Final Project: Report

Data Visualisation

Project Group 03:
Hamza Sheikh, Gabriel Sampaio Da Silva Diogo

1   Introduction

At present, data visualisation is used extensively in the corporate world. Considered to be an art and science, data visualisation is "certain to change the corporate landscape over the next few years" [1]. It was decided as a group that we would produce visualisations based on data concerning British society to make conclusions on topics more relevant to the living standards in England; or more specifically, London.

2   Setting up the Workspace

The first few weeks of beginning the final project consisted of setting up the workspace for the group to work in. This consisted of gathering the data that we would be visualising, understanding what libraries we will be required to have access to in order to implement our ideas for visualisation and also familiarising ourselves with the library.

2.1   Gathering the Data

Our data was gathered from DATA.GOV.UK [Beta] (https://data.gov.uk). The data that were to be visualised were as follows:

- Unemployment Over Time
- Workforce Jobs Over Time
- Percentage of Students Pursuing Higher Further Education
- Statistics on Obesity, Physical Activity and Diet (England 2017)
- Information Exclaiming Salary Mobility amongst boroughs in London

As our objective was to produce visualisations concerning the living standards of British society, we thought it would be appropriate to make educated conclusions, based off of these sets of data.

2.2   Libraries

In order to implement complex and somewhat sophisticated visualisations, it was imperative to look at the specialist libraries that JavaScript has to offer in relation to data visualisation [2]. Coming across libraries such as 'Leaflet', 'Processing.js', 'VEGA' and many more; we came to find that the most popular library in terms of data visualisation was in fact 'D3.js' [3].

The reason behind our decision to use the 'D3.js' library was because it was extremely flexible with the web, which is very similar to 'p5.js'; a library that we were able to familiarise ourselves with in the initial part of the course. 'D3.js' allows for the manipulation of the HTML, CSS and SVG elements of the web. Having this flexibility with the DOM is a huge advantage as it allows for the visualisation to look like anything you want. It was unanimous that we would be implementing both 'p5.js' and 'D3.js' in order to visualise our data.

2.3   Planning and Development

The initial plan for the project was discussed during the first meeting. It was decided that communication would be practiced using 'Slack'; an online service offering a lot of IRC-like features and also most-importantly, file sharing features. Milestones were put in place so that there was a good track of where the project was at that moment in time.

As the project progressed further, there was an evident need to update our accessibility to file modification as there were many changes being made together and having to constantly share files over 'Slack' became pretty tedious. Hence, we took to GitHub, a web-based hosting service for version control using git. Basically, this allowed for the existence of a repository that held our files and saved changes being made. Instead of constantly sharing files over Slack, this was a frictionless way of developing the project.

3   Software Features

In terms of visualisations, five data sets were visualised. These consisted of bar charts, line graphs and pie charts. However, some data sets were representing multiple data; hence a more flexible approach was practiced – making use of multiple axes or including interactivity to represent the data. For example, it was not practical to assume the reading of the data from the bar chart as the exact figure cannot be read from the axis; a feature that displays the reading once the user hovers over the data was included in some of the visualisations. The application also makes use of a navigation bar that allows the user to access the different data visualisations.

4   Implementation

4.1   Efficiency

In order to have an efficient working environment and to have a good idea of all different files and folders, it was imperative that the folder was organised so that everything was clear. For example, the data sets that were being used were organised in a separate folder containing the data.

In terms of coding practice and programming experience, some parts of the code were having to be written multiple times or over multiple files. To increase efficiency, these lines of code became reusable once they were being called as appropriate functions.

4.2   Error Handling and Minimisation

Error handling and minimisation was taken into consideration so that progress would not be hindered as the application was developed. For example, general algebraic expressions were implemented (e.g. 'salary.js': line 59) so that it can be applied to various data sets instead of just singular sets of data; reducing the chances of running into errors.

However, in terms of how to tackle potential unforeseen problems, it was important to understand the options that were available to assist the group in overcoming these problems. For example, it was discussed that any issues would be tackled together so that efficiency is maximised.

More specifically, during moments in which the application suddenly blanks out and the code just typed somehow broke the application – we would refer to the console.log and find the error that we were facing. This was a good way of debugging the project and ensuring there was an understanding behind the reason as to why the program is behaving in the way it is.

4.3   Modularity

Each visualisation was saved in a separate '.js' file and called separately. The practice of having separate files in place for each visualisation proved to be a very smart idea as it was very easy to work in these sections. For example, having this project in one big file would be extremely impractical and frustrating to have everything clustered in one file. Furthermore, having this modularity allowed for ease of encapsulation so that code can be used from the other files instead of having those functions in the same file.

5   Evaluation

This project has turned out to be both challenging as well as interesting. With each stage of the project presenting its own problems to be overcome.

For example, becoming familiar with the 'D3.js' library [4] was a gradual process and some implementations were picked up on the go. One specific example would be understanding the 'd3-time-format' [5] – a module that allows for the manipulation of time formats. It was also imperative to further our existing understanding of the HTML, CSS and SVG side of programming. Data visualisation in 'D3.js' is made possible through the HTML, CSS and SVG elements of the web, thus being able to understand these elements makes things much easier.

Hamza Sheikh, Gabriel Sampaio Da Silva Diogo

Resources

Data:

- https://data.london.gov.uk/dataset/unemployment-rate-region
- https://data.london.gov.uk/dataset/workforce-jobs
- https://data.london.gov.uk/dataset/further-education-and-higher-education-destinations-ks5-students-borough-and-insti?q=further%20edu
- https://data.gov.uk/dataset/statistics_on_obesity_physical_activity_and_diet_england/resource/faa0747c-7557-4ce1-97db-8a40a07671e5
- https://data.london.gov.uk/dataset/earnings-workplace-borough

Libraries:

- P5.js
  https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.6.0/p5.js

- D3.js
  https://d3js.org/d3.v4.min.js

- jQuery
  https://code.jquery.com/jquery-3.3.1.min.js

- Bootstrap
  https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js

References

[1]   Sas.com. (2018). *Data Visualization: What it is and why it matters*. [Online]. Available: https://www.sas.com/en_gb/insights/big-data/data-visualization.html#dmtoday [Accessed 7 Apr. 2018].

[2]   Sonalake. (2018). *Popular open source JavaScript frameworks for data visualisation.* [Online]. Available: https://sonalake.com/latest/popular-open-source-javascript-frameworks-for-data-visualisation/ [Accessed 7 Apr. 2018].

[3]   Bostock, M. (2018). D3.js - Data-Driven Documents. [Online]. D3js.org. Available: https://d3js.org/ [Accessed 1 Mar. 2018].

[4]   Bostock, M. (2018). D3. [online] Available: https://github.com/d3 [Accessed 1 Mar. 2018].

[5]   GitHub. (2018). *d3/d3-time-format.* [Online]. Available: https://github.com/d3/d3-time-format [Accessed 8 Mar. 2018].