Introduction to Git and GitHub with R

2025-04-16

Introduction

This tutorial introduces Git and GitHub and walks you through setting them up for R projects. You'll learn how to track changes to your code, and publish your work online.

What is Git and GitHub?

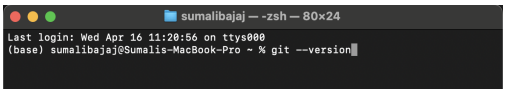
Git is a free and open-source version control system. It allows you to track changes in your code and collaborate with others.

GitHub is a web-based platform that hosts your Git repositories online. It provides tools for collaboration, sharing, and working with others on coding projects.

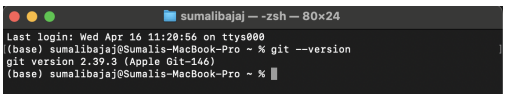
Step 1: Installing Git

1. Visit the official Git website: https://git-scm.com
2. Download the installer for your operating system (Windows, macOS, or Linux)
3. Run the installer and follow the default prompts.
4. After installation, open the terminal or command prompt and run:

```
git --version
```



5. You should see output like:



Step 2: Creating a GitHub account

1. Visit https://github.com
2. Click Sign up.
3. Enter your email, password, and username.
4. Verify your email address.

Step 3: Installing R and RStudio

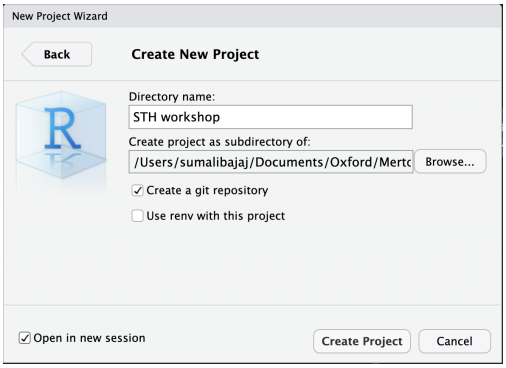
1. Download R from: https://cran.r-project.org
2. Install R following the prompts for your operating system.
3. Download RStudio Desktop (free version) from: https://posit.co/download/rstudio-desktop/
4. Install RStudio.
5. Open RStudio to ensure it launches correctly.

Step 4: Installing GitHub desktop GUI

1. Download GitHub Desktop from: https://desktop.github.com
2. Install and open the application.
3. Sign in with your GitHub credentials.
4. You can now manage repositories with a graphical interface instead of using the terminal.

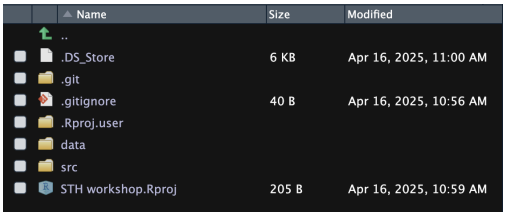
Step 5: Creating an R project locally and initializing a Git repository

1. Open RStudio
2. Go to: File > New Project > New Directory > New Project
3. Choose a project name and select the folder location.
4. Make sure to check the box: "Create a git repository"
5. Click Create Project
6. This will create a new folder with Git tracking enabled.



Step 6: Creating folders

1. Go to your folder where you have created the R project
2. Add two folders `data` and `src`. Here you will add your datasets and code scripts
3. Check that these show up in your R studio, under the files section



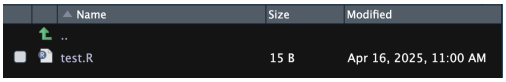
Step 7: Creating a test R script

In RStudio, go to File > New File > R Script

Add the following code:

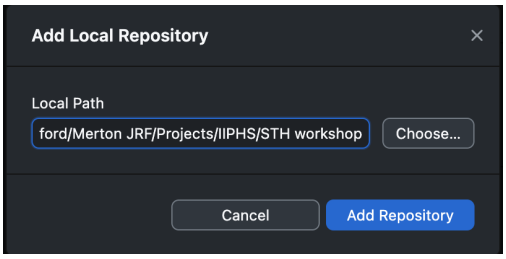
```
print("Hello GitHub!")
```

Save the file inside your R project directory as `test.R`

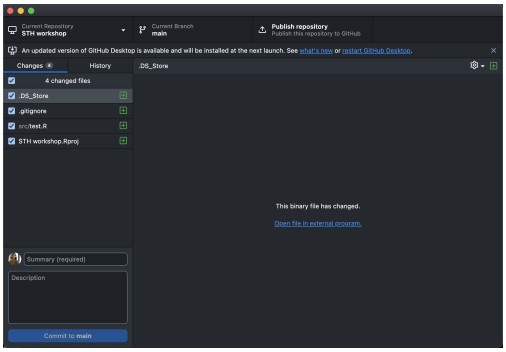


Step 8: *Committing* changes using GitHub desktop

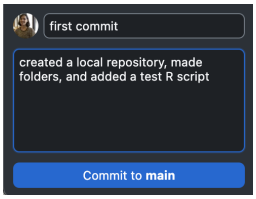
1. Open GitHub desktop
2. Click File > Add Local Repository



3. Navigate to your R project folder and select it
4. You will see your script (`test.R`) listed as a changed file, along with some additional changes



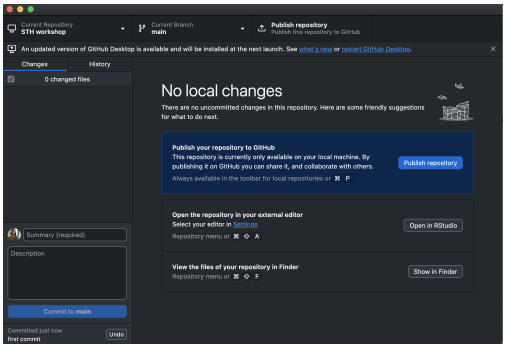
5. Type a commit message like:



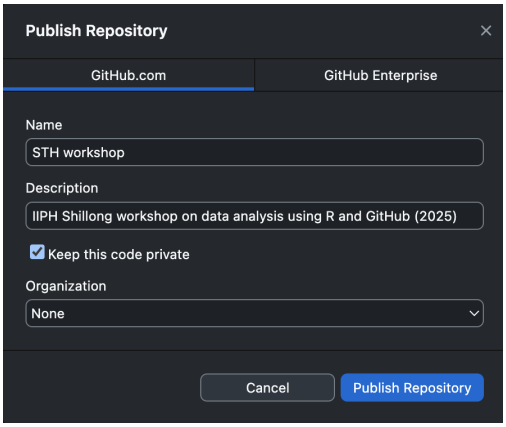
6. Click Commit to main

Step 9: *Pushing* changes to GitHub using GitHub desktop

1. After committing, click the Publish repository button in the top-right corner



2. Choose the repository name (or use the same name as your project)

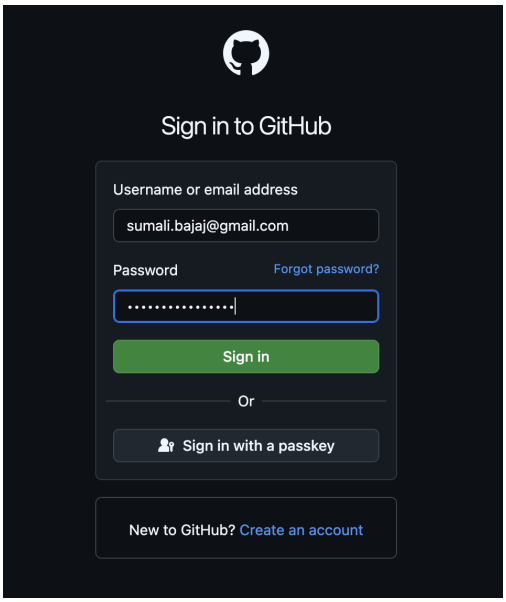


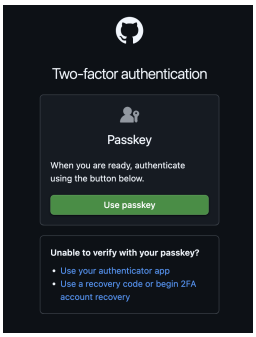
3. Choose whether to make it public or private
4. Click Publish Repository

Your project is now on GitHub!

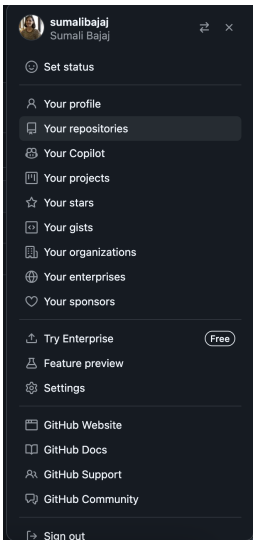
Step 10: Check if your repository exists on GitHub

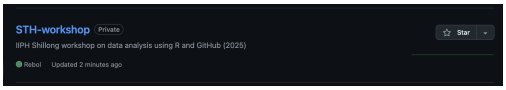
1. Visit https://github.com/YOUR_USERNAME



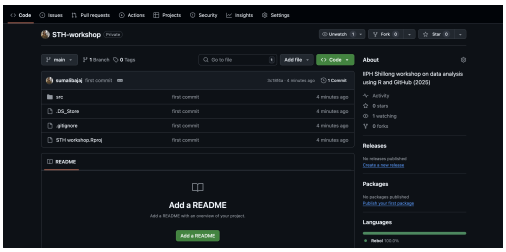


2. You should see your repository listed



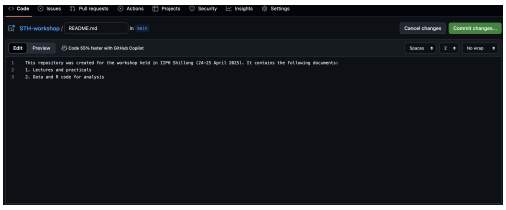


3. Click on it to view the contents and confirm that your script was pushed successfully

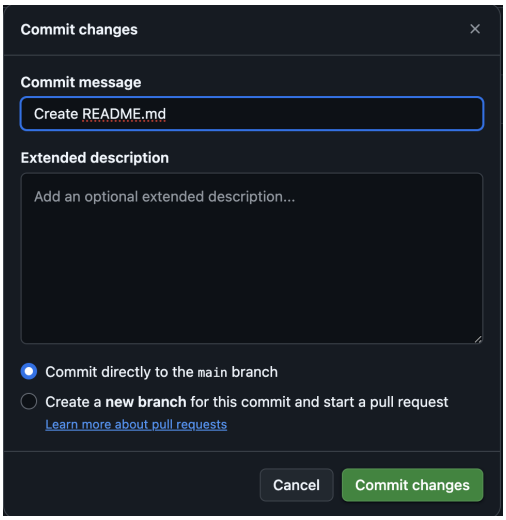


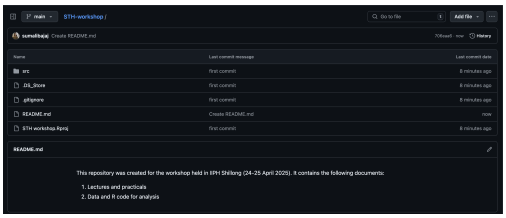
Step 11: Add a README on GitHub

1. Click on Add a README
2. Add a simple description, for example:



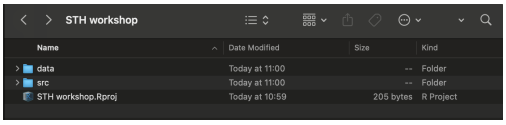
3. Scroll down and click Commit new file

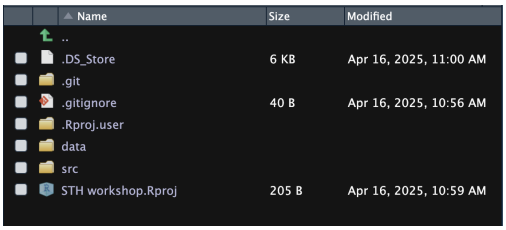




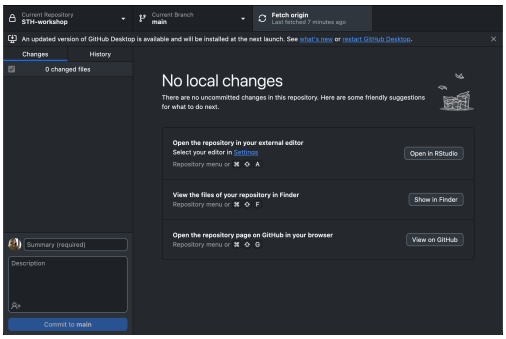
Step 12: *Pull* changes to your local computer

1. Go to your folder on your computer and R studio, and confirm that a README.md file doesn't already exist

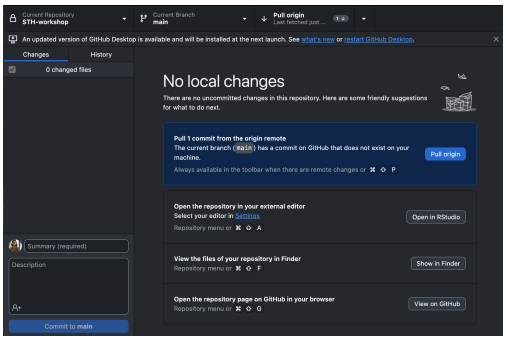




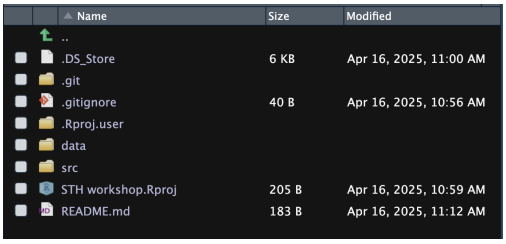
2. Open GitHub Desktop
3. Make sure your project is selected (look in the top-left corner)
4. Click the Fetch origin button



5. Then click Pull origin



6. The new README.md file will now appear in your local R project folder.



Practical exercises

(I) Branching and merging using GitHub desktop

Branching lets you make changes without affecting your main code. This is great for trying out features or testing ideas, and then *merging* with your main code when ready.

A. Create a new branch

1. Open GitHub Desktop
2. In the top toolbar, click on Current Branch > New Branch
3. Name your branch: feature-hello-update
4. Click Create Branch

You are now working in a new branch.

B. Modify a script

1. In RStudio, open `test.R`, and update it:

```
print("Hello from my new branch!")
```

2. Save the file.

C. Commit the change

1. Go back to GitHub Desktop
2. Stage and commit the changes with message like: "Updated message in feature branch"

D. Merge branch into main

1. In GitHub Desktop, switch back to main (Current Branch > main)
2. Click Branch > Merge into Current Branch
3. Choose feature-hello-update
4. Click Merge feature-hello-update into main
5. Click Push origin to update GitHub

(II) Cloning an existing repository with teaching materials

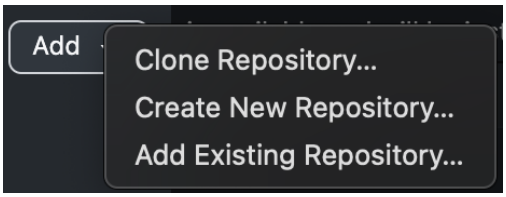
In this exercise, you'll learn how to *clone* a repository from GitHub to your local computer. This is how you'll access the teaching materials provided for this workshop.

Note: The actual data file will be sent to you by email. The repository contains all other files exercises, solutions, and shapefiles for mapping.

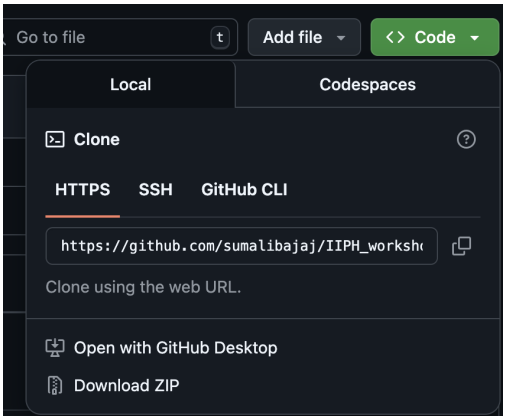
1. Visit the teaching repository online: https://github.com/sumalibajaj/IIPH_workshop_material

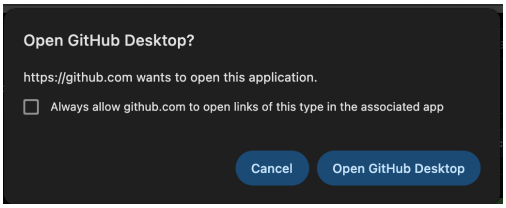
You should see a page listing all the contents of the repository.

2. On the repository page, click the green **Code** button near the top right

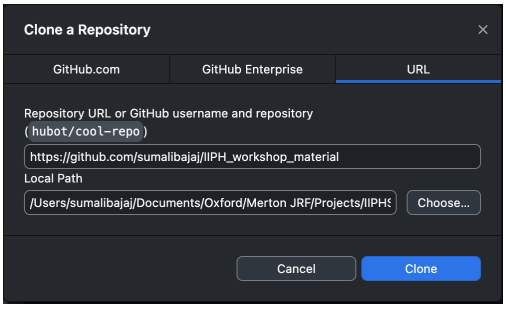


3. From the dropdown menu, click "Open with GitHub Desktop"





4. Choose the local path where you'd like the project saved and click "Clone"



This downloads all project files to your computer at the path you provided.

Important note about workflow

Now that you've cloned the teaching materials repository to access the files.

DO NOT try to push changes to the original teaching repository (you don't have permission).

What you should do

1. Copy the shapefiles and from the cloned `teaching-materials` repo to your own GitHub-linked project folder (the one you created earlier).
2. Use that folder to write your own R scripts and do the exercises.
3. Commit and push regularly to your own GitHub repository using GitHub Desktop.

The teaching repo is just for reference — your own work should live in your own project and repo!