Applied Statistical Analysis I

GitHub workflow

Elena Karagianni, PhD Candidate
karagiae@tcd.ie
Dr Redmond Scales, Teaching Fellow
rscales@tcd.ie

📅 September 17, 2025

Department of Political Science, Trinity College Dublin

Today's class

- Resolve R/Python set-up issues
- Get familiar with GitHub
- Mock Problem Set upload

Version Control and Git

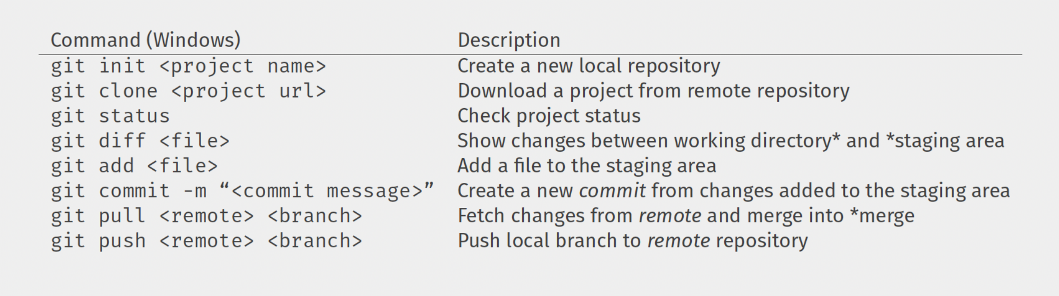
- Version control systems (VCSs) allow automatic tracking of changes in files and collaboration.
- Git is one of several major version control systems (VCSs, see also Mercurial, Subversion).
- GitHub is an online hosting platform for projects that use Git for version control.

Installing Git on Mac: Homebrew

- First open the terminal (command + space 'terminal')
- Type 'git –version'
- If version number appears you're all good!
- If git is not installed go to **https://git-scm.com/downloads**
- Install **Homebrew** (paste command to terminal)
- Press enter and enter in mac password
- Will be promted to install x-code tools - press 'enter'
- Run the promted commands in as instructed
- Now install git - type $ `brew install git`
- check the git version: git –version

Some useful Git commands

For a more detailed cheatsheet: Git Cheatsheet



Command (Windows)	Description
`git init <project name>`	Create a new local repository
`git clone <project url>`	Download a project from remote repository
`git status`	Check project status
`git diff <file>`	Show changes between working directory* and *staging area
`git add <file>`	Add a file to the staging area
`git commit -m “<commit message>”`	Create a new *commit* from changes added to the staging area
`git pull <remote> <branch>`	Fetch changes from *remote* and merge into *merge
`git push <remote> <branch>`	Push local branch to *remote* repository

Let’s clear up some things

- Git and GitHub sound similar but are different things:
    - Git is the tool that makes version history on your computer.
    - GitHub is like Google Drive; it lets you sync and share those histories online.

Local Git Repository

- Let's create a test local directory together.
- Open your CLI/Terminal
- Type `mkdir test`
- Go into the newly created directory by typing `cd test`
- To make Git track changes run `git init` command **in** this directory.
- Congratulations! You now have a test local repository!

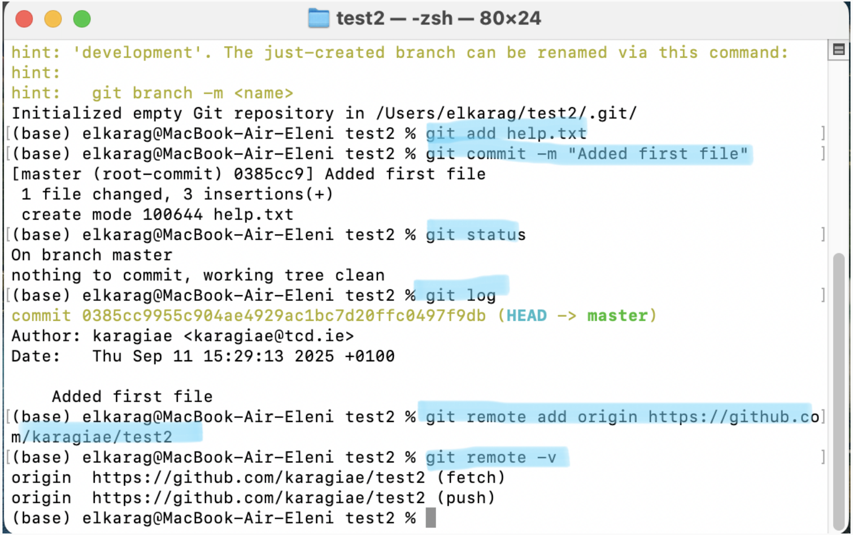
Making a commit: Creation to Staging

- Let’s upload a simple .txt file.
- Open your text editor of choice (Notepad, Sublime Text, Atom, Visual Studio Code...)
- Create a file called ’text.txt’ in your local ’test’ repository.
- Type whatever you like in this file
- Add this file to your staging area (make Git aware of its existence) by running `git add test.txt` command

Making a commit: Staging to Committing

- Commit this file to your local repository by running `git commit -m "Added first file"`
- Note that all files that were added at the previous stage with `git add <file>` would be committed.
- Check status of your repository by running `git status` (it should say ‘nothing to commit, working tree clean’)
- Check history of your repository by running `git log` and make sure that you see your commit

A screenshot:



```
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/elkarag/test2/.git/
(base) elkarag@MacBook-Air-Eleni test2 % git add help.txt
(base) elkarag@MacBook-Air-Eleni test2 % git commit -m "Added first file"
[master (root-commit) 0385cc9] Added first file
 1 file changed, 3 insertions(+)
 create mode 100644 help.txt
(base) elkarag@MacBook-Air-Eleni test2 % git status
On branch master
nothing to commit, working tree clean
(base) elkarag@MacBook-Air-Eleni test2 % git log
commit 0385cc9955c904ae4929ac1bc7d20ffc0497f9db (HEAD -> master)
Author: karagiae <karagiae@tcd.ie>
Date:   Thu Sep 11 15:29:13 2025 +0100

    Added first file
(base) elkarag@MacBook-Air-Eleni test2 % git remote add origin https://github.co
m/karagiae/test2
(base) elkarag@MacBook-Air-Eleni test2 % git remote -v
origin  https://github.com/karagiae/test2 (fetch)
origin  https://github.com/karagiae/test2 (push)
(base) elkarag@MacBook-Air-Eleni test2 % 
```

GitHub

- Hosting platform for projects that rely on Git for version control
- Bought by Microsoft in 2018
- Provides extensive tools for collaborative development and search functionality
- Helpful for troubleshooting more narrow problems (check GitHub issues of the package/library that you have a problem with)
- GitHub is far from the only platform for hosting Git projects

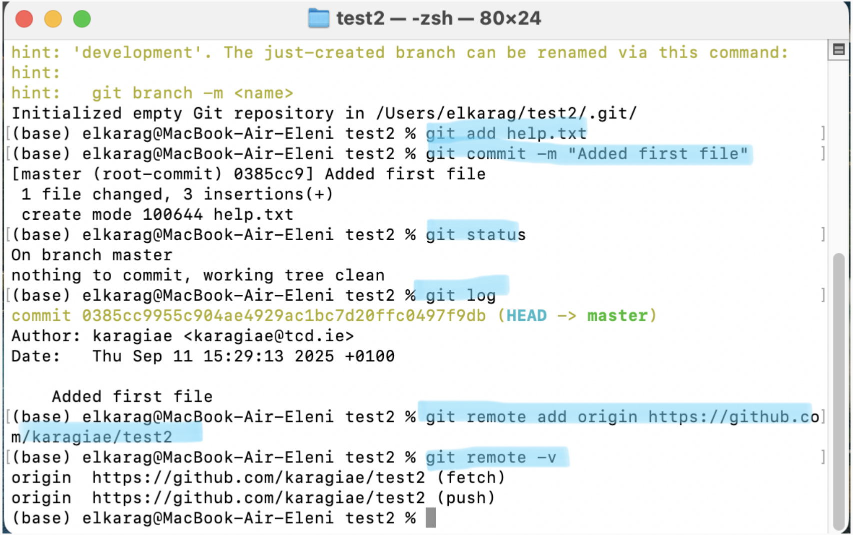
Creating remote repository on GitHub

- Register and login into your account on GitHub
- Create a new GitHub repository (choose private)
- You should see a similar page with the project URL of the form: **https://github.com/¡username¿/¡repository_name¿.git**

Synchronising local git repository with GitHub

- Go to your local Git repository (the one created in the previous step)
- Add link from your local Git repository to remote repository on GitHub by running: `git remote add origin <project_url>` where:
    - `git remote add` is the command
    - `origin` is the name given to this link (¡remote¿) and
    - `<project_url>` is the URL of the repository on GitHub
- Check the status of links between your local Git repository and remotes by running `git remote -v` where:
    - `git remote` is the command and
    - `-v` is the argument ‘verbose’

A screenshot:



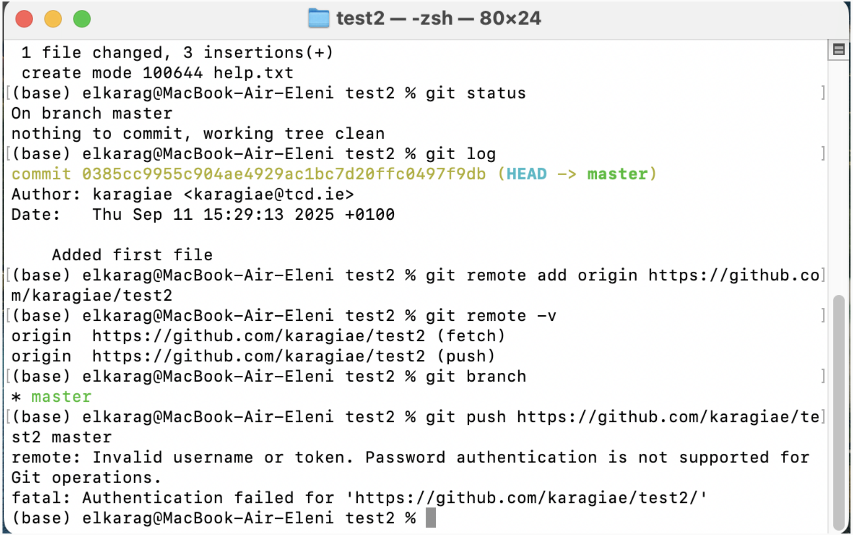
```
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/elkarag/test2/.git/
[(base) elkarag@MacBook-Air-Eleni test2 % git add help.txt                       ]
[(base) elkarag@MacBook-Air-Eleni test2 % git commit -m "Added first file"       ]
[master (root-commit) 0385cc9] Added first file
 1 file changed, 3 insertions(+)
 create mode 100644 help.txt
[(base) elkarag@MacBook-Air-Eleni test2 % git status                             ]
On branch master
nothing to commit, working tree clean
[(base) elkarag@MacBook-Air-Eleni test2 % git log                                ]
commit 0385cc9955c904ae4929ac1bc7d20ffc0497f9db (HEAD -> master)
Author: karagiae <karagiae@tcd.ie>
Date:   Thu Sep 11 15:29:13 2025 +0100

    Added first file
[(base) elkarag@MacBook-Air-Eleni test2 % git remote add origin https://github.co]
m/karagiae/test2
[(base) elkarag@MacBook-Air-Eleni test2 % git remote -v                          ]
origin  https://github.com/karagiae/test2 (fetch)
origin  https://github.com/karagiae/test2 (push)
(base) elkarag@MacBook-Air-Eleni test2 % 
```

Pushing local Git changes to GitHub

- Your local Git repo is now linked to remote repo hosted on GitHub
- Let's bring the changes made locally to the remote repository
- We will use the `git push` command for that
- One last thing to check before doing so is which branch we're on
- Run `git branch` to see name of branch you're on ('master' or 'main')
- Finally, run `git push <remote><branch>` (e.g., `git push origin master` where:
    - `git push` is the command
    - `<remote>` is the name of the remote link, and
    - `<branch>` is the name of the branch
- Visit your GitHub repository to check that your commit is reflected there

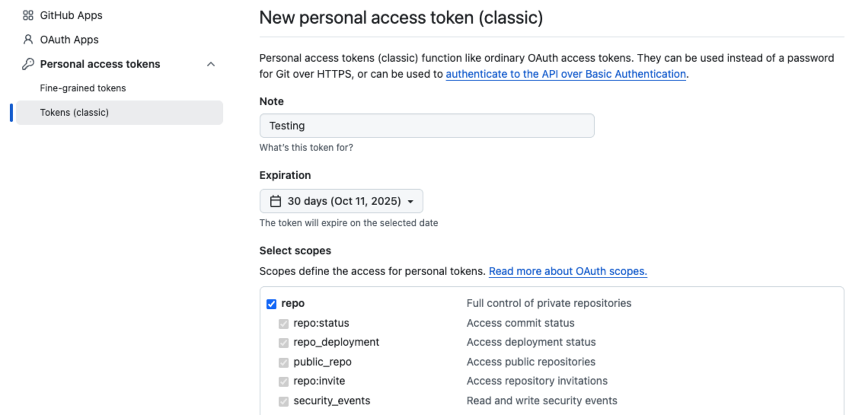
A screenshot:



```
 1 file changed, 3 insertions(+)
 create mode 100644 help.txt
[(base) elkarag@MacBook-Air-Eleni test2 % git status                          ]
On branch master
nothing to commit, working tree clean
[(base) elkarag@MacBook-Air-Eleni test2 % git log                             ]
commit 0385cc9955c904ae4929ac1bc7d20ffc0497f9db (HEAD -> master)
Author: karagiae <karagiae@tcd.ie>
Date:   Thu Sep 11 15:29:13 2025 +0100

    Added first file
[(base) elkarag@MacBook-Air-Eleni test2 % git remote add origin https://github.co]
m/karagiae/test2
[(base) elkarag@MacBook-Air-Eleni test2 % git remote -v                       ]
origin  https://github.com/karagiae/test2 (fetch)
origin  https://github.com/karagiae/test2 (push)
[(base) elkarag@MacBook-Air-Eleni test2 % git branch                          ]
* master
[(base) elkarag@MacBook-Air-Eleni test2 % git push https://github.com/karagiae/te]
st2 master
remote: Invalid username or token. Password authentication is not supported for
Git operations.
fatal: Authentication failed for 'https://github.com/karagiae/test2/'
(base) elkarag@MacBook-Air-Eleni test2 % 
```

Token access



New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

Testing

What's this token for?

Expiration

30 days (Oct 11, 2025)

The token will expire on the selected date

Select scopes

Scopes define the access for personal tokens. Read more about OAuth scopes.

☑ **repo**	Full control of private repositories
☑ repo:status	Access commit status
☑ repo_deployment	Access deployment status
☑ public_repo	Access public repositories
☑ repo:invite	Access repository invitations
☑ security_events	Read and write security events

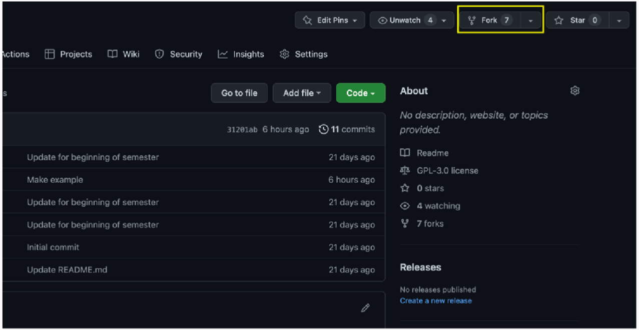
After getting your token

- Type `git push -u origin`
- If this did not work, type this: `git push --set upstream origin master/main`
- You will be asked to put your username
- A lock symbol will show up. Copy your token, **click** on the terminal window, paste. Nothing will show up, don't worry - just press Enter.

Cloning module repository

- All module materials are hosted on GitHub in this repo.
- You can clone this repository to your local machine by running:
  `git clone https://github.com/ASDS-TCD/StatsI_2025`
- This will create a folder called ‘https://github.com/ASDS-TCD/StatsI_2025’ within the directory where you ran this command.
- To keep up to date with change in the remote repository you can run `git pull origin main` where:
  - ‘origin’ is the remote address of the repository
  - ‘main’ is the name of the branch

Using GitHub and GitHub Desktop (Easier)



31201ab 6 hours ago 11 commits

Update for beginning of semester 21 days ago

Make example 6 hours ago

Update for beginning of semester 21 days ago

Update for beginning of semester 21 days ago

Initial commit 21 days ago

Update README.md 21 days ago

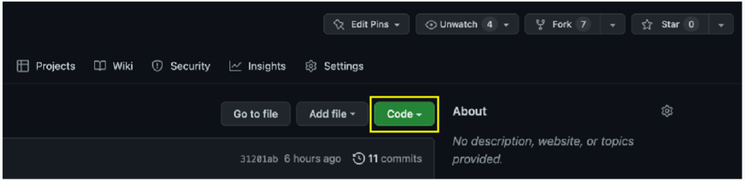
No description, website, or topics provided.

No releases published

- Set up a GitHub account at **https://github.com/join**
- You need to *fork* the *parent* repository (make your own copy of the folder)
- To do this go to the main folder **StatsI**$_2$025*andclick***fork**

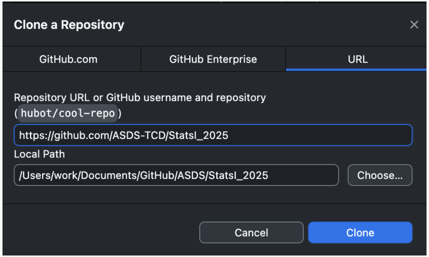
- **Now you will have your own repository that you control thats linked to the**

GitHub Desktop: Link online repository to local repository



- With the desktop version downloaded, let’s now link your online repository with your local repository.
- You’ll want to do this by click the green ‘Code’ button, and then selecting to open it in GitHub desktop

GitHub Desktop: Link online repository to local repository



Repository URL or GitHub username and repository
(`hubot/cool-repo`)

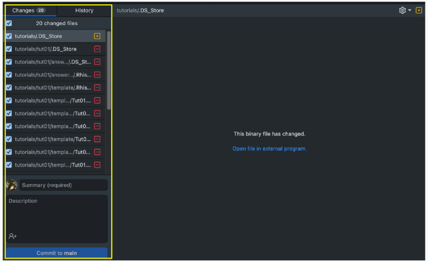
https://github.com/ASDS-TCD/StatsI_2025

Local Path

/Users/work/Documents/GitHub/ASDS/StatsI_2025

- You will be transfered over to GitHub Desktop
- Make sure that your folder is placed in Documents - GitHub - StatsI_2025
- If asked, do not contribute to parent directory
- Now you have a completely up-to-date version of my files from the parent directory, and a synchronized folder between your local files and your online repository.

Posting Homework to GitHub



☑ 20 changed files

- ☑ tutorials/.DS_Store
- ☑ tutorials/tut01/.DS_Store
- ☑ tutorials/tut01/answ.../.DS_St...
- ☑ tutorials/tut01/answer.../.Rhis...
- ☑ tutorials/tut01/template/.Rhis...
- ☑ tutorials/tut01/templ.../Tut01....
- ☑ tutorials/tut01/templa.../Tut0...
- ☑ tutorials/tut01/templa.../Tut0...
- ☑ tutorials/tut01/template/Tut0...
- ☑ tutorials/tut01/templa.../Tut0...
- ☑ tutorials/tut01/templ.../Tut01....

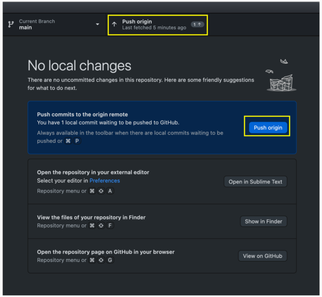
tutorials/.DS_Store

This binary file has changed.

Open file in external program.

- Let's say you finish your homework, and you now want to post your answers to GitHub. Assuming that you're editing your files in your local GitHub folder, GitHub desktop will automatically recognize changes that you have made
- All you need to do is write a summary. Once you write something, anything, to remind yourself of what changes you've made (maybe like 'Post PS01'), then you'll 'commit to main

GitHub Desktop: Pushing Changes



No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Push commits to the origin remote
You have 1 local commit waiting to be pushed to GitHub.
Always available in the toolbar when there are local commits waiting to be pushed or ⌘ P

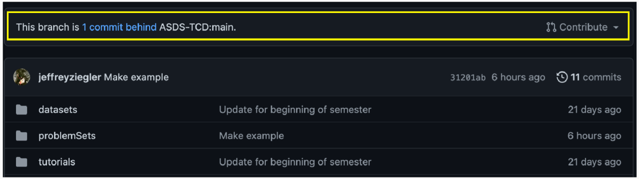
Open the repository in your external editor
Select your editor in Preferences
Repository menu or ⌘ ⇧ A

View the files of your repository in Finder
Repository menu or ⌘ ⇧ F

Open the repository page on GitHub in your browser
Repository menu or ⌘ ⇧ G

- Commit puts your changes in Git purgatory
- You need to finish by ’pushing to origin’
- This will sync your local files with your online files

GitHub: Fetching new files



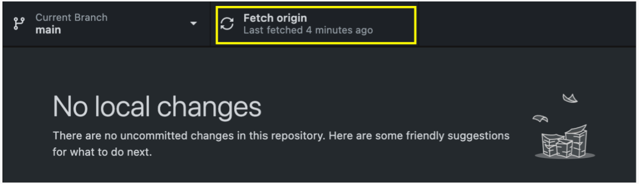
This branch is 1 commit behind ASDS-TCD:main.

jeffreyziegler Make example 31201ab 6 hours ago 11 commits

datasets	Update for beginning of semester	21 days ago
problemSets	Make example	6 hours ago
tutorials	Update for beginning of semester	21 days ago

- You will need to get updated homework from the parent file
- You will see that I have files you don't have, so you will need to sync
- We then use GitHub desktop to sync your online repository with your local files

GitHub: Fetching new files from origin

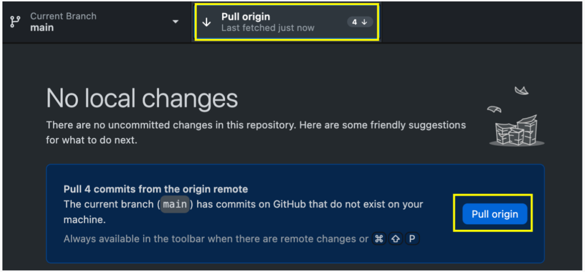


No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

- To sync you need to **fetch origin**
- You will see there are files that are different

GitHub: Pulling to origin



No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Pull 4 commits from the origin remote

The current branch (`main`) has commits on GitHub that do not exist on your machine.

Always available in the toolbar when there are remote changes or ⌘ ⇧ P

- Once you 'fetch' from origin, it will notice that there are files (in this instance, 4) that are different. You just need to click 'pull origin', which will add the missing files from your online repository to your local folder.
- You can convince yourself that it worked by checking your 'Documents' folder where your local GitHub files are located