Reproducibility and Open Science

Follow along at: https://gordonwatts.github.io/ros-roadshow

$ 37.8M for 5 years: "Moore-Sloan Data Science Environments"

Additional funding from

- Washington Research Foundation
- National Science Foundation

Reproducibility and Open Science Working Group:

- https://reproduciblescience.org/
- Mailing list: reproducible@uw.edu

- Goal: Stimulate discussion and share ideas
  - Types of reproducibility
  - Tools for reproducibility
- Data: archiving, curation, sharing
- Code: scripting, versioning, collaborating, sharing, publishing
- Publication: open access

Use scripts, not GUIs, for data analysis and visualization.

Use version control / provenance tracking tools.

Archive code and data used for published results.

Why?

- Ability to check results in prior publication,
- Ability to build on your own past research of your own (or students / collaborators).
- Easily modify tables/figures to satisfy referees, etc.

Use scripts, not GUIs, for data analysis and visualization.

Use version control / provenance tracking tools.

Archive code and data used for published results.

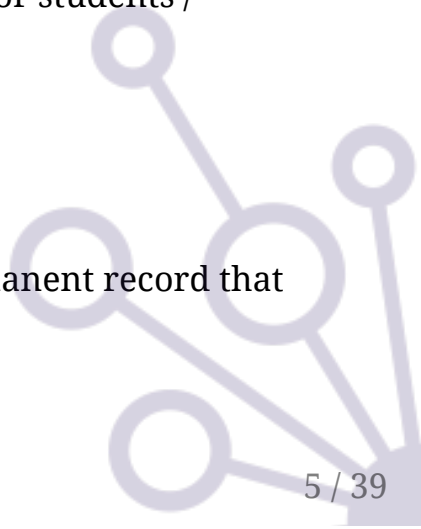
Why?

- Ability to check results in prior publication,
- Ability to build on your own past research of your own (or students / collaborators).
- Easily modify tables/figures to satisfy referees, etc.

Auditable Research:

Even if code and data are not shared, there should be a permanent record that can be checked.

Analogous to lab notebooks.



Allowing others to reproduce your results.

(Readers, referees, researchers down the hall...)

Why?

- Verifying scientific integrity of results.
- Aids in understanding ideas, implementing methods
- Increases impact of work.

Allowing others to reproduce your results.

(Readers, referees, researchers down the hall...)

Why?

- Verifying scientific integrity of results.
- Aids in understanding ideas, implementing methods
- Increases impact of work.

"An article about computational result is advertising, not scholarship. The actual scholarship is the full software environment, code and data, that produced the result."

Buckheit and Donoho (1995)

Traditional research in Mathematics is reproducible...

- A paper containing a new theorem cannot be published without the proof.

Traditional research in Mathematics is reproducible...

- A paper containing a new theorem cannot be published without the proof.

It wasn't always so...

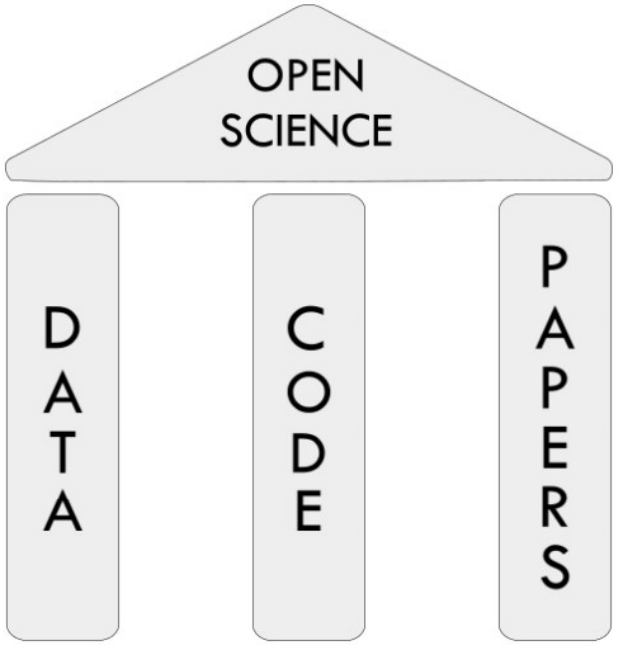
There is no . . . mathematician so expert in his science, as to place entire confidence in any truth immediately upon his discovery of it. . . . Every time he runs over his proofs, his confidence encreases; but still more by the approbation of his friends; and is raised to its utmost perfection by the universal assent and applauses of the learned world.

- David Hume, 1739

Many arguments against publishing code might be applied to proofs in an alternate universe...

"Top Ten Reasons To Not Share Your Code (and why you should anyway)", SIAM News, April, 2013

- The proof is too ugly to show anyone else.
- I didn't work out all the details.
- I didn't actually prove the theorem - my student did.
- Giving the proof to my competitors would be unfair to me.
- The proof is valuable intellectual property.
- Etc.



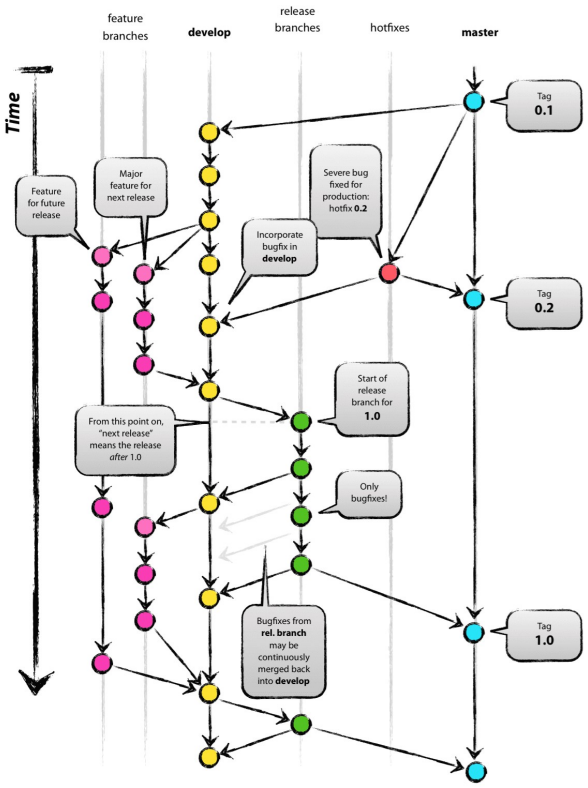
Gorgolewski and Poldrack (2016)

The broader open source software community has worked out a lot of the issues around making code available and broadly useful.

The broader open source software community has worked out a lot of the issues around making code available and broadly useful.

- Version control

http://www.phdcomics.com/comics/archive.php?comicid=1531


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

The broader open source software community has worked out a lot of the issues around making code available and broadly useful.

- Version control

The broader open source software community has worked out a lot of the issues around making code available and broadly useful.

- Version control
- Automated software testing

Write code that checks that our code does what we expect it to do

Write code that checks that our code does what we expect it to do

We all do this anyway...

Write code that checks that our code does what we expect it to do

We all do this anyway…

Formalize this and keep running the tests every time you make changes to the software

Write code that checks that our code does what we expect it to do

We all do this anyway...

Formalize this and keep running the tests every time you make changes to the software

Continuous integration

Write code that checks that our code does what we expect it to do

We all do this anyway...

Formalize this and keep running the tests every time you make changes to the software

Continuous integration

Why not design your analysis to run in this envrionment as well?

- No hand art
- Parameters and configurations tracked
- Results tracked as artifacts and log files
- Results computer accessible

The broader open source software community has worked out a lot of the issues around making code available and broadly useful.

- Version control
- Automated software testing

The broader open source software community has worked out a lot of the issues around making code available and broadly useful.

- Version control
- Automated software testing
- Software licensing

http://www.astrobetter.com/blog/2014/03/10/the-whys-and-hows-of-licensing-scientific-code/

http://www.astrobetter.com/blog/2014/03/10/the-whys-and-hows-of-licensing-scientific-code/

- Code without a license is *closed code*

http://www.astrobetter.com/blog/2014/03/10/the-whys-and-hows-of-licensing-scientific-code/

- Code without a license is *closed code*
- Use a license that is broadly compatible (*do not make up your own license!*)

http://www.astrobetter.com/blog/2014/03/10/the-whys-and-hows-of-licensing-scientific-code/

- Code without a license is *closed code*
- Use a license that is broadly compatible (*do not make up your own license!*)
- Consider using a permissive (e.g, BSD) license, rather than a "copyleft" license

http://www.astrobetter.com/blog/2014/03/10/the-whys-and-hows-of-licensing-scientific-code/

- Code without a license is *closed code*
- Use a license that is broadly compatible (*do not make up your own license!*)
- Consider using a permissive (e.g, BSD) license, rather than a "copyleft" license

Licensing makes your software useful to others, while maintaining your rights as the creator of the software.

To proceed in the academic career ladder, we need signals that our work is meaningful and useful

Especially pertinent if some aspects of your software work are not captured by traditional peer-reviewed publications

Software papers give you a line in your CV, and allow others to cite their dependence on your software (independently from their inspiration by your findings).

https://www.software.ac.uk/which-journals-should-i-publish-my-software

https://www.software.ac.uk/which-journals-should-i-publish-my-software

Journal of Open Source Software

https://www.software.ac.uk/which-journals-should-i-publish-my-software

Journal of Open Source Software

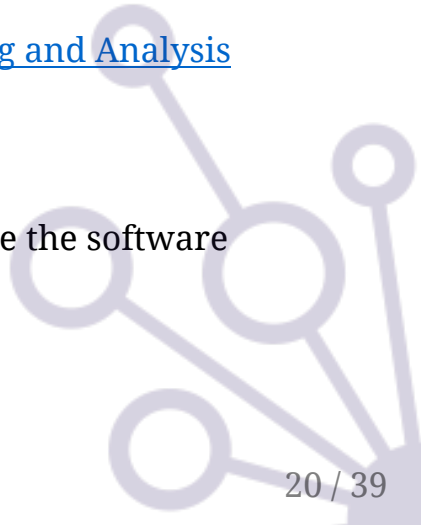
How to cite software

https://github.com/uwescience/citing_software

We did something like this at the recent Advanced Computing and Analysis Techniques in Physics Research conference.

Daniel Katz's talk contians further examples.

All submissions for the ACAT proceedings will be asked to cite the software directly using these guidelines.



Data Curation

Ten Simple Rules for the Care and Feeding of Scientific Data

by Alyssa Goodman, Alberto Pepe , Alexander W. Blocker, Christine L. Borgman, Kyle Cranmer, Merce Crosas, Rosanne Di Stefano, Yolanda Gil, Paul Groth, Margaret Hedstrom, David W. Hogg, Vinay Kashyap, Ashish Mahabal, Aneta Siemiginowska, Aleksandra Slavkovic,

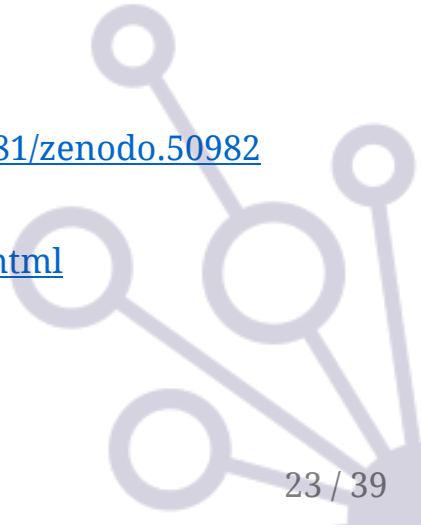
PLOS Computational Biology 10(2014), e1003542.
http://dx.doi.org/10.1371/journal.pcbi.1003542

- Rule 2. Share Your Data Online, with a Permanent Identifier (e.g. DOI)
- Rule 4. Publish Workflow as Context
- Rule 5. Link Your Data to Your Publications as Often as Possible
- Rule 6. Publish Your Code (Even the Small Bits)
- Rule 7. State How You Want to Get Credit
- Rule 8. Foster and Use Data Repositories

- Open Science Framework: https://osf.io/

  Slides by Kara Woo in eScience Reproducibility and Open Science Seminar

- UW ResearchWorks: https://researchworks.lib.washington.edu/
  - Ex: Human neuroimaging data, https://digital.lib.washington.edu/researchworks/handle/1773/33311
- figshare: https://figshare.com/
- Zenodo: https://zenodo.org/
  - Ex: Clawpack Version 5.3.1 at http://dx.doi.org/10.5281/zenodo.50982
  - Ex: Code, data, and Jupyter notebooks for a paper: http://faculty.washington.edu/rjl/pubs/KLslip/index.html



Geosciences:

- DesignSafe: https://www.designsafe-ci.org/
- Community Surface Dynamics Modeling System (CSDMS): http://csdms.colorado.edu Data and model repositories, Web interface to some models

Neuroscience:

- Collaboration in Computational Neuroscience: https://crcns.org/
- Open fMRI: https://openfmri.org/

Sharing Detailed Research Data Is Associated with Increased Citation Rate

Piwowar HA, Day RS, Fridsma DB (2007) PLoS ONE 2(3): e308.
http://dx.doi.org/10.1371/journal.pone.0000308

Sharing Detailed Research Data Is Associated with Increased Citation Rate

Piwowar HA, Day RS, Fridsma DB (2007) PLoS ONE 2(3): e308.
http://dx.doi.org/10.1371/journal.pone.0000308

A collection of links on the topic: http://opcit.eprints.org/oacitation-biblio.html

- Organize your data in a manner that will make sharing easy.

- Organize your data in a manner that will make sharing easy.
- Develop your software using git/Github. Use private repos during development, if you must (https://education.github.com/)

- Organize your data in a manner that will make sharing easy.
- Develop your software using git/Github. Use private repos during development, if you must (https://education.github.com/)
- Use tools that facilitate open communication around code, data and results.

Jupyter

A notebook format that supports reproducibility by interweaving code, data and figures.

40 different languages are supported, including Julia, Python and R, and many others (Matlab too!).

http://dx.doi.org/10.1371/journal.pone.0123272

Evaluating the Accuracy of Diffusion MRI Models in White Matter

http://dx.doi.org/10.1371/journal.pone.0123272

Code: https://github.com/vistalab/osmosis

Notebooks:
https://github.com/vistalab/osmosis/tree/master/doc/paper_figures

Data: https://purl.stanford.edu/ng782rw8378

To run these notebooks, you have to install all my dependencies.

To run these notebooks, you have to install all my dependencies.

To reproduce my results, you have to download my code, and my data, to your machine.

To run these notebooks, you have to install all my dependencies.

To reproduce my results, you have to download my code, and my data, to your machine.

If my code has compiled components, you'll need to compile it.

To run these notebooks, you have to install all my dependencies.

To reproduce my results, you have to download my code, and my data, to your machine.

If my code has compiled components, you'll need to compile it.

If you happen to have a different operating system, different compiler, different libraries, etc... we might be out of luck!

Virtualization

- Package code along with complete environment
- E.g., VirtualBox, VMware, etc.
- Docker

Virtualization

- Package code along with complete environment
- E.g., VirtualBox, VMware, etc.
- Docker

Cloud computing

- E.g., Amazon EC2, Windows Azure, etc. + VM

Virtualization

- Package code along with complete environment
- E.g., VirtualBox, VMware, etc.
- Docker

Cloud computing

- E.g., Amazon EC2, Windows Azure, etc. + VM

Web platforms for running code

- E.g., RunMyCode.org, wakari.io
- SageMathCloud: https://cloud.sagemath.com

http://mybinder.org

Developed by the Jeremy Freeman's Lab at Janelia Farms

Provisions a GitHub repository as a cloud-computing environment

http://mybinder.org

Developed by the Jeremy Freeman's Lab at Janelia Farms

Provisions a GitHub repository as a cloud-computing environment

For example, here is a binder that will run the LIGO analysis that confirmed the existence of gravitational waves (The Github repository is here).

I'll address more complex workflows later

Publish in open access journals

Use preprint servers:

Make your work available *before it is published*

https://arxiv.org/ http://biorxiv.org/

Publish in open access journals

Use preprint servers:

Make your work available *before it is published*

https://arxiv.org/ http://biorxiv.org/

Provides access to your work

Publish in open access journals

Use preprint servers:

Make your work available *before it is published*

https://arxiv.org/ http://biorxiv.org/

Provides access to your work

Establishes precedence

- Reproducibility is a cornerstone of science.

- Reproducibility is a cornerstone of science.
- Think about reproducibility when you start your project and bake it in.

- Reproducibility is a cornerstone of science.
- Think about reproducibility when you start your project and bake it in.
- Make your data, code and papers open and available, so that others can build on your work.

- Reproducibility is a cornerstone of science.
- Think about reproducibility when you start your project and bake it in.
- Make your data, code and papers open and available, so that others can build on your work.
- Come and talk to us!

Reproducibility and Open Science Working Group:

- https://reproduciblescience.org/
- Mailing list: reproducible@uw.edu, https://mailman11.u.washington.edu/mailman/listinfo/reproducible

Come to our office hours!

http://escience.washington.edu/office-hours/

We're eager to hear! And you can post issues/questions here:
https://github.com/rjleveque/2016-ros-amath/issues

More materials

https://medium.com/@lorenabarba/barba-group-reproducibility-syllabus-e3757ee635cf#.x1w245xvg

- List of 10 recommended tutorials
- https://help.github.com/categories/bootcamp/
- http://git-scm.com/book/en/Getting-Started-Git-Basics
- Github online tutorial

More general resources, including Git:

- Software Carpentry
- Code Academy