Assignment 04: Data Visualization, Quarto, and Git

Data Science for Public Policy

Aaron R. Williams and Alena Stern - Georgetown University

Deliverable: There are three deliverables to submit for this assignment:

1. index.html
2. the .qmd file with your R code
3. the URL of the GitHub repository

You may need to place the .qmd and .html file into a folder, and compress (or zip) that folder, in order to submit it to Canvas.

Points: 11

Lab partner: Each student can pick one lab partner from their section and submit one assignment between the two of them. Lab partners are expected to complete every exercise together and should spend time together in-person or on Zoom. Dividing and conquering is not an acceptable approach.

Plagiarism on homework or projects will be dealt with to the full extent allowed by Georgetown policy (see http://honorcouncil.georgetown.edu).

Setup

Note: If you are working with a partner, you will follow the instructions as outlined in the class exercise. If you are working alone, you can skip steps 2 and 7.

1. Only **one** partner should go to GitHub.com and log into your account. Click on the green button that says "New" to create a new GitHub repository and a create a repository called assignment04.

2. The partner that created the repo should now add the other partner to their repository. Under Settings select "Collaborators" under the Access menu on the left. Select the "Add people" button and add your collaborators by entering their GitHub user name.
3. Now both sets of partners should clone the repo to the location where they want the assignment04 folder.
4. Only **one** partner should create a new R Project, choose the "existing directory" option, and select the `assignment04` folder they just created by cloning their GitHub repo.
5. That same partner should create an index.qmd file inside the assignment04 folder, copy the YAML header from the `template_html.qmd` file on canvas into the index.qmd file (replacing the default YAML header), and add both their name and their partner's name in the "author" option in the YAML header. The partner should render the index.qmd file.
6. The same partner should `git add`, `git commit -m "<message>"` and `git push` index.qmd, index.html, and the .Rproj to GitHub following the instructions in exercise 04.
7. The other partner should make sure they're inside their assignment04 folder using `pwd` and `cd` on the command line. Then, they should run `git pull` to copy the changes the first partner just pushed to github to their local machine. They can now open the `.Rproj` file their partner created and make edits to `index.qmd`.
8. One partner should create (or edit if you created it with the repo) a file called README.md in the assignment04 folder and provide a title and authors in the README. Please push this file to GitHub.
9. Once that's done, you're all set up for assignment04! Remember to push changes to GitHub often and to always run `git pull` **before** making edits to shared files.

Please note: Setting up and using Git and GitHub for the first time can be a challenge. I strongly recommend you start this assignment very early and come to office hours or otherwise reach out with any issues.

Assignment Description

This assignment differs from previous assignments, in that you are tasked with finding and analyzing a dataset of your own choosing. This will result in you creating and submitting a single .html file, containing your code, a series of four visualizations made with ggplot2, and a brief discussion of what those visualizations mean.

First, find a public dataset available from the web, relevant to one of your policy interests. Add this data to a folder called 'data' in the same folder as your Quarto file. You should also create a .gitignore file and add the file suffix of your data (like .csv, .xlsx, or .sas7bdat) to the .gitignore, so git will ignore your data. You can list the full file name to ignore individual files or use code like `*.csv` to ignore all .csvs. Properly ignored files will not show up in `git status`.

Then analyze the data, storing all your code and writing in your .qmd file, following the requirements described below. There are three graded components of this analysis (Read all instructions before starting, as you must use git as you create the graphs).

1. Git Repo and Commits (2 Points)

Use `git` thoroughly for this assignment. This means, at a minimum, you must commit when you first start the project and after the completion of each graph (a bare minimum of five commits). When you submit the assignment, all commits must have been pushed to GitHub. This means there will be a record of the commits on the public GitHub repository. You must submit the GitHub URL as part of the assignment, and will be graded on having committed your code as you worked on the assignment.

2. Quarto document (1 points)

Create and clearly format an Quarto document for this analysis. Specifically:

- Include the YAML header provided in `template_html.qmd` on Canvas;
- Use a hyperlink to link to the source of your data;
- Use appropriate headers to signify each visualization;
- Include code chunks for data manipulation and visualization so I can understand the code and analysis you ran to create the graphs;
- Hide warning messages and unnecessary printing of data.

3. Create a .gitignore (1 point)

A gitignore file specifies intentionally untracked files that Git should ignore. Files that you add to the gitignore file cannot be pushed to GitHub. This is useful to avoid accidentally pushing files that you don't want to GitHub - like data files or files storing passwords and other credentials.

1. Create a new text file by selecting `File -> New File -> Text File` in the top menu.
2. Save the file as `.gitignore` in the root directory of your repo (this is the same folder your RStudio project is located in). You may get a message that "Names that begin with a dot "." are reserved for the system." - click 'Use "." ' to save the file.
3. GitHub has a set of gitignore templates for different programming languages that are prepopulated with common files to add to gitignore. Copy the contents of the R gitignore template into your `.gitignore` file. Add the file path to the data file you used for assignment 04 to your gitignore file and save the file.
4. Try to `git add` your data file. What message do you receive in the terminal?
5. Push your `.gitignore` file to GitHub.

4. Four ggplot2 graphs (4 points)

Your analysis should have four data visualizations of distinct graph types, made with ggplot2. Across all four graphs, use a total of:

- Six different aesthetics (i.e. set inside `aes()`);
- Six different non-aesthetic options (i.e. set outside `aes()`);
- Five different geoms;
- Two different scales (meaning change the default scale used for at least two aesthetics).

Further, each graph must include:

- Correct usage of all visual encodings;
- Appropriate data sourcing (hint, check out the `labs()` function);
- Proper labeling of all visual encodings;
- An appropriate title and subtitle;
- The code you used to generate each graphic, right above the graph (Quarto should make this easy).

The data visualizations should be clear and polished enough to go in a report. Checkout fivethirtyeight or the Urban Institute for inspiration.

5. Written narrative & interpretation of visualizations (2 Points)

Write at least three to five sentences about each graph in your document, describing what it says and how it informs relevant policy topics. You should (to the ex-

tent possible) write this as a narrative that ties together all of visualizations. You will be graded on the interpretation of the visualizations, and their relevance to the policy topic.

Part 6: GitHub pages (1 point)

GitHub pages offers free web hosting.

1. Make sure you've called your .qmd and rendered .html document index.qmd and index.html.
2. In your GitHub repo, go to Settings > Pages.
3. Set the source to `main` and click save. You now have a website!
4. Make sure your final `index.qmd` and `index.html` files are pushed to GitHub. Go to your GitHub page and observe that your changes are automatically reflected in your page (note that it may take a moment for the GitHub to deploy the latest changes).
5. Add the URL to README.md and make sure the updated README is on your main branch on GitHub.

It isn't required, but consider learning more about figures and cross references to improve your Quarto document.

Submission

Upon completion of the assignment, render the .qmd file to .html, push the .qmd and .html to GitHub and submit both to Canvas, along with the URL of your GitHub Repository. You may need to put the URL in a submission comment.

Tips

- Quarto can be challenging at first. Render early and render often.
- Git/GitHub can be challenging at first. Commit early and commit often.
- If you forget `-m` with `git commit`, use this highly viewed Stack Overflow post to escape VIM.