Git tutorial

Dmitry Petrov

May 14
2018

Overview

- Why git
- How git
- Git gud

Why use Git and GitHub

— Allows to save and track changes in your code

— Useful tool for collaboration: issues, code reviews
(i.e. https://github.com/scikit-learn/scikit-learn/issues)

— Great for releasing your code and getting feedback for it

— Useful for teaching (i.e. http://cs231n.github.io/)

— Socializing (personal page and project following)

— Integration and apps (i.e. ZenHub and Slack integrations)

How to git — basics

Basic operations in Git are:

1. Init (init repo)
2. Add (add files)
3. Commit (commit)
4. Pull (get data from repo)
5. Push (push changes to repo)

Advanced Git operations are:

1. Branching
2. Merging
3. Rebasing

Useful link: https://www.quora.com/What-is-git-and-why-should-I-use-it

How to git — hands-on

— We will go through https://try.github.io/ (15 min)

— Shameless promo https://github.com/USC-IGC

Git gud — some links

— Git best practices (https://github.com/holidayextras/culture/blob/master/git-best-practices.md)

— GitHub student pack (https://education.github.com/pack)

— Tutorial on branching (http://nvie.com/posts/a-successful-git-branching-model/)

— You may consider contributing to open source in your free time (example of guidlines https://github.com/nipy/dipy/blob/master/CONTRIBUTING.md)

Git tutorial — conclusion

— Version control (i.e. Git) and its infrastructure (i.e. GitHub) is a useful tool to master for a modern computational scientist

— It helps in teaching, reproducible results sharing and socializing

— Cons: open-source contributions are very time consuming and as of now, most scientific KPI doesn’t take them into account (but it changes slowly)

Thank you!

✿*∗˵╰༼✪‿✪༽╯˵∗*✿