

Lab 04: Scaffolding reproducible research

Sicheng Wang

Table of contents

1	Reproducible research	1
2	Project management (Taking Lab-04 Repository as an Example)	3
3	Assessment	7

1 Reproducible research

1.1 What is reproducible research?

- **Reproducible Research** In academic setting, reproducible research refers to work that can be replicated by other scholars following the steps outlined in the **Methodology** part, resulting in the same results as the original author. As outlined in [Recipe #4](#), the concept of reproducible research specifically refers to reproducible projects, which aims to mitigate the impact of varying computing environments by documenting every step, typically through files like a README file outlining the project's objectives and production steps, as well as a DESCRIPTION file detailing the computing environment and packages utilized. This approach is particularly valuable for collaborative work, as it allows others to test the reproducibility and applicability of the work across different settings.
- **Computing Environment and Project Structure** The computing environment includes hardware (see [Figure 1](#)), the operating system (see [Figure 2](#)), and apps (see [Figure 3](#)), whereas the project structure refers to the files and folders contained in our project.



Figure 1: Computer Hardware from [javaTpoint](#)



Figure 2: Operating System from [Welcome to CS](#)



Figure 3: Computer Hardware from [CODERUS](#)

- **Git and GitHub** Git is an app which download in individual' computer which could be used to track changes in your local repositories while Github is a web-based which allows individuals to share repositories. When connecting them by RStudio, the repository one made could upload to one's GitHub account to let others get it by cloning to their own computers or forking to their GitHub account. And people who cloning or forking others' repositories could also editing in their own computer or in their GitHub account which would not influence the original author's repository. They could use Git app to upload(push) to their own forking repository or replacing the original one if the original author permit that through pull request. Besides, the tracking function of Git makes the changes available to individuals by committing button. Under the repository in GitHub, one could also report issues which shows the interactive function of GitHub. Though the above functions, GitHub and Git help our project be more reducible.

1.2 Why it is important for modern science?

- **In academic**, research is interconnected; scholars build upon previous studies. Moreover, interactions among scholars worldwide are becoming increasingly common. Reproducible studies facilitate the review of others' work and enable scholars to offer more informed suggestions by validating previous research. Additionally, reproducible research can help prevent academic misconduct. For scholars collaborating with peers, reproducible research aids in tracking each others' edits and contributions.
- **For the general public involved in projects**, reproducible research simplifies collaboration, particularly in light of recent significant shifts in work patterns where remote work has become prevalent. With reproducible research, individuals can collaborate more efficiently regardless of their location, enabling them to work remotely with ease.

2 Project management (Taking Lab-04 Repository as an Example)

2.1 Forking

In GitHub, if you want to copy someone else's repository without downloading it to your local computer, you utilize the **fork** option located below the repository, as shown in Figure 4:



Figure 4: Fork

2.2 Cloning

When you choose to download repositories from GitHub, whether they are owned by you or shared by others, you can use the `clone` option, as depicted below in Figure 5:

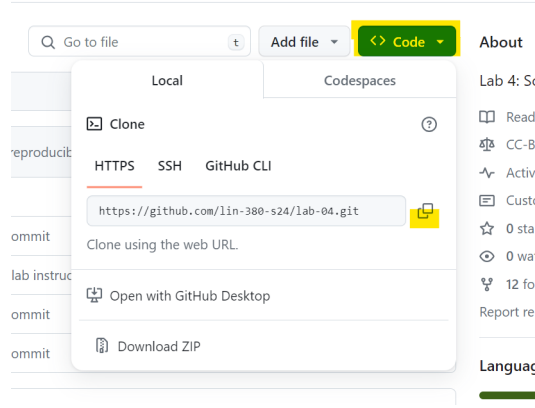


Figure 5: Clone

2.3 Editing

Then, we can make modifications to the cloned repository on our computer, such as adjusting file content, adding or deleting files, and performing other actions according to the aims of our project.

2.4 Committing

After making adjustments to your local repository, you may want to save your changes. While you can use **Render on Save** for immediate saving, for long-term projects, you might want to revert to earlier versions. In such cases, you would utilize the `commit` function, which helps track changes, especially during collaboration with others. This function is facilitated by Git. Below is where you can find `commit` when working with the project in RStudio, see Figure 6:

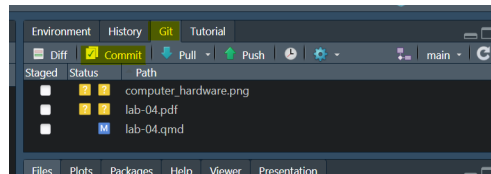


Figure 6: Commit

You can also provide a `commit message` to remind yourself of the changes made in one specific version, as shown in Figure 7:

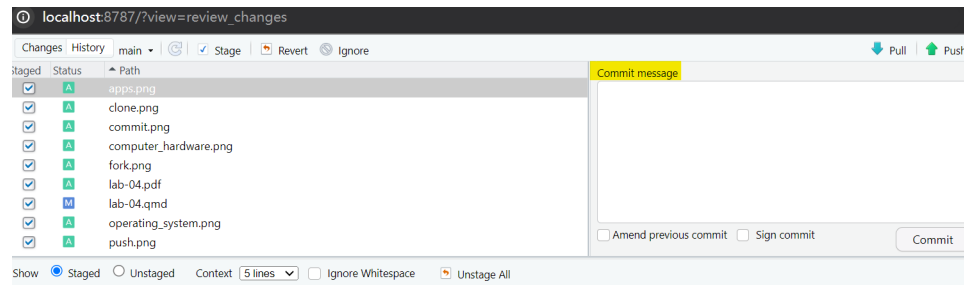


Figure 7: Commit Message

P.S.

- The symbol ? indicates that the file does not exist in the repository you cloned from GitHub.
- M indicates that the file exists in the original repository but has been modified by you.

2.5 Pushing

This step is for individuals who want to update the original repository on GitHub. This is particularly useful when you want to ensure that the repository you have shared with others is up-to-date, or when you are collaborating with others. The `push` button allows you to upload the revised local repository to your GitHub account. In RStudio's Git pane, you can find the `push` button, as shown in Figure 8:

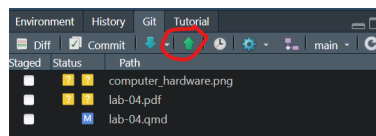


Figure 8: Push

2.6 Previous Methods **Scenario A**

See in Figure 9:

boardmix

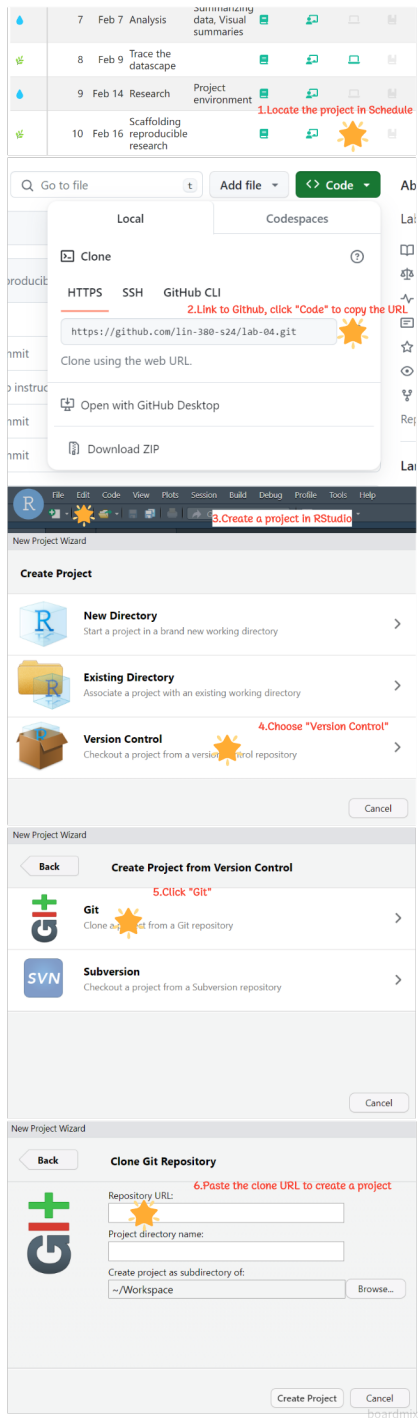


Figure 9: Scenario A

3 Assessment

3.1 What did you learn?

- Reproducible projects entail both the computing environment (hardware, operating system, and software) and the project structure (organization of files and folders within the project).
- To ensure reproducibility in our projects, it's essential to include basic information such as the project's purpose, the source of data, the packages installed in the project, and so on.
- A repository is a managed project, while a corpus is a collection of samples that may include diverse variables.
- Git and GitHub serve to backup and share projects, enhancing the reproducibility of our work. Git is an application downloaded onto a computer to track changes in local repositories, while GitHub is website-based for sharing remote repositories with others.
- Several features offered by Github and Git: clone (to copy the repository to the local computer), fork (to copy the repository to the GitHub account), commit (to track changes made on the local computer), push (to upload the revised local repository to the repository in the GitHub account), and pull request (to request replacing others' original repository on GitHub by pushing your revised local repository).

3.2 What did you find most/ least challenging?

I find [Lab-04](#) to be relatively straightforward compared to previous labs, although the most challenging aspect for me was pushing my revised local repository to my GitHub account. Initially, I encountered an error message stating **The requested URL returned error**. With assistance from the professor, I was able to resolve this issue by reconnecting my GitHub account with my Git account through RStudio. However, prior to today's class, I mistakenly believed that I needed to use **Git Bash** to change my Git username and set my email address, which left me confused. Now I understand that I can change my username through RStudio's terminal pane.

3.3 What resources did you consult?

- For question one - the inability to push my local repository to my GitHub account, I sought assistance from ChatGPT and the professor. ChatGPT suggested four potential issues: credential problems, authorization problems, operating system (OS) issues, and spelling errors. However, I was unsure which one I was experiencing, so I followed the

professor's advice to reconnect my Git account and GitHub account, which ultimately resolved the issue for me.

- For the second question - changing my Git username and adding an email address in Git, I attempted to find a solution by watching a video titled [How to Install and Configure Git and GitHub on Windows 11](#). However, the video used an app called `test.js` during the process, which I felt might not be suitable for my situation. As a result, I decided to wait until the professor covered this topic in today's class.

3.4 What more would you like to know about reproducible research and/ or Git and GitHub?

Yesterday, when I checked today's [Lab-04](#), I noticed something related to **creating issues** in the repository. However, when I forked the repository to my GitHub account, I couldn't find the **issues** tab, only the **pull requests** tab was visible. To gain clarity, I watched a video titled [Creating issues in GitHub Projects](#), hoping it would provide insight into locating the **issues** tab, even though it primarily focused on creating issues in GitHub Projects. Unfortunately, the video didn't address the issue directly because it directly went to this page, as shown in Figure 10. Then, I continued my search and found that **pull requests** are closely related to **issues** when I conducted a Google search, as shown in Figure 11. By clicking on the **pull requests** tab in my GitHub account, I accidentally stumbled upon the **issues** tab, as seen in Figure 12. Additionally, I was curious about some basic functions of GitHub, so I watched a tutorial titled [GitHub Tutorial](#). However, this tutorial primarily focused on downloading the GitHub desktop app to connect to a remote repository in the GitHub web interface, which differs from what we learned using Git. It also covered how to create a remote repository in GitHub, similar to [Scenario C](#). However, it didn't delve into functions such as discussions, pull requests, or issues.

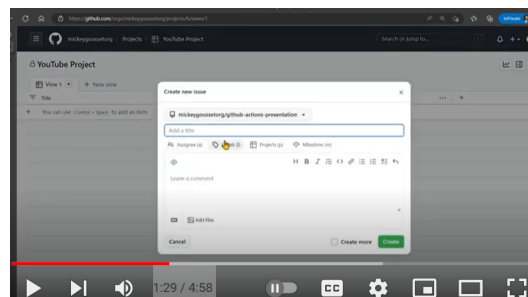


Figure 10: The Screenshot

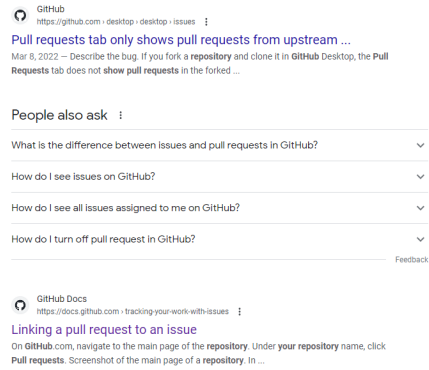


Figure 11: Google Search

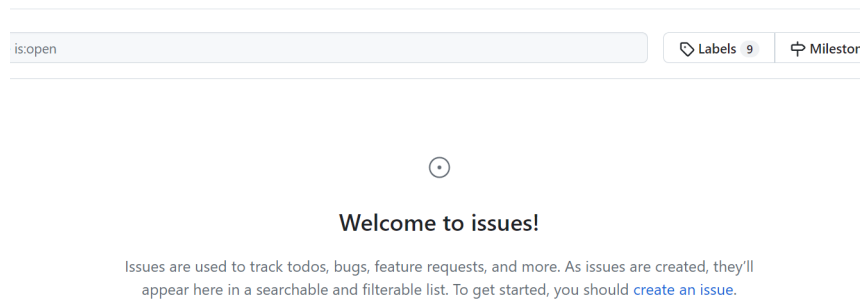


Figure 12: Issues