CS/INFO 3300; INFO 5100
Homework 9
Due 11:59pm Wednesday, November 10

Goals: Demonstrate effective visualization design skills by making a new chart

Your work should be in the form of an HTML file called index.html or index.htm with one <p> element per problem. For this homework we will be using d3.js. **In the <head> section of your file, please import d3 using this tag: <script src="https://d3js.org/d3.v7.min.js"></script>**

Create a zip file containing your **HTML file and associated data files** and upload it to CMS before the deadline. Submissions that do not include data files may be penalized. Your submission will be graded using a Python web server run in a parent directory containing your zip file contents along with many other students' submissions.

1. For this homework you have a choice between several different multivariate datasets. Your goal is to visualize the dataset of your choosing using whatever visual metaphor you feel is appropriate for the task. You do not need to visualize all of the datasets – submissions that visualize more than one will not receive any extra credit. Your visualization does not need to be incredibly complicated or show 10 data attributes at once – simple but well-designed visualizations will receive full credit.

As we are reaching the end of the term, many individual design and implementation decisions will be left to you. We will provide a general list of requirements you must meet with your submission. **This assignment will be graded on a completion basis. If you meet the design criteria listed, you will receive full credit for the assignment.**

Dataset options:

a) **hip-hop-rankings.csv** – rankings from a BBC survey of music critics which attempted to find the best hip-hop songs of all time (github link).
b) **passwords.csv** – the most popular passwords in public account data and leaks, as visualized by Information is Beautiful, with strength and popularity scores (google doc)
c) **board-games.csv** – a dump of board games between 1950 and 2016 that had over 50 ratings on Board Game Geek, as visualized by Five Thirty Eight (github link)
d) **powerlifting.csv** – a dataset of participants in International Powerlifting Federation events, as visualized by Elias Oziolor using the Open Powerlifting repository (github link)
e) **adoptable-dogs.csv** – a dataset of dogs that were listed for adoption in September, 2019, as visualized by The Pudding using the Petfinder API (github link)
f) **bob-ross.csv** – a list of all of Bob Ross's paintings by episode, with boolean values indicating different subject matter, as visualized by Five Thirty Eight (github link) (this one may require post-processing to aggregate or use of d3.nest to create summary data)

Step 0:

Take a look at the datasets. While the Github links and articles will give some insight into the datasets we've provided, you can learn a lot from just inspecting the headers of the .csv files. Examine the different data attributes (columns) available to you and start hypothesizing about what kinds of things you might want to show for the data. Check out the articles to get some more ideas (you are welcome to make your own version of one of the charts presented in the articles but be sure to use your own design sensibilities here). Pick one dataset to visualize. If necessary, post-process, clean, or trim your dataset so that it is easier to work with. Some of the datasets are quite large, and you are not required to visualize all of the data.

Step 1:

In a <p> tag, describe which dataset you chose and why. Identify one hypothesis you hope to answer through your visualization (e.g. "Does age impact the performance of powerlifters negatively or positively?") or one particular kind of insight you want users to learn (e.g. "I want users to learn more about the overall distribution of dog breeds available for adoption"). Your question should be specific and connect to the visualization you develop in Step 2.

Step 2:

Visualize the data. You can choose whatever visual metaphor you feel best conveys the hypothesis/insight you identified in Step 1 (e.g. bar chart, scatterplot, choropleth). No matter which dataset and visualization you choose to work with, you must meet the following general criteria in your design and implementation:

- You must **visualize at least 3 different data attributes at once**. Latitude and longitude count as 1 attribute together.
- Your visualization must have **one interactive "drill-down" component** that helps users investigate points or categories of data in more detail. In the past we have provided this sort of functionality through click and hover information, but you can choose whatever interaction you feel is appropriate. *Mouseover labels are sufficient for full credit*.
- You must provide **axis labels and annotations** to help users make sense of your visualization. At the very least, you must add axis labels and markers to position/length channels if you are making a bar or scatterplot and a graticule if you are making a geographic view. You are not required to add a color legend, though it is recommended.
- The visualization must load efficiently (<5s) and the code must execute properly. Exceptions can be thrown in the console, but the visualization must work.
- Choose effective visual channels for your design. The use of volume is discouraged.
- Use space well and consider carefully what kinds of scales you use and how you add whitespace. The visualization should be **no larger than 900px wide x 800px** tall.
- If you feel necessary, you have the option of importing topoJSON and using JS built-in methods (as in past course notes). You are free to use the topoJSON files provided in the course repository to visualize data geographically. You may not import other libraries. Make sure to include the topoJSON files in your ZIP if you choose to use them.
- You may use <canvas> elements for this assignment but be aware that it will make the interactive portion especially tricky.
- If you face a trade-off between less-than-ideal options, feel free to include an extra <p> tag describing these issues.