Reproducibility Workshop Report

University of Washington

May 8, 2014

Table of Contents

Overview

The first Reproducibility Workshop sponsored by the eScience Institute and the Moore/Sloan Data Science Environment was held at the University of Washington on May 8, 2014. The event attracted a large crowd, with more than 90 participants pre-registered.

A wide variety of departments and colleges on campus were represented by faculty, students, and staff, including for example Aquatic and Fishery Sciences, Applied Mathematics, Archaeology, Bioengineering, Computer Science and Engineering, Human Centered Design and Engineering, Institute for Health Metrics and Evaluation, Linguistics, Oceanography, Mathematics, Pathobiology, and Statistics.

There were also many participants from off campus, including from Sage Bionetworks, Boeing, Fred Hutchinson Cancer Research Center, Institute for Systems Biology, Mozilla Science Lab, and Pacific Northwest National Laboratory.

Members of the DSE Working Group on Reproducibility and Open Science from our partner institutions also participated: Julian Freire from NYU and Fernando Perez and Philip Stark from Berkeley, along with Chris Mentzel from the Moore Foundation.

The detailed schedule of the day with links to slides used by speakers and panelists appears in Appendix A and on the workshop webpage.The morning program was aimed at a broad audience, with general discussion of issues surrounding reproducibility and some of the currently-available software tools and technologies for tackling this problem.

The day started with a lead-off talk by Stephen Friend, President of Sage Bionetworks, located at the Fred Hutchinson Cancer Research Center in Seattle. Dr. Friend gave several motivating examples of the ways in which encouraging open collaboration, data sharing, and code reuse can lead to better science and new discoveries in cancer research. He also discussed difficulties with the current incentive and reward structure in biomedical research (and academics more generally), and some ways in which collaboration and sharing can be incentivized.

This talk was followed by two panel discussions -- the first on some of the issues with undertaking reproducible research, and the second with more focus on some of the tools that are available to facilitate reproducibility.

In the afternoon, the focus was on next steps that should be taken, in particular in relation to the Data Science Environments (DSE) project, and started with a panel discussion on current activities at UW, NYU, and UCB, followed by break-out groups to discuss strategies and next steps. The break-out group topics were:

1. What are "best practices" regarding code and data?
2. What are next steps towards education and training?
3. How do we assess progress in reproducibility on campus?
4. What are next steps for building community engagement?

Detailed summaries from the breakouts can be found in the next section and slides are posted on the workshop webpage.

Highlights

A few highlights of the discussions and working group reports follows. For more details on some of these topics, please see the reports summarizing the discussion and conclusions of each group, which are included in the next section.

- It is important to keep in mind that reproducibility is a means and not the goal in itself. [The real goals are better science and greater productivity.] The real goals include increasing the productivity and impact of researchers and the quality and credibility of the resulting science. Stressing the benefits to individuals and research groups and providing additional incentives may be more effective in winning converts than focusing too much on reproducibility for its own sake.

- Informal personal engagement may be the key to breaking through the chicken-and-egg problem that changing the culture requires new reward structures, which require policy changes, which in turn require cultural changes. Although helping guide top-down policy changes in journals or grant funding is also important, the real change in culture will have to be bottom up as researchers and students learn to appreciate the true value. Sharing personal success stories and talking informally with students and colleagues can help build a community in which reproducibility is valued.

- A short set of best practices and recommended tools should be prepared for postdocs and graduate students arriving this summer. This document can evolve over the next several years but it is important to start with something concrete. A draft set was discussed by break-out group 1 and will be further refined.

- It is important to teach good coding and research habits as early as possible in the curriculum -- testing frameworks, documentation, and version control should be integrated into many courses and taught as part of the modern scientific method, along with good practices for managing, curating, and archiving data.

- In addition to incorporating these topics into the undergraduate curriculum, a short course for all incoming graduate students and postdocs would be helpful. Opportunities should also be available for more senior researchers to learn new tools, at least to gain an appreciation for the tools available and the value of using them, to help build support at all levels.

- A curated collection of resources on these topics would be very useful for students and instructors, and for researchers more generally. A joint effort between the partner institutions would be most efficient.

- A course taught by Philip Stark and Aaron Culich at Berkeley last year could serve as a model for other courses, including one planned at UW next Spring. GitHub repository is here: https://github.com/stat157

- A draft survey to help assess the current state of reproducibility on campus was discussed by break-out group 3. Other assessment means were also explored, including ways to assess curriculum and grant proposals.

Breakout Group Reports

Best Practices

Participants

Joe Hellerstein (UW, Google), moderator
Nat Goodman (ISB)
Andrew Markiel (UW School of Medicine)
William Stein (UW Math) wstein@uw.edu
Fernando Perez (UC Berkeley)
Michael Miller (ISB)
Dick Kreisberg (ISB)
Jim White (UW)
Mauricio Del Razo (UW)
Jim W. (Boeing)
Randy LeVeque (UW)

Session Summary

At the beginning, everyone provided a short sentence (on the order of ten words) that expressed what they considered most important about best practices for reproducibility. This formed the basis for our discussions. The responses clustered into the following groupings.

1. *Open access, ontologies, clean description of models*. The theme here is that reproducible studies must provide key data/code in a well that these artifacts can be interpreted by others. This often necessitates metadata such as specifications in well defined ontologies. Or it may be that publications describing the study's methods are sufficiently descriptive that others can reproduce the code and data. The later might be communicated through a set of equations.
2. *Provenance, traceability*. The discussion here focused on how code and data artifacts are transformed from a public, shared state to that used in a study. Traceability was viewed as a core part of provenance. But the view is that provenance also includes a management layer and potentially other considerations as well.
3. *Mindset of mistakes, regression testing, verification, code reviews, comprehensible code, documented processes, testing for similar behavior*. The central idea here is that there must be a way to verify that the computational parts of a study produce the published results. This is required to provide a validation of the scientific work. Also, this is needed so that others can build on the work of the study. Many concepts from software engineering come into play, such as code reviews, unit tests, and system tests.
4. *Extensibility, scalability.* This relates mostly to reproducibility being a means to the bigger goals of extensibility and scalability. The thinking was that a good study should be structured in a way so that modest extensions can be done without major revisions to the code or data. Also, it should be possible to scale the study to larger data sets (within some scope).
5. *Tools & solutions, especially notebook technology and engineering incentives for researchers to ensure their work is reproducible*. Notebook technology (e.g., iPython, RStudio, Mathematica, SageMath) provide an easy way to capture and replicate the sequence of

interactions that led to a result. A challenge here is to be able to edit the wrong paths and/or restructure a complex analysis so that others can examine the process that led to the result and replicate it. The thinking about incentives is that there's extra work required to make a study replicable; we need to engineer incentives and/or provide appropriate communication to motivate others.

6. *Goals/objectives*. We identified three reasons why it's important for studies to be reproducible: (1) scientific credibility, (2) ability to build on the work of others, and (3) more efficient science (since common tools are used and it's possible to rely on the results of a reproducible study). An example of (3) is Rob Knight at the University of CO who invests considerable time with his lab in code reviews and other elements of computational reproducibility. A further comment here is that *not* having a reproducible study can provide a false illusion of speed. That is, early on, it seems that the work is going faster because there is no "drag" from testing, documentation, and other considerations. But later on, the science because more error prone and ends up re-doing work because of the lack of considerations for reproducibility.
7. *Relationship to software engineering practice.* There are core parts of reproducibility that coincide with (or align closely) with many elements of good software engineering practice. Examples include: writing readable code; the use of unit tests; continuous testing; and code reviews. There are extensive resources on the web that can be leveraged here, both training and tools (e.g., "linting" tools).

Characterizing the Reproducibility of a Study

Previously, we had proposed four levels of reproducibility in this write-up. One feedback on this was that we should consider a multidimensional structure instead. A first pass on this was to incorporate the following dimensions of a study.

1. Does/does not have explicit models/algorithms
2. Data: original data, similar data, reconstructs the experiments
3. Code: same implementation, comparable algorithms, equivalent science
4. Environment: special purpose, generally available

The thought is that we'd specify different levels of reproducibility for each of the four dimensions as a way to characterize the reproducibility of a study. The appropriate levels for each dimension will depend on the reproducibility objectives such as scientific verifiability and extensibility by other researchers. For example, a researcher interested in understanding the sensitivity of a result to the choice of algorithms might choose a high level of reproducibility for dimensions 2-4 (e.g., a VM with all data in it is the 'exact reproduction'), and explore variations for dimension 1 (e.g., alternative statistical models).

A draft Best Practices document that came about from many hours of discussion after the workshop that incorporates some of the ideas above can be found in Appendix B.

Education & Training

Participants

David Beck (UW eScience & ChemE), moderator
Abraham Flaxman (UW IHME)

Juliana Freire (NYU)
Emilia Gan (UW Pathobiology)
Andrew Gartland (FHCRC)
Philip Stark (Department of Statistics, UC Berkeley)
Kaitlin Thaney (Mozilla Science Lab)
Stephanie Wright (UW Libraries)

Session Summary

The breakout group's discussion revolved around three aspects of education and training for reproducible research: (1) when should the concepts be introduced; (2) who is responsible for teaching & training; and (3) what concepts, skills, technologies, etc. should be taught. The session was fortunate to have representatives from both formal and informal education and training, early career students and post-docs, and data management. Some of the ideas for the three aspects are described below.

1. *When should the concepts be introduced?* **Early**, early, early! Ideally at the time when we introduce students to the scientific method.

a. Reproducibility is a central concept for the scientific method. However, once is not enough! **Undergraduate** classes need to be part of the solution as should **graduate** classes. At least once as an upper level undergraduate, students should be exposed to the ideas, technologies, etc.

b. One idea was that all incoming graduate students participate in a course that covers these topics.

c. One quote that kept coming up (courtesy of Philip Stark) involved teaching 'puppies' because it was going to be very hard to teach 'old dogs.' Established faculty spend their time 'writing 12 proposals a year and don't have time to learn new technologies.' Other than good research hygiene, one of the identified needs for (re)training 'old dogs' is to ensure that they value reproducibility even if they themselves do not re-tool to work reproducibly. If they value, encourage, and even insist on reproducible practices in their labs, and if the appreciate young faculty's efforts towards open and reproducible science are appreciated during promotion and tenure review, fields will eventually adopt reproducible practices.

d. A great avenue for education and training for more established faculty could be external programs such as the Software Carpentry workshop. These also serve to synchronize faculty and students.

e. Some options for less formal training include: Mentorship and continuing engagement after workshops such as offering to train participants as instructor / evangelists.

2. *Who is responsible for education & training?* **Everyone!** Domain sciences *vs.* computer sciences and statistics need to cover aspects of reproducible research hygiene. E.g. Version control should be taught from day one in computer sciences classes and computational statistics courses (for majors and non-majors).

a. Where possible, the concepts should be integrated into existing classes so that students are exposed repeatedly and naturally. In this way, the approaches are not additive to doing science; rather, they are **an integral part of science.**

b. Similarly, the tools should be introduced at the same time as the concepts.

c. For domain students, it is vital that the statistics and computing classes they take have reproducibility education and training components as part of the syllabi, and expect students to work reproducibly. By way of example, one way to integrate the use of good tools into the classroom is to have students submit homework through git pull requests.

d. More academic libraries are taking on roles in data management services and can offer workshops and classes in this area, as well as include relevant information on resources and best practices in student orientations.

3. *What concepts, skills, technologies, etc. should be taught?* It is important to acknowledge that answering this question occupied most of the time for the breakout session. The discussion was extremely fruitful and the diverse backgrounds of the participants offered many opinions. The core list of concepts that seemed appropriate across domains follows, though it is not complete:

a. Collaboration & sharing. Computing has become collaborative. Students should understand how to share code, computational environments, and data, both conceptually and practically. Having the ability to give and receive a code review was cited as a solid example.

b. Communication, presentation & visualization. Being able to present your methods accurately and succinctly enables others to reproduce your work. Similarly, effective communication of results has an important role.

c. Data skills. Practical aspects and conceptual motivations should be presented for: data curation, data structures and data management. Curating, cleaning and transforming data are critical steps in many research fields. Structuring data for efficient handling, processing and parsing is essential as are the tools for above.

d. Technologies & tools. Generally speaking, the wide array of technologies and tools need to be presented. It was difficult to enumerate a specific set though some come through from the Best Practices breakout session.

e. Testing. Getting students into the mentality of 'checking their work' and establishing procedures for testing workflows, algorithms, and code. Embracing the ideas of error handling and test units.

f. Documentation. Separate from (b), documentation reflects the need to keep digital notebooks of ideas, good coding style and comments, etc. How do you document for an audience that has not been identified in timeless ways?

What can we do now?

A quick list of some of the immediate actions that can be taken to advance the state of education and training was generated at the end of the break out. Essentially, this revolved around establishing a curated collection of resources for students and instructors. Curation is necessary as not all resources are appropriate for all skill levels, domains, etc., and not all resources are equally useful or authoritative. Impedance mismatches between a resource and consumer have the potential to frustrate individuals resulting in a lost opportunity. Students and instructors can browse such a resource list organized by various facets, e.g. Good for beginners, Focused on biology, Data management oriented, etc. In addition, there is need for an infrastructure for sharing syllabi and modular components of the education and training process that can be integrated into existing classes or implemented as entirely new classes.

Final thoughts…

One of the common threads in the breakout session was Berkley's STAT 157 developed and taught by Philip Stark. The class is a project-oriented tour of the concepts, sociology and tools that are part of reproducible research hygiene aimed at upper level undergraduates. The GitHub repository is here: https://github.com/stat157 . The current syllabus is here: http://www.stat.berkeley/edu/~stark/Teach/ReproDataSci/reproDataSci_syllabus.pdf

Building Community Engagements

Participants

Ben Marwick, from the UW Anthropology department
Alice Racca, from the UW Bioengineering department
Steven Roberts, from the UW School of Aquatic & Fishery Sciences
Eric Stephan, from the Pacific Northwest National Laboratory
Jake VanderPlas, from the UW eScience institute

After some discussion, we agreed that building momentum for reproducibility in research is a bit of a chicken-and-egg problem. Changing the culture requires new reward structures. New reward structures require policy changes. And of course, policy changes require a changing of the culture. So how do we break-in to this cycle?

After a bit of discussion, we collectively came to the conclusion that *informal personal engagement* is the key to breaking this cycle: building a community committed to reproducibility in their work will start with individuals sharing stories of their own success in applying this framework, whether that is in formal classes or workshops, or in informal one-on-one conversations with colleagues and students.

Nevertheless, everyone can play a role in engaging the community with the importance of reproducibility in research. Though we feel the role of the individual researcher is the most important, there are also important roles to play for universities, for journals, and for funding agencies as well.

Session Summary

Universities

The universities can encourage reproducibility by explicitly supporting faculty who emphasize reproducibility in their research, and by providing educational resources for students and faculty on the subject. One important piece is to provide a set of guidelines for best practices in research. We see the UW Green Laboratory Program as an analogy for a **UW Open Science Program** that would create a list of principles-based criteria to score and certify the reproducibility of work done by a researcher, lab, or project as bronze, silver or gold (or with Mozilla Open Badges, which might make it easier for our program to be used by other places). Other programs that we'd model this on are the Berkeley Initiative for Transparency in the Social Sciences and Project TIER (Teaching Integrity in Empirical Research) at Haverford College. The specific tools and workflows are probably best determined at the level of individual researchers/labs/projects, as each field has its own field-dependent needs. This Open Science Program would provide some basic training, information and resources (incorporating materials from Software Carpentry Bootcamps) to help people improve the reproducibility of their research, and contribute to changing the norms of doing science to place a greater value on reproducibility. This kind of formal system of certification would create a campus-wide community of researchers who value reproducibility, and a network for sharing ideas, providing support and discovering solutions. Like the Green Lab Program, the Open Science Program would be a simple, non-binding program with the main goal being to increase awareness of, and give formal recognition to, reproducibility and openness. A graduate level seminar would be convened once per year to audit people holding certifications and advise applicants, this would also serve to train graduate students in reproducibility.

Journals

Because journals have a large degree of visibility, they can take a leadership role in this area and have a broad impact, as we've seen in recent years through efforts by PLOS ONE, Nature, and others. One method we felt might be effective is to follow the model used in the journal *Psychological Science*, providing "badges" (produced by the Open Science Foundation) to papers which meet certain criteria: in this case, meeting a high bar with regards to reproducibility or openness in the research methods. *Biostatistics* uses a similar system of kite marks on articles to indicate degrees of reproducibility. Another important role the journals could play is to provide, or encourage the use of, persistent and referenceable IDs for data and code, which would make them be first-class research products alongside papers.

Funding Agencies

Because funding agencies control the purse strings, their policy decisions can have wide impacts. One thing we continued to come back to was the need for reproducibility to be valued in grant applications. For example, the NSF recently began including a "data management plan" as part of proposals. This points in the right direction, though some feel that lack of enforcement has made this less effective. Another example of change being effected from the top is the recent executive order requiring openness in government data: our PNNL representative mentioned how much this order had changed the culture at the labs regarding openness of data, which is an important component of reproducibility. We also recognized that the long-term sharing of data often requires extra resources, and the funding agencies should recognize this and provide for these expenses.

Conclusion

All told, everyone in the scientific community has a role to play in encouraging reproducibility in research methods. We think that while top-down efforts from universities, journals, and funding agencies have an important role to play, the real change will come bottom up, from **informal personal engagement** between researchers and their advisors, students, and colleagues.

Assessment

Participants

Michael Brooks (HCDE)
Greg Nelson (CSE)
Dan Halperin (eScience/CSE)
Bill Howe (eScience)
Micaela Parker (eScience/Oceanography)
Abraham Flaxman (Global Health)

Session Summary

Our breakout session was concerned with brainstorming and discussion on ways to assess progress in reproducibility on campus. Much of the assessment breakout session was focused on refining a survey instrument (drafted by Michael Brooks in the Ethnography and Evaluation Working Group). The survey is intended to capture current perceptions and understandings of scientific reproducibility, and could be used as a baseline measurement, distributed to scientists across disciplines and career levels to answer several broad questions:

- How do scientists perceive and understand reproducibility in their field?
- How reproducible is current research?
- What are barriers to improving reproducibility?
- What are strategies currently employed to improve reproducibility?

We walked through the survey questions together and discussed possible improvements. Questions were reworded for clarity and refactored for more precise measurement. Several intriguing variations on the survey strategy were proposed. For example, it might be useful to recruit participants for a survey by finding recent publications from UW or other institutions and contacting the authors. The authors could then answer the survey questions with respect to that particular recent publication (e.g. "How much effort would it take you to reproduce *this* research?"). Within five years, a study oriented around recent publications might be able to find changes in perceptions and practices related to reproducibility.

The initial draft of the survey we discussed in the breakout included open-ended questions to elicit participant definitions of reproducibility, strategies, and barriers. Although useful in unknown domains where the answers are difficult to anticipate, these are difficult to analyze and compare over time. It was also suggested that we begin by distributing a smaller preliminary survey with more open-ended questions. We could then use these answers to create a new survey with more targeted multiple-choice questions which we would distribute to a much larger population to measure progress.

The breakout session also discussed several alternative assessment strategies, such as using a questionnaire or other reporting method to track how much reproducibility was supported or discussed during office hours at the Data Science Studio at UW. We also considered how we could assess changes in how reproducibility is represented in courses at UW, by analyzing course syllabi and other materials or by surveying and interviewing instructors. Finally, it was suggested we do content analysis on successful grant proposals (if these can be obtained) over the five years of the initiative to see how many proposals discuss reproducibility and in what ways.

Appendix A: Schedule

8:30 Coffee available

9:00 Introductory remarks

9:15 Keynote talk: Stephen Friend, President, Sage Bionetworks

10:15 Panel 1 - "Issues of reproducibility"
Moderator: Dan Halperin
Andy Connolly, Astronomy
Abraham Flaxman, Institute for Health Metrics and Evaluation
Eric Klavins, Electrical Engineering

11:00 Coffee break

11:15 Panel 2 - "Some tools and solutions"
Moderator: Jake Vanderplas
Juliana Freire, Computer Science and Engineering, NYU
Philip Stark, Statistics, Berkeley
William Stein, Mathematics

12:00 Lunch on your own

1:30 Panel 3 - "Next steps in Moore/Sloan partnership"
Moderator: Bill Howe
Juliana Freire, Computer Science and Engineering, NYU
Randy LeVeque, Applied Mathematics
Fernando Perez, Henry H. Wheeler Jr. Brain Imaging Center, Berkeley

2:15 Organize break-out groups

2:30 Break out groups -- tentative topics:
What are "best practices" regarding code and data?
Next steps towards education and training?
How do we assess the current state of reproducibility on campus?
Next steps for building community engagement?

3:30-4:15 Reconvene, report back, and discuss

4:30-5:30 Reception

Appendix B: Draft Best Practices

Guidelines for Reproducible and Open Science
Moore/Sloan Data Science Environment (DSE)

Our working definition for reproducible research is that a research result can be replicated by another investigator. This note summarizes best practices to facilitate reproducible research. It is expected that all research conducted with funding from the DSE will be performed in accordance with these guidelines to the extent possible.

Goals

The primary goals motivating these guidelines are:

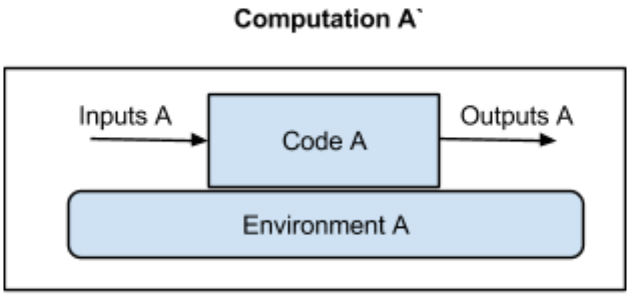
- To achieve greater scientific validity and integrity by making it easier to verify published results.
- To provide the evidence needed to evaluate the reliability of scientific claims.
- To increase productivity of current and future researchers on funded projects.
- To increase the impact of the research performed, software developed, and papers published.
- To help promote data and code as first class research products.
- To increase access to and usability of research products by other researchers.
- To use the DSE as a test bed for developing and promoting tools and cultural changes across a broad spectrum of academic disciplines.

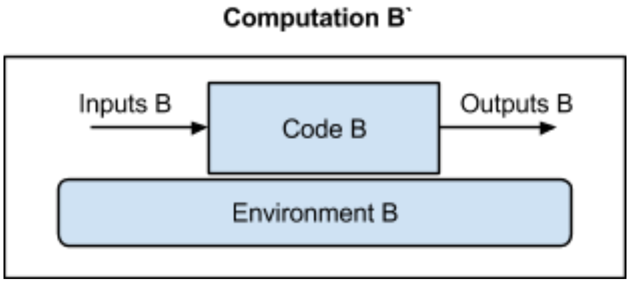
The last goal is an important aspect of the DSE project and so feedback regarding difficulties in following these guidelines, or suggestions for improving them, are always welcome. These guidelines are expected to evolve over the next several years to better meet these goals. Since the overarching goals are greater productivity, increased impact, and better science, it is hoped that researchers will find these guidelines useful to follow in all their work, and will view them as valuable in achieving their own research goals (rather than as an onerous requirement for this particular funding source).

It is also hoped that other research groups and labs around campus and beyond will find it valuable to adopt similar guidelines. Feedback, suggestions, and reports of success or failure are encouraged from any source and can be send to DSE_repo_WG@uw.edu.

Requirements

A simple example helps with identifying the requirements for reproducibility. Consider two computations, A and B. Each computation has its inputs, code (sequence of data transformations/computations), and the environment in which the computation is performed (e.g., operating system, versions of software libraries, visualization packages). Our premise is that if the inputs, code, and environment are equivalent, then the outputs of A should be equivalent to the outputs of B. We use equivalent here to allow for differences that do not have a material impact on the outputs.





Based on these observations, we have the following requirements:

- Manage versions of code and data to ensure that equivalent code and data are used to reproduce results.
- Code should be automated (no manual steps) so that it can be reproduced.
- Share data and code so that others can build on research results.
- Provide clear documentation (or even automation) to show how the computational environment can be created from known components.

Other aspects of reproducibility are also important, e.g. the ability to take a method or algorithm described in a paper and independently turn it into code that gives essentially the same results, but here we focus on the ability to preserve the Inputs, Code, and Environment in such a way that the published outputs can be reliably reproduced.

Best Practices

Version control

- Use version control for all code and documents describing software, workflow, data provenance, etc. Get in the habit of doing this for all projects, whether or not you intend to eventually share it with others.
- Git is the recommended version control system. Back up your repositories by maintaining clones on other computers or in the cloud. Using Github is recommended since it greatly facilitates collaboration and eventual sharing. For work in progress, private Github repositories are available through the eScience Github account or by requesting a free academic upgrade from Github. Bitbucket, Gitlab and other services can also be used to host private git repositories. To request access to the eScience Github account, please contact info@escience.washington.edu.

Replicable computations

- Use scripts rather than GUIs with data analysis or visualization tools in order in insure that you can reproduce the results. Keep the scripts under version control.
- When appropriate, use notebook environments as an easy way to capture and replicate the sequence of instructions (interspersed with documentation and commentary) that led to a result. Suggested tools include IPython notebooks, Sage Worksheets, RStudio (all open source), or Mathematica, Maple, Matlab Publish.

- More generally, document your code so that you can decipher it later and others can understand what you have done. Use literate programming tools to facilitate this when possible. Some examples include doxygen, CWEB, Sphinx.
- When suitable, use workflow management systems to track a series of experiments performed, versions of code used, data provenance, etc. Suggested tools include VisTrails, Taverna, Galaxy.

Data provenance, sharing and archiving

- Data used in publications (and associated metadata) should be available to readers of the publication - subject of course to privacy requirements or related issues, but sharing should be the default. Where possible, seek to provide a DOI for data in your publications.
- Data should be deposited in archives that are appropriate for the discipline, data size, and the nature of access. For example, biological sequence data should be deposited in the appropriate NCBI archive whereas small data could be placed in github. For larger data that does not have a discipline related archive, the UW ResearchWorks repository can be used which will also generate a DOI. To deposit data in ResearchWorks, send email to rworks@uw.edu. Other general repositories such as figshare or ZENODO may also be useful. (SQLShare?)
- Metadata and provenance of data (e.g. original source, date acquired, etc.) should be recorded and archived with the data.

Code sharing and archiving

- Code developed as part of the research (e.g. to illustrate a new algorithm or to perform data analysis) should be made available to readers.
- Perform internal code reviews and replication studies within your research group to insure that you have archived and adequately documented all code and data needed to reproduce published results. This is also a good way to catch errors and oversights that could negatively impact your scientific reputation or future productivity. Moreover, sharing and reviewing code within small groups can reveal new tips and tricks that accelerate software development.
- Insure that your code and data can be properly cited. Many repositories issue a DOI, for example.
- Make it clear what the rules are for others to use your code or data, e.g. by attaching a suitable license to code. This increases the chances that others will build on it, and that you will get the proper credit.

Replicable environment

- The computation environment should be documented with sufficient detail so that others can create an equivalent environment. It is preferred to use automated tools such as Vagrant for Virtual Machines or make for Unix.
- Use virtualization when appropriate to archive the environment in which a code runs, e.g. VirtualBox (which is free) or VMWare, or machine images on cloud computing platforms such as AWS or Windows Azure.

Other resources

- Ten Simple Rules for the Care and Feeding of Scientific Data
  http://www.ploscompbiol.org/article/info:doi/10.1371/journal.pcbi.1003542

- Literate Programming Tools
  http://www.literateprogramming.com/tools.html

- Detailed guide to using vagrant for science here:
  http://hplgit.github.io/vagrantbox/doc/pub/._vagrant_box001.html

- Thoughts on metadata annotation & standards
  https://github.com/mozillascience/code-research-object/issues/2#issuecomment-35610035