

Assignment 1: D3.js

Submission date : Sat, 9 Feb, 10pm

Type : Individual Marks : 5

1. Objective

The purpose of the assignment is to learn visualization through D3.js. You are to choose D3 visualizations best suited for the data and tasks listed in sections below.

2. Data

There is a data file provided with this assignment **<results.csv>**.

About Dataset:

- A team has performed several experiments to test the efficiency of video streaming methods. For better validation of results, tests are conducted using **4 network conditions/profiles** with **7 video samples** under **3 buffer capacity configurations**.
- In the first configuration, the total duration of video that can be stored in the buffer is either 30 seconds or 60 seconds based on the duration of the video. If the video is more than 10 minutes, the buffer capacity is 60 seconds, else, it is 30 seconds.
- In the other two configurations, the buffer capacity is fixed and not dependent on video duration, i.e., 120 seconds and 240 seconds.
- Video sample V1, V4, and V5 are less than 10 minutes and remaining (V2, V3, V6, and V7) are more than 10 minutes. The results are stored in **result.csv** file.
- The result csv file has following attributes **<profile, sample, method, quality, change, inefficiency, stall, numStall, avgStall, overflow, numOverflow, qoe, bufSize>**. The definitions for the attributes are:
 - **profile**: Network profile used for testing.
 - **sample**: Video sample on which test is conducted.
 - **method**: Method used for streaming.
 - **quality**: Average quality played (in Kbps).
 - **change**: Changes in quality during the playback.
 - **inefficiency**: Inefficiency of the method to fully utilize the available bandwidth.
 - **stall**: Total stall duration (in seconds) during the playback.
 - **numStall**: Number of stalls happened during the playback.
 - **avgStall**: Average stall duration (in seconds) during the playback.
 - **overflow**: The duration (in seconds) for which buffer was full.
 - **numOverflow**: Number of times when buffer was full.

- **qoe**: Quality of experience during the playback.
- **bufSize**: Buffer configuration, i.e., the maximum content that can be buffered/ buffer capacity (in seconds). ***Do note that there are three configurations : 30 seconds and 60 seconds come under the same buffer configuration category. The other two configurations are 120 seconds and 240 seconds.***

You can process the data to derive new values.

3. Task

Based on the description given above, plot the visualization for following queries. Feel free to use your choice of plot (bar plot, histogram, stacked plot, X-Y plot, etc.) to clearly visualize the results:

1. Draw plots to show the Average quality and number of changes in quality for a method. Each buffer configuration should have separate plot.
2. Draw a single plot to show the Average QoE for a method grouped by different buffer configuration. The plot should be in such a way that we can compare the performance of the method with itself for different buffer configuration as well as with different methods within a buffer configuration.
3. Draw plots to show the correlation between inefficiency and quality for all methods in different buffer configurations.
4. We would like to know the methods which have the minimum number of stalls for video V7 under different network profiles. Draw appropriate plot for it.

Note: You may replace any of the above tasks with a visualization/insight you find on your own for the given data. You are free to submit upto **two** such replacement visualizations.

We have listed few visualizations in Section 5 below for reference. You are allowed to use/modify the provided code samples for the assignment purpose. You are encouraged to explore D3 visualizations beyond what is mentioned here.

4. Submission

The deadline for submitting the assignment is **Saturday 9 Feb, 10pm.**

Submit a file labeled **<your matric number>_D3Assignment.zip**

containing (i) your code (HTML & JS files with d3.min.js). The code should be self contained and able to run without external dependencies.

(ii) a readme file with a simple explanation of your choices for visual encoding(for each of the four visualizations) and insights you gained(particularly for two visualizations you might submit to replace any of

the given tasks) and any extra information on e.g. how to run your code. Submit it in **IVLE Files /Student Submission /Assignment 1** Folder.

5. Sample Visualizations

Pie chart - <https://bl.ocks.org/mbostock/3887235>

Heat map - <http://bl.ocks.org/tjdecke/5558084>

Grouped bar chart - <https://bl.ocks.org/mbostock/3887051>

Line graph - <https://bl.ocks.org/mbostock/3883245>

Bubble Chart - <https://bl.ocks.org/mbostock/4063269>

Stacked Area Chart - <https://bl.ocks.org/mbostock/3885211>

6. Grading Rubrics

The maximum obtainable points for this assignment is 40 points. Each visualization will be graded for addressing the given requirement, meaningful representation and visualization appeal.

Each visualization will be marked on **10 points**.

Expressiveness
including explanations about visual encoding ; about insights gained and why the visualization is chosen.
Effectiveness
including Visualization appeal and clarity; Use of Labels, legends, user interactivity (tooltips or data-manipulation animation)

Tip : Use the developer tools (F12) and `console.log()` regularly when writing visualization code, it will help with debugging.

Send your queries to cs5346.tutor@gmail.com.