Extra Resources and GitHub

Will Calandra

Extra Resources

Thank you very much for attending my R modules! I really appreciate your time and investment in yourself to learn a new skill! I encourage you to take on datasets and create your own projects just like we did together in the course. If you have any questions or want any coding/career advice, please feel free to reach out to Will Calandra at wcc44@georgetown.edu (mailto:wcc44@georgetown.edu)! I am always available to help out a fellow peer in the data community :)

While we covered the basics and got a taste of advanced modeling during the R module, below is a list of resources I encourage you to explore as you continue your journey in the R programming language. As with anything, continual practice leads to the best results!

- R for Data Science - Hadley Wickham: https://r4ds.had.co.nz/ (https://r4ds.had.co.nz/)
    - A great free book that walks you through the basics of coding and modeling in R!
- RStudio Cheatsheets: https://www.rstudio.com/resources/cheatsheets/ (https://www.rstudio.com/resources/cheatsheets/)
    - The most essential information compiled all in one place (and on one page)! You can get very far with these cheatsheets…
- Tidyverse website: https://www.tidyverse.org/packages/ (https://www.tidyverse.org/packages/)
    - The tidyverse is critical to learn, but good news, it makes life so easy! Here you will find documentation to all of its libraries
    - In particular, use the link for ggplot here for data visualization help: https://ggplot2.tidyverse.org/ (https://ggplot2.tidyverse.org/).
- YaRrr! The Pirate's Guide to R: https://bookdown.org/ndphillips/YaRrr/ (https://bookdown.org/ndphillips/YaRrr/)
    - This book attempts to make R funny (they try pretty hard, cringe), but chapter 5 and beyond covers PLENTY of both beginner and advanced techniques - there is great work on data structures
- RStudio Books: https://www.rstudio.com/resources/books/ (https://www.rstudio.com/resources/books/)
    - More books on R! Start with the first one though (R for Data Science).
- R resources - Paul Vanderlaken: https://paulvanderlaken.com/2017/08/10/r-resources-cheatsheets-tutorials-books/#introductory (https://paulvanderlaken.com/2017/08/10/r-resources-cheatsheets-tutorials-books/#introductory)
    - Comprehensive list of more than you could ask for!
- DataCamp, Coursera, freeCodeCamp, etc.
    - A bunch of online free resources for video learners! Just search R courses and find the teacher you like!!
- Kaggle: https://www.kaggle.com/ (https://www.kaggle.com/)
    - A great online community full of datasets and coding solutions available for download!
    - Create a free account and filter datasets for ".csv" files
- UCI Machine Learning Repository - https://archive.ics.uci.edu/ml/index.php (https://archive.ics.uci.edu/ml/index.php)
    - Another great, free website with datasets! Use the filters cleverly to get the type of dataset you want for your analysis!

- KDNuggets' list of datasets - https://www.kdnuggets.com/datasets/index.html (https://www.kdnuggets.com/datasets/index.html)
    - A huge list of datasets here - just google which one you'd like and have fun!

GitHub Instructions

A step-by-step guide to submit files through GitHub…

1 - Create a GitHub account and share your username with us (with Will Calandra, wcc44@georgetown.edu (mailto:wcc44@georgetown.edu))! We will then invite you to our repository so you can make changes (i.e. upload your files).

2 - Download GitHub Desktop! This will make the process easier than using your terminal. This link walks you through the steps to get started: https://docs.github.com/en/desktop/installing-and-configuring-github-desktop/overview/getting-started-with-github-desktop (https://docs.github.com/en/desktop/installing-and-configuring-github-desktop/overview/getting-started-with-github-desktop)

3 - Once installed and authorized, open GitHub Desktop and go to "File," "Clone Repository," and change the local path to an empty folder in your directory (this part is critical: if you are getting the "git can only clone to empty folders" case, you need to change the folder name on your local machine). Click "Clone," and if successful, in GitHub Desktop, your should be in our repository!

4 - Now, some terminology. "Cloning" a repo means that you took our repository from online and downloaded it to your computer. Notice now in your GitHub Desktop we have the current branch as main - this is good. A "branch" in GitHub is just like a folder on your computer… it's a shared directory online. We will only be using the main branch, but tech companies can get organized by having different branches for different things. Notice how you also have "fetch origin" in GitHub Desktop. The repo will automatically update online, but not on your computer. As such, to get the most recent version, click "fetch origin" every time you open GitHub Desktop.

5 - Now, we'll connect an "external editor," or just another way to say IDE to GitHub Desktop. IDE (or integrated development environment) is a fancy way to say what platform you are coding in. We will use IDEs for their file management features to make this process quick and easy. I recommend downloading Visual Studio Code (link: https://code.visualstudio.com/download (https://code.visualstudio.com/download)); the rest of the instructions will be based off of that.

6 - Open Visual Studio Code (VSCode). In a new window, click "Clone Git Repository." Then click "GitHub," "Allow," and authorize VSCode to use your GitHub account. Now close VSCode and go back to GitHub Desktop.

7 - In GitHub Desktop, click "File", "Options," "Integrations," and change the external editor to Visual Studio Code.

8 - Now, back in the home page of GitHub Desktop, click "Open in Visual Studio Code" under "Open the repository in your external editor."

9 - Now, in VSCode, notice "Subs" in your top left! Hover over it, click the "New Folder," and add a folder with your first initial last name (ex: wcalandra). Now, go to "File," "Open File," and select the file you want to publish (make sure this file's name is your lastname_file for example). When opened, it should show up at the top of your VSCode as a tab. Click and drag this tab into your folder. Then the file should be in there, and on the left in VSCode, both your folder and the file should be highlighted green!

10 - Go back to GitHub Desktop. Notice the changes? This means that it tracked your changes to the repository. Now it's time to authorize these changes, or what Git calls, "make a commit." Next to your GitHub logo on the left in GitHub Desktop, type what you are doing (ex: committing first file), and feel free to type in the description as well

if needed (this part is optional). Then click “commit to main,” which will formally make the change in the online repository (in the main branch).

11 - However, it is not completed until you click “Push Origin.” This pushes your changes to the online repo. If done correctly, go to your browser and look at our repo. You should see your changes uploaded successfully! Congrats!

12 - Pretty neat, right! Collaboration, code review, maintenance, etc. are all done in Git by many companies around the world. Familiarize yourself with Git and its terminology - it is a very useful skill IRL! If you have any questions, comments, or are interested in not using GitHub Desktop but trying to connect your terminal with our repo instead, reach out to Will Calandra (netid: wcc44).