Final Project: Reproducible Research

Short Proposal and Group Members Due: February via email

In-class presentations: February 27 and 27

Writeup and Code Due: March 1 by midnight on GitHub

Project Description:

The goal of this project is to showcase your knowledge of R by applying it to a research project of your own. This is not a research methods course, so the quality of the research is ancillary to the quality of your programming. You will be graded on the parts: coding, writeup, and an in-class presentation.

Coding (70%)

The code for the project should have following components:

1. Data wrangling (20%)

   You must use a minimum of *three* datasets, at least one of which should be retrieved automatically from the web using APIs or web scraping. All processing of the data should be handled by your code, including all merging and reshaping. Any automatic data retrieval must have an option to toggle accessing the web off if the data is already downloaded. This is where you can showcase your abilities practiced in homework 1.

2. Plotting (20%)

   From that data, you will create a minimum of *two static plots* using ggplot, and *two interactive Shiny plots*. Your Shiny does not have to be shared on shinyapps.io. The skills used here will roughly correspond to your work on homework 2.

3. Text processing (10%)

   You will now introduce some form of text analysis, similar to that of homework 3. While it should relate to your broader question, this may be distinct data from what you created in part 1, and the results of it may be used in your plotting or analysis.

4. Analysis (10%)

   Then you will fit a model to your data and report basic results. As this is not a statistics or econometrics class, the model you choose and the validity of your results are not terribly important; fitting an OLS model with fixed effects that has insignificant p-values will not be penalized. The goal is to show you can prepare your data through the previous steps to have it ready for model fitting.

5. Reproducibility (10%)

   The project and files should be structured and documented so that someone could fork your repository and reproduce your results. This means that your README should document the order in which codes should be run, and what needs to be edited (e.g., where the user should set their path) by the user. If a dataset is retrieved automatically, then the final results do not have to reproduce exactly but the code should run smoothly even if the underlying data changes.

Writeup (15%)

You will then spend *no more than 2-3 pages* writing up your project. You should describe your research question, then discuss the approach you took and the coding involved, including discussing any weaknesses or difficulties encountered. Finish with a brief discussion of results, and how this could be fleshed out in future research. The primary purpose of this writeup is to inform me of what I am reading before I look at your code.

The top of your writeup should include the names of all group members and Github user IDs.

Presentation (15%)

You will give a *10-minute in-class presentation* on the project in the last 2 sections of class. The presentation will largely mirror the structure of the writeup, but be more focused on discussing the research question and results as opposed to explaining the details of the coding. You should cover where the data come from, discuss the motivation and research question you are interested in, and explain what methods you used. You talk about discuss each static figure and your regression model, and demo at least one of the Shiny apps. Finally, you should conclude with a summary of your current findings and potential directions for future work. If you are working in a pair, each member should present for about half the time.

Instructions:

You may work on this project alone, or in pairs. All pairings must be formed *on GitHub classrooms* before any work is done - it is not possible to join one after. You should send me an email listing who you are working with and a short, informal proposal (2 paragraphs) by February 1.

It is required that you use GitHub, and we may use your past commits to understand your thought process for partial credit. If you working in a pair, note that as we are grading we will be looking for multiple commits per individual throughout the project. Expectations for the scope of the project will be higher for pairs than for individuals, and the division of labor should be approximately evenly across both individuals. While we will lean toward giving the same grade for both students, it is possible that individuals may receive different grades based on the commit history.

Your final repository must contain the following:

1. README file summarizing project and code
2. Your R code and commit history. Split your code up into multiple files associated with the categories listed above: data.R, staticplot.R, shinyapp.R, textprocess.R, and model.R.
3. A folder named "data" that contains the initial, *unmodified* dataframes you download and the final versions of the dataframe(s) you built. For the dataset which is automatically retrieved from online, you should archive a version of the data you pulled in the "data" folder, and indicate in the data.R code where a user could replace the data retrieval with the path of the archived data. If the dataset is too large to be hosted on Github, it can hosted on Drive or Dropbox and the link should be provided in the README file as well as indicated in the data.R code.
4. A folder named "images" that contains saved .png versions of your static plots
5. Your writeup in markdown format, named as "writeup.md."

Suggestions and Tips

- I encourage you to create a project on a subject that is relevant to your interests, and other Harris classes. If your research idea is a much larger project, think of how you can develop a basic framework for it using this project, which can then later be expanded into a proper research project.
- If you feel stuck coming up with research ideas, feel free to contact me or one of the TAs so we can discuss your interests and make suggestions.
- You may use libraries and methods we did not go over in class, but ones that we did go over should be preferred if they duplicate the functionality. Remember all citation rules from the academic dishonesty policy in the syllabus.
- Effort put into organizing your code and making it readable, by, for example, following the Tidyverse Style Guide, and good usage of dplyr, functions, variable names, and comments will be rewarded.
- Similarly, your GitHub repo should be organized - do not leave useless files there (e.g. DS_store files), and keep things in folders.
- The entire point of reproducible research is to make it possible for others (and for a future you who has had time to forget what you did and why) to understand, replicate, and modify your work. Keeping this in mind as you work will be good for your grade, and helpful to you in the future if you expand on the project.
- Free shinyapps.io pages will run slowly, particularly if your data is large. Keep this in mind when planning your Shiny app, especially if you have large shape files.

Project Checklist

By February 1:

- □ If working in a pair, list who you will be working with on GitHub classrooms
- □ Email with 2 paragraph proposal and whether you will be working in an individual or in a pair
- □ Sign up for a presentation slot

In-class presentation (February 27 or 29):

- □ Motivation and research question
- □ Data source and methods
- □ 2 static figures
- □ Regression model
- □ Demo of 1 Shiny app
- □ Conclusion and next steps
- □ If working in pair, both individuals should present

Project submission (by March 1):

- □ All commits are documented in GitHub. For pairings, it should be evident from the commit history that work was divided approximately evenly across both individuals.
- □ README.md
  - □ Original data source and description of data
  - □ Links to any large data files hosted on Dropbox or Drive and description of where they are used
  - □ What each code does and in what order they should be run
  - □ Explain what a user needs to modify in each code to replicate (e.g., path name)
  - □ Date created and author(s), version of R used, packages required, and package versions
  - □ Links to Shiny apps
- □ data.R file
  - □ Option to use archived version of retrieved dataset
- □ staticplot.R
- □ shinyapp.R
- □ textprocess.R
- □ model.R
- □ writeup.md
- □ “data” folder
  - □ Raw data
  - □ Archived version of retrieved data
  - □ Final, cleaned data
- □ “images” folder
  - □ .png versions of static plots