

'Git'-ing better: Collaborating on your research with version control and GitHub

Daiva & Luke

2015-04-17

Welcome back!

Purpose:

To review Git in more detail, give you more practice exercises that mimic what you may do in your day-to-day research activities, and show you how GitHub can be used for collaboration.

Caveat: We aren't here to teach statistics

Need help with stats? Use these resources!

- U of T Statistical Consulting Services ([click here](#))
- <http://www.stackoverflow.com>
- <http://stats.stackexchange.com>

Notes and help during this workshop

Go to this website:

<https://etherpad.mozilla.org/camWorkshops>

Version control refresher

Version control refresher

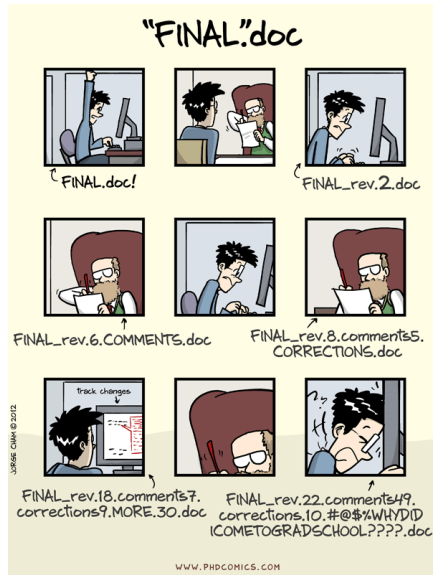


Figure 1:

What is version control¹ (VC)

- Keeps a history of everything done in a folder
- Like a big track changes or logbook (basic science)
- Can revert to previous change, try new things out
- Don't have to worry about losing what you wrote!

¹ See the Git website ([click here](#)) for more detail.

What is version control¹ (VC)

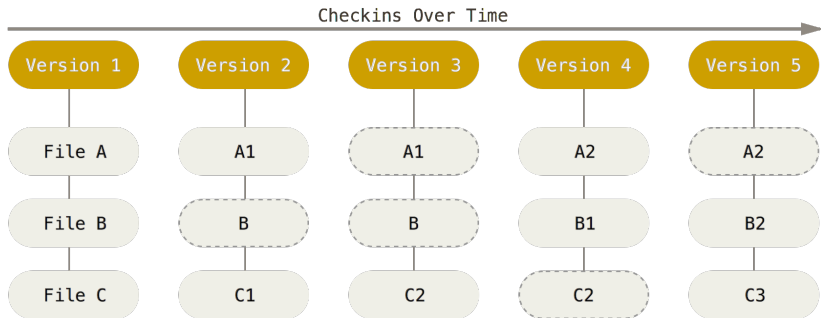
- Keeps a history of everything done in a folder
- Like a big track changes or logbook (basic science)
- Can revert to previous change, try new things out
- Don't have to worry about losing what you wrote!

Importance of VC!

- Future of journals and retractions
- Requests for data and code
- Transparency, scientific rigor
- Protect against accusations of fraud

¹ See the Git website ([click here](#)) for more detail.

Visualization of VC²



²Taken from the Git site ([click here](#))

Using Git

Before we start... The command line is **not** something to be afraid of!! Open up the terminal (Mac/Linux) or Git Bash (Windows).

Using Git

Before we start... The command line is **not** something to be afraid of!! Open up the terminal (Mac/Linux) or Git Bash (Windows).

Open up your terminal (Mac or Linux) or Git-Bash (Windows).

```
git config --global user.name "Your Name"  
git config --global user.email "you@some.domain"  
git config --global color.ui "auto"  
git config --global core.editor "your_editor"  
git config --list
```

Download our GitHub repository

GitHub³ is a place to store your git repo for several reasons:

- ① As a backup
- ② To use across computers
- ③ To share with others

³Or <http://BitBucket.org>

Download our GitHub repository

GitHub³ is a place to store your git repo for several reasons:

- ① As a backup
- ② To use across computers
- ③ To share with others

In your terminal/Git-Bash, run:

```
cd ~  
git clone \  
    https://github.com/codeasmanuscript/gitWorkshop.git  
cd gitWorkshop/part2
```

³Or <http://BitBucket.org>

Download our GitHub repository

GitHub³ is a place to store your git repo for several reasons:

- ① As a backup
- ② To use across computers
- ③ To share with others

In your terminal/Git-Bash, run:

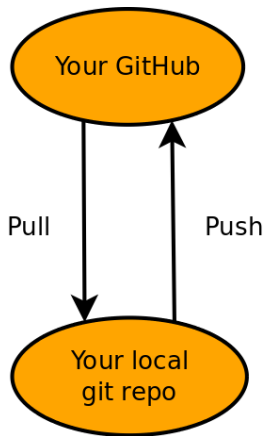
```
cd ~  
git clone \  
    https://github.com/codeasmanuscript/gitWorkshop.git  
cd gitWorkshop/part2
```

Check out the `cheatsheet.html` file.

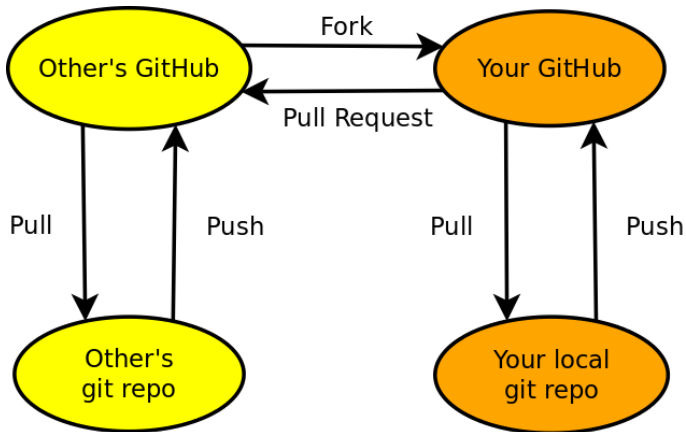
³Or <http://BitBucket.org>

Create a GitHub account

How GitHub works



How GitHub Forking (collaboration) works



Live coding – Create git repo from scratch and push to GitHub

Tip: Most of the time you don't really need to remember the code, just google search it! Or use our cheatsheet!! :D

Main Exercise – Collaborate on a project

Main Exercise – Collaborate on a project

- ① Create a git repository in a new folder or your own research folder
- ② Create a SAS (or R) file to practice in or add your own files
- ③ Run basic stats (means) or change a file in your own folder
- ④ Add and commit your changes to Git
- ⑤ Create a GitHub repo and push your changes to it*
- ⑥ Fork a friend's repo on GitHub (or someone fork yours) and clone onto your computer
- ⑦ Change a friend's files
- ⑧ Push the changes and make a pull request
- ⑨ Check out your own pull requests!