EDAV FINAL PROJECT due Thursday, April 20, 2017, 11:59pm

GENERAL INFORMATION

GOAL
The goal of this project is to perform an exploratory data analysis / create visualizations of a data set of your choosing, in order to gain preliminary insights on questions of interest to you.

TEAMS
You may work alone or in teams of up to 4 people. Grading will be by team; more is expected of larger teams. Information will be provided on CourseWorks on how to sign up as a team.

DATA
Choose a data set that is not on the beaten track, that is, one that is not included in R (or similar), nor used in Kaggle (or similar) competitions, nor relatively well-known through some other forum. You will begin working with the dataset in Homework #4, to be assigned soon, due Tues, March 28.

ANALYSIS
You have a lot of freedom to choose what to do, as long as you restrict yourselves to *exploratory* techniques (rather than modeling / prediction approaches). In addition, your analysis must be clearly *documented* and *reproducible* (more on that below).

FEEDBACK
At any point, you may ask the TAs--Ian (iak2119) and Bridget (blr2147)--or me (jtr13) for advice. Our primary role in this regard will be to provide general guidance on your choice of dataset / topic / direction. As always, you are encouraged to post specific questions to Piazza, particularly coding questions and issues. You may also volunteer to discuss your project with the class in order to get feedback--if you'd like to do this, email me to schedule a date.

PEER REVIEW
A portion of your grade is based on the feedback you give to other groups. After the due date, you will be assigned projects to review, which you need complete by **Monday, May 1, 11:59pm**. More specific details on what you need to do will be provided at that time. Your grade is *not* directly based on the feedback you receive. It will be determined by the instructor and TAs.

REPORT FORMAT
Your project should be submitted to CourseWorks as a **nb.html** or **.ipynb** file, with graphs / output rendered. Any material that cannot be included in the notebook format, such as certain interactive visualizations, should be clearly referenced, ideally by providing a link in your notebook to an online visualization. You will lose points if we have trouble reading your file, need to ask you to resubmit with graphs visible, if links are broken, or if we have other difficulties accessing your materials due to factors that are in your control.

- #1 piece of advice: don't wait to start writing. Your overall project will undoubtedly be better if you give up trying to get that last graph perfect or the last bit of analysis done and *get to the writing*!
- Note that you can run Python in R and vice-versa, on a by chunk basis, though with some limitations.
- Formats other than notebooks are ok too, as long as the code is accessible. If the code is not included, provide a link to the location of the code on Github or elsewhere. That means a link to the specific file or folder with the code (not just a note that the "code is on Github.") If you cannot include the code at all since, for example, you used an online tool to create a graph, state that, and give a brief explanation of what the tool does and where it can be found.

- Using Markdown + code chunks is supposed to make combining code, text and graphs easier. If it is making it more difficult, you are probably trying to do something that isn't well suited to the toolset. Focus on the text and graphs, not the formatting.

REPORT OUTLINE

Your report should include the following sections, with subtitles ("Introduction", etc.) as indicated:

1. Introduction

In this section, explain why you chose this topic, and the questions you are interested in studying. Include a brief description of how you found the data, and clear instructions on where the reader can find the data.

2. Team

List team members and a description of how each contributed to the project. (If you're working alone, briefly describe the stages of the project.)

3. Analysis of Data Quality

Provide a detailed, well-organized description of data quality, including textual description, graphs, and code.

4. Executive Summary

Provide a short **nontechnical** summary of the most revealing findings of your analysis with no more than 3 static graphs or one interactive graph (or link), written for a nontechnical audience. The length should be approximately 2 pages (if we were using pages...) Do not show code, and take extra care to clean up your graphs, ensuring that best practices for presentation are followed.

- Note: the tips below are not intended to be a complete list of everything we've covered this semester on designing a successful graph. It's meant to help you avoid some common problems.
- Title, axis labels, tick mark labels, and legends should be comprehensible (easy to understand) and legible (easy to read / decipher).
- Tick marks should not be labeled in scientific notation or with long strings of zeros, such as 3000000000. Instead, convert to smaller numbers and change the units: 3000000000 becomes "3" and the axis label "billions of views".
- Units should be intuitive (Extreme example: an axis labeled in month/day/year format is intuitive, one labeled in seconds since January 1, 1970 is not.)
- The font size should be large enough to read clearly. The default in ggplot2 is generally too small. You can easily change it by passing the base font size to the theme, such as `+ theme_grey(16).` (The default base font size is 11.)
- The order of items on the axes and legends is logical. (Alphabetical is often *not* logical.)
- Colors should be color vision deficiency friendly.
- If a legend is taking up too much space on the right, move it to the bottom.
- If categorical variable levels are long, set up the graph so the categorical variable is on the y-axis and the names are horizontal. A better option, if possible, is to shorten the names of the levels.
- Not all EDA graphs lend themselves to presentation, either because the graph form is hard to understand without practice or it's not well labeled. The labeling problem can be solved by adding text in an image editor. The downside is that it is not reproducible. If you want to go this route, Paintbrush is a free and simple bitmap image editor for the Mac: https://paintbrush.sourceforge.io/ There are many other options.

- Err on the side of simplicity. Don't, for example, overuse color when it's not necessary. Ask yourself: does color make this graph any clearer? If it doesn't, leave it out.
- Test your graphs on nontechnical friends and family and ask for feedback.

5. Main Analysis
Provide a detailed, well-organized description of your findings, including textual description, graphs, and code. Your focus should be on both the results and the process. Include, as reasonable and relevant, approaches that didn't work, challenges, the data cleaning process, etc.

- The guidelines for the Executive Summary above do **NOT** apply to exploratory data analysis. Your main concern is designing graphs that reveal patterns and trends.
- As noted in Hmk #4, do not use circles, that is: bubbles, pie charts, or polar coordinates.
- Use stacked bar charts sparingly. Try grouped bar charts and faceting as alternatives, and only choose stacked bar charts if they truly do a better job than the alternatives for observing patterns.

6. Conclusion
Discuss limitations and future directions, lessons learned.

A note on style:
You are encouraged to be as intellectually honest as possible. That means pointing out flaws in your work, detailing obstacles, disagreements, decision points, etc. -- the kinds of "behind-the-scene" things that are important but often left out of reports. You may use the first person ("I"/"We") or specific team members' names, as relevant.

Grading Rubric

Section	**Points**
Introduction (including choice of data set, questions), Team description	10
Analysis of Data Quality	10
Executive Summary (focus on quality of presentation choices / techniques)	20
Main Analysis (focus on quality of EDA choices / techniques)	35
Conclusion	5
General	
Reproducibility (will be discussed more in class), sources cited • reader can clearly follow the process of data collection, cleaning, analysis and visualization either by providing code or links to code, *and explanation* ("code is on Github" is not sufficient.) • sources are cited with links (bibliographies not necessary)	10
Technical flawlessness (files open, links work, code runs, etc.)	10
TOTAL*	**100**

*** If late, 10 points will be deducted per day. Plagiarism of any kind will not be tolerated and will result in a grade of 0 for the project.**