Assignment 2

Due Date: 06/18/2023

In this assignment, you will design an interface that shows visualizations including some methods you have learned from the exercises. Choose the data domain and the interactive visualization techniques that best describe the data chosen. You will describe the features of your application, reasons for your choice of visualizations and interaction techniques, observation of data from your application in a report. You should deeply consider why the interaction techniques you choose are effective in the context of your data domain. D3 is recommended to use as you practice throughout the course, yet you are welcomed to use any type of programming language and framework for implementation. (Please do not use the software that does not require programming like Tableau; your task is to code the program.)

System Guidelines

- Design an interface with a clear layout
- Implement at least 3 different views, each containing a different visualization component (e.g. 3 bar charts only count as 1 component)
- Include interactive techniques like brushing, painting, scaling and linking. Interaction within one view must trigger a change in a different view. At least two views need to be linked bidirectionally.
- Include at least one filtering option
- Feel free to use multiple datasets

Point deduction will be made when the result suffers from:

- Omission of any requirement described above
- Errors or broken features
- Questionable visual encodings
- Confusing interface design or unhelpful interactions
- Lack of exploratory interaction techniques
- Incomplete report

Report Guidelines

You should also submit the report including the following information.

- Teaser image of your application
- Description of your data including a URL linking to the source of your data and data preprocessing pipeline
- Features of your application with the detailed description
- Design rationale why you choose the particular visual encodings, interactions, and animation techniques
- Usage scenarios how users use your system
- Observation that can be obtained from your system
- An explanation to run your code (for any installation, create requirement.txt) as an appendix

Extra Credit

- High quality of the final interface
- Implementation of advanced functionalities that exceed the project's basic requirements
- Outstanding and comprehensive explanation
- Showcasing a demo website

Here I provide two websites with several examples:

https://www.omnisci.com/demos

https://www.students.cs.ubc.ca/~cs-436v/22Jan/fame/

These are just examples that you can get a sense of implementation, you do not need to care too much about the styles such as visual effects.

Please submit a zip file named *studentID_firstname.zip* including all the materials needed to implement your code and the report in a pdf format. Any submission past the deadline will result in 0 points.