How To Use the McCoy Lab Github Repo

Ijeamaka Anyene

2/2/2020

Introduction

Welcome to the McCoy Lab github repo! I'm excited that you are a part of the lab and that you are starting (or continuing) your github journey with us.

Being part of this github repo is more than just neverending `git commit -m "new changes"` (this will be funnier after you use git for a while), but it also keeping a consistent file and folder structure. By doing so, this will allow our team to be able to learn from each other, but also allows *you* to better collaborate with *future you*! It will also allow you others on the team to be able to understand what you did and easily identify the files they need.

What is this guide?

This HTML file (created using Rmarkdown) is a quick overview of:
1. Tenents of keeping your file naming convention and structure tidy
2. Tips for not accidentally releasing secure data into the wild

It is **not** a comprehensive how-to-guide of navigating github. Why not? Because so many people did a better job of explaining it than I ever could.

I recommend reading Happy Git and Github for the useR by Jenny Bryan to get you started with Git! There is a lot of incredibly helpful and well explained information in there.

Keeping Confidential Data Safe

Often, the McCoy Lab is working with de-identified patient data. Here are some key tips to ensure that confidential and senstive data is not exposed:

No Data on Github

Do not upload the actual dataset to github. You can do this by either remembering to never add and committ your data file. You can also update your .gitignore file to ignore your data.

No Results on Github

Keep your comments code specific, not data specific. Review your comments - do not refer to participants IDs or their information. This can be utilized to re-identify participants. BAD: User ID XXXX's HIV viral count information needs to be reviewed

For some projects, your results may also need to be confidential! If this is you, do not refer to your results or raw data in your data. Review your rmarkdown to make sure your tables are not being printed.

File Naming Conventions

There are three main principles naiming files. The file names should be 1) Machine readable, 2) Human readable, and 3) Play well with default ordering.

Machine Readable

You should avoid spaces, punctuation, accented characters. You should also se deliminators. "_" between words and "-" to make long chunks of words more understandable.

Human Readable

Name files based on content. This makes it easy to figure out what a file is for, purely based on name.

Plays well with default ordering

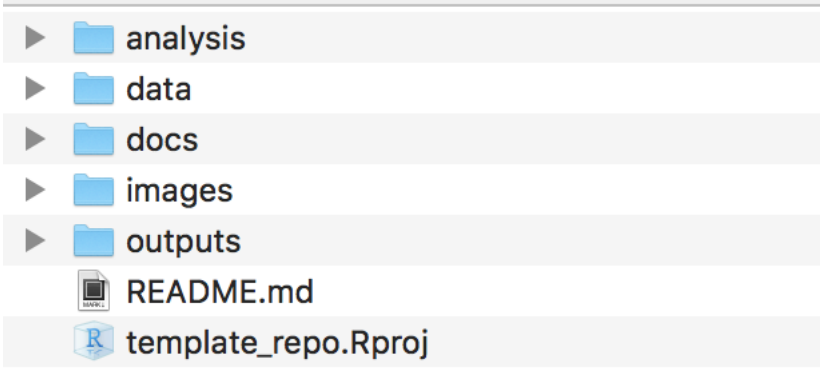
You should put numeric data first in the file name. When using numeric data: 1) Left pad numbers with zeros, and 2) Use YYY-MM-DD for dates.

Repository Structures

There are multiple strategies! The most important thing is you **pick** a strategy and then you consistently **use** said strategy.

Overview

Here is a picture of how I usually like to set up my repo for analysis projects.

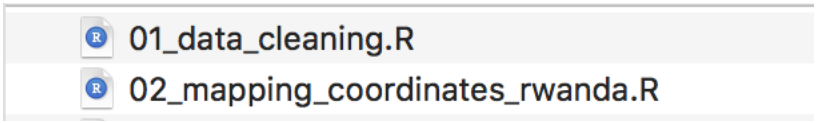


Detailed Tabs

Rproj file You should set up an R project file for your repo! It is a feature that comes along with RStudio that helps you keep everything organized. It's main power is the elimination of the need for changing your working directory or setting absolite file paths.

Data file This file and its contents *should not* be pushed to github. However it is an important file to have. I personally like to put my raw data and cleaned data in this file - but have them clearly labeled as such.

Scripts / Analysis Typically contains all of my scripts written in R. You want your scripts to be broken out into distinguishable chunks based on what you accomplish with each script. They should also be numbered by which order they need to be run in.



Outputs This file typically contains a myriad of information. It can be tables, graphics, etc. Depending on what it contains - if the results are confidential or sensitive participant information - you should not push it to github.

Docs
This is another file that most likely should not be pushed to github.

Acknowledgements

Much of this information in this document is thanks to:
* Fiona Grimm - Towards open health analytics: our guide to sharing code safely on Github
* Jenny Bryan - Reproducible Science Workshop —