Git for research projects

Contents

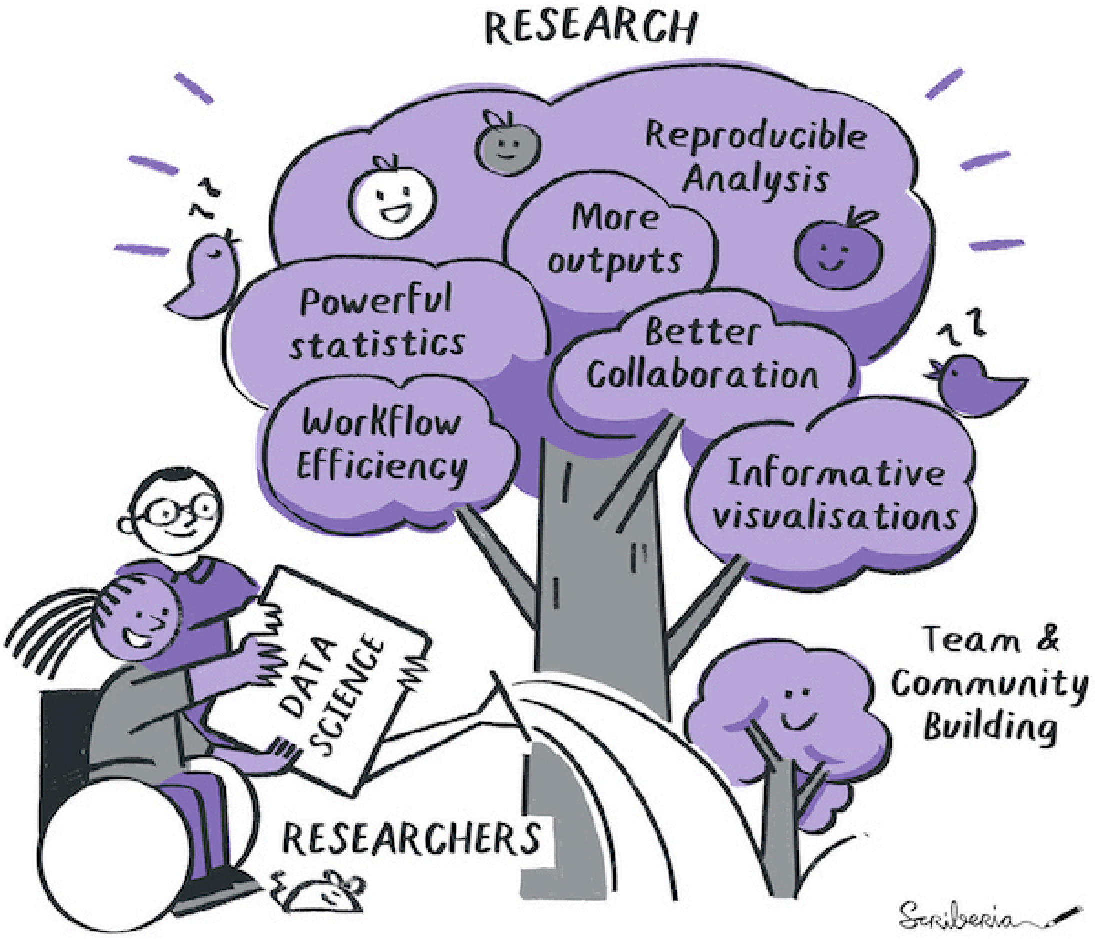

Fig. 29 Data science practices can leverage the potential of the research workflow, in order to produce better research in less time. *The Turing Way* project illustration by Scriberia. Used under a CC-BY 4.0 licence. DOI: 10.5281/zenodo.3332807.

Because each research project has a data science component, there are clear advantages to use data science practices for the management of all the data produced during research. In particular, the use of Git and GitHub is very appealing. However, GitHub is not enough to handle all research projects:

- Data versioning needs special care, see the section on data version control.
- Specific folder structure help in the workflow.
- They may be some legal issue to use an american tool for your data.

You may refer to a carpentry workshop related to this topic.

Potential

Here is a non-exhaustive list of features that a Git/GitHub workflow bring to data science projects, and that would be useful for research projects:

- Backup data by pushing the data to a Git platform, toward a public or private repository.
- Easily use different computers to work on the same project (with yourself of with collaborators).
- Keep track of contributions.
- Facilitate the use of folder templates to help with files organisation, see Data Organisation.
- Use Git platforms tools for project management.
- Use Git platforms for outreach, even when the repository is private (using the Wiki).
- Create an associated website under the same organisation on the Git platform.

Issues

As described in the general section about Git, Git does not work well when there are a lot of data, or when the data are large. When you expect the project to get large, one needs to set a different tooling to avoid creating unpractical repositories. Some of these tools makes it more difficult to access or see you files, so it is important to plan in advance what tool will best suits your need. See the section on Data Version Control for more detailed explanations.

Briefly, in order to use Git when there are lots or large files, one needs to split the data in different repositories, and have these repositories use the git-annex

technology.

Tools

We encourage you to use a Git platform that is provided as an open infrastructure. In many university, you will have access to a GitLab platform (which works very similarly to GitHub). Alternatively, you may want to install your own instance of one of the more lightweight open source Git platform (gogs, GitLea, GIN).

If you have many or large files, you will need to use the Git submodules and git-annex technologies. If you do, we encourage you to look into DataLad and follow the progresses of the GIN-Tonic project. Be prepared to invest some time learning how to use these tools.

Fictive example

Max has created a folder following a standard structure, they uses datalad to create submodules for each experiment, where they will save their datasets. Using datalad, the git-annex technology is used to save the file content outside of the Git repository at every push. They got their own GIN platform where the git repository and git-annexed content is saved, and backed up. Their collaborators have access to the whole data, either via the browser interface or using some command line tool. The GIN repositories are linked to a GitLab issue, so that the team is using advanced project management tools offered by GitLab. The data analysis code is also set in a submodule, where git-annex is not allowed.

After working for a couple of years on the project, together with their collaborators, Max has written a paper where they could link both the data and the analysis code, which was made public by archiving the Git repositories and the git-annexed data on the university library service.

While this use case is already possible, it requires to use the command line (to use datalad), and get a GIN instance installed (the public GIN instance is meant only for neuroscience data).