

Using Git and Github with R

Bonnie McLean

07/06/2021

Git and github for version control - overview

- Git is the software that does version control for you.
- Github is the website where the different versions of your projects *get stored and saved*.
- Projects are stored online in github in *repositories*, which are linked to local folders on your computer (including R project folders). You can “push” changes from your local computer to the online git repository, to save your work at different stages.

How it works/Git language

You make your repository on github, and then when you want to add stuff to the repository/update your files at different stages in your work, you do the following:

1. **“Stage”** some files for commit: this is where you select which files you want to save a record of *in their current state*.
2. Make the **“commit”**, with a *commit message*. When you make a commit, you are saving a record of your selected files in their current state. You should accompany your commits with a message, describing the current state of the file(s). E.g. ‘just got X working’ or ‘just imported some new data’, etc.
3. **“Push”** your changes to github: this is where you upload your commits to github, where they will be saved permanently. If you don’t do this step, and your computer breaks, you won’t be able to access the latest versions of your files, but if you *pushed* your changes before your computer broke, you can get all your work back from github.

You can also *pull* changes from github. This is useful when you are collaborating with multiple people on a single project, as you can *pull* the changes that they have made to the project to your own computer.

Before you start

Install Git

1. Visit the git website and download the relevant version of git for your operating system.
2. Click on the downloaded file to install it. You can accept most of the defaults, but when you get to the bit about ‘Adjusting the name of the initial branch in new repositories’, I want you to select the option to ‘Override the default branch name for new repositories’, and the specified name should be “main”. Github already uses “main” for new repositories, so it’s just more convenient if your git does as well.

Make a github account

Make an account on github

Set your git identity

On some people's computers, this step wasn't necessary, but for others it was. So, if you have trouble in the later steps, come back and do this then try again. Or just do this anyway to be on the safe side.

When you installed git, it should also have installed a program called `gitbash`. If you have gitbash, open that up. If not, you can also just open up Command Prompt (on Windows) or Terminal (on a mac). Then type the following, pressing enter after each line:

1. `git config --global user.name "YOURNAME"`
2. `git config --global user.email "your-email"`

Use the same email address you used to make your github account.

Now you should be ready to use git with R!

Using git with R

The easiest way is if you make a github repository first, then make an R project from that repository (rather than the other way around).

1. Log into your github account on github.com, and click the + symbol on the top right, then choose 'Make a new repository'.
2. Make the github repository. Choose a name for the repository, make it "private" (if you don't want other people to be able to see your work; you can make it public later when the work is finished if you want, by going into the settings for the repository), and check the box to initialise it with a README file.

Github will make the repository and take you to the page with the repository. Copy the url for this webpage with the repository.

Then, start up R.

1. In R, go File > New Project > Version Control > Git.
2. In the Repository URL box, paste the URL to your github repo.
3. It will make a new project. Lets add a markdown notebook to this project, called finalreport.Rmd.
4. Then, in your R sidepanel on the top right where you can see tabs like 'Environment' 'History' etc., you should also now see a tab 'Git'. Click on that tab.
5. You will see a bunch of files. These are files that you have changed, where the changes haven't been uploaded to github yet. Tick the box next to each file to "stage" it for a commit. Then press the button at the top that says "Commit".
6. A new window will open up. And you will see a box where you can write a "Commit message". Write a message like "first upload of files", then press 'Commit'.
7. You will see git do some stuff. Once it's done, just close the git window.
8. You will see a message that "Your branch is ahead of origin/main by 1 commit". This means although we've staged our changes for commit, we haven't added these changes to github yet. To do this, press the button in the top right that says "Push".
9. R will upload your changes to the repository on github! You can check by going back to your github repository; you should now see your files there, with your first commit message.

Great, now you know how to save your report at different stages when you are working on the final assignment!