Homework 1

Due: Oct 10, 11:59PM

There are many ways to produce meaningful data visualizations. Choosing the right format with the right tools is critical to delivering your message. In this homework, we will use a simple data set and explore various mainstream data visualization tools and understand their benefits and drawbacks.

In this homework, you will use the UC San Diego Admission Dataset extracted from here. However, instead of using this dataset, you will use a simpler version. We have generated this data in two formats, JSON and CSV. Please use the following link to download them:

1. JSON file: https://gist.github.com/dingmei/e889c092928a3de2ad9eb7076b88ae83
2. CSV file: https://gist.github.com/dingmei/d1ab5dd8ff9597d71b3fba3dc26347fe

Tools:

1. Google G Suite
2. Infogram
3. Tableau Desktop*
4. D3.js
5. Highcharts*

Instructions for each tool:

1) Create 3 sets of charts: bar chat, line chart, pie chart.
   Take a screenshot for each chart and insert them into your submission.
2) Write a short analysis of each tool. Perform a short comparative analysis focusing on the pros and cons of each.
3) When creating the charts, you do not need to use every single field in this dataset when creating the visualizations. Pick the fields you think are appropriate.

*** Additional Instructions for Tableau Desktop:**

1) Download Tableau Desktop via https://public.tableau.com/en-us/s/ (you don't necessarily need to sign up for an account, you can enter your email and download the installation file directly)
2) After importing data, make sure the data type of everything in data source is "Number(whole)" except "Measure Names"

3) Think about what data should be Dimensions and what data should be Measures (hint: think about independent and dependent variables)
4) Google any problems you encounter! (and then feel free to reach out to the TAs)

*** Additional Instructions for Web-Based solutions:**
Cross-Origin Resource Sharing (CORS) is a mechanism that uses additional HTTP headers to tell browsers to give a web application running at one origin, access to selected resources from a different “origin”. A web application executes a cross-origin HTTP request when it requests a resource that has a different origin (domain, protocol, or port) from its own.

To avoid running into CORS issues, you might consider hosting your webpage on a server (For instance, Apache or Nginx are some of the full-fledged web-servers, or even Python’s SimpleHTTP server. If you run with Node.js, consider using a library from npm that can help serve static files, as opposed to writing it all out from scratch)

Make sure to use Chrome’s Developer Tools to debug as you work through the assignment! (or the node command line tool depending on your environment)

Submission:
Submissions will be made through GradeScope. Include a one page report with the writeup for each tool. Also include a link to your GitHub repository in your writeup that contains any code and configuration files you might have created for this assignment.