Best practices

"An article about a computational result is advertising, not knowledge. The actual knowledge is the full software environment, code and data, that produced the result"[1]

Reproducible research

It is expected that papers in science describe the results and provide a sufficiently clear protocol to allow successful repetition and extension of analyses based on original data[2]. Keep in mind the following elements/tools:

- ❑ Data - Dropbox[1] and Drive[2] to make available very simple and small data. Some projects could require specialized repositories that support the FAIR data guiding principles: Data should be Findable, Accessible, Interoperable and Re-usable[3].
- ❑ Code and documentation - Definitely Jupyter Noteboot[3] is the most functional tool in literate computing.
- ❑ Version control repository - Bitbucket[4] is indicated for private projects and GitHub[5] is a more versatile tool.
- ❑ Environment - It depends of your project complexity. Docker[6] is a powerfull tool but be carefull: "Do no use a cannon to kill a fly".
- ❑ Final paper - LaTeX[7] system is the recommended tool for producing scientific documents. Overleaf[8] platform facilitates Latex usage, online collaborative work and integration with your literate computing tool.

Useful links

[1] https://www.dropbox.com
[2] https://www.google.com/drive/
[3] [Starting] http://jupyter.org/install.html
[3] [Moving on] https://www.datacamp.com/community/tutorials/tutorial-jupyter-notebook
[4] https://bitbucket.org/product
[5] https://github.com/
[6] [Starting] https://www.docker.com/
[6] [Linux installer] https://docs.docker.com/engine/installation/linux/ubuntu/
[7] [Starting] https://en.wikibooks.org/wiki/LaTeX
[7] [Moving on] https://tobi.oetiker.ch/lshort/lshort.pdf
[8] https://www.overleaf.com/

Recipe to make an executable paper

In order to do a reproducible research, a serie of steps and elements must be fullfilled. These steps and elements depend of the research type and size, but in general, it must be observed:

Preparing your data

- ❑ The most important thing is to think from the beginning in reproducibility. All of your project decisions, including variable's names, versions, tool selection, data naming, must be oriented toward reproducibility.
- ❑ Your data must be anonymized leaving file names without identification.
- ❑ Leave instructions about preparation of your dataset, including naming and location, in order to be loaded by the code. These instructions go well in the readme file on your repository or in a plain text file inside your dataset.

Developing your code

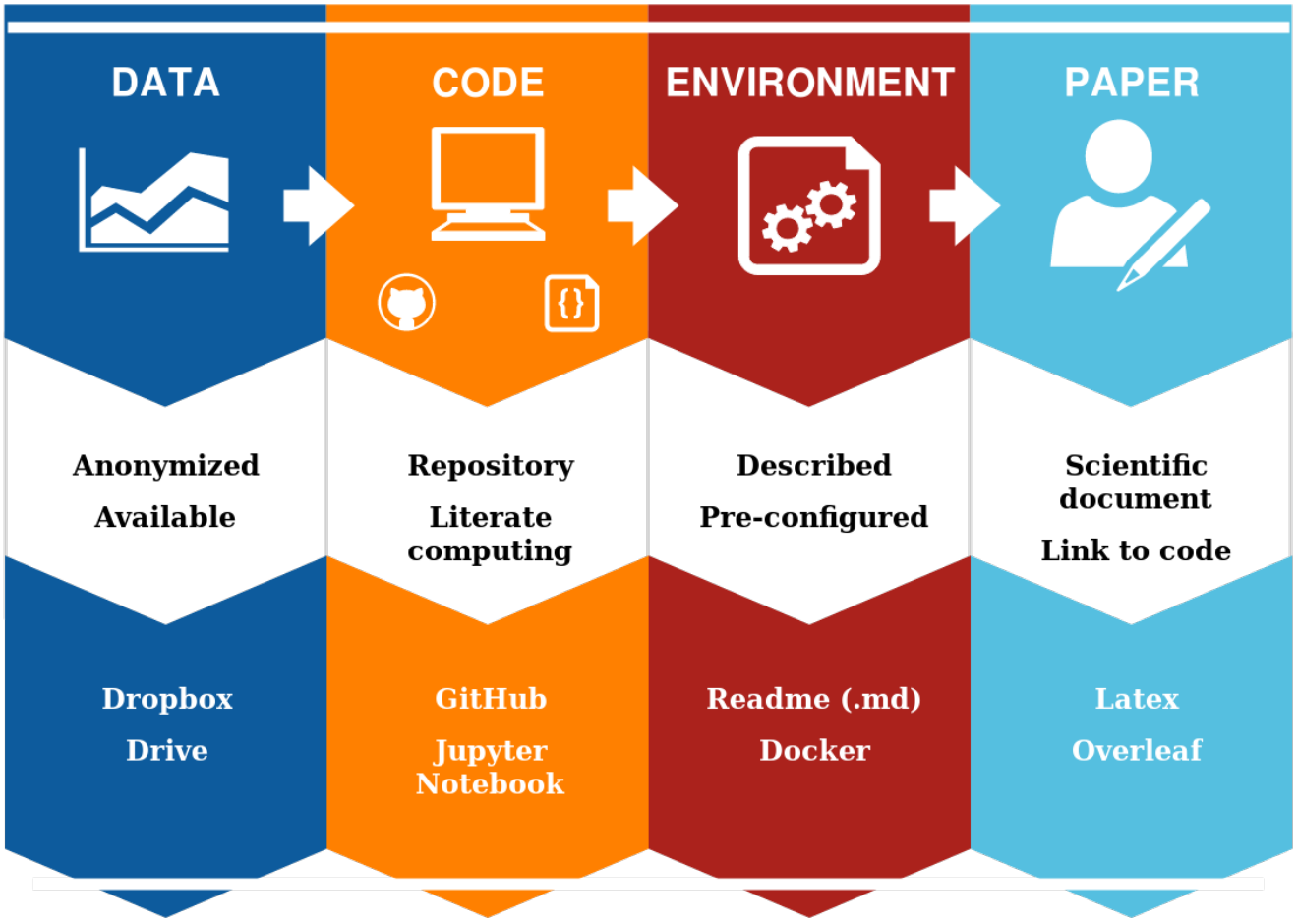
- ❑ Create a repository in some version control repository tool and clone this locally. It is recommended work in two directories: <deliver> (for released versions of your code) and <dev> (for developing notebooks) located in root directory of your repository. Libraries and script functions can stay in your <dev> directory.
- ❑ Name your scripts or notebooks following a determined sequence, ex: <data_initials_idfile> (<data> is the script creation data, <initials> is the initials of the coder's name and <idfile> is a very short description).
- ❑ Use your literate computing preferred tool for coding, providing detailed description of the program logic, insights and partial conclusions about your code in a natural language. When developing your code, focus on the structure of your final paper.
- ❑ As possible, leave comments along your code in order to facilitate more specific understanding of variables, functions, operations and code blocks in general.
- ❑ Keeps correctly identified functional versions of your code.
- ❑ Always use commit and merge git commands for updating your files online. Avoid make these operations manually.

Make available environment

- ❑ It is important specify your environment (necessary software, libraries and dependencies) for maximizing reproducibility. Leave a readme file in your repository root holding this information. The user who wants to reproduce your work will try to emulate your environment as much as possible.
- ❑ In this readme file, add instructions for installing all necessary software and dependencies in a recipe-wise.
- ❑ If you have many dependencies and special libraries and you want to guarantee full reproducibility try to create an environment in a container platform.

Writing final paper

- ❑ The paper is the facade of your project. So, maintaining a reduced version of your literate code focused in the main results, figures and line of thought. If the reader wants to know more details he will go behind of the explicit code. Explicit code doesn't go here but pseudo-code is allowed.
- ❑ From your literate code, you will need to add/format some issues to make the final paper: format according with document type, figures, internal and external references, tables, and captions.



Do and Don't

- ❑ A workflow design can help to understanding your project. Try to make each component of your workflow an equivalent script, function or notebook. Tag input and output files.
- ❑ Use variable names that help to understanding their role in the code.
- ❑ Always prefer supported and official software over external, privative or unusual libraries or package.
- ❑ Saving and making available partial variables, tables and analytic data will increase reproducibility. As possible, provide raw data and analytic data.
- ❑ Leave the git structure ready to be cloned and executed.
- ❑ More information on: https://github.com/wilomaku/Reproducibility_best_practices/blob/master/Learned_lessons.ipynb

Neuroimaging case[4]

- ❑ DICOM data is, in general, not very suitable for reproducibility. Try convert to nifti format. This format eliminates excess of information maintaining data anonymized.
- ❑ Since results are dependent of the image capture parameters provide information like: number of subjects (quantity and population type), scanner, acquisition plane, image resolution, slices used and acquisition specific information (gradient echo, spin echo, TE, TR, flip angle, acquisition time).
- ❑ Pre-processing steps can be arduous to be reproduced. A detailed step-by-step guide with parameters setting and software configuration should be very useful.
- ❑ Follow a standarized data structure, like BIDS (http://bids.neuroimaging.io/) for gain data usability.
- ❑ Remember that information belongs to humans. Consent documents are necessary for data sharing.

References

[1] David L Donoho. An invitation to reproducible computational research. *Biostatistics*, 11(3):385–388, 2010.

[2] Geir Kjetil Sandve et al. Ten simple rules for reproducible computational research. *PLoS Comput Biol*, 9(10):e1003285, 2013.

[3] Mark D Wilkinson et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3, 2016.

[4] Thomas E Nichols et al. Best practices in data analysis and sharing in neuroimaging using MRI. *Nature Neuroscience*, 20(3):299–303, 2017.

This Best Practices guide to reproducible research was developed as part of Tools for reproducible research course at School of Electrical and Computer Engineering - University of Campinas (Brazil-SP). This guide is licensed under the Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit http://creativecommons.org/licenses/by/3.0/. Author: William Javier García Herrera (wjgarciah@unal.edu.co).