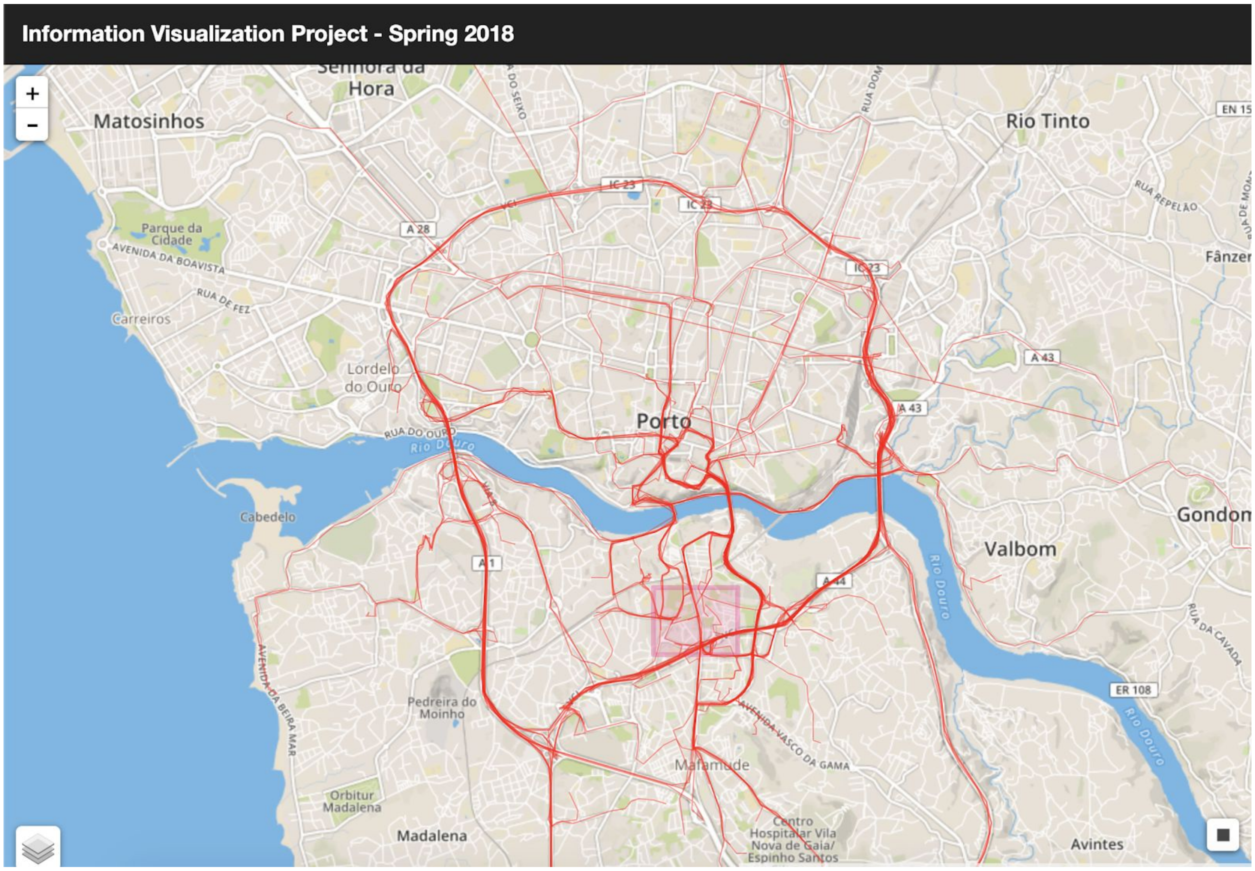
Information Visualization Project Report

Introduction:

Dataset consists of taxi trajectory data in Porto recorded as a series of positions associated with the vehicle attributes (**Av_Speed, Distance, Duration, Start_time, End_time, Max_Speed, Min_Speed, Street_Names, Taxi_ID, Trip_ID**) over urban road networks. A code with a web interface consisting of a map of Porto is provided. The following image shows the webpage:



Here, the red lines indicate the streets in Porto on which the taxis operated. We can select a particular area in Porto with a red rectangle

Goals:

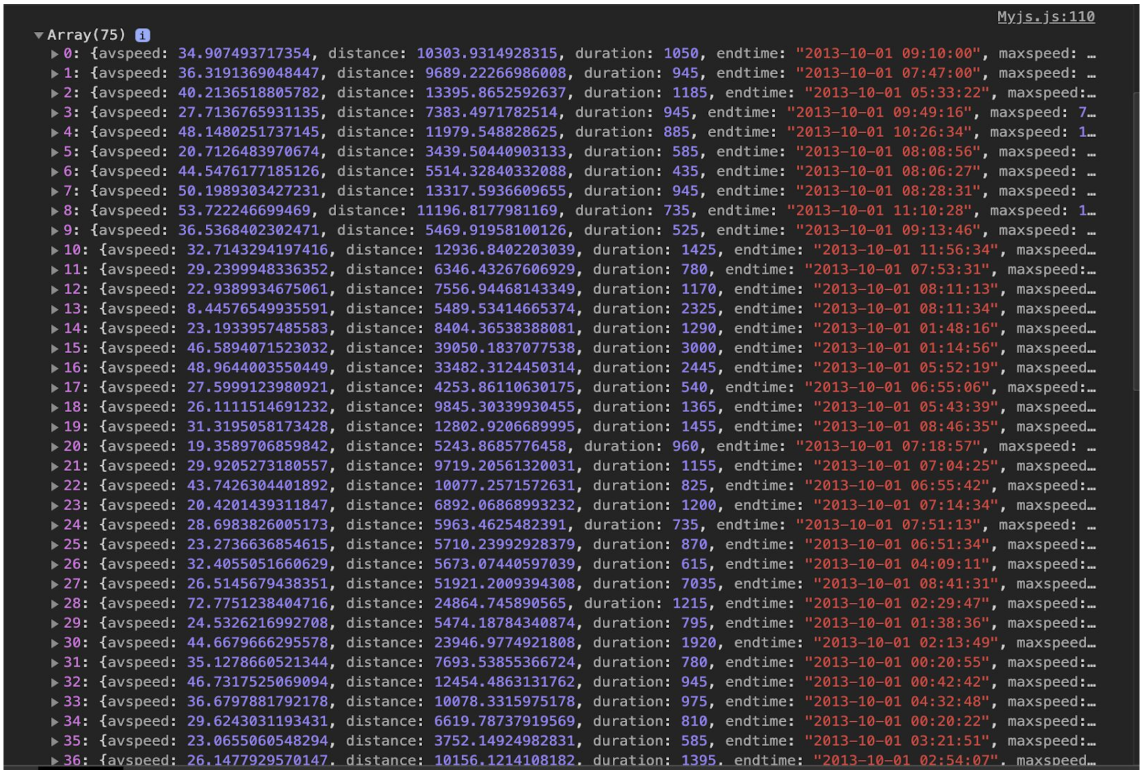
To implement a visual analytic system for taxi trajectory data that support data exploration and analytical reasoning with interactive visual interfaces. The system should help the user to conduct visual analytics tasks through an iterative, exploratory process.

Tasks:

1. Show the top pick-up and drop-off street names with its relation by using an Interactive Chord Diagram or Sankey Diagram.
2. Show the most frequent street names by using an Interactive Word Cloud.
3. Provide an interactive Scatter-Matrix to show the relationship between different attributes. For instance, Av_Speed, Distance and Duration.

Methods:

The dataset with the taxi trajectory information is outputted in a variable, ***results*** by developing an R-tree in the console. The following image highlights the dataset on the console:



```
                                                                                                 Myjs.js:110
▼Array(75) i
  ▶0: {avspeed: 34.907493717354, distance: 10303.9314928315, duration: 1050, endtime: "2013-10-01 09:10:00", maxspeed: …
  ▶1: {avspeed: 36.3191369048447, distance: 9689.22266986008, duration: 945, endtime: "2013-10-01 07:47:00", maxspeed: …
  ▶2: {avspeed: 40.2136518805782, distance: 13395.8652592637, duration: 1185, endtime: "2013-10-01 05:33:22", maxspeed:…
  ▶3: {avspeed: 27.7136765931135, distance: 7383.4971782514, duration: 945, endtime: "2013-10-01 09:49:16", maxspeed: 7…
  ▶4: {avspeed: 48.1480251737145, distance: 11979.548828625, duration: 885, endtime: "2013-10-01 10:26:34", maxspeed: 1…
  ▶5: {avspeed: 20.7126483970674, distance: 3439.50440903133, duration: 585, endtime: "2013-10-01 08:08:56", maxspeed: …
  ▶6: {avspeed: 44.5476177185126, distance: 5514.32840332088, duration: 435, endtime: "2013-10-01 08:06:27", maxspeed: …
  ▶7: {avspeed: 50.1989303427231, distance: 13317.5936609655, duration: 945, endtime: "2013-10-01 08:28:31", maxspeed: …
  ▶8: {avspeed: 53.722246699469, distance: 11196.8177981169, duration: 735, endtime: "2013-10-01 11:10:28", maxspeed: 1…
  ▶9: {avspeed: 36.5368402302471, distance: 5469.91958100126, duration: 525, endtime: "2013-10-01 09:13:46", maxspeed: …
  ▶10: {avspeed: 32.7143294197416, distance: 12936.8402203039, duration: 1425, endtime: "2013-10-01 11:56:34", maxspeed…
  ▶11: {avspeed: 29.2399948336352, distance: 6346.43267606929, duration: 780, endtime: "2013-10-01 07:53:31", maxspeed:…
  ▶12: {avspeed: 22.9389934675061, distance: 7556.94468143349, duration: 1170, endtime: "2013-10-01 08:11:13", maxspeed…
  ▶13: {avspeed: 8.44576549935591, distance: 5489.53414665374, duration: 2325, endtime: "2013-10-01 08:11:34", maxspeed…
  ▶14: {avspeed: 23.1933957485583, distance: 8404.36538388081, duration: 1290, endtime: "2013-10-01 01:48:16", maxspeed…
  ▶15: {avspeed: 46.5894071523032, distance: 39050.1837077538, duration: 3000, endtime: "2013-10-01 01:14:56", maxspeed…
  ▶16: {avspeed: 48.9644003550449, distance: 33482.3124450314, duration: 2445, endtime: "2013-10-01 05:52:19", maxspeed…
  ▶17: {avspeed: 27.5999123980921, distance: 4253.86110630175, duration: 540, endtime: "2013-10-01 06:55:06", maxspeed:…
  ▶18: {avspeed: 26.1111514691232, distance: 9845.30339930455, duration: 1365, endtime: "2013-10-01 05:43:39", maxspeed…
  ▶19: {avspeed: 31.3195058173428, distance: 12802.9206689995, duration: 1455, endtime: "2013-10-01 08:46:35", maxspeed…
  ▶20: {avspeed: 19.3589706859842, distance: 5243.8685776458, duration: 960, endtime: "2013-10-01 07:18:57", maxspeed: …
  ▶21: {avspeed: 29.9205273180557, distance: 9719.20561320031, duration: 1155, endtime: "2013-10-01 07:04:25", maxspeed…
  ▶22: {avspeed: 43.7426304401892, distance: 10077.2571572631, duration: 825, endtime: "2013-10-01 06:55:42", maxspeed:…
  ▶23: {avspeed: 20.4201439311847, distance: 6892.06868993232, duration: 1200, endtime: "2013-10-01 07:14:34", maxspeed…
  ▶24: {avspeed: 28.6983826005173, distance: 5963.4625482391, duration: 735, endtime: "2013-10-01 07:51:13", maxspeed: …
  ▶25: {avspeed: 23.2736636854615, distance: 5710.23992928379, duration: 870, endtime: "2013-10-01 06:51:34", maxspeed:…
  ▶26: {avspeed: 32.4055051660629, distance: 5673.07440597039, duration: 615, endtime: "2013-10-01 04:09:11", maxspeed:…
  ▶27: {avspeed: 26.5145679438351, distance: 51921.2009394308, duration: 7035, endtime: "2013-10-01 08:41:31", maxspeed…
  ▶28: {avspeed: 72.7751238404716, distance: 24864.745890565, duration: 1215, endtime: "2013-10-01 02:29:47", maxspeed:…
  ▶29: {avspeed: 24.5326216992708, distance: 5474.18784340874, duration: 795, endtime: "2013-10-01 01:38:36", maxspeed:…
  ▶30: {avspeed: 44.6679666295578, distance: 23946.9774921808, duration: 1920, endtime: "2013-10-01 02:13:49", maxspeed…
  ▶31: {avspeed: 35.1278660521344, distance: 7693.53855366724, duration: 780, endtime: "2013-10-01 00:20:55", maxspeed:…
  ▶32: {avspeed: 46.7317525069094, distance: 12454.4863131762, duration: 945, endtime: "2013-10-01 00:42:42", maxspeed:…
  ▶33: {avspeed: 36.6797881792178, distance: 10078.3315975178, duration: 975, endtime: "2013-10-01 04:32:48", maxspeed:…
  ▶34: {avspeed: 29.6243031193431, distance: 6619.78737919569, duration: 810, endtime: "2013-10-01 00:20:22", maxspeed:…
  ▶35: {avspeed: 23.0655060548294, distance: 3752.14924982831, duration: 585, endtime: "2013-10-01 03:21:51", maxspeed:…
  ▶36: {avspeed: 26.1477929570147, distance: 10156.1214108182, duration: 1395, endtime: "2013-10-01 02:54:07", maxspeed…
```

Sankey Diagram:

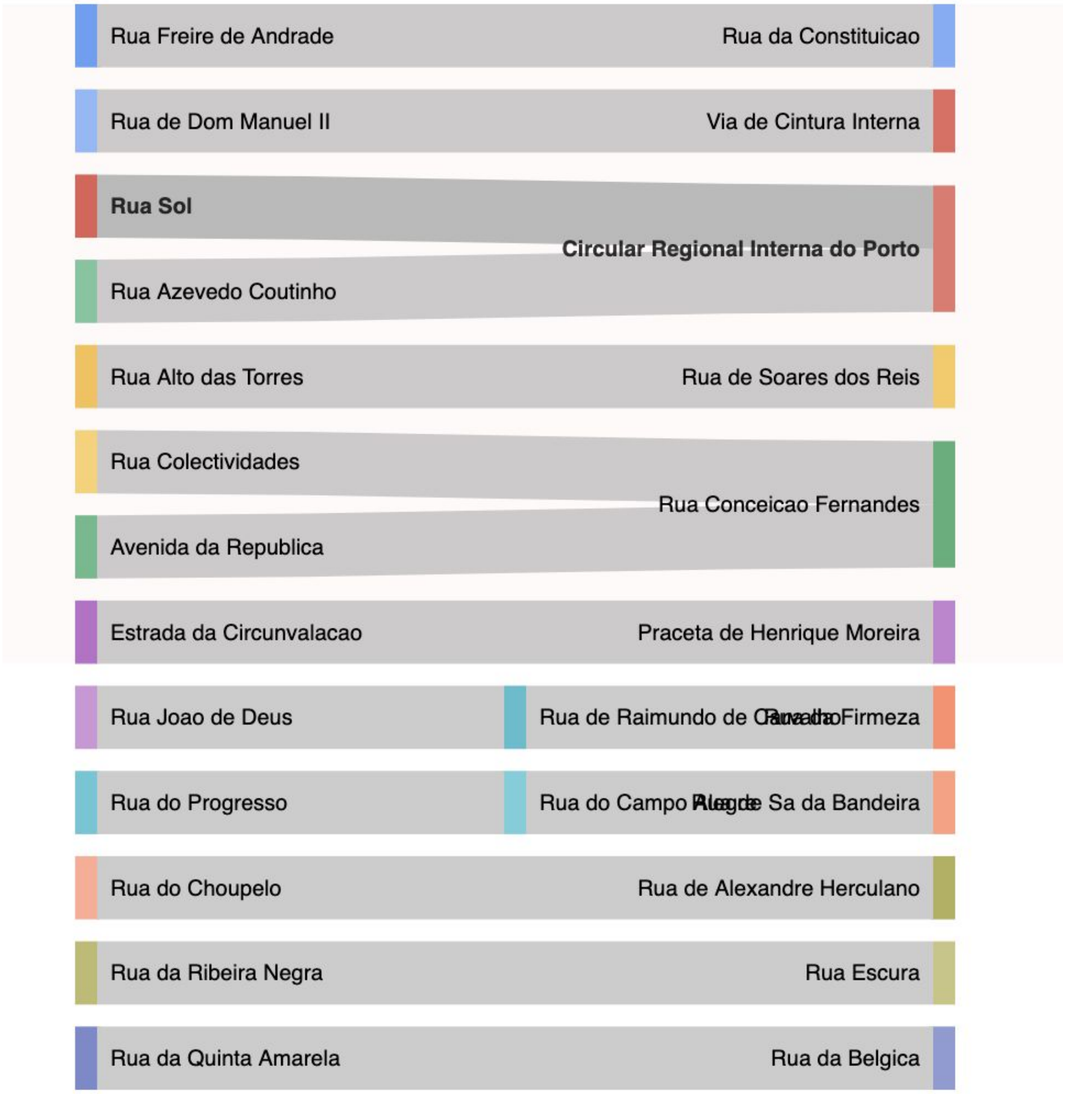
A Sankey diagram is a visualization used to depict a flow from one set of values to another. The things being connected are called nodes and the connections are called links. Sankeys are best used when you want to show a many-to-many mapping between two domains (e.g., universities and majors) or multiple paths through a set of stages (for instance, Google Analytics uses Sankey to show how traffic flows from pages to other pages on your web site). For the curious, they're named after Captain Sankey, who created a diagram of steam engine efficiency that used arrows having widths proportional to heat loss. [REF]

```
<html>
  <head>
    <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
    <script type="text/javascript">
      google.charts.load('current', {'packages':['sankey']});
      google.charts.setOnLoadCallback(drawChart);

      function drawChart() {
        var data = new google.visualization.DataTable();
        data.addColumn('string', 'From');
        data.addColumn('string', 'To');
        data.addColumn('number', 'Weight');
        data.addRows([
          [ 'A', 'X', 5 ],
          [ 'A', 'Y', 7 ],
          [ 'A', 'Z', 6 ],
          [ 'B', 'X', 2 ],
          [ 'B', 'Y', 9 ],
          [ 'B', 'Z', 4 ]
        ]);
        // Sets chart options.
        var options = {
          width: 600,
        };

        // Instantiates and draws our chart, passing in some options.
        var chart = new google.visualization.Sankey(document.getElementById('sankey_basic'));
        chart.draw(data, options);
      }
    </script>
  </head>
  <body>
    <div id="sankey_basic" style="width: 900px; height: 300px;"></div>
  </body>
</html>
```

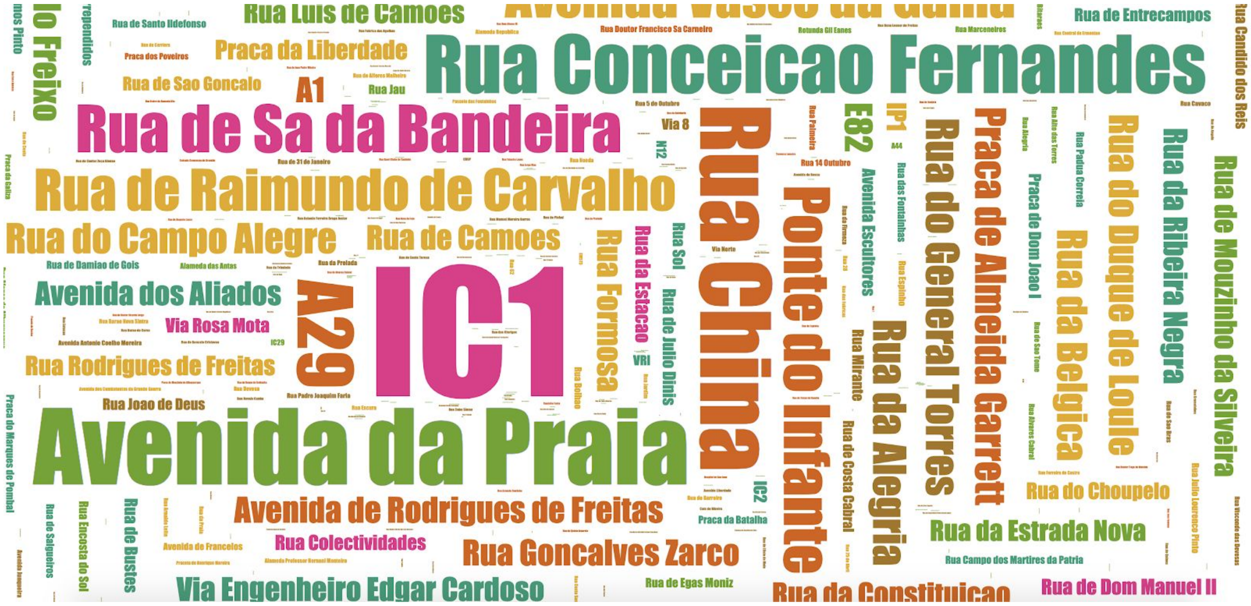
The above code provides a simple Sankey diagram connecting the nodes A to X having a weight of 5, A to Y having a weight of 7, and so on. Our results matrix consists of a key ***streetnames*** which stores the start of the trip and the destination of the trip. For weights, we calculate the number of times a taxi takes the same route. **The following image displays the Sankey diagram provided to the users by our system**:



Word Cloud

Word clouds (also known as text clouds or tag clouds) work in a simple way: the more a specific word appears in a source of textual data (such as a speech, blog post, or database), the bigger and bolder it appears in the word cloud. A word cloud is a collection, or cluster, of words depicted in different sizes. The bigger and bolder the word appears, the more often it's mentioned within a given text and the more important it is. [REF].

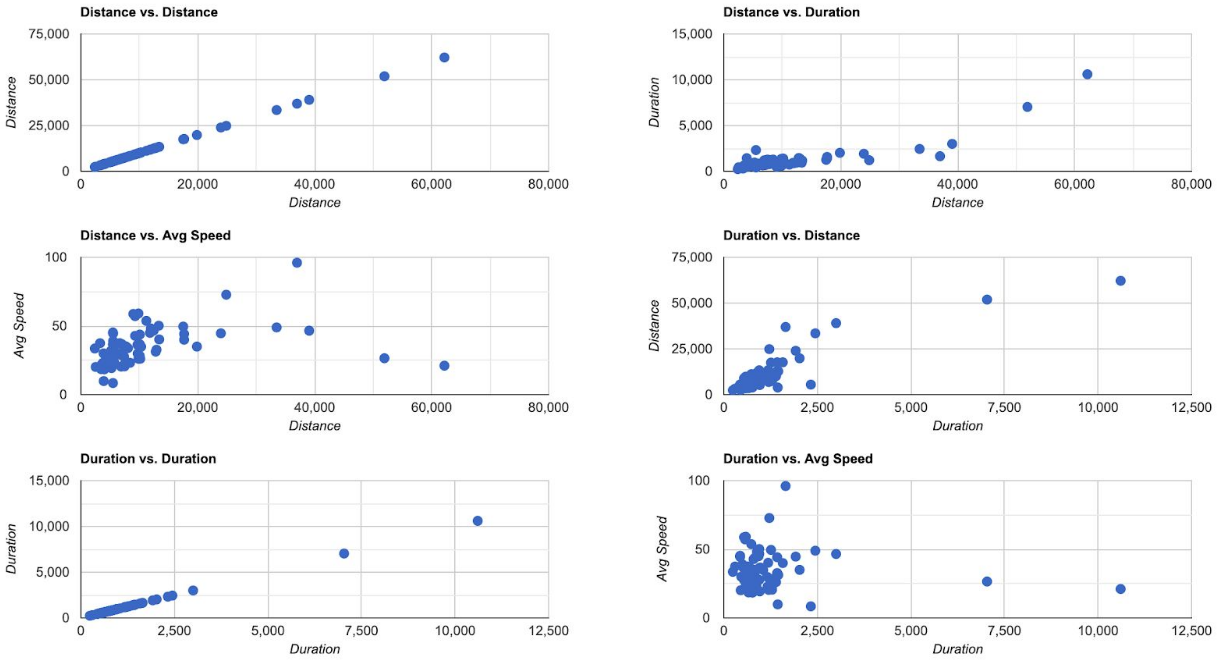
The following image displays the Word Cloud provided to the users by our system:



Scatter Matrix:

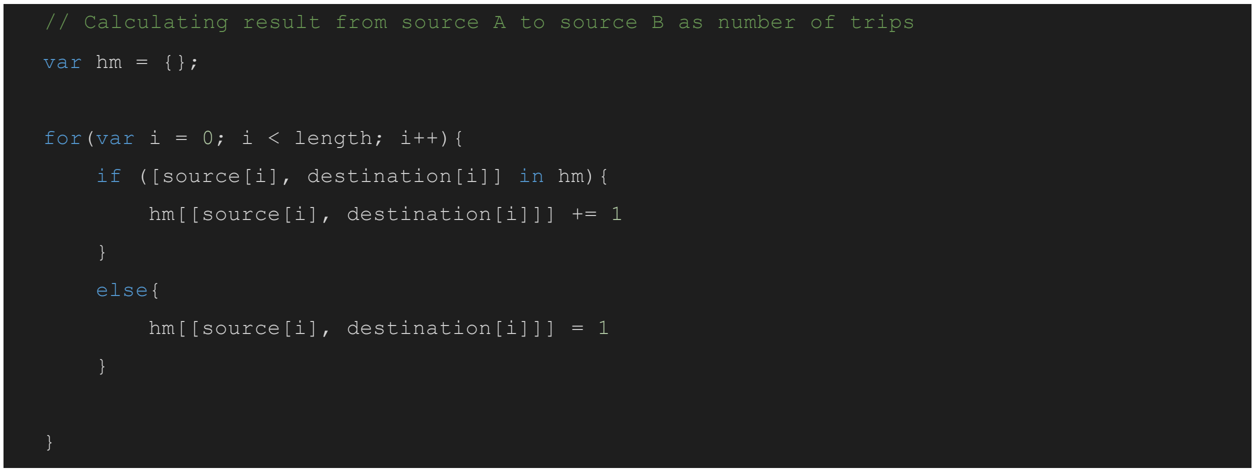
A scatter matrix consists of several pair-wise scatter plots of variables presented in a matrix format. It can be used to determine whether the variables are correlated and whether the correlation is positive or negative.[REF]

The following image displays the Scatter Matrix provided to the users by our system:



Observations and Conclusions:

The Sankey diagram provides users an efficient visual representation of a taxi path and the number of times taxis traverse on the same path. This calculation of weights required a ***hashmap*** data structure to store the weights. Here is the code snippet used:



```
// Calculating result from source A to source B as number of trips
var hm = {};

for(var i = 0; i < length; i++){
    if ([source[i], destination[i]] in hm){
        hm[[source[i], destination[i]]] += 1
    }
    else{
        hm[[source[i], destination[i]]] = 1
    }

}
```

The Word Cloud provides users an efficient way to analyze the most frequently traversed street names. Each word or phrase represents a street in Porto. The size of the word tells us the rate at which the street was busy.

The Scatter matrix has different scatter plots associated with distance traveled by taxis, the duration of time taken by taxis and the average speed of taxis. Each blue dot denotes a trip id. Hovering over the dots provides us with attributes such as average speed, average distance, etc. associated with the trip.

References:

Note: The codes for all three diagrams are referenced from online sources and adapted to take input of the attributes of our results matrix.

- https://bl.ocks.org/blockspring/847a40e23f68d6d7e8b5
- https://observablehq.com/@d3/brushable-scatterplot-matrix
- https://github.com/ManjunathBirajdar/UrbanTrajectoryViz
- https://developers.google.com/chart/interactive/docs/gallery/sankey