

# Lab 04: Scaffolding Reproducible Research

Breanna Laws

## 1 Reproducible Research

### 1.1 What it is:

Reproducible research is the idea that a project can be replicated at any point in time by any person to perfection. For example, one can clone a public project in Git and perfectly copy someone's code because it is all completely available to them. Essentially, it is about sharing data with ease. It involves two components: the computing environment and the project structure.

### 1.2 “Computing Environment” vs “Project Structure”:

The computing environment is the hardware, operating system, and software used for a project. The project structure is the organization of the files and folders that make up a project. The project structure is nested within the computer environment.

### 1.3 How GitHub supports reproducible research:

GitHub is a project management site that allows users to communicate with one another, as well as clone the repositories of others. It supports reproducible research by actively allowing people to replicate the work of others through the use of code sharing.

## 2 Project Management

### 2.1 Steps for Labs:

#### 2.1.1 Forking:

To fork a project, go to the repository you are looking to fork on GitHub. In the upper right-hand corner, click on the drop-down menu under “fork.” Select “create new fork.” You will get the option to rename the repository, but that is not necessary. You now have your own editable repository which contains the information in the original repository. You can compare these at any time while making edits.

#### 2.1.2 Cloning:

To clone a lab, press the green “<> Code” button in the right-hand corner. Clone using the web URL under the HTTPS tab (should be the tab that automatically opens). Copy the web URL.

Go into RStudio and hit “File -> New Project -> Version Control.” Then, select “Git” and paste in the repository URL you copied previously. The name of the lab should auto-load, but if not, name the project according to the “README” instructions.

#### 2.1.3 Editing:

Editing can now occur as normal. You can change things in text documents, create new documents within the project, etc. As you save your edits, they will be noted under the “Git” tab in the top right section of RStudio (also contains tabs for “Environment,” “History,” and “Tutorial.” Here, you can commit your changes and have them reflected on your GitHub account.

#### 2.1.4 Committing:

To commit changes, go to the “Git” tab in the top right section of RStudio and select what you want to commit. This tab should have kept track of all of your saved changes. Select the documents, press commit, and then a new tab will open. This allows you to type up what you changed so that you can better keep track of your edits. Once you are done typing up your comments, hit “commit” and then close the window. Now, to have these changes reflected in GitHub, you will want to “push” to changes to the repository.

### **2.1.5 Pushing Changes to a Repository:**

To push the changes to the repository, you will have wanted to have already committed the changes. Now, the “Git” tab will tell you how far ahead you are of the original repository you forked from. Press the green “Push” button to make the changes reflect in your forked repository on GitHub. This should run automatically. You can close the window once it is done pushing.

## **3 Assessing my Progress**

I feel a lot more confident after doing this lab in my ability to work between RStudio and GitHub. I feel well set up for the next lab and I am excited to see what's next :-)