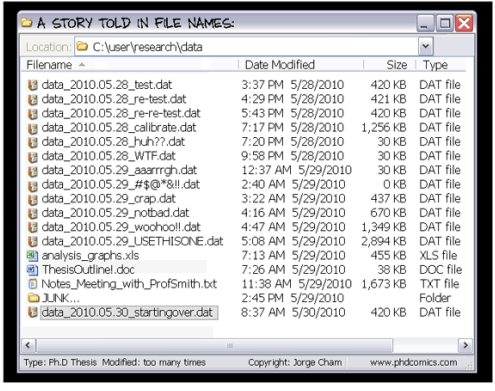
Introduction to Git

Hendrik Ranocha

2023-10-31

Version control

- track changes to files, share it with others
- originally developed for software development — cornerstone of reproducible research nowadays
- `main` copy of code in repository (backup, history, blame, . . .)



Credit: `https://phdcomics.com/comics/archive/phd052810s.gif`

First steps with Git

Before you start coding:

- `git config -global user.name "[firstname lastname]"`
- `git config -global user.email "[valid-email]"`
  Choose the same email address as on GitHub!

Getting started:

- Example `https://github.com/ranocha/2023-RSE_in_Julia`
- You can use the GitHub online interface for simple changes
- `git clone [url]`
- `git --help`
- `git log` — what are all these numbers?

Basic workflow in your own repository

- `git status`
- `git checkout main`
- `git pull`
- `git branch [new-branch-name]`
- `git checkout [branch-name]`
- `git add [files]`
- `git commit -m "commit message"`
- `git push`
- `git checkout main`
- `git merge [branch-name]`

Basic workflow when contributing to another repository

- `git status`
- `git checkout main`
- `git pull`
- `git branch [new-branch-name]`
- `git checkout [branch-name]`
- `git add [files]`
- `git commit -m "commit message"`
- Fork the repository on GitHub and `git remote add myfork [url]`
- `git push -u myfork`
- Create a pull request (PR) on Github
- Wait for code review and improve the code
- Let the maintainer merge your PR

Git workflows 1: centralized workflow

- Basically one branch: `main`
- Keep your changes in local commits till some feature is ready
- If ready, directly push to `main`; no PRs, no reviews
- Conflicts: fix locally (push not allowed anyway), use `git pull --rebase`
- Good for: small teams, small projects, projects that are anyway reviewed over and over again
- Example: LaTeX papers, reports, theses
  - Do not use very long lines! (I typically use $\sim$ 80 characters)
  - Maybe: Put each sentence in separate line (I don't)
  - Maybe: Put each section in separate file (I don't)

Git workflows 2: feature branch workflow

- Each feature (or bugfix) in separate branch
- Push feature branch to remote, use descriptive name
- `main` should never contain broken code
- Protect direct push to `main`
- PR with review to merge from feature branch to `main`
- Rebase feature branch on `main` if necessary
- Delete remote branch once merged and no longer needed
- Good for: small teams, small projects, prototyping, websites (continuous deployment), documentation
- Example: Trixi.jl

Git workflows 3: bigger open source projects

Similar as previous workflow but

- Develop on `main`
- New branch for every release cycle, forked from `main`
- Backport bugfixes to LTS, current stable release
- Example: `https://github.com/julialang/julia`

When you start contributing

- Make a small, self-contained PR with a clear description
- Use GitHub's features to link to issues, PRs etc. via `#number`
- Look for "good first issue" labels, e.g., `https://github.com/JuliaLang/julia/issues?q=is%3Aopen+is%3Aissue+label%3A%22good+first+issue%22`

Ressources

- Git cheatsheet by GitHub:
  `https://education.github.com/git-cheat-sheet-education.pdf`
- RSE Summer School
  `http://www.simtech-summerschool.de/material/2_tue/git/slides.html`
- Official documentation `https://git-scm.com/doc`
- RSE with Python online book
  `https://merely-useful.tech/py-rse/git-cmdline.html` etc.

Exercises

- Create a GitHub repository `My_RSE_in_Julia.jl` (why `.jl`?)
- Implement the trapezoidal rule to approximate the integral $\int_a^b f(x)\,\mathrm{d}x$ as

$$\int_a^b f(x)\,\mathrm{d}x \approx \sum_{k=1}^{N} \frac{f(x_{k-1}) + f(x_k)}{2} \Delta x$$

  with $\Delta x = (b-a)/N$, $x_0 = a$, $x_{k+1} = x_k + \Delta x$
- Test (and plot) the convergence of your code, e.g., using

$$\int_0^1 \cos(200x)\,\mathrm{d}x, \quad \oint_{|z|=1} \frac{\cos(z)}{z}\,\mathrm{d}z = \int_0^{2\pi} \frac{\cos(\mathrm{e}^{2\varphi\mathrm{i}})}{\mathrm{e}^{2\varphi\mathrm{i}}} \mathrm{i}\mathrm{e}^{2\varphi\mathrm{i}}\,\mathrm{d}\varphi = 2\pi\mathrm{i},$$

$$\int_0^2 f(x)\,\mathrm{d}x \text{ with } f(x) = (1, x, x^2, x^3)^T, \quad \int_{-1}^{2} x\mathrm{e}^x\,\mathrm{d}x = \left[(x-1)\mathrm{e}^x\right]_{-1}^{2}$$

- Make your results reproducible (`README.md`, `Project.toml`, ...)