to funding agency and journal policies have led to an increased demand for practices and procedures related to data deposit and storage. For example, since 2008, researchers awarded funding from the Canadian Institute for Health Research (CIHR) have been required to deposit bioinformatics, atomic, and molecular coordinate data into recognized public databases (Government of Canada 2006). In 2012, the *American Journal of Political Science* (*AJPS*) mandated all empirical data underlying the analyses of published papers be made available online. The journal updated the policy in 2015 to include all of the documentation and code that would allow for the replication of the findings and conclusions reported in the article (Jacoby, Lafferty-Hess, and Christian 2017). Though the *AJPS* policy uses the term replication rather than reproduction, it appears as if they are testing the code using the data supplied from the authors, which suggests that their aim is reproducibility.

Jacoby, Lafferty-Hess, and Christian (2017) report that it takes approximately eight hours for one person to replicate the analyses and curate the materials for a manuscript published in the *AJPS*. They also estimate that the journal's replication and verification process add an average of 53 days to the publication workflow because, in many cases, authors need to verify and resubmit their materials (Jacoby, Lafferty-Hess, and Christian 2017).

This evidence suggests that the earlier a curator is involved with a research project, the easier their tasks will be. This process is known as 'active data curation', where curators are embedded with a research team to coordinate and implement best practices and prepare the data for reuse as it is being created (Akmon et al. 2017; Digital Curation Centre n.d.). While active data curation is ideal, it is not always feasible. It is as likely that a curator's involvement will begin after the project has been completed.

Curating data for reproducibility takes time, but it also requires curators to have subject-specific knowledge and computational experience. One of the key questions for curators is whether or not the curator should be responsible for ensuring the reproducibility of the project.

It is likely that library professionals, who have established expertise in the organization and description of complex collections, will be increasingly tasked with curatorial responsibilities, especially as funder policies develop and are implemented. In the next section, we describe the elements of computationally reproducible research. These elements have been distilled into an evaluation framework that can assist researchers as they collect their data, or curators as they prepare the data for deposit.

Elements of Computationally Reproducible Research

Notwithstanding the variation of research datasets within and between disciplinary fields, there exists a core set of characteristics minimally required to support computational reproducibility. Many publications and online sources have undertaken discussions of best practices for computational reproducibility (see Benureau and Rougier 2018; Broman n.d.; Sandve et al. 2013; White et al. 2013; Wilson et al. 2014), which can be distilled into a core set of reproducible elements. While it is recommended that reproducible methods be applied throughout the research workflow, there is an opportunity for these elements to be evaluated and implemented post-hoc by a researcher or data curator to improve the reproducibility of a final dataset. Together, these elements should allow someone who is unfamiliar with a dataset to examine it, understand protocols, and reproduce the workflow with minimal effort. While these elements of reproducibility are not necessarily mutually exclusive and exist as part of the spectrum discussed above, for the purposes of discussion and evaluation, it can be helpful to compartmentalize them.

Organization

Reproducibility begins with good file and directory organization. It is recommended that all files relevant to a single project are stored under a common root directory (Broman n.d.; Noble 2009). Within that directory, folders and files should use brief but clear descriptive language for naming (Borer et al. 2009). Noble (2009) provides a sample directory structure, with sub-directories for data, code, results, and documents, variations of which have been recommended (e.g., the TIER Protocol). Regardless of project structure suitability, sub-directory folders should clearly distinguish inputs from output files in the analysis. In particular, data, source code, and results should be clearly labelled, and raw data should be separated from processed data (Broman n.d.). An organized directory should also make use of version control practices as a means of documenting and tracking changes to project files in a systematic and transparent manner (Kitzes, Turek, and Deniz 2017). Although not strictly required for reproducibility, this disposition can make it easier for a reviewer to understand the history of changes, allow them to return to an earlier state to find any bugs, and guide them in making modifications without worrying about breaking code (Broman n.d.; Noble 2009). Finally, an organized file directory should always contain a readme file, ideally placed in the main directory, that explains the sub-directory structure, file contents, and processes (Broman n.d.).

Documentation

One of the main tenets in reproducibility is for research to be “independently understandable” (Peer and Wykstra 2016, 7); in other words, that a researcher need not be present for another researcher in the same field to rerun and understand their workflow and outputs. To achieve this ideal for computational reproducibility, to the extent possible, workflows should be scripted to avoid any manual intervention on the part of the reuser, as manual procedures are not only inefficient and error prone, they are also hard to troubleshoot and to reproduce (Sandve et al. 2013). Therefore, everything from data cleaning and file conversion to analyses should be scripted. To support seamless reusability, scripts should use relative paths (Broman n.d.; Noble 2009) and be stitched together into a workflow whose execution is automated by a master script (Kitzes, Turek, and Deniz 2017).

As Benureau and Rougier (2018) note, it is impossible to write future-proof code, as technology is evolving so quickly, it cannot be known how systems and software will change. The simpler solution is to make it possible for a reuser to recreate the original execution environment. For this to be possible, explicit documentation of both the execution environment used to run the software and all requirements and dependencies of systems, software, and libraries should be clearly noted in the code and/or readme file (Benureau and Rougier 2018). While time passing can make software obsolete, it also fades the memories of researchers. For that reason alone, it is important to annotate source code to explain the intended operations, which supports reproducibility for the original research team while also supporting future reusers (Benureau and Rougier 2018).

While source code reflects actions taken in a workflow, it does not explain reasoning behind those decisions. The latter should be commented in the code, or included as separate supporting documentation (Peer and Wykstra 2016).

Files and formats

A copy of the raw, unprocessed data should be shared alongside the code that describes any transformation or processes applied to prepare it for analysis. These steps will allow other researchers to assess the process by which values used for analysis were obtained (White et al. 2013). This is not always possible, for example, when using data collected from an organization or another researcher. In this case, sufficient supporting documentation describing processes should be given. If data was gathered from an online repository, a permanent link to the source should be provided (Broman n.d.). As noted above, to reproduce a given analysis, specific versions of programs may be required. It is not always a simple task to obtain older versions of software, as they may have restrictive licenses, or exist on hardware that is obsolete; when possible, archiving the exact versions of programs used should be considered (Sandve et al. 2013). File formats as well as software can become obsolete, therefore best practice is to save data files in a non-proprietary file format to promote future reusability (Borer et al. 2009; Peer and Wykstra 2016). Finally, like code, data files themselves should be described with enough documentation and metadata to enable reuse (Peer and Wykstra 2016; White et al. 2013).

Sharing and Licensing

To facilitate effective data sharing, datasets must be discoverable and accessible. They should therefore be deposited to a well-established repository that registers persistent identifiers. The latter should be referenced directly in any associated publications using the dataset (Benureau and Rougier 2018; White et al. 2013). It is important to specify how others are permitted to use, modify, and distribute your work. A number of established licenses exist for software and data that can be applied in academic and research environments to enable permissive reuse (Morin et al. 2012).

The Framework

The evaluation framework, shown in Table 1, was created in order to introduce these elements of computational reproducibility to a wide audience. The framework is structured as a series of questions that a researcher or curator would consider in reviewing a dataset’s reproducibility. In the framework, space is provided next to each question for the reviewer to record their findings. This framework was originally presented as part of a workshop for the Canadian Data Curation Forum; additional materials, including exercises, can be found in the conference repository (Khair, Sawchuk, and Zhang n.d.).

This framework is designed for use at any stage in the research lifecycle, though it would have the most impact at the beginning of a project, especially if accompanied by a data management plan. We anticipate that curators, even those that are completely new to the practice, will find value in this work, whether it is used for teaching, research, or active curation. This tool is not meant to be prescriptive; we encourage curators to do whatever they can to improve and promote computational reproducibility as they develop their skills and knowledge in this domain.

Though this framework specifically supports computational reproducibility, it can be used to accompany other tools that support curation on a broader scale. One prime example is the series of 'CURATED' checklists developed by the Data Curation Network (DCN) (n.d.) in the United States. These checklists form part of a larger workflow supporting and standardizing curation among partner institutions of the DCN. Each letter in 'CURATED' stands for a unique step in their curation process: Check, Understand, Request, Augment, Transform, Evaluate, and Document (Data Curation Network n.d.). The Check stage involves conducting an appraisal of the files submitted for curation, and the Understand stage includes tasks such as running files and code. Our computational reproducibility framework could augment or accompany these stages, especially for datasets that rely substantially on software and code.

Table 1: Reproducibility Framework

Organization	**Yes / No / Maybe? (explain if necessary)**
Are all files encapsulated within one directory?	
Is the sub-directory structure clear and easy to navigate? • Are the names of subdirectories self-explanatory? • Is the raw / input data separated from the derived data? • Is the data separated from the code? • Are any outputs (figures, tables) provided? Are they contained in their own subdirectory?	
Are file names self-explanatory, or described clearly in the documentation? If not, how could they be improved?	
Are there multiple versions of files? If yes, are versions clearly enumerated?	
Is there a README file? • If yes, does it specify author contact information, file contents, directory overview, dependencies, etc.? What other information could it provide to improve reproducibility?	

Table 1: Reproducibility Framework (continued)

Document Software	**Yes / No / Maybe?** **(explain if necessary)**
Is the software environment specified?	
Are dependencies needed to run scripts specified clearly, or have they been packaged and included?	
Are relative paths used in scripts (vs. absolute paths)?	
Are all file conversions, data cleaning, and analysis steps documented within scripts?	
Is the execution of all code automated by a master script?	
Are decisions behind data cleaning, analysis, and other scripts well documented within the code as annotations, or as a reproducible report (e.g. R markdown (*.Rmd))?	
Document Data	**Yes / No / Maybe?** **(explain if necessary)**
Are the raw data provided? If only processed data are provided, is there sufficient description to understand transformations made to raw data?	
Are all data files necessary to rerun analyses provided? If not, are links to containing repositories specified?	
Are data provided in open file formats?	
Is sufficient documentation provided to understand the data? (e.g. data dictionary, code book)	
Licensing and Sharing	**Yes / No / Maybe?** **(explain if necessary)**
Is a license specified for the software? (for e.g., either in a README file or a separate license text file?)	
Is a license specified for the data?	
Is the repository(ies) containing the data and code registered with a unique DOI?	
Are the repository(ies) and published article cross linked using metadata?	

Challenges

While reducing the concept of computational reproducibility to a set of key elements might be helpful from a pedagogical perspective, it may also have the unintended side effect of making this work appear simpler or easier than it really is. We recognize how implementing these elements into a research program is "easier said than done" for a host of reasons.

First, software stacks for even relatively simple workflows can exist upon a mountain of dependencies, which Boettiger (2015) terms "dependency hell." Unlike traditional scholarly outputs, software is executable, iterative, and interdependent (Chassanoff et al. 2018). As Huff (2017) notes, the first (and often last) obstacle for reuse, is often simply getting the workflow running on another machine.

Proprietary software and file formats can be problematic for curators because it is complex to determine how the data has been encoded (Rimkus et al. 2014). When key parts of the research workflow are locked up in a 'black box,' it is almost impossible to examine a project in its entirety (Morin et al. 2012), let alone reproduce it. Researchers use proprietary software for many reasons, including ease of use, security, affordability, availability, reliability, and disciplinary preference (Singh, Bansal, and Jha 2015). The use of proprietary formats is positive at times, for the widespread adoption of a particular type of software can indicate its potential for longevity (Rimkus et al. 2014).

Success often relies on effective documentation, which takes time that most researchers are lacking. Moreover, when present, effective documentation can still lack precision due to the fact that researchers often rely on ready-made software packages (Boettiger 2015). Even when effective documentation is provided, when separated from inputs and outputs, the components and moving parts included in a single dataset can be challenging to piece together (Chassanoff et al. 2018). For that reason, integrating a workflow into a Jupyter Notebook or R Markdown report can be very useful for reproducibility.

A significant obstacle to reproducibility not yet acknowledged concerns creators and reusers themselves. Researchers all bring different skills and experiences to their research, and as research becomes more collaborative and team-based, often the lowest common denominator tools are used, usually at the expense of reproducibility (Huff 2017).

Conclusion

While the ultimate end-goal may be computational reproducibility, neither researchers nor curators should let a lack of perfection stop them from taking incremental steps to increase the possibility for reuse. For researchers, the first and easiest task is to improve documentation, ideally beginning the project with a data management plan. While open-source software is always preferable, any software can be understood if it is documented in detail, including the computing environment and dependencies. Researchers should be encouraged to consult with a research data management specialist or librarian as early as possible and, granted budget allowances, include that person as a collaborator or co-author.

For curators, the old adage applies, the enemy of progress is perfection. Each curation project will pose its own unique challenges. However, the framework presented in this paper provides hope in sharing the foundational knowledge and encouragement needed to increase reproducibility in the curatorial process, regardless of experience or computational ability.

Supplemental Content

Reproducibility Framework

An online supplement to this article can be found at "Curating Data Sets for Reproducibility Workshop" at https://data-curation.github.io/cdcf-workshop2B.

References

Akmon, Dharma, Margaret Hedstrom, James D. Myers, Anna Ovchinnikova, and Inna Kouper. 2017. "Building Tools to Support Active Curation: Lessons Learned from SEAD." *International Journal of Digital Curation* 12(2): 76–85. https://doi.org/10.2218/ijdc.v12i2.552

Akmon, Dharma, Ann Zimmerman, Morgan Daniels, and Margaret Hedstrom. 2011. "The Application of Archival Concepts to a Data-Intensive Environment: Working with Scientists to Understand Data Management and Preservation Needs." *Archival Science* 11(3–4): 329–348. https://doi.org/10.1007/s10502-011-9151-4

Barnes, Nick. 2010. "Publish Your Computer Code: It Is Good Enough." *Nature* 467(7317): 753–753. https://doi.org/10/cj8t6n

Beaulieu-Jones, Brett K., and Casey S. Greene. 2017. "Reproducibility of Computational Workflows Is Automated Using Continuous Analysis." *Nature Biotechnology* 35(4): 342–346. https://doi.org/10/f9ttx6

Benureau, Fabien C.Y., and Nicolas P. Rougier. 2018. "Re-Run, Repeat, Reproduce, Reuse, Replicate: Transforming Code into Scientific Contributions." *Frontiers in Neuroinformatics* 11(January). https://doi.org/10/ggb79t

Boettiger, Carl. 2015. "An Introduction to Docker for Reproducible Research." *ACM SIGOPS Operating Systems Review* 49(1): 71–79. https://doi.org/10/gdz6f9

Borer, Elizabeth T., Eric W. Seabloom, Matthew B. Jones, and Mark Schildhauer. 2009. "Some Simple Guidelines for Effective Data Management." *The Bulletin of the Ecological Society of America* 90(2): 205–214. https://doi.org/10/b2sn4j

Borgman, Christine L. 2012. "The Conundrum of Sharing Research Data." *Journal of the American Society for Information Science and Technology* 63(6): 1059–1078. https://doi.org/10.1002/asi.22634

Broman, Karl. n.d. "Initial Steps toward Reproducible Research." Steps Towards Reproducible Research. Accessed December 3, 2019. https://kbroman.org/steps2rr

Chassanoff, Alexandra, Yasmin Al Noamany, Katherine Thornton, and John Borghi. 2018. “Software Curation in Research Libraries: Practice and Promise.” Journal *of Librarianship and Scholarly Communication* 6(1). https://doi.org/10.7710/2162-3309.2239

Claerbout, Jon. n.d. “Reproducible Computational Research: A History of Hurdles, Mostly Overcome.” Accessed February 11, 2021. http://sepwww.stanford.edu/sep/jon/reproducible.html

Clary, Erin, Jason Brodeur, Lee Wilson, Jeff Moon, and Shahira Khair. 2020. “Conceptualizing a National Approach to Data Curation Services in Canada.” Zenodo. https://doi.org/10.5281/zenodo.3894935

CURE Consortium. 2017. “Defining ‘Reproducibility’.” Published November 27, 2017. https://cure.web.unc.edu/defining-reproducibility

Dat Project. 2018. “Is Open Science Ready for Software Containers?” *Dat Project Blog*. Published January 26, 2018. https://blog.datproject.org/challenges-of-decentralized-hpc-containerization

Data Curation Network. n.d. “DCN Curation Workflow.” Accessed June 16, 2021. https://datacurationnetwork.org/outputs/workflows

Digital Curation Centre. n.d. “What Is Digital Curation?” Accessed August 6, 2021. https://www.dcc.ac.uk/about/digital-curation

Edwards, Paul N., Matthew S. Mayernik, Archer L. Batcheller, Geoffrey C. Bowker, and Christine L. Borgman. 2011. “Science Friction: Data, Metadata, and Collaboration.” *Social Studies of Science* 41(5): 667–690. https://doi.org/10.1177/0306312711413314

Elliott, Clark. 1974. “Experimental Data as a Source for the History of Science.” *The American Archivist* 37(1): 27–35. https://doi.org/10.17723/aarc.37.1.98681h774661j223

Fear, Kathleen. 2015. “Building Outreach on Assessment: Researcher Compliance with Journal Policies for Data Sharing.” *Bulletin of the Association for Information Science and Technology* 41(6): 18–21. https://doi.org/10.1002/bult.2015.1720410609

Freire, Juliana, Philippe Bonnet, and Dennis Shasha. 2012. “Computational Reproducibility: State-of-the-Art, Challenges, and Database Research Opportunities.” In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, 593–596. https://doi.org/10.1145/2213836.2213908

Goodman, Alyssa, Alberto Pepe, Alexander W. Blocker, Christine L. Borgman, Kyle Cranmer, Merce Crosas, Rosanne Di Stefano, Yolanda Gil, Paul Groth, and Margaret Hedstrom. 2014. “Ten Simple Rules for the Care and Feeding of Scientific Data.” *PLoS Comput Biol* 10(4): e1003542. https://doi.org/10/sjk

Government of Canada. 2016. “Tri-Agency Statement of Principles on Digital Data Management.” Published December 21, 2016. http://www.science.gc.ca/eic/site/063.nsf/eng/h_83F7624E.html

———. n.d. “Tri-Agency Research Data Management Policy.” Accessed March 17, 2021. http://www.science.gc.ca/eic/site/063.nsf/eng/h_97610.html

Government of Canada, Canadian Institutes of Health Research. 2006. “Tri-Agency Open Access Policy on Publications - CIHR.” Published August 15, 2006. https://cihr-irsc.gc.ca/e/32005.html

Gray, Jim, David T. Liu, Maria Nieto-Santisteban, Alex Szalay, David J. DeWitt, and Gerd Heber. 2005. “Scientific Data Management in the Coming Decade.” *Acm Sigmod Record* 34(4): 34–41. https://doi.org/10.1145/1107499.1107503

Hale, Jeff. 2019. "Learn Enough Docker to Be Useful." *Medium*. Published January 9, 2019. https://towardsdatascience.com/learn-enough-docker-to-be-useful-b7ba70caeb4b

Harvey, Douglas Ross, and Gillian Oliver. 2016. *Digital Curation*. ALA Neal-Schuman. https://doi.org/10.1080/19322909.2017.1338056

Hatton, Les, and Michiel van Genuchten. 2019. "Computational Reproducibility: The Elephant in the Room." *IEEE Software* 36(2): 137–144. https://doi.org/10/ggkvtr

Hinsen, Konrad. 2018. "Reusable Versus Re-Editable Code." *Computing in Science & Engineering* 20(3): 78–83. https://doi.org/10.1109/MCSE.2018.03202636

Huff, Kathryn D. 2017. "Lessons Learned." In *The Practice of Reproducible Research: Case Studies and Lessons from the Data-Intensive Sciences*, edited by Justin Kitzes, Daniel Turek, and Fatma Deniz, 42–59. University of California Press. https://doi.org/10.1525/9780520967779

Ioannidis, John P.A. 2014. "How to Make More Published Research True." *PLoS Medicine* 11(10). https://doi.org/10/gfc87k

Jacoby, William G., Sophia Lafferty-Hess, and Thu-Mai Christian. 2017. "Should Journals Be Responsible for Reproducibility?" *Inside Higher Ed*. Published July 17, 2017. https://www.insidehighered.com/blogs/rethinking-research/should-journals-be-responsible-reproducibility

Johnston, Lisa R., Jake Carlson, Cynthia Hudson-Vitale, Heidi Imker, Wendy Kozlowski, Robert Olendorf, and Claire Stewart. 2017. "Data Curation Network: A Cross-Institutional Staffing Model for Curating Research Data." University of Minnesota Digital Conservancy. https://hdl.handle.net/11299/188654

Khair, Shahira, Sandra Sawchuk, and Qian Zhang. n.d. "Curating Data Sets for Reproducibility." Reproducible Research. Accessed March 19, 2021. https://data-curation.github.io/cdcf-workshop2B

King, Gary. 2011. "Ensuring the Data-Rich Future of the Social Sciences." *Science* 331(6018): 719–721. https://doi.org/10.1126/science.1197872

Kitzes, Justin, Daniel Turek, and Fatma Deniz. 2017. *The Practice of Reproducible Research: Case Studies and Lessons from the Data-Intensive Sciences*. University of California Press. https://www.practicereproducibleresearch.org

Kouper, Inna, Kathleen Fear, Mayu Ishida, Christine Kollen, and Sarah Christine Williams. 2017. "Research Data Services Maturity in Academic Libraries." In *Curating Research Data: Practical Strategies for Your Digital Repository*, 1: 153–170. Association of College and Research Libraries. https://doi.org/10.14288/1.0343479

Krier, Laura, and Carly A. Strasser. 2014. *Data Management for Libraries: A LITA Guide*. American Library Association.

Macneil, Rory. 2018. "Electronic Notebooks as Data Curation Tools 2: Optimizing the ELN-to-Repository Workflow." *ResearchSpace (blog)*. Published March 15, 2018. https://www.researchspace.com/electronic-notebooks-as-data-curation-tools-2-optimizing-the-eln-to-repository-workflow

McCoach, D. Betsy, Jennifer N Dineen, Sandra M Chafouleas, and Amy Briesch. 2020. "Reproducibility in the Era of Big Data: Lessons for Developing Robust Data Management and Data Analysis Procedures." In *Big Data Meets Survey Science: A Collection of Innovative Methods*, 625–655. Wiley. https://doi.org/10.1002/9781118976357

Morin, Andrew, Jennifer Urban, Paul D. Adams, Ian Foster, Andrej Sali, David Baker, and Piotr Sliz. 2012. "Shining Light into Black Boxes." *Science* 336(6078): 159–160. https://doi.org/10/m5t

National Academies of Sciences, Engineering, and Medicine. 2019. *Reproducibility and Replicability in Science*. National Academies Press. https://doi.org/10.17226/25303

Noble, William Stafford. 2009. "A Quick Guide to Organizing Computational Biology Projects." *PLoS Comput Biol* 5(7): e1000424. https://doi.org/10/fbbpkn

Palmer, Carole L., Nicholas M. Weber, Trevor Muñoz, and Allen H. Renear. 2013. "Foundations of Data Curation: The Pedagogy and Practice of 'Purposeful Work' with Research Data." *Archive Journal (blog)*. June 2013. http://dev.archivejournal.net/?p=4819

Peer, Limor, and Stephanie Wykstra. 2016. "New Curation Software: Step-by-Step Preparation of Social Science Data and Code for Publication and Preservation." *IASSIST Quarterly* 39(4): 6. https://doi.org/10/ggkvtp

Peng, Roger D. 2011. "Reproducible Research in Computational Science." *Science* 334(6060): 1226–1227. https://doi.org/10/fdv356

Piccolo, Stephen R., and Michael B. Frampton. 2016. "Tools and Techniques for Computational Reproducibility." *Gigascience* 5(1): 30–30. https://doi.org/10/gfs3cq

Piwowar, Heather A, Roger S Day, and Douglas B Fridsma. 2007. "Sharing Detailed Research Data Is Associated with Increased Citation Rate." *PloS One* 2(3): e308. https://doi.org/10/apv

Project TIER. n.d. "TIER Protocol 3.0." Accessed June 16, 2021. https://www.projecttier.org/tier-protocol/specifications-3-0

Rasmussen, Karsten Boye. 2018. "Failure as the Treatment for Transforming Complexity to Complicatedness." *IASSIST Quarterly* 42(4): 1–2. https://doi.org/10.29173/iq949

Rimkus, Kyle, Thomas Padilla, Tracy Popp, and Greer Martin. 2014. "Digital Preservation File Format Policies of ARL Member Libraries: An Analysis." *D-Lib Magazine* 20(3/4). https://doi.org/10.1045/march2014-rimkus.

Sandve, Geir Kjetil, Anton Nekrutenko, James Taylor, and Eivind Hovig. 2013. "Ten Simple Rules for Reproducible Computational Research." *PLoS Comput Biol* 9(10): e1003285. https://doi.org/10/pjb

Singh, Amandeep, R.K. Bansal, and Neetu Jha. 2015. "Open Source Software vs Proprietary Software." *International Journal of Computer Applications* 114(18). https://doi.org/10/gh4jxn

Steeves, Vicky. 2017. "Reproducibility Librarianship." *Collaborative Librarianship* 9(2): 4. https://digitalcommons.du.edu/collaborativelibrarianship/vol9/iss2/4

Steeves, Vicky, Rémi Rampin, and Fernando Chirigati. 2018. "Using ReproZip for Reproducibility and Library Services." *IASSIST Quarterly* 42(1): 14–14. https://doi.org/10/gf9hw5

Stodden, Victoria. 2010. "The Scientific Method in Practice: Reproducibility in the Computational Sciences." *SSRN Electronic Journal*. https://doi.org/10/fzmph2

Stodden, Victoria. 2012. "Reproducible Research for Scientific Computing: Tools and Strategies for Changing the Culture." *Computing in Science & Engineering* 14(4): 13–17. https://doi.org/10.1109/MCSE.2012.38

Stodden, Victoria, Jonathan Borwein, and David H. Bailey. 2013. "'Setting the Default to Reproducible' in Computational Science Research." *SIAM News* 46(5): 4–6. http://stodden.net/icerm_report.pdf

Stodden, Victoria, Friedrich Leisch, and Roger D Peng. 2014. *Implementing Reproducible Research*. CRC Press. https://doi.org/10.1201/b16868

Tatman, Rachael, Jake VanderPlas, and Sohier Dane. 2018. “A Practical Taxonomy of Reproducibility for Machine Learning Research.” June. https://openreview.net/forum?id=B1eYYK5QgX

Turp, Clara, Lee Wilson, Julienne Pascoe, and Alex Garnett. 2020. “The Fast and the FRDR: Improving Metadata for Data Discovery in Canada.” *Publications* 8(2): 25. https://doi.org/10/gh4tbp

Varcoe, Colleen, Annette J. Browne, Sabrina Wong, and Victoria L. Smye. 2009. “Harms and Benefits: Collecting Ethnicity Data in a Clinical Context.” *Social Science & Medicine* 68(9): 1659–1666. https://doi.org/10/cd4nhd

White, Ethan P., Elita Baldridge, Zachary T. Brym, Kenneth J. Locey, Daniel J. McGlinn, and Sarah R. Supp. 2013. “Nine Simple Ways to Make It Easier to (Re) Use Your Data.” *Ideas in Ecology and Evolution* 6(2). https://doi.org/10/gfj86j

Wilkinson, Mark D., Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, and Philip E. Bourne. 2016. “The FAIR Guiding Principles for Scientific Data Management and Stewardship.” *Scientific Data* 3. https://doi.org/10.1038/sdata.2016.18

Wilson, Greg, D.A. Aruliah, C. Titus Brown, Neil P. Chue Hong, Matt Davis, Richard T. Guy, Steven H. D. Haddock, et al. 2014. “Best Practices for Scientific Computing.” *PLOS Biology* 12(1): e1001745. https://doi.org/10/qtt

Witt, Michael. 2008. “Institutional Repositories and Research Data Curation in a Distributed Environment.” *Library Trends* 57(2): 191–201. https://doi.org/10.1353/lib.0.0029