

STA2201H Methods of Applied Statistics II

Monica Alexander

Introduction to GitHub

Git

- ▶ Git is a version control system
- ▶ The system tracks changes you make to git repositories ('repos')
- ▶ Think of repos as folders
- ▶ In order for file versions to be tracked, they need to be **committed** to the git repo
- ▶ Think of committing as like saving, but with slightly more steps

GitHub

<https://github.com/>

- ▶ A hosting service for git repos
- ▶ You can sign up for free, and host an unlimited number of public or private repos
- ▶ You will be submitting lab exercises via GitHub, so you need to set up an account!

The simplest Git/GitHub workflow

- ▶ New repo on GitHub
- ▶ Clone onto local computer
- ▶ Do work on local computer
- ▶ Save
- ▶ Add and commit to git repo
- ▶ Push to GitHub (this means your new work will appear on the GitHub website)

If you are working on your own, on one computer, this is it!

Git/GitHub

- ▶ If you are working on a couple of different computers / servers, you may also need to **pull** from GitHub to update any new work done elsewhere
- ▶ Git is designed for collaborative work. More complicated workflows have branches, pull requests, merges
- ▶ More later (time permitting)

Git

For now, you just need to learn

- ▶ clone
- ▶ status
- ▶ add
- ▶ commit
- ▶ push
- ▶ pull

Steps on GitHub

1. Create an account on GitHub
2. Click the new repository green button
3. Name it something sensible (e.g. STA2201H-applied-stats), select private, select initialize with README, click create
4. Settings → Collaborators → Add MJAlexander as a collaborator

Steps to clone on your computer

Disclaimer: I use the terminal and will show you these steps.

You are welcome to use the GitHub Desktop:

<https://desktop.github.com/>

Steps to clone on your computer

1. open terminal window
2. cd into place you want to save the folder
3. `git clone`
`https://github.com/yourusername/yourrepo`

Steps to add work, commit and push

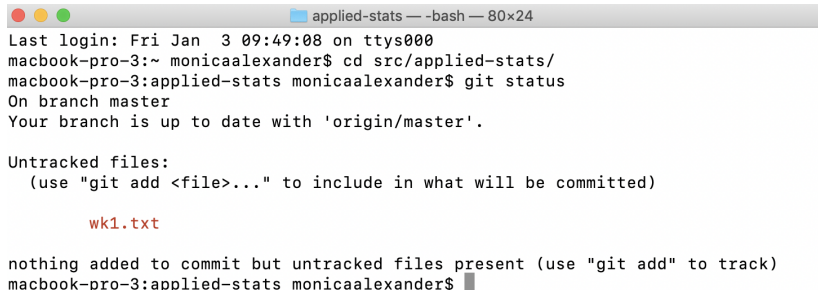
Save some work:

- ▶ Create a text file with your name and favorite thing to do in Toronto
- ▶ Save it as something like `wk1.txt` to the git repo folder (as you would normally save something)

Steps to add work, commit and push

Commit work

- ▶ open terminal window
- ▶ cd into git repo
- ▶ git status should show uncommitted work

A screenshot of a macOS terminal window. The title bar shows three colored window control buttons (red, yellow, green) on the left, followed by a folder icon and the text 'applied-stats — -bash — 80x24'. The terminal text shows a login session on 'Fri Jan 3 09:49:08 on ttys000'. The user 'monicaalexander' is at 'macbook-pro-3'. They navigate to 'src/applied-stats/' and run 'git status'. The output shows they are on the 'master' branch and up to date with 'origin/master'. It then lists 'Untracked files:' as 'wk1.txt' (in red text) and provides instructions to use 'git add' to track it. The prompt ends with 'macbook-pro-3:applied-stats monicaalexander\$' and a cursor.

```
Last login: Fri Jan  3 09:49:08 on ttys000
macbook-pro-3:~ monicaalexander$ cd src/applied-stats/
macbook-pro-3:applied-stats monicaalexander$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        wk1.txt

nothing added to commit but untracked files present (use "git add" to track)
macbook-pro-3:applied-stats monicaalexander$
```

Steps to add work, commit and push

- ▶ `git add wk1.txt` adds the file to staging area
- ▶ do a `git status` again to see what's going on

```
macbook-pro-3:applied-stats monicaalexander$ git add wk1.txt
```

```
macbook-pro-3:applied-stats monicaalexander$ git status
```

```
On branch master
```

```
Your branch is up to date with 'origin/master'.
```

```
Changes to be committed:
```

```
(use "git reset HEAD <file>..." to unstage)
```

```
    new file:   wk1.txt
```

```
macbook-pro-3:applied-stats monicaalexander$ █
```

Steps to add work, commit and push

- ▶ `git commit -m "adding wk1 file"` commits the file (`-m` gives the option of a message)
- ▶ do a `git status` again to see what's going on

```
[macbook-pro-3:applied-stats monicaalexander$ git commit -m "adding wk1 file"
[master 96c178f] adding wk1 file
1 file changed, 1 insertion(+)
create mode 100644 wk1.txt
[macbook-pro-3:applied-stats monicaalexander$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
macbook-pro-3:applied-stats monicaalexander$ █
```

Steps to add work, commit and push

Push to Github

- ▶ `git push`

```
macbook-pro-3:applied-stats monicaalexander$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 335 bytes | 335.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/MJAlexander/applied-stats
   8488fdc..96c178f  master -> master
macbook-pro-3:applied-stats monicaalexander$ █
```

- ▶ Now check your repo on GitHub to check the new work is there

This week's lab assessment

- ▶ Make a repo and add me as a collaborator
- ▶ Add a text file with your name and what you recommend doing in Toronto
- ▶ Push changes to GitHub