Published October 12, 2015

Opinion and Policy

Core Ideas

- A significant portion of present-day geoscience research is computational.
- Science would benefit from greater transparency in computational research.
- *Vadose Zone Journal* is launching a Reproducible Research program.
- Code and data underlying a research article will be published alongside articles.

T.H. Skaggs, U.S. Salinity Laboratory, 450 W. Big Springs Rd., Riverside, CA 92507, USA. M.H. Young, Bureau of Economic Geology, Jackson School of Geosciences, University of Texas at Austin, Austin, TX, USA. J.A. Vrugt, Dep. of Civil and Environmental Engineering, University of California, Irvine, CA, USA. *Corresponding author (todd.skaggs@ars.usda.gov).

Vadose Zone J.
doi:10.2136/vzj2015.06.0088
Received 12 June 2015.
Accepted 15 Aug. 2015.
Open access article

Reproducible Research in Vadose Zone Sciences

T.H. Skaggs,* M.H. Young, and J.A. Vrugt

A significant portion of present-day soil and Earth science research is computational, involving complex data analysis pipelines, advanced mathematical and statistical models, and sophisticated computer codes. Opportunities for scientific progress are greatly diminished if reproducing and building on published research is difficult or impossible due to the complexity of these computational systems. *Vadose Zone Journal* (VZJ) is launching a Reproducible Research (RR) program in which code and data underlying a research article will be published alongside the article, thereby enabling readers to analyze data in a manner similar to that presented in the article and build on results in future research and applications. In this article, we discuss reproducible research, its background and use across other disciplines, its value to the scientific community, and its implementation in VZJ.

Abbreviations: NIH, National Institutes of Health; RR, Reproducible Research; VZJ, *Vadose Zone Journal*.

A hallmark of the scientific method is that research results must be reproducible. Although the reproducibility requirement has always existed, technological advances over the last few decades have changed the way science is practiced and communicated, creating for researchers and publishers new opportunities and challenges with respect to openness and reproducibility.

One set of opportunities involves increased reuse of experimental data. The internet and related information technologies have allowed greater archiving and sharing of environmental and geoscience data. Data sharing makes the validation of scientific findings possible, lessens the need for wasteful duplication of research efforts, and facilitates new data synthesis and aggregation activities. A number of environmental and geoscience publishers have promoted data sharing through the introduction of "dataset" articles and journals that focus on digital data archives (e.g., Hornberger, 1994; Pfeiffenberger and Carlson, 2011; Nature Publishing Group, 2014; Gregorich, 2015). Novel data sharing opportunities also arise from long-term observational networks such as LTER (http://www.lternet.edu, accessed 4 Sept. 2015), NEON (http://www.neoninc.org, accessed 4 Sept. 2015), FLUXNET (http://fluxnet.ornl.gov, accessed 4 Sept. 2015), LTAR (http://www.ars.usda.gov/ltar, accessed 4 Sept. 2015), CZO (http://criticalzone.org/national, accessed 4 Sept. 2015), TERENO (http://teodoor.icg.kfa-juelich.de/overview-en, accessed 4 Sept. 2015), and various monitored watersheds (e.g., Reynolds Creek Experimental Watershed, Idaho [Marks, 2001]). These networks are creating new possibilities for evaluating agroecosystem data, including assessments of reproducibility across geographical locations and time.

Yet, beyond these considerations of experimental data and field observations, we recognize that modern computing technologies have created entirely new dimensions to the issue of research reproducibility (Yale Law School Round Table on Data and Code Sharing, 2010; Peng, 2011; Stodden et al., 2014). A significant portion of present-day scientific research is computational, involving complex data analysis pipelines, elaborate mathematical and statistical models, and sophisticated computer codes or scripts. In many published papers, the computational methods are integral to the research results being reported,

but they are often not fully explained or described, partly due to constraints imposed by the format of the traditional research paper, and possibly also due to the reluctance of authors to share intellectual property. As a consequence, research results are difficult to reproduce, either for purposes of verifying their correctness, or for building on results in future research and applications.

Reproducible research (RR) has in recent years become a prominent issue in several academic fields, including biomedical research, where the irreproducibility of a great number of pharmacology and cancer research studies has been well publicized (Prinz et al., 2011; Begley and Ellis, 2012), and in fields featuring lengthy computations with complex algorithms (e.g., bioinformatics, data science). The journals *Science* and *Nature* have both published numerous articles about reproducibility and editorialized in favor of funding and academic publication policies that promote open science and reproducibility (see, for example, the archive at http://www.nature.com/nature/focus/reproducibility, accessed 4 Sept. 2015). In 2014, the National Institutes of Health (NIH) released a set of reproducible research guidelines and principles (http://www.nih.gov/about/reporting-preclinical-research.htm, accessed 4 Sept. 2015) which have been endorsed by about 70 journals and societies. The guidelines are being widely adopted. NIH-funded researchers in the future will be required to publish only in journals adhering to these guidelines. Of course, we cannot predict how funding agencies in the United States, the European Union, and elsewhere might expand these guidelines for use in Earth sciences research, but we doubt that notions of openness and transparency are fleeting.

For decades, computational research has been important to vadose zone science and engineering, and has become central to many research programs. Computational models are used not only for prediction purposes, but also for basic understanding and explanation. Models are needed to fill knowledge gaps in vadose zone states, fluxes, and parameters, many of which are not directly observable. Even where geoscience data exist (such as in the archive and dataset publications noted above), those "observational" data are, in fact, often heavily processed products, obtained only after a series of transformations have been applied to raw measurements (Easterbrook, 2014). For many sensors used in vadose zone investigations, particularly remote sensors, it may be hard to separate computational models from observational data (Easterbrook, 2014).

The complexity of computational vadose zone research is only expected to further increase with continued advances in numerical models (spatial + temporal resolution), availability of new direct and indirect measurement technologies, use of increasingly larger data sets (big data), and utilization of distributed computing resources (among others). As computational procedures become more complex, opportunities for scientific progress are greatly diminished if reproducing and building on published results is difficult or impossible. The advancement of vadose zone science will be greatly facilitated by ensuring that published research is reproducible.

If one accepts the goal of creating reproducible computational research for vadose zone science and engineering, then what is to be done? What is required? Stodden et al. (2014) summarize some general requirements for reproducible computational science:

> In computational science, reproducibility requires that one make code and data available to others so that they may analyze the original data in a similar manner as in the original publication. This task requires that the analysis be done in such a way that preserves the code and data, and permits their distribution in a format that is generally readable, and a platform be available to the author on which the data and code can be distributed widely. Both data and code need to be licensed permissively enough so that others can reproduce the work without a substantial legal burden.

While consensus is growing around general principles such as these (see also Box 1), the literature on RR suggests that, in practice, implementing reproducible computational science poses challenges (Easterbrook, 2014; Stodden et al., 2014). Some researchers have expressed reluctance to disclose code and data for various reasons (Box 2), including software licensing (Box 3). Questions exist about appropriate goals, definitions, and objectives (Box 4). New tools (Box 5) and platforms (Box 6) are needed to better leverage

Box 1: Requirements for open computational science.

One concise vision for the requirements of reproducible research is given by the five C's of Nick Barnes's Science Code Manifesto:

Code: All source code written specifically to process data for a published paper must be available to the reviewers and readers of the paper.

Copyright: The copyright ownership and license of any released source code must be clearly stated.

Citation: Researchers who use or adapt science source code in their research must credit the code's creators in resulting publications.

Credit: Software contributions must be included in systems of scientific assessment, credit, and recognition.

Curation: Source code must remain available, linked to related materials, for the useful lifetime of the publication.

(source: http://sciencecodemanifesto.org, accessed 4 Sept. 2015)

Box 2: Reasons researchers may be reluctant or unwilling to disclose code and data.

No direct benefits or incentives (citations, promotion)

Too much effort required to "clean" code and data

Loss of competitive advantage over other researchers

Possibility of having to answer questions from users

Intellectual property issues

Research uses proprietary code or data

(Borgman, 2007; Stodden, 2010; Hurlin et al., 2014)

previous research efforts, and, most importantly, a community of researchers needs to be developed who embrace and follow best practices and guidelines for open, reproducible computational research (Millman and Perez, 2014).

In response to the growing need for computational reproducibility, many publications are now requiring that authors pledge to make code and data available. For example, *Nature* "now mandates that when code is central to reaching a paper's conclusions, we require a statement describing whether that code is available and setting out any restrictions on accessibility. Editors will insist on availability where they consider it appropriate: any practical issues preventing code sharing will be evaluated by the editors, who reserve the right to decline a paper if important code is unavailable" (Nature Editors, 2014). We anticipate other journals will follow suit.

Reproducible Research in *Vadose Zone Journal*

While such policies are a step in the right direction, relying on authors to maintain code and data archives is a potential drawback. A more interesting approach, and one that is consistent with VZJ as an online journal, is to publish the data and code in an electronic format as part of, or as a supplement to, a published article.

Box 3: Open Source Licenses, Intellectual Property.

Intellectual property issues surrounding supplemental data and computer code have been cited as one impediment to developing open research practices (Hurlin et al., 2014; Borgman, 2007).

A number of open-source software licenses are in wide use. Open source licenses differ with respect to what is permitted in terms of using, modifying, or redistributing software. Open source licenses can be categorized as either "copyleft" or "permissive." Copyleft licenses allow users to modify and redistribute code, but they require that any derivative works also have a copyleft license. Non-copyleft licenses are termed "permissive" and allow derivative works to have other licenses, including non-open source (proprietary) licenses. Commercial use is allowed by (most) copyleft and permissive licenses, so long as the terms of the license are followed. The most common copyleft license is the **GNU General Public License**, whereas common permissive licenses include the **MIT**, **BSD**, and **Apache** licenses (see http://opensource.org/licenses for details).

Creative Commons Licenses are a set of licenses spanning the full range of permissiveness with respect to copying, reuse, modification, and redistribution. Although creative commons licenses can be used for source code, they are most often used for other kinds of creative digital works. The "Attribution" creative commons license (aka "**CC-BY**") has been suggested as appropriate for code and data associated with open research publications (Stodden, 2014). The CC-BY license lets others distribute, modify, and build on a work, including commercial development, as long as they credit the original work. The **CC-BY-NC** license is similar except that only non-commercial uses are permitted (see: http://creativecommons.org).

"**Public domain**" is a legal term and not a software license. "Public domain" means that a code or work is not under copyright.

Box 4: Notes on terminology and goals.

A distinction can be made between *replication* and *reproduction* (Drummond, 2009). In general, *replication* refers to obtaining the same result from the same experiment under the same conditions, whereas *reproduction* refers to obtaining the same result from a different experiment (Gomez et al., 2010). However, this terminology is not applied universally (Easterbrook, 2014; Leek and Peng, 2015). Some authors use the terms interchangeably and do not distinguish between the two concepts, whereas others observe the conceptual distinction but apply different definitions.

In computational science, "reproducible research" has come to refer to the idea that code and data from scientific publications are available so that others can similarly analyze the data, reproduce key results from the original paper, and use the code and data in developing future research (Stodden et al., 2014).

But what does it mean to "reproduce key results"? Does that mean exact reproduction down the bit-level? Or does it mean something less stringent; say, reproducing numbers within some error bound? Different scientific endeavors may require different definitions (Murray-Rust and Murray-Rust, 2014). Going forward, the vadose zone scientific community will need to identify the standards and practices for reproducibility that best serve to advance the science.

Box 5: Programming languages and systems.

The basic tool for computational research is computer code, and compared with a few decades ago, scientific programmers have a variety of languages and environments to choose from. Statically typed, complied languages such **Fortran** and **C/C++** still usually provide the highest performance, and remain dominant in many problem domains. But nowadays much research is also done using higher-level, dynamically typed languages/systems that may have longer execution times but generally require less development time due to the availability of a large number of built-in, well-tested, high-level functions and libraries (which often exist "under the hood" as compiled Fortran or C). Well-known proprietary commercial packages such as **MATLAB** and **Mathematica** are examples, but open source alternatives are available, including **Python** (www.python.org, accessed 4 Sept. 2015), a general purpose language for which a large collection of scientific libraries exist (www.scipy.org, accessed 4 Sept. 2015); **R** (www.r-project.org, accessed 4 Sept. 2015), a widely used computing language and environment for statistical computing; **Julia** (julialang.org, accessed 4 Sept. 2015), a relatively new language designed specifically for technical computing; and **GNU Octave** (gnu.org/software/octave, accessed 4 Sept. 2015), a high-level interpreted language that is similar to MATLAB and primarily intended for numerical computations. Although any computing system can be used to develop reproducible research, the use of higher-level, reusable code and systems has been recommended as leading to more reliable scientific software (Wilson et al., 2014).

Box 6: Platforms for sharing code.

Several possibilities exist for packaging and sharing code so that it can studied and executed by other researchers. One format gaining in popularity is the notebook-style development environment. Notebooks permit the mixing of executable computer code, text, equations, and graphics into a single document. A popular open source implementation is the web browser-based Jupyter notebook (http://jupyter.org, accessed 4 Sept. 2015), which evolved from the IPython notebook (Pérez and Granger, 2007). The notebook document structure utilizes input and output cells, similarly to the well-known Mathematica notebook. The Jupyter notebook can be used with several programming languages including Python, Julia, R, Haskell, and Ruby. *Nature* (Shen, 2014) recently promoted the Jupyter notebook as a possible platform for reproducible research. Another open notebook environment is Beaker (http://beakernotebook.com, accessed 4 Sept. 2015), which similarly supports multiple languages.

As an example, a Jupyter notebook reproducing results and figures from Skaggs et al. (2014) can be viewed at: https://github.com/thskaggs/2014-steady-state/blob/master/steadystate.ipynb (accessed 4 Sept. 2015).

Vadose Zone Journal is launching a Reproducible Research pilot program in which code and data underlying a research article will be published alongside the article, thereby enabling readers to analyze data in a manner similar to that presented in the article and to build on results in future research and applications. *Vadose Zone Journal* is hence establishing a new category of article termed *Reproducible Research*. Papers with the RR designation will consist of two parts: (i) a standard VZJ article and (ii) a supplemental file archive containing code, data, and metadata, the latter including information on the content and licensing of the RR file archive, as well as requirements and instructions for using code and data. Any type of VZJ article (e.g., original research papers, technical notes, reviews and analyses, comments, letters to the editor, priority communications) featuring data and/or computational analyses can be published as an RR article. The only requirement is that code and data underlying the article be included as part of the publication. Analyses and data do not have to be complicated, although they can be. An RR file archive might be as simple as a single spreadsheet with some cell formulas performing data transformations, or it might be significantly more complicated, such as complex computer codes performing model simulations or analyzing large data sets hosted on remote repositories. Although there is a preference that source codes be included, there are situations where that will not be possible due to licensing or other intellectual property considerations. In such cases, executable code may be acceptable, or if third party software packages (libraries, models, environments) are needed, then input files and/or scripts for the software can be given along with a specification (version number, etc.) of the required packages. Note that VZJ is not a software journal in the manner of, say, *Environmental Modeling & Software*. Rather, the focus of both VZJ and the RR program is original interdisciplinary research.

Papers submitted to VZJ with the RR designation are reviewed under the same terms and conditions as all submitted papers. Authors selecting RR during the submission process can upload the RR file archive along with the draft manuscript. Alternatively, authors can indicate at the time of submission an interest in participating in the RR program, and then work with a designated Associated Editor to develop an RR supplemental file to be uploaded at a later time. In this case, the article peer review will proceed separately and without delay, independent of RR archive preparation. In a sense, the treatment of RR material is similar to "Supplemental Information," which is now increasingly being used by authors to complement and better contextualize the main manuscript. The goal is to permit authors who are unaware or unfamiliar with the RR program to participate without any delay in manuscript review and publication. All RR file archives will be reviewed to verify that the archive is readable and understandable, and that code is technically sound and can be used to reproduce key results from the paper.

File archives smaller than 10 MB will be hosted for free on ACSESS servers alongside the published article. Archives between 10 and 100 MB can be hosted on the journal website with a one-time charge (See https://dl.sciencesocieties.org/files/publications/vzj/rr-archive-pricing-schedule.pdf for current pricing details). Anything above 100 MB will have to be stored on an external, publically accessible repository (institutional repositories or another acceptable repository such as Dryad). In the case of an external repository, the journal will host (for free) a small archive containing metadata and information about the file archive. Authors will be asked to ensure that externally hosted files remain accessible for a period of at least 5 years. For additional details on the RR pilot program, see https://dl.sciencesocieties.org/publications/vzj/author-instructions-reproducible-research .

We will work toward a RR program for VZJ that can aid in developing a community of researchers who embrace and follow best practices for open, reproducible, computational research. The advancement of vadose science and engineering requires that we build on published results. Individual authors should benefit too, inasmuch as making their research more accessible should lead to earlier and more frequent citations of their work.

References

Begley, C.G., and L.M. Ellis. 2012. Drug development: Raise standards for preclinical cancer research. Nature 483(7391):531–533. doi:10.1038/483531a

Borgman, C.L. 2007. Scholarship in the Digital Age: Information, Infrastructure, and the Internet. MIT Press, Cambridge, MA.

Drummond, C. 2009. Replicability is not reproducibility: Nor is it good science. Proceedings of the Evaluation Methods for Machine Learning Workshop 26th International Conference for Machine Learning, Montreal, Quebec, Canada.

Easterbrook, S.M. 2014. Open code for open science? Nat. Geosci. 7:779–781. doi:10.1038/ngeo2283

Gomez, O.S., N. Juristo, and S. Vegas. 2010. Replication, Reproduction and Re-analysis: Three ways for verifying experimental findings. Proceedings of the 1st international workshop on replication in empirical software engineering research (RESER 2010), Cape Town, South Africa.

Gregorich, E. 2015. Dataset papers. J. Environ. Qual. 44:1. doi:10.2134/jeq2014.01.0001ed

Hornberger, G.M. 1994. Data and analysis note: A new type of article for Water Resources Research. Water Resour. Res. 30:3241–3242. doi:10.1029/94WR01878

Hurlin, C., C. Perignon, and V. Stodden. 2014. RunMyCode.org: A research-reproducibility tool for computational sciences. In: V. Stodden, F. Leisch, and R.D. Peng, editors, Implementing reproducible research. CRC Press, Boca Raton, FL. p. 367–381.

Leek, J.T., and R.D. Peng. 2015. Opinion: Reproducible research can still be wrong: Adopting a prevention approach. Proc. Natl. Acad. Sci. USA 112:1645–1646. doi:10.1073/pnas.1421412111

Marks, D. 2001. Introduction to special section: Reynolds Creek Experimental Watershed. Water Resour. Res. 37:2817. doi:10.1029/2001WR000941

Millman, K.J., and F. Perez. 2014. Developing open-source scientific practice. In: V. Stodden, F. Leisch, and R.D. Peng, editors, Implementing reproducible research. CRC Press, Boca Raton, FL. p. 149–183.

Murray-Rust, P., and D. Murray-Rust. 2014. Reproducible physical science and the declaration. In: V. Stodden, F. Leisch, and R.D. Peng, editors, Implementing reproducible research. CRC Press, Boca Raton, FL. p. 113–145.

Nature Editors. 2014. Code share. Nature 514:536. doi:10.1038/514536a

Nature Publishing Group. 2014. More bang for your byte. Scientific Data 1. doi:10.1038/sdata.2014.10

Peng, R.D. 2011. Reproducible research in computational science. Science 334:1226–1227 10.1126/science.1213847. doi:10.1126/science.1213847

Pérez, F., and B.E. Granger. 2007. IPython: A system for interactive scientific computing. Comput. Sci. Eng. 9(3):21–29. doi:10.1109/MCSE.2007.53

Pfeiffenberger, H., and D. Carlson. 2011. "Earth System Science Data" (ESSD)— A peer reviewed journal for publication of data. D-Lib Magazine 17(1/2). doi:10.1045/january2011-pfeiffenberger.

Prinz, F., T. Schlange, and K. Asadullah. 2011. Believe it or not: How much can we rely on published data on potential drug targets? Nat. Rev. Drug Discov. 10(9):712. doi:10.1038/nrd3439-c1

Shen, H. 2014. Interactive notebooks: Sharing the code. Nature 515:151–152. doi:10.1038/515151a

Skaggs, T.H., R.G. Anderson, D.L. Corwin, and D.L. Suarez. 2014. Analytical steady-state solutions for water-limited cropping systems using saline irrigation water. Water Resour. Res. 50:9656–9674. doi:10.1002/2014WR016058

Stodden, V. 2010. The scientific method in practice: Reproducibility in the computational sciences. MIT Sloan School Working Paper 4773-10. Available at http://papers.ssrn.com/abstract=1550193 (accessed 4 Sept. 2015). MIT, Cambridge, MA.

Stodden, V. 2014. What computational scientists need to know about intellectual property law: A primer. In: V. Stodden, F. Leisch, and R.D. Peng, editors, Implementing reproducible research. CRC Press, Boca Raton, FL. p. 325–339.

Stodden, V., F. Leisch, and R.D. Peng, editors. 2014. Implementing reproducible research. CRC Press, Boca Raton, FL.

Wilson, G., D.A. Aruliah, C.T. Brown, N.P. Chue Hong, M. Davis, R.T. Guy, S.H.D. Haddock, K.D. Huff, I.M. Mitchell, M.D. Plumbley, B. Waugh, E.P. White, and P. Wilson. 2014. Best practices for scientific computing. PLoS Biol. 12(1):E1001745. doi:10.1371/journal.pbio.1001745

Yale Law School Round Table on Data and Code Sharing. 2010. Reproducible research. Comput. Sci. Eng. 12:8–13 doi:10.1109/MCSE.2010.113.