### PROJECT REPORT

### URBAN TRAJECTORY VISUALIZATION

Team 20

Rishi Koushal Naganathan

Sathyaram Venkatesan

Abinandaraj Rajendran

### **INTRODUCTION:**

The project involves a visual analytical system for taxi trajectory data that supports data exploration and analytical reasoning with interactive visual interfaces. The data includes the taxi route information from the city of New York over a certain period of time. It involves latitude, longitude, trip information (distance, speed etc.) and the tripid. Using these factors, we were provided with a prototype to begin with. It involved the area selection in the map to display the trajectories. Our goal was to create three level-1 diagrams (Scatter matrix, word cloud and chord diagram).

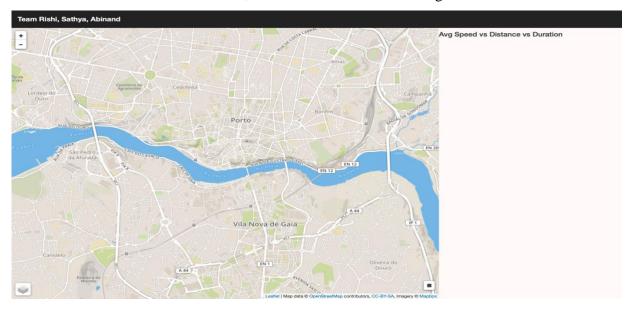
## PREREQUISITES AND INSTALLATION:

- Download 200 OK chrome extension that helps our folder to act as a local server. Then point out the downloaded folder to the 200 OK and click on the link. Once you deploy, you can drag and drop by selecting the small square in the right bottom corner. The visualizations appear on the right.
- Selecting a random area helps the system to understand the number of taxis and the trips in that selected portion. Now the region will help to feed the diagrams by their requirements that ultimately ends with interactive diagrams.
- Whenever a user selects a region, it develops the trajectory that involves specific area. Here, the red lines show the path of the trip and the small yellow dot describes the point of origin or destination. The trajectory of every taxi in the selected region looks appears as red line. Every red line in the diagram represents a trip across the streets from origin to destination. Thus, the yellow small dot helps to point out the ends of the trip segment.

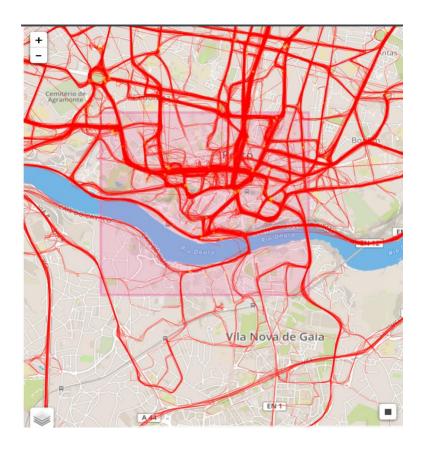
#### **WORKFLOW:**

The following are the steps to look into project. Whenever a user enters.

Step 1: Once the local server is switched on, the UI look like the following



Step 2: The user needs to select the area on the map, it displays the list pf trajectories like the following.



**Step 3:** Once the selection, the system automatically generates 3 diagrams. Scatterplot, word cloud and chord diagram. The user can play with the data points to see the interaction.



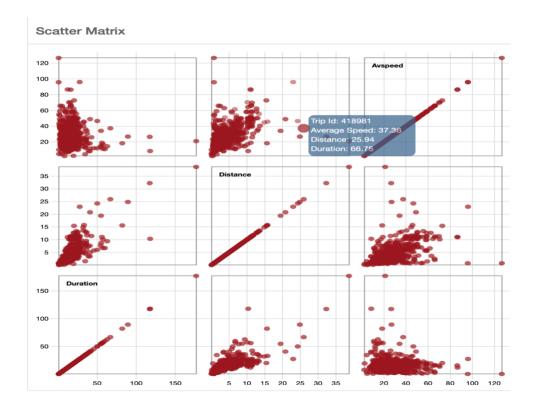
### **DESCRIPTION & EXPLANATION:**

Since there are 3 diagrams generated by the system, each have a specific meaning and reason that they were generated.

### **SCATTER MATRIX:**

The scatter matrix involves 3 different inputs from the user selection. It has the distance, duration and the average speed of that particular area. Here, it plots every taxi's inputs in a scatter plot such that it is spread across the x and y axis. It also has a mouse hover interaction that enables the user to see what that particular plot represents. Each record in trips contains average speed, distance and duration as values. Thus, passing them through the function helps to generate a scatter matrix

# function DrawScatterMatrix(trips)



User can click to see the specific taxi trajectory. It looks like:



The black ring indicates the origin point of the trip. Also, when the trip was selected, the word cloud also changes correspondingly.

## **WORD CLOUD:**

Word cloud is generally a visual representation of available words, here in our case, we took the word as the street names among the user selection. For a given json file, every specific taxi id has the list of street names that it has travelled. So, few taxi trips have single street, and few has around 100. So, the word cloud involves

all the street names that were selected. The street names were sorted with respect to the frequency of occurrences. Thus, the word cloud is a visualization of all the street names that the user has selected. Colors represent just the selection of the user. Word cloud used the following function:

## function DrawWordCloud(trips)

This gets the input of trips selected and the frequency count of the street names are also calculated. Duplicate or repetitive street names are ignored as the word cloud involved unique name.

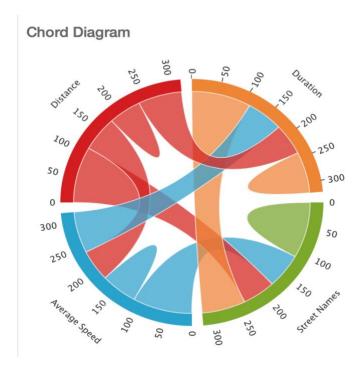


Word cloud changes when a specific point is selected in the scatter matrix.

### **CHORD DIAGRAM:**

Here, the chord diagram helps us to understand the values present in the selected area in terms of number of street names, avg. distance, avg. duration and average speed. Each data for the chord diagram will consist of the top 4 street names from the given selection. Thus, average speed, duration, distance with the top 4 street names with the most frequency is considered. The values will change only after every selection in the map and not specifically for a particular trip. The values as discussed in the scatter matrix will be present in the inputs that are arranged radically such that their representation will look like:

## function DrawChord(trips)



## **REFERENCES:**

Following are the reference for the project. The code snippets, reference papers and video/ blog tutorials from various sources like blocks.org, charts for d3.js, GitHub and stack overflow are listed below:

- Blocks.org for chord diagram Link 1
- Word cloud using d3.js Link 2
- Scatter matrix code snippet Link 3
- Chord Diagram using zender Link 4
- D3.js for responsive word cloud Link 5
- Visual index for d3.js charts Link 6
- Interactive data visualization using charts Link 7