Motor Vehicle Collisions in New York City

How can we produce useful visualisations to identify critical collisions dedicated to improving traffic safety?

Emmanuel Tan Sheng Wei, Gan Shao Hong, Wang Shurui, Yong Wei Theen, Yu Mengting

01 **Introduction**

- Problem Statement
- Motivation
- Literature Review

02 **Dataset**

- Data Source
- Tools and Resources

03 **Visualisation**

- Tasks
- Method
- Result & Discussion
- Demo

04 **Conclusion**

- Project Milestones/Schedule
- Future Work

01

INTRODUCTION

PROBLEM STATEMENT

Topic:

Motor Vehicle Collisions in New York City

Title:

How can we produce useful visualisations to identify critical collisions and trends dedicated to improving traffic safety?

MOTIVATION

Singapore

New York City

THE STRAITS TIMES

5 killed in Tanjong Pagar crash: 29-year-old driver believed to have sped before crashing into shophouse

Population: **8.6** million

Households with cars : **1.4** million

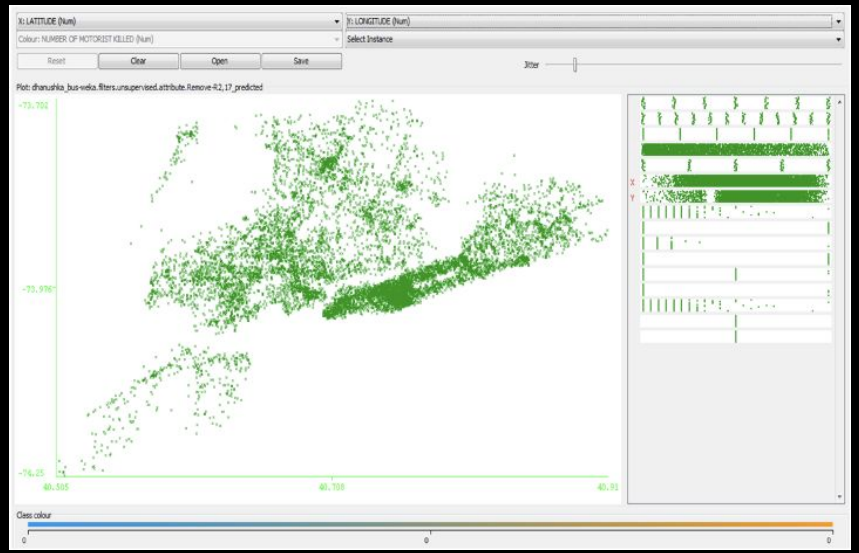
Worst Traffic in the US in 2021

303 car accidents daily in 2020

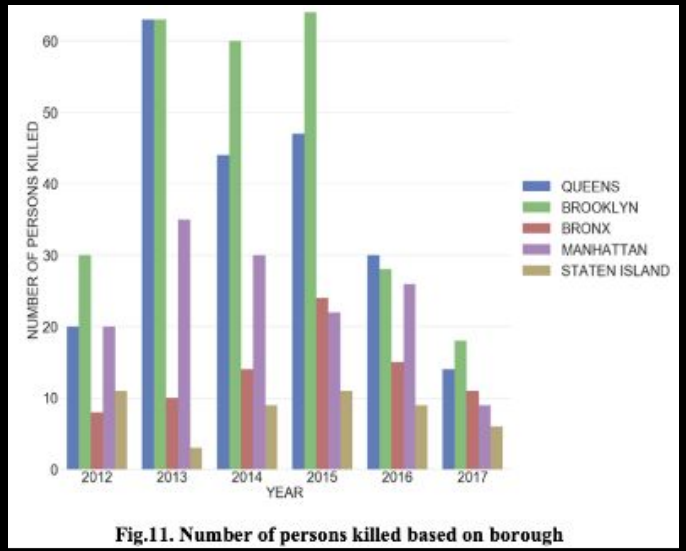
LITERATURE REVIEW What has been done?

Abeyratne et al. (2020):

Applying Big Data Analytics on Motor Vehicle Collision Predictions in New York City



Representing latitude and longitude on a heatmap

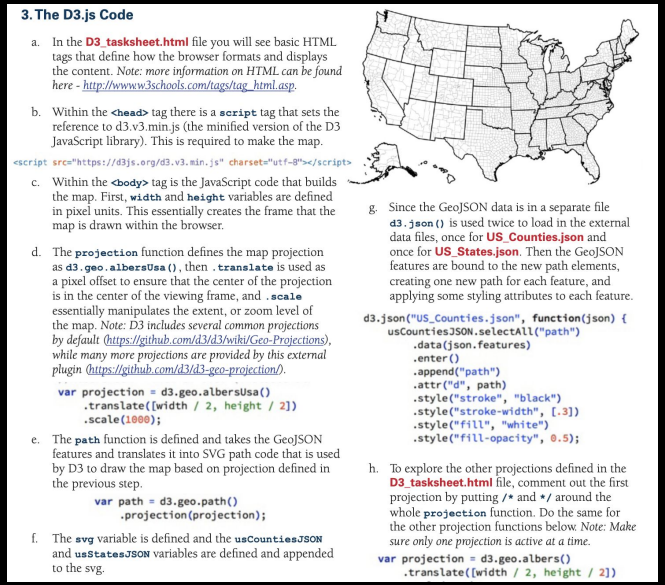


Representing the number of persons killed based on borough and year

LITERATURE REVIEW What needs to be done?

Hanson et al. (2016):

D3.js: Introduction to Mapping



3. The D3.js Code

a. In the **D3_tasksheet.html** file you will see basic HTML tags that define how the browser formats and displays the content. *Note: more information on HTML can be found here - http://wwww3schools.com/tags/tag_html.asp.*

b. Within the **<head>** tag there is a **script** tag that sets the reference to d3.v3.min.js (the minified version of the D3 JavaScript library). This is required to make the map.

```
<script src="https://d3js.org/d3.v3.min.js" charset="utf-8"></script>
```

c. Within the **<body>** tag is the JavaScript code that builds the map. First, **width** and **height** variables are defined in pixel units. This essentially creates the frame that the map is drawn within the browser.

d. The **projection** function defines the map projection as **d3.geo.albersUsa()**, then **.translate** is used as a pixel offset to ensure that the center of the projection is in the center of the viewing frame, and **.scale** essentially manipulates the extent, or zoom level of the map. *Note: D3 includes several common projections by default (https://github.com/d3/d3/wiki/Geo-Projections), while many more projections are provided by this external plugin (https://github.com/d3/d3-geo-projection/).*

```
var projection = d3.geo.albersUsa()
    .translate([width / 2, height / 2])
    .scale(1000);
```

e. The **path** function is defined and takes the GeoJSON features and translates it into SVG path code that is used by D3 to draw the map based on projection defined in the previous step.

```
var path = d3.geo.path()
    .projection(projection);
```

f. The **svg** variable is defined and the **usCountiesJSON** and **usStatesJSON** variables are defined and appended to the svg.

g. Since the GeoJSON data is in a separate file **d3.json()** is used twice to load in the external data files, once for **US_Counties.json** and once for **US_States.json**. Then the GeoJSON features are bound to the new path elements, creating one new path for each feature, and applying some styling attributes to each feature.

```
d3.json("US_Counties.json", function(json) {
    usCountiesJSON.selectAll("path")
        .data(json.features)
        .enter()
        .append("path")
        .attr("d", path)
        .style("stroke", "black")
        .style("stroke-width", [.3])
        .style("fill", "white")
        .style("fill-opacity", 0.5);
```

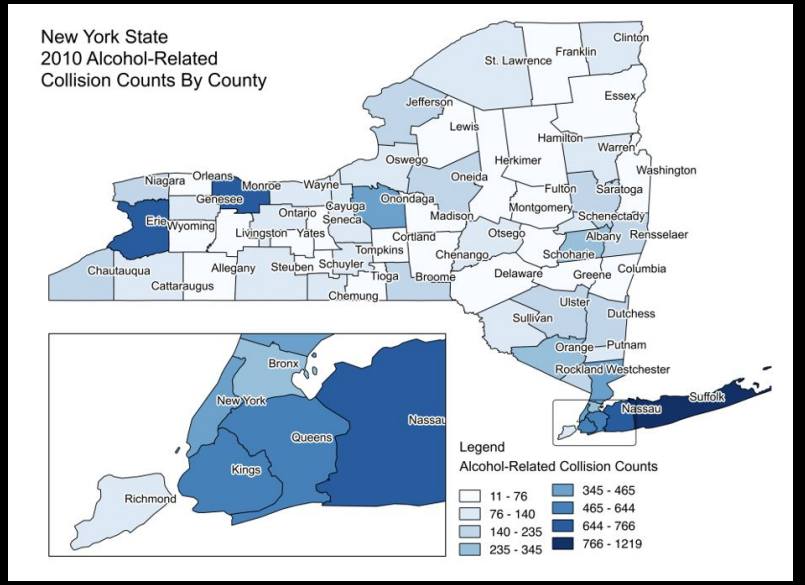
h. To explore the other projections defined in the **D3_tasksheet.html** file, comment out the first projection by putting **/*** and ***/** around the whole **projection** function. Do the same for the other projection functions below. *Note: Make sure only one projection is active at a time.*

```
var projection = d3.geo.albers()
    .translate([width / 2, height / 2])
```

Detailed steps on how to create a map with D3.js

Peck J L. (2017):

New York City Drunk Driving After Uber

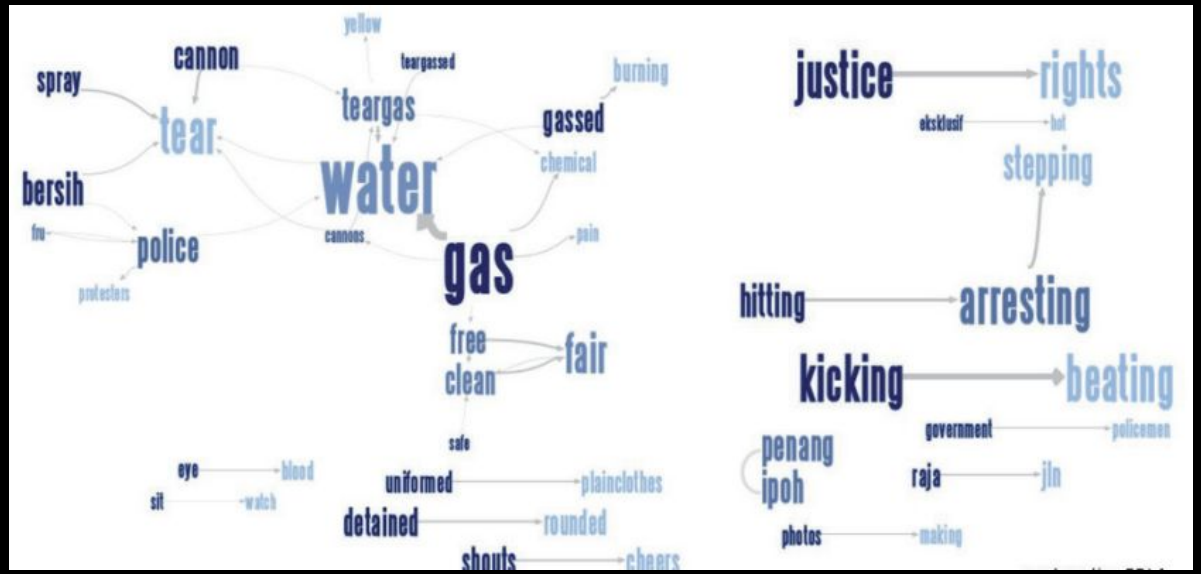


Representing count of alcohol-related collisions by county in New York State with a choropleth map

LITERATURE REVIEW What needs to be done?

Lim and Merlyna (2018):

Sticks and Stones, Clicks and Phones: Contextualizing the Role of Digital Media in the Politics of Transformation.



Phrase Nets can be used to visualise relationships between various words used in tweets to understand the role of social media in politics transformation

TOOLS AND RESOURCES USED

- Generate simple graphs to understand our dataset
- Decide on which types of graphs would work best with our dataset.
- Allow us to recognize interesting trends and patterns with less overhead.

- Transition
- Map Projection
- Path
- Colour Mapping with legend
- Research on Scale:
  - scaleQuantile() vs scaleQuantize()
  - scaleBand() vs scaleLinear()

rSlider.js

02

DATASET

OUR DATA

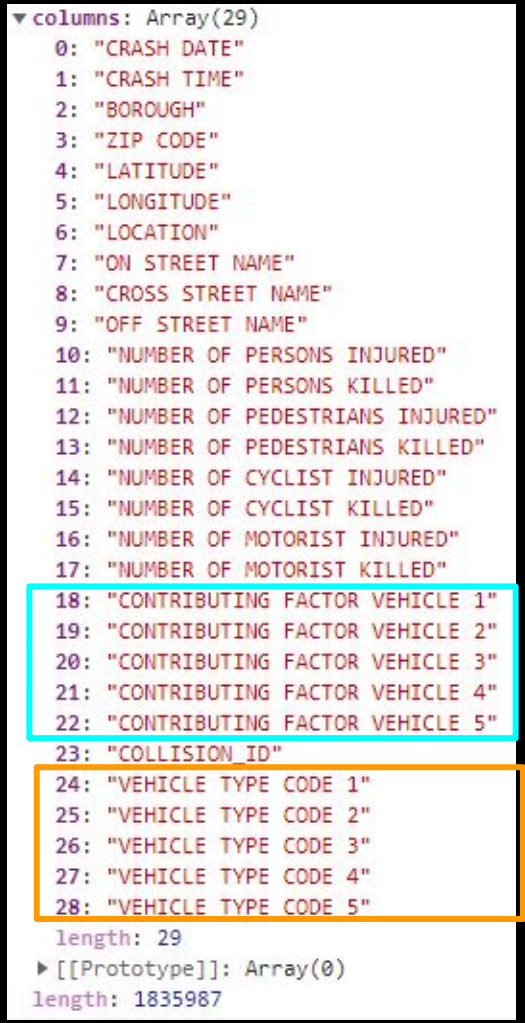
opendata.cityofnewyork.us

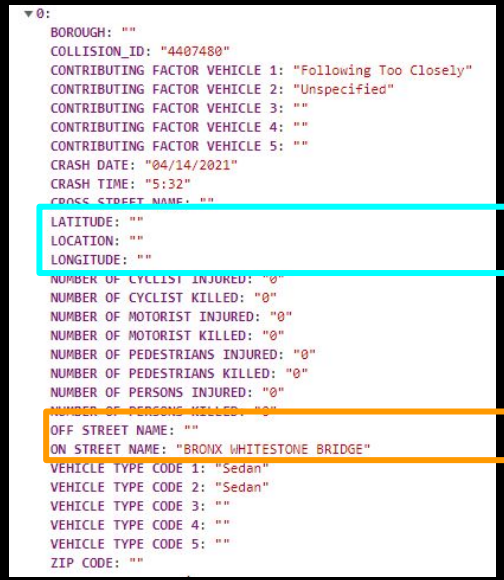
Raw Dataset:
https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95

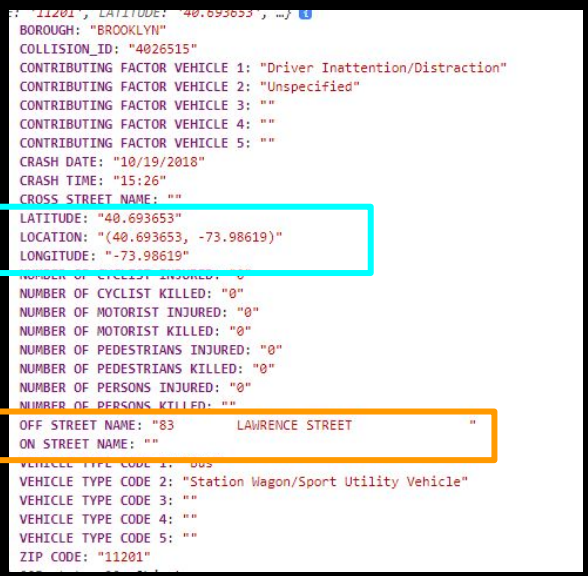
Borough Boundary Map (GeoJSON):
https://data.cityofnewyork.us/City-Government/Borough-Boundaries/tqmj-j8zm

Raw Dataset

- 1,835,987 data points (Excel row limit: 1,048,576)
  - Either process the data in Javascript or using Python to populate the processed data
- 29 Columns





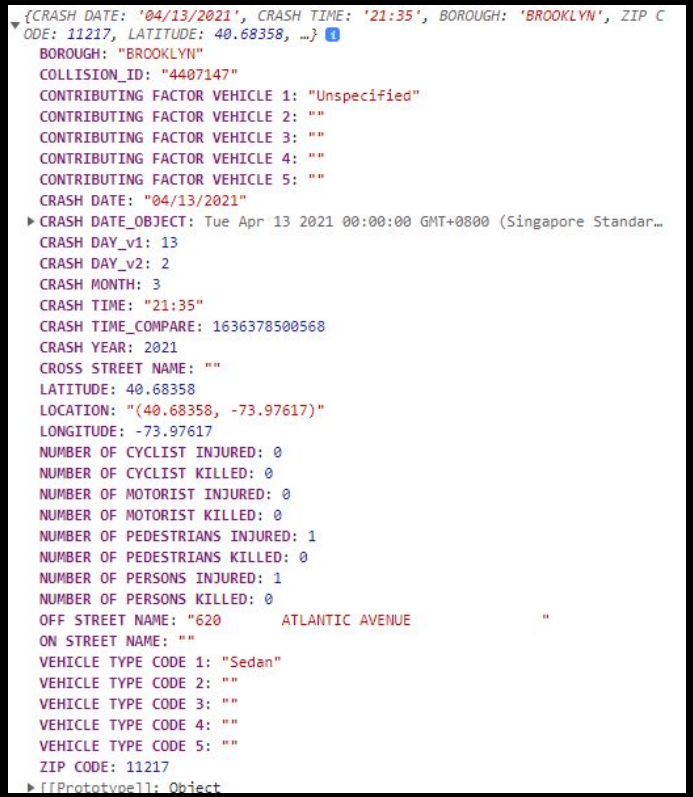


Load.js

- Rescope the data points
  - Only include those that has valid Borough and Location (Lat, Long)
  - Original Length: 1,835,987; Filtered Length: 1,233,491

- Process time
- *(“CRASH DATE” e.g. “04/13/2021”), (“CRASH TIME” e.g. “21:35”)*
  - Month, Day and Year
  - Date Object for comparison

- Conversion of String to Int and empty string to 0
  - ```
    (!!d[i]['NUMBER OF PERSONS KILLED']) ?
    parseInt(d[i]['NUMBER OF PERSONS KILLED']) : 0
    ```


    ```

Vehicle Collision Contributing Factor

- Many contributing factors are similar in nature
  - “Unsafe Speed….Unsafe Lane Changing….Failure to Keep Right”
  - “Brakes Defective… Accelerator Defective…Tire Failure/Inadequate”
  - “Fell Asleep…Lost Consciousness…Fatigued/Drowsy”

- Group factors that are similar
  - “Improper Driving” - [“Unsafe Speed”, “Unsafe Lane Changing”…]
  - “Defects in Vehicle” - [“Brakes Defective”, “Accelerator Defective”…]
  - “Driver unconscious” - [“Fell Asleep”, “Lost Consciousness”…]

Relationships between collisions

EDA:

- Empty records (NaN) for boroughs, Vehicle Type Code 1 & 2 are dropped

	BOROUGH	VEHICLE TYPE CODE 1	VEHICLE TYPE CODE 2	VEHICLE TYPE CODE 3	VEHICLE TYPE CODE 4	VEHICLE TYPE CODE 5	NUMBE O PERSON INJURE
6	QUEENS	Sedan	Sedan	NaN	NaN	NaN	0
13	BROOKLYN	❖MBU	Taxi	NaN	NaN	NaN	0
17	BROOKLYN	Sedan	Sedan	NaN	NaN	NaN	1
18	STATEN ISLAND	Station Wagon/Sport Utility	Station Wagon/Sport Utility	NaN	NaN	NaN	7

- Sum all the injuries and deaths per collision Drop rows where sum = 0.

NUMBER OF PEDESTRIANS INJURED	NUMBER OF PEDESTRIANS KILLED	NUMBER OF CYCLIST INJURED	NUMBER OF CYCLIST KILLED	NUMBER OF MOTORIST INJURED	NUMBER OF MOTORIST KILLED	weight
0	0	0	0	0	0	0.0
0	0	0	0	0	0	0.0
0	0	0	0	1	0	2.0
0	0	0	0	7	0	14.0

- Calculated frequency of vehicles being involved across boroughs & log transform

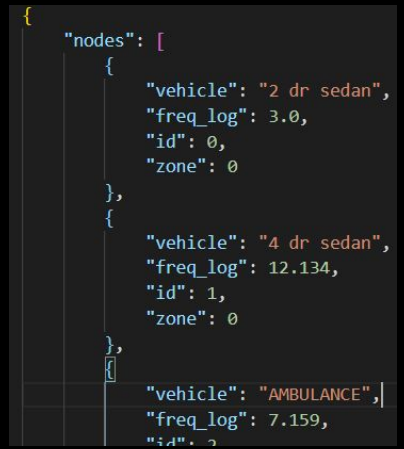
BOROUGH	type	vehicle	freq	freq_log
BRONX	VEHICLE TYPE CODE 1	2 dr sedan	1	3.000
BRONX	VEHICLE TYPE CODE 1	4 dr sedan	21	12.134
BRONX	VEHICLE TYPE CODE 1	AMBULANCE	4	7.159
BRONX	VEHICLE TYPE CODE 1	Ambulance	2	5.079
BRONX	VEHICLE TYPE CODE 1	BICYCLE	1	3.000

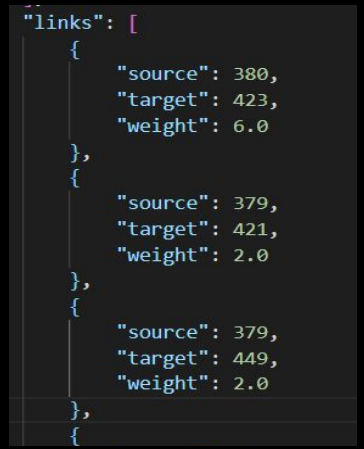
- Identified the ‘source’ and ‘target’ in each collision

BOROUGH	VEHICLE TYPE CODE 1	VEHICLE TYPE CODE 2	VEHICLE TYPE CODE 3	VEHICLE TYPE CODE 4	VEHICLE TYPE CODE 5	weight	ids	ids combined	source	target
QUEENS	Station Wagon/Sport Utility Vehicle	Taxi	NaN	NaN	NaN	6.0	[380, 423]	(380, 423)	380	423
QUEENS	Sedan	Station Wagon/Sport Utility Vehicle	Station Wagon/Sport Utility Vehicle	NaN	NaN	2.0	[379, 421, 449]	(379, 421)	379	421
QUEENS	Sedan	Station Wagon/Sport Utility Vehicle	Station Wagon/Sport Utility Vehicle	NaN	NaN	2.0	[379, 421, 449]	(379, 449)	379	449
QUEENS	Sedan	Station Wagon/Sport Utility Vehicle	Station Wagon/Sport Utility Vehicle	NaN	NaN	2.0	[379, 421, 449]	(421, 449)	421	449
BROOKLYN	Bike	Station Wagon/Sport Utility Vehicle	NaN	NaN	NaN	2.0	[112, 187]	(112, 187)	112	187

Data source (Summary)

1. Raw Dataset (Motor_Vehicle_Collisions_-_Crashes.csv)
2. Borough Boundary GeoJSON (Borough Boundaries.geojson)
   a. Values will be appended to the geojson.features to access and create map visualization laters
3. Python EDA script
4. Json Files - Nodes, Relationships (Links) and collectively node links





Challenges

Challenges:

- Dataset was too large
  - Cannot process through Excel and view the whole dataset at a glance

- Had many inconsistencies

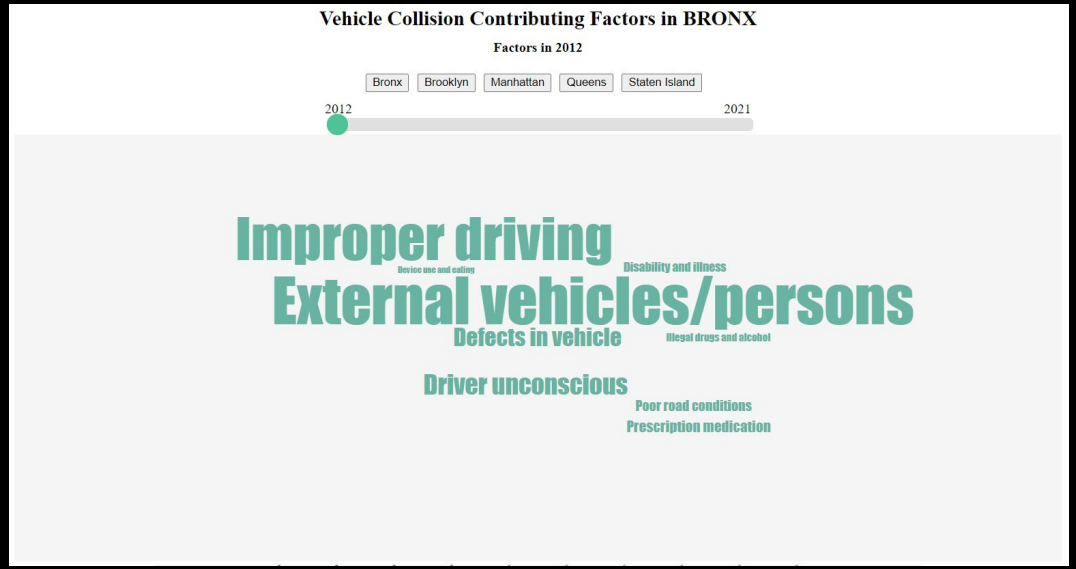
03

VISUALISATION

TASKS

No.	TASK
1.	Identify prominent factors that contribute to vehicle collisions and how they change over time for each borough
2.	Examine the number of collisions between different boroughs at different years, months, days and hours
3.	Explore the sum and average of injuries and deaths across different personnel groups at different years, months, days, days of week, and hours
4.	Visualise the severity between collisions by injuries and deaths across different boroughs
5.	Examine casualty rate and number of collisions across different boroughs

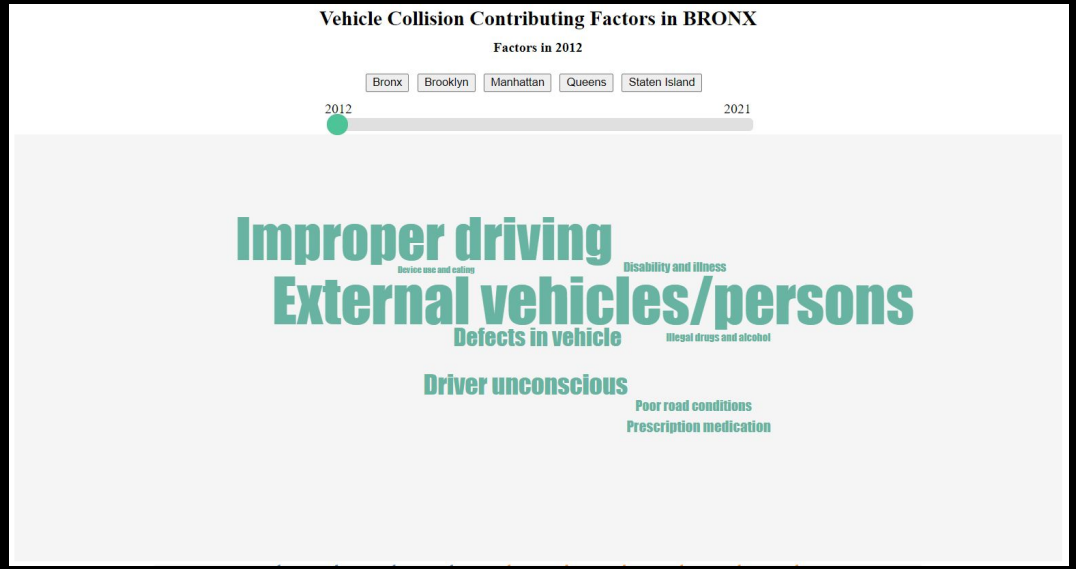
METHOD: WordCloud



Visual Encodings:

- Buttons to filter by borough
- Slider to filter across year
- Font size: Larger for higher occurrence, Smaller for lower occurrence

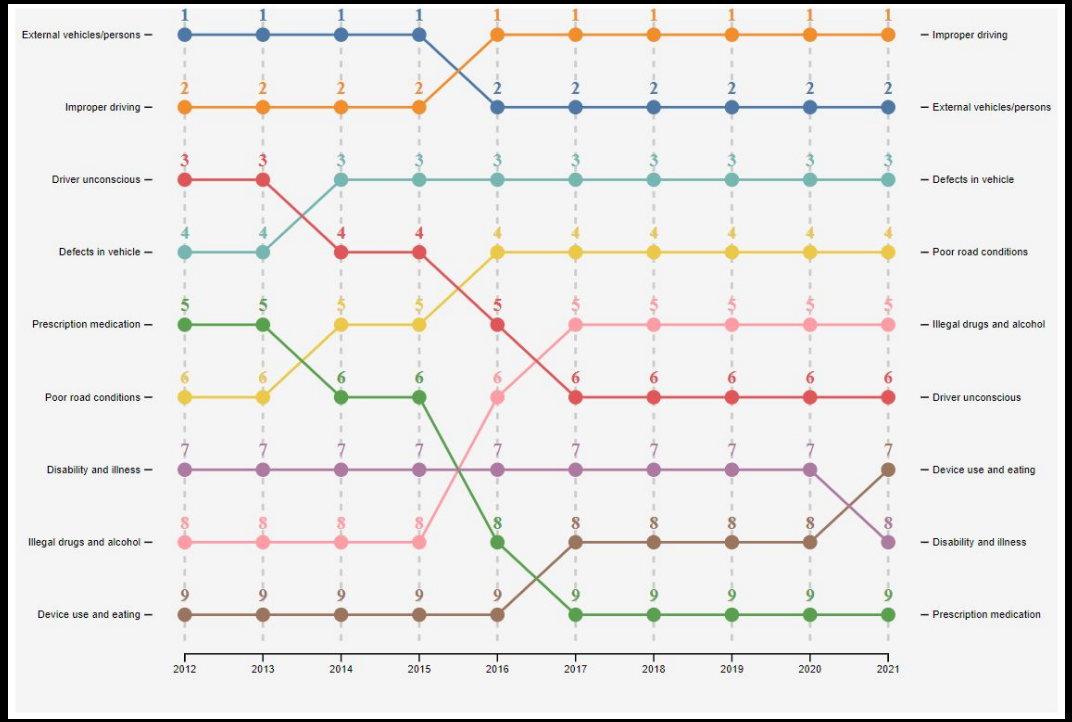
METHOD: WordCloud



Challenges:

- Ensuring word cloud can be displayed due to different font sizes and limiting width of svg
- Font size of factors are not proportionate enough to see difference in sizes especially for the ones lower in occurrence, as occurrences can range from 1 to >5000
- Found a better way to visualise factors by their occurrence (Bump Chart)

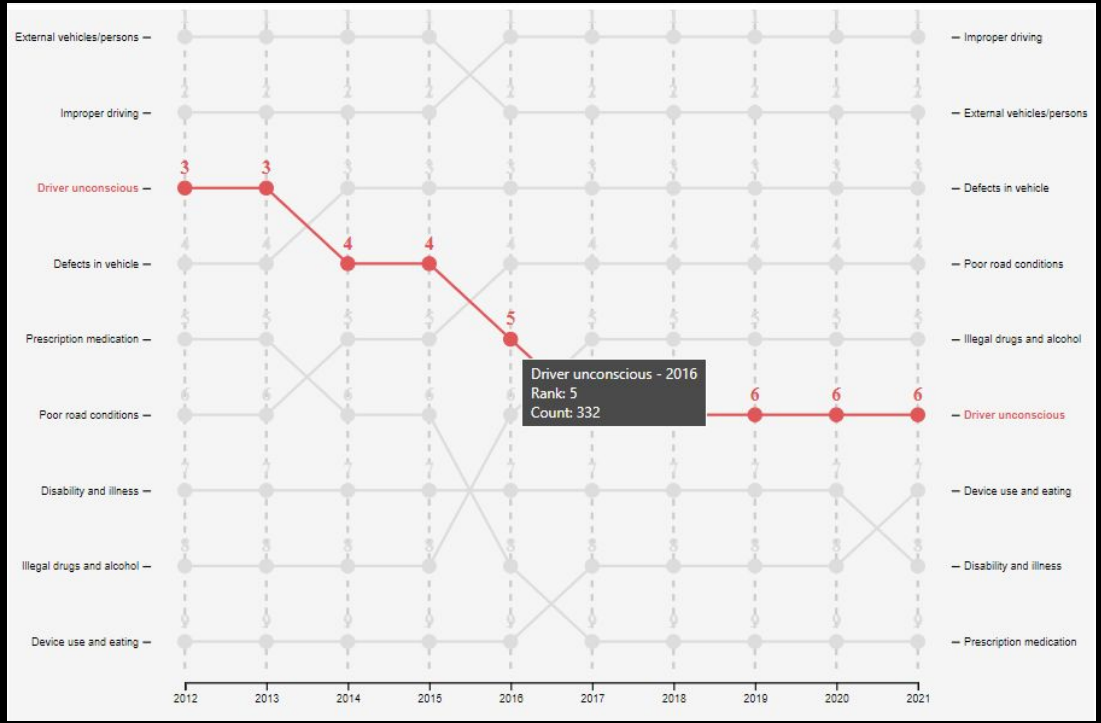
METHOD: Bump Chart



Visual Encodings:

- X Axis: Year intervals
- Y Axis: Vehicle collision contributing factors by ranking
- Numbers indicate ranking in occurrence: 1 - highest, 9 - lowest
- Different colours for each factor line

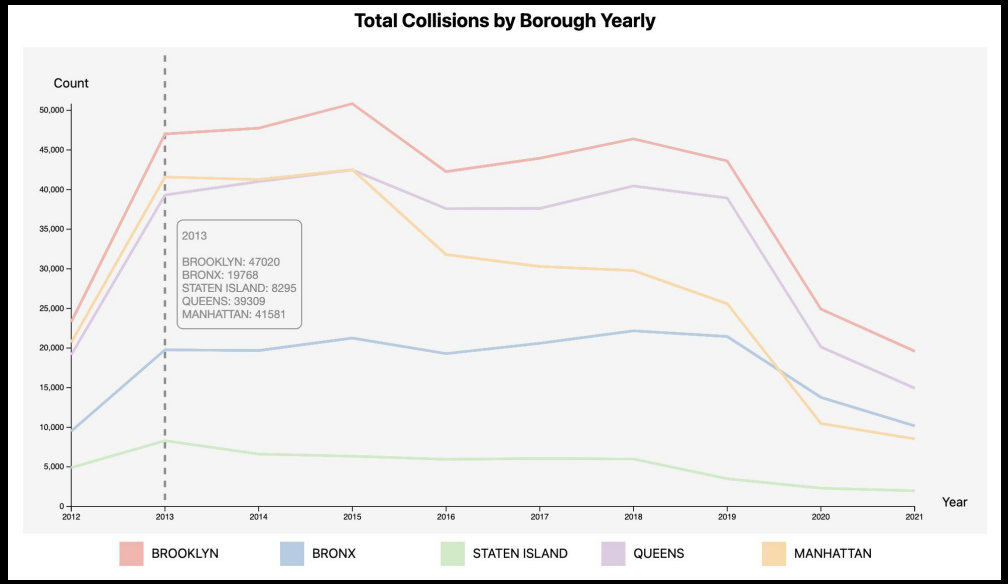
METHOD: Bump Chart



Visual Encodings:

- Tooltip on each point to indicate the factor, rank, and occurrence
- Line is focused on hover reduce visual clutter

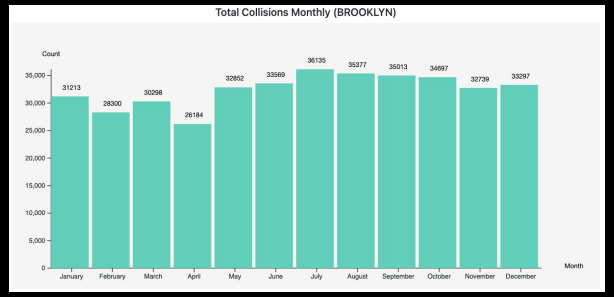
METHOD: Line Chart

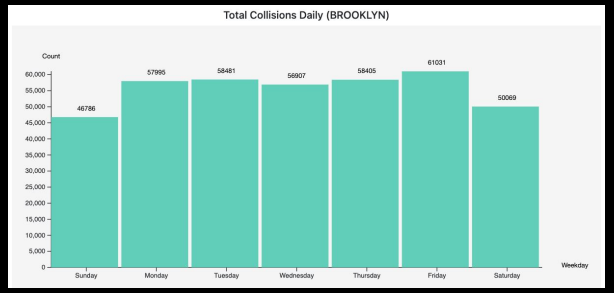


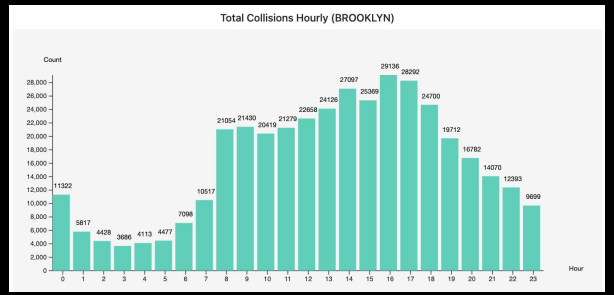
Visual Encodings:

- Total Collision Count against Year
- Categorical Colour Map for each borough
- Tooltip for specific year across all boroughs

METHOD: Bar Chart



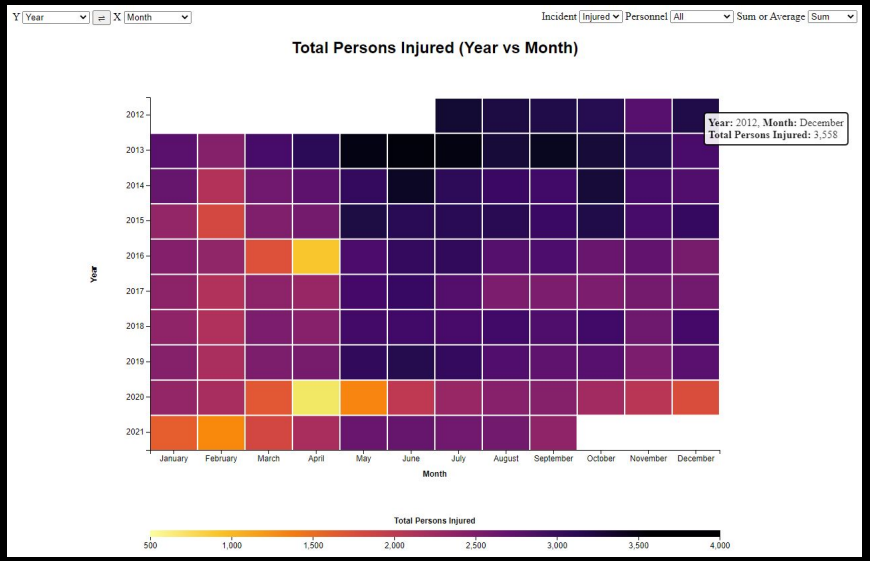




Visual Encodings:

- Total count against time dimension (Month, Day of Week, Date)
- Buttons to filter by borough

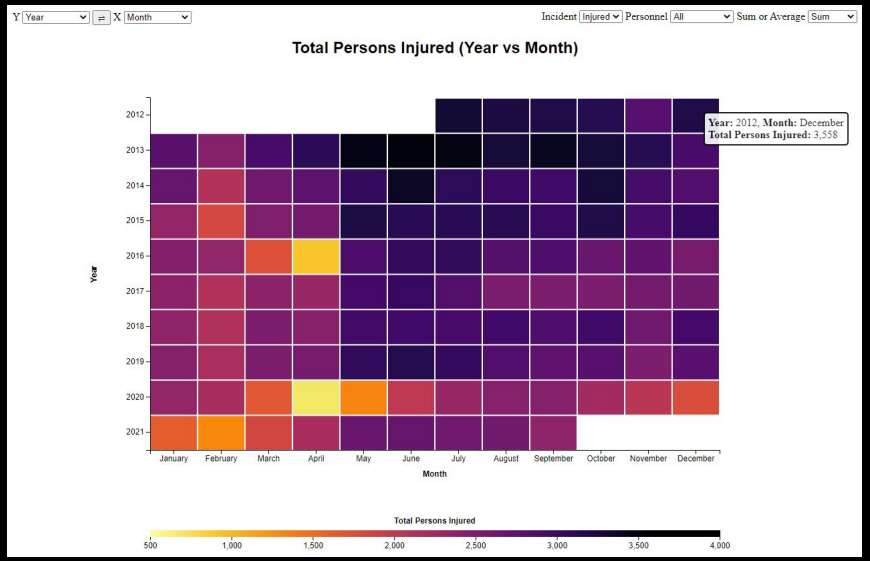
METHOD: Heatmap



Visual Encodings:

- X and Y Axis: Year, Month, Day, Day of Week, or Hour
- Values: Sum/Average of Injuries/Deaths by Personnel Type (Total, Pedestrian, Cyclist, Motorist)
- Cells: Sequential Colour Map, Lighter colour for lower value; Darker colour for higher value
- Tooltip to get specific values by hovering over cell

METHOD: Heatmap



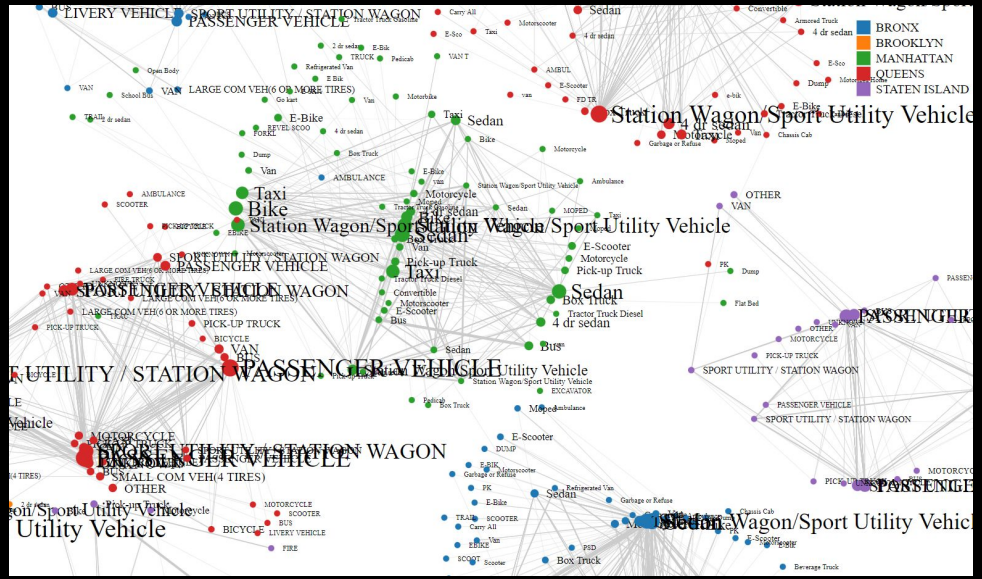
Challenges:

- Deciding on the best visualization that is simple and intuitive
  - Chose current visualization
- Changing dimensions and values of the visualization through dropdown menus
  - d3.flatRollup() and d3.transition()

METHOD: VehicleNet

Visual Encodings:

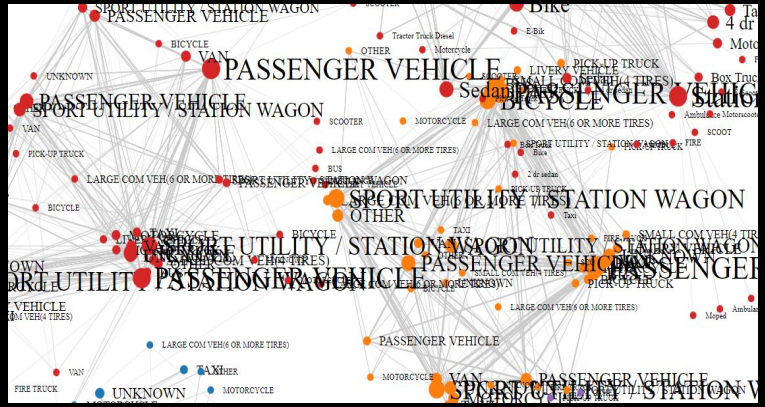
- Node-link diagram: Force Directed Graph
- Connectivity through vertex/node with one or more edges connecting it to another node
- Text Labels for Vehicles
- Legends for Boroughs
- Size encoding for node attributes (frequency of vehicle).
- Size encoding for link attributes (severity of collision).
- Colour encoding for node attribute (boroughs)
- Simulation of physical forces
- Drag & Pull on nodes to interact with individual graphs



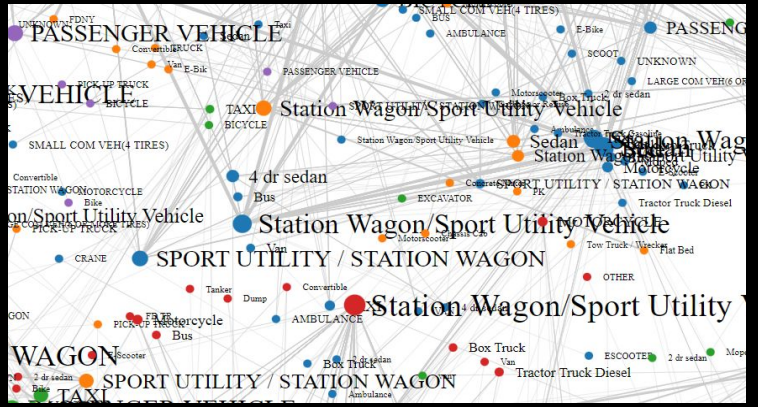
METHOD: VehicleNet

Challenges:

- Computationally Expensive & Scalability
- Tightly interconnected group of nodes with many links forms a visual clump
- Solution: Stratified Sampling & Finding a balance between node size and link distance.



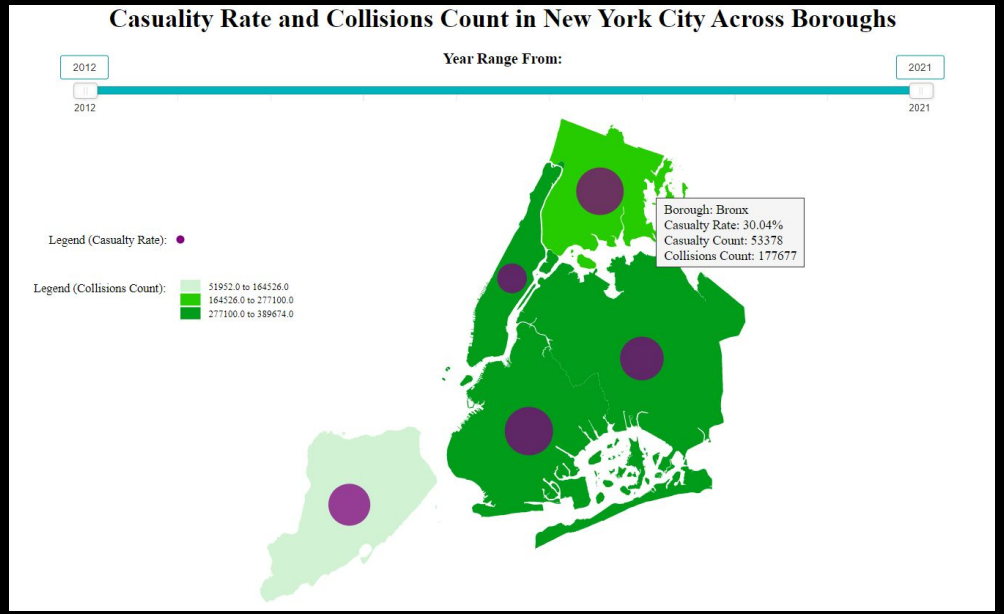
Larger Text & Shorter node links



Smaller Text & Longer node links

- Ambiguity in spatial proximity might mislead users
- FDG layouts are often non-deterministic

METHOD: Map



Visual Encodings:

- Map color represent Collisions count
- Size of circle represent Casualty Rate
- Slider to choose from a specific year or period
- Tooltips for easy reference to data points

The color and size encoding will change according to the dataset which is filtered via the year slider.

The use of Transition for smoother interaction

METHOD: Map

Challenges:

- Tried to create a Density Map but too many data points (over 1m) to be added into the svg, resulting in the webpage to crash (preferably not to render them)
  - https://stackoverflow.com/questions/18244995/d3-how-to-show-large-dataset

- Still trying to combine both Borough geojson with the detailed NYC roadmap geojson.
  - Each of the geojson specialise in different features.
  - An alternative that the team found is to create a bounding box in the detailed NYC roadmap geojson but we might not be able to use the Borough features in our analysis.

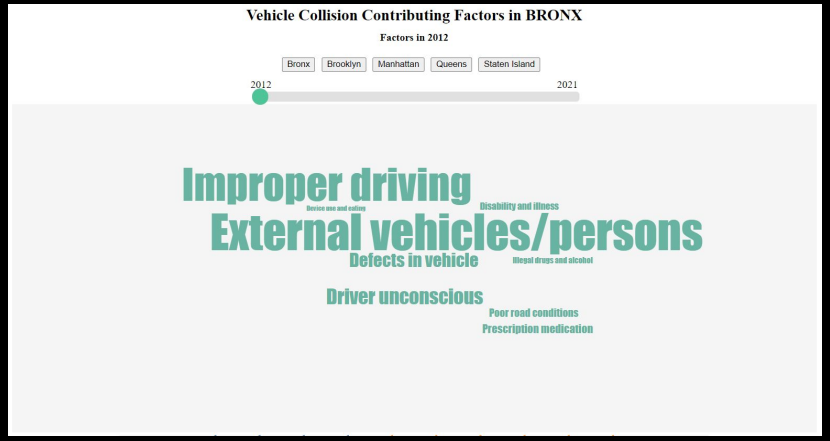
METHOD: Map

Challenges:

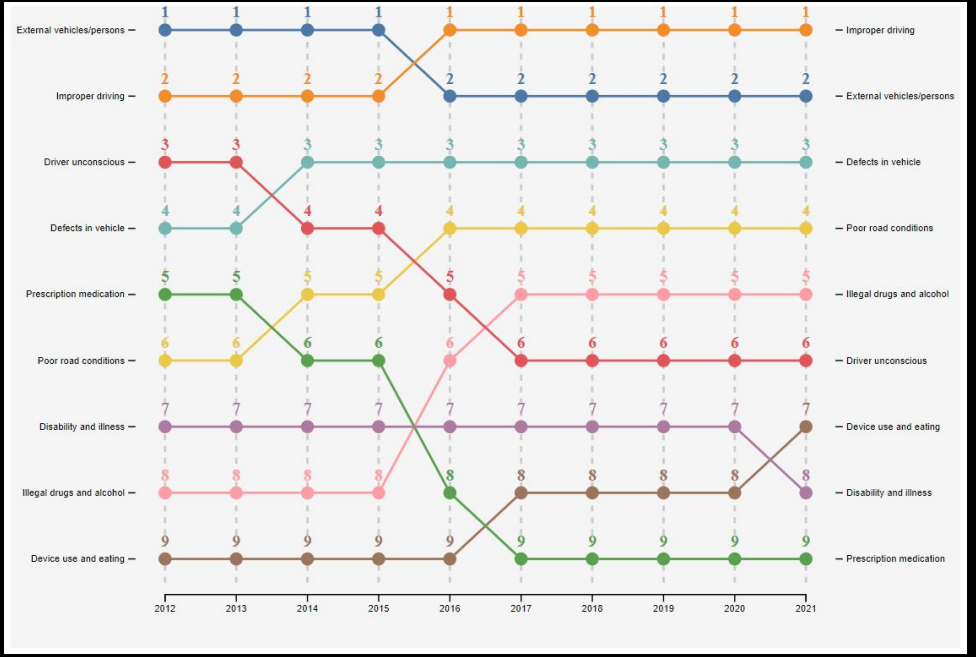
- Casualty Rate is similar to each other and might not show a big difference in terms of size.
  - Tried log-transformation with multiplier (FAIL)
  - Current size is based on proportion of the highest casualty rate with multiplier

- The color and size (data) is based across boroughs and might not be every effective in comparing across years

RESULTS AND DISCUSSION



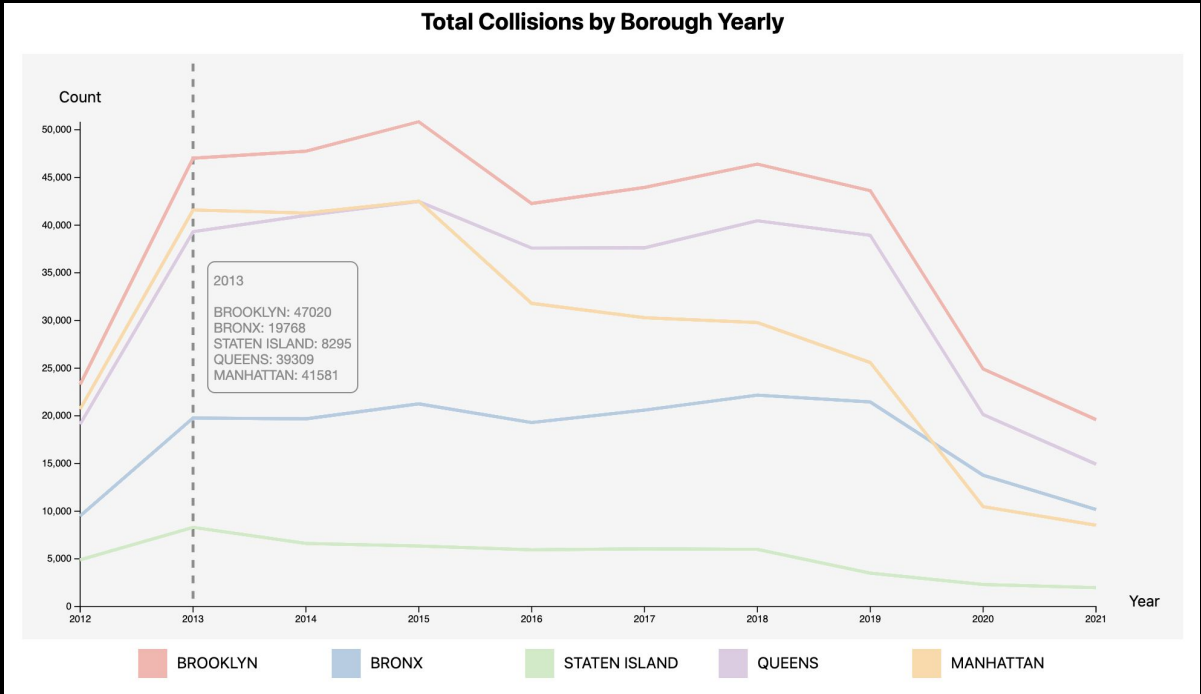
Word Cloud on the contributing factors for Bronx in 2012



Rankings of contributing factors over time for Bronx

Task 1	Identify prominent factors that contribute to vehicle collisions and how they change over time for each borough

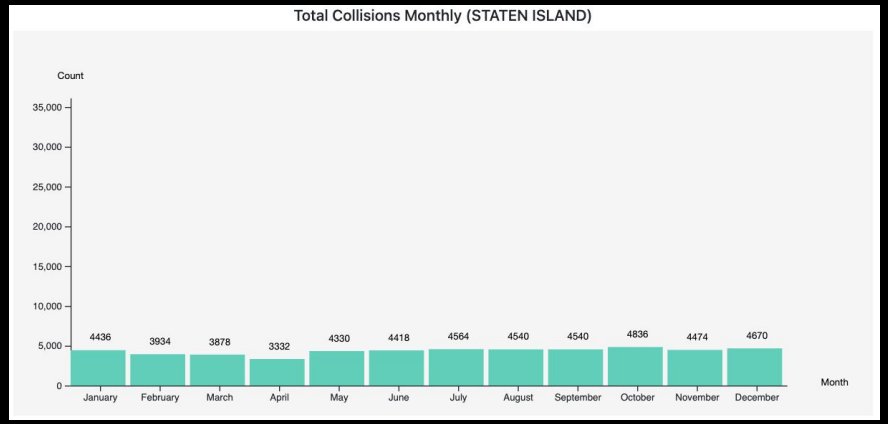
RESULTS AND DISCUSSION



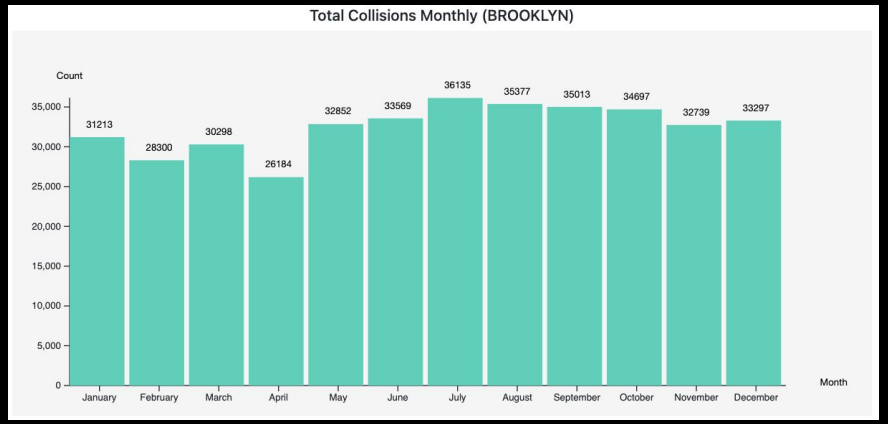
Total collisions by borough yearly

Task 2	Examine the number of collisions between different boroughs at different years, months, days and hours.

RESULTS AND DISCUSSION



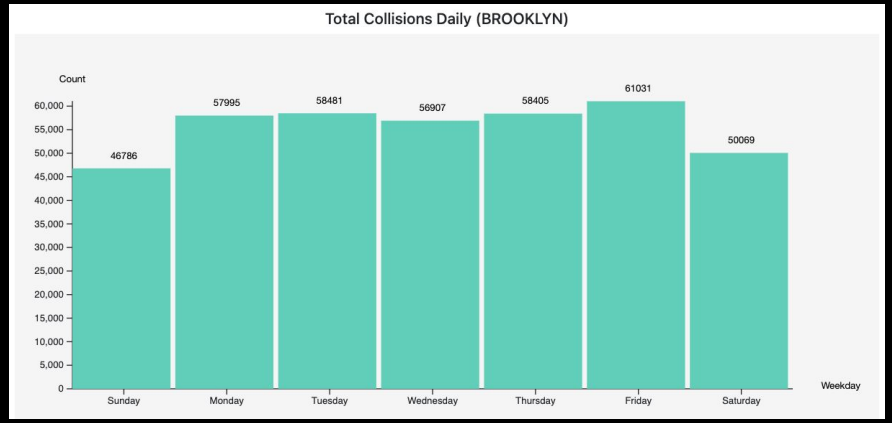
Total collisions monthly for Staten Island



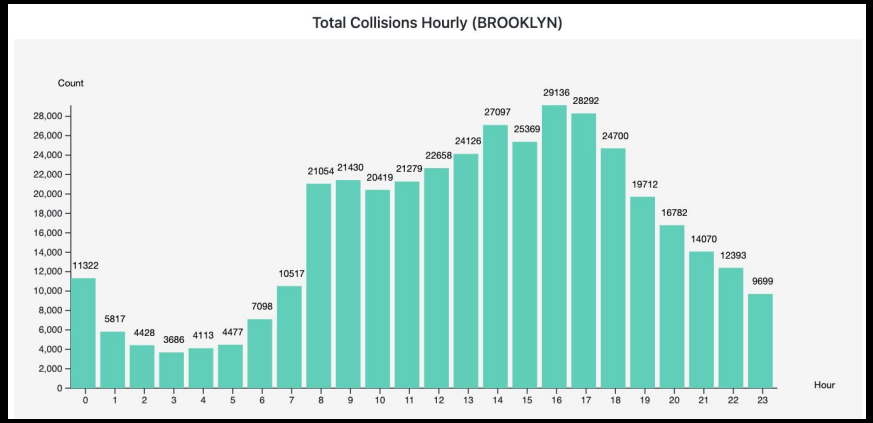
Total collisions monthly for Brooklyn

Task 2	Examine the number of collisions between different boroughs at different years, months, days and hours.

RESULTS AND DISCUSSION



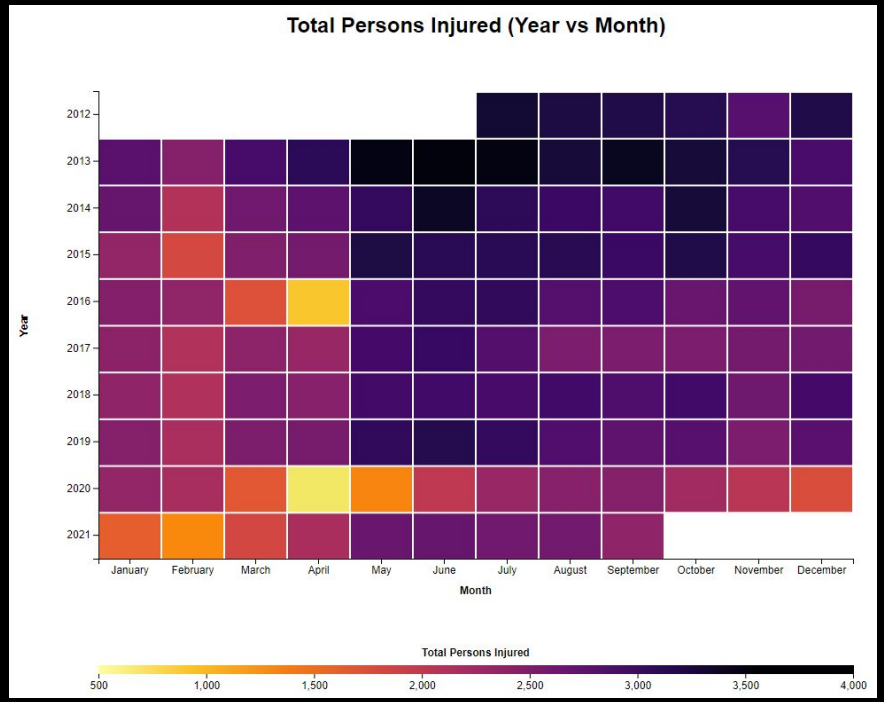
Total collisions by day of week for Brooklyn

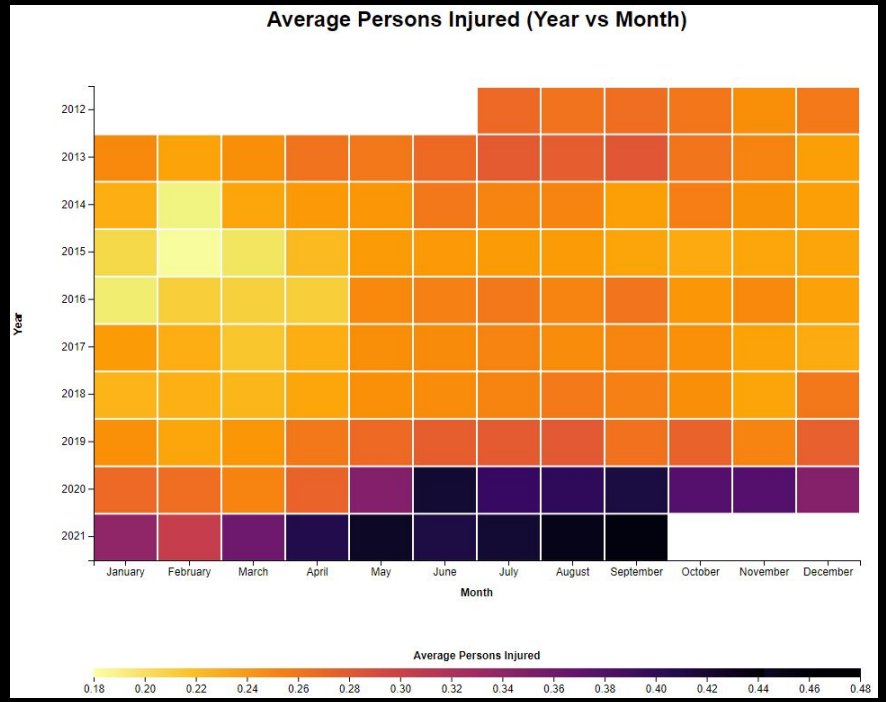


Total collisions by hour for Brooklyn

Task 2	Examine the number of collisions between different boroughs at different years, months, days and hours.

RESULTS AND DISCUSSION

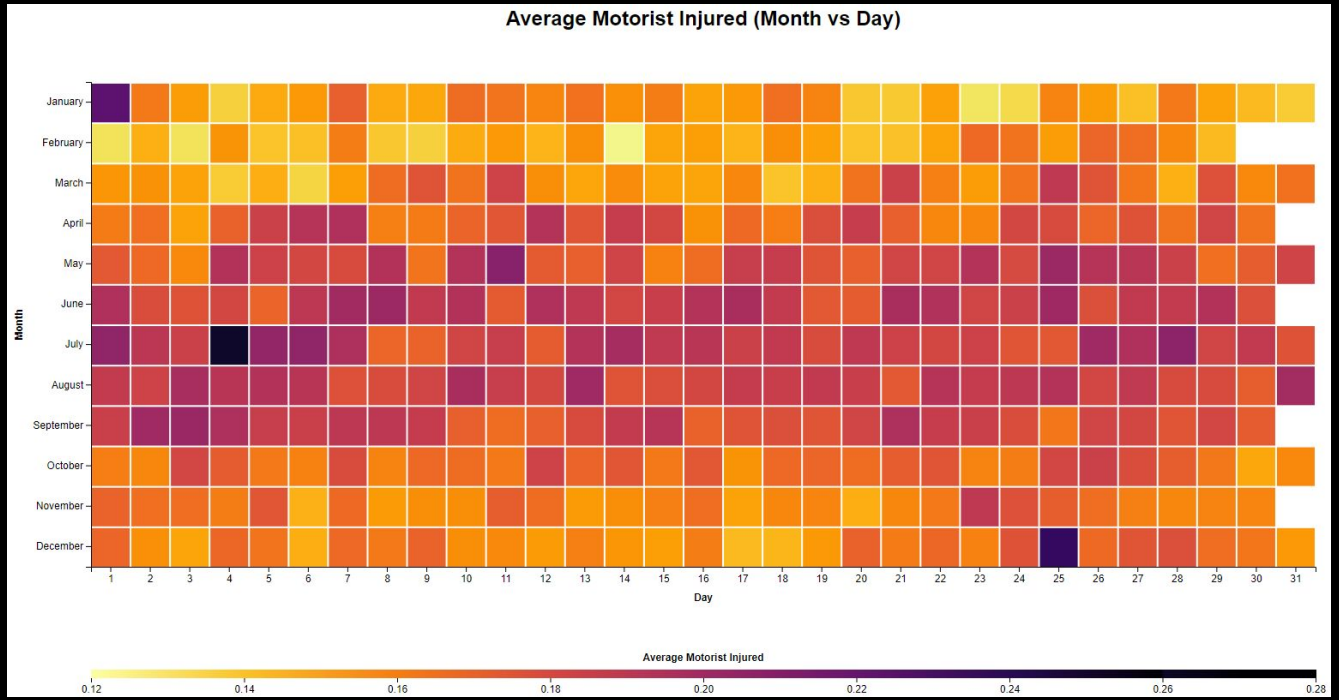




Comparison between Total and Average Persons Injured (Year vs Month)

Task 3	Exploring the sum and average of injuries and deaths across different personnel groups at different years, months, days, days of week, and hours.

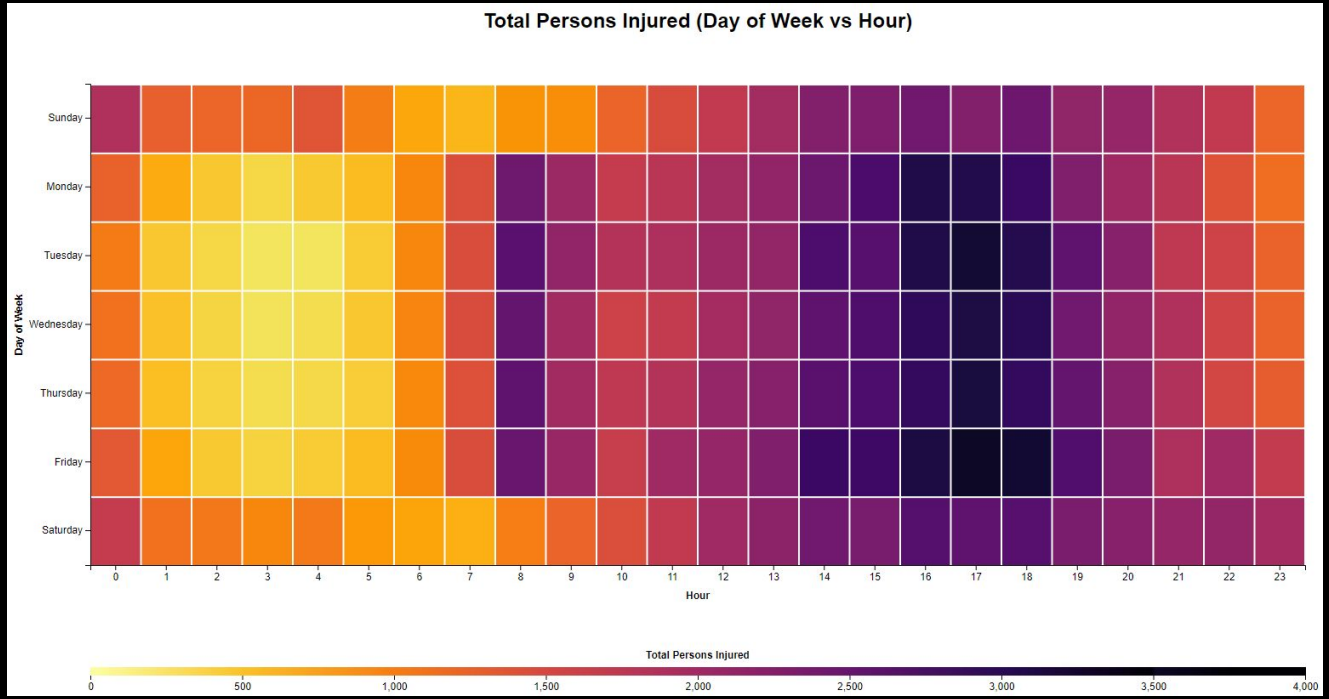
RESULTS AND DISCUSSION



Average Motorist Injured (Month vs Day)

Task 3	Exploring the sum and average of injuries and deaths across different personnel groups at different years, months, days, days of week, and hours.

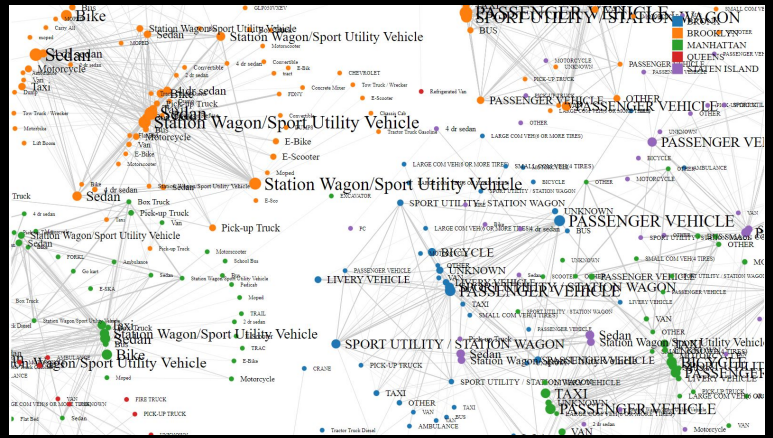
RESULTS AND DISCUSSION



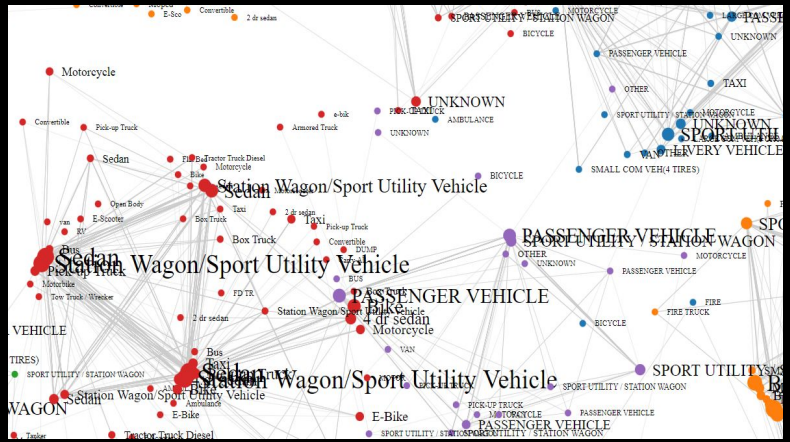
Total Persons Injured (Day of Week vs Hour)

Task 3	Exploring the sum and average of injuries and deaths across different personnel groups at different years, months, days, days of week, and hours.

RESULTS AND DISCUSSION



Collisions in Brooklyn, Bronx & Manhattan

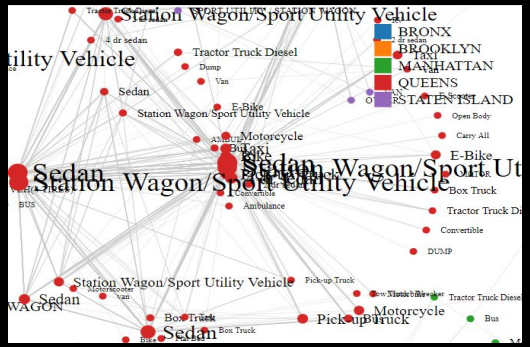


Collisions in Queens, Staten Island

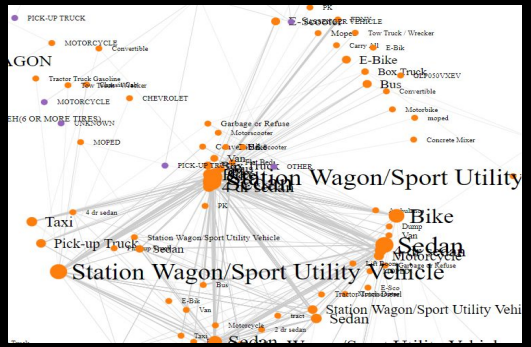
Sedan, Sports Wagon/ Utility Vehicle, Passenger Vehicle, Taxi and Bike are most commonly involved in collisions

Task 4	Visualise the severity between collisions by injuries and deaths across different boroughs

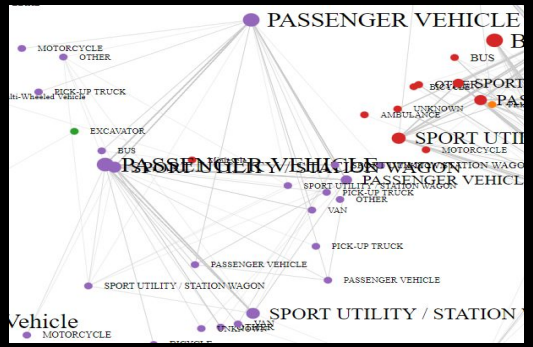
RESULTS AND DISCUSSION



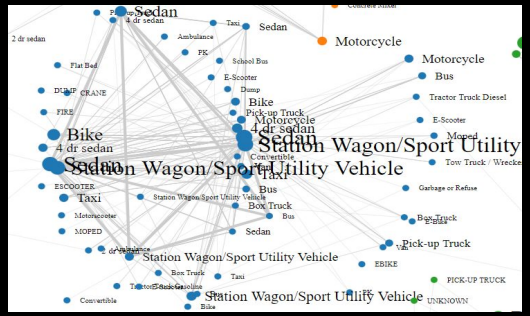
Queens



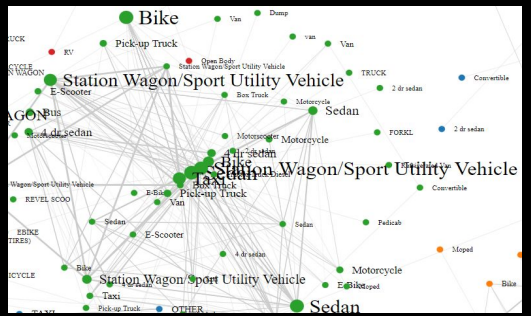
Brooklyn



Staten Island



Bronx

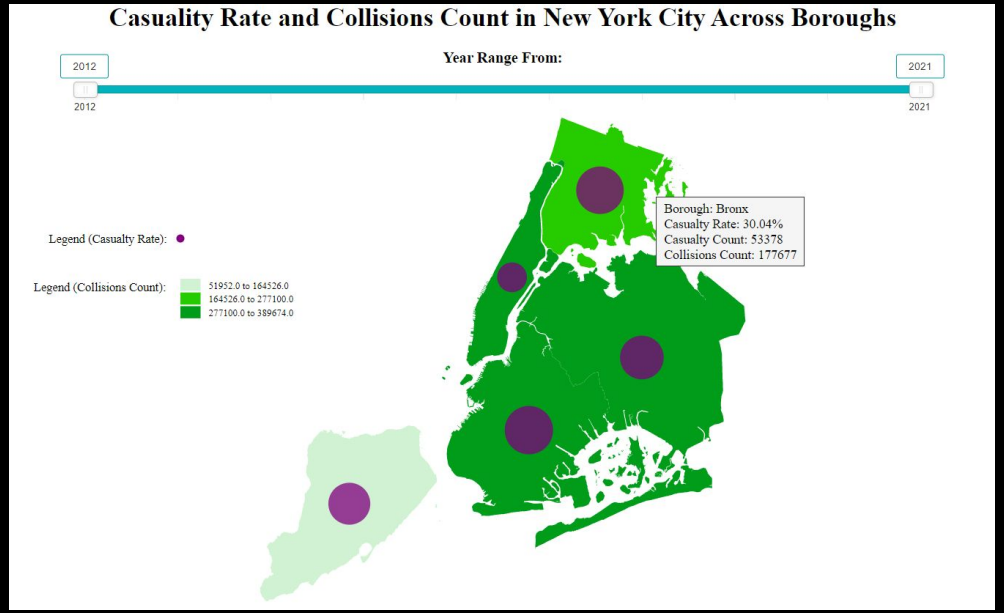


Manhattan

Difference in severity across different boroughs

Task 4	Visualise the severity between collisions by injuries and deaths across different boroughs

RESULTS AND DISCUSSION



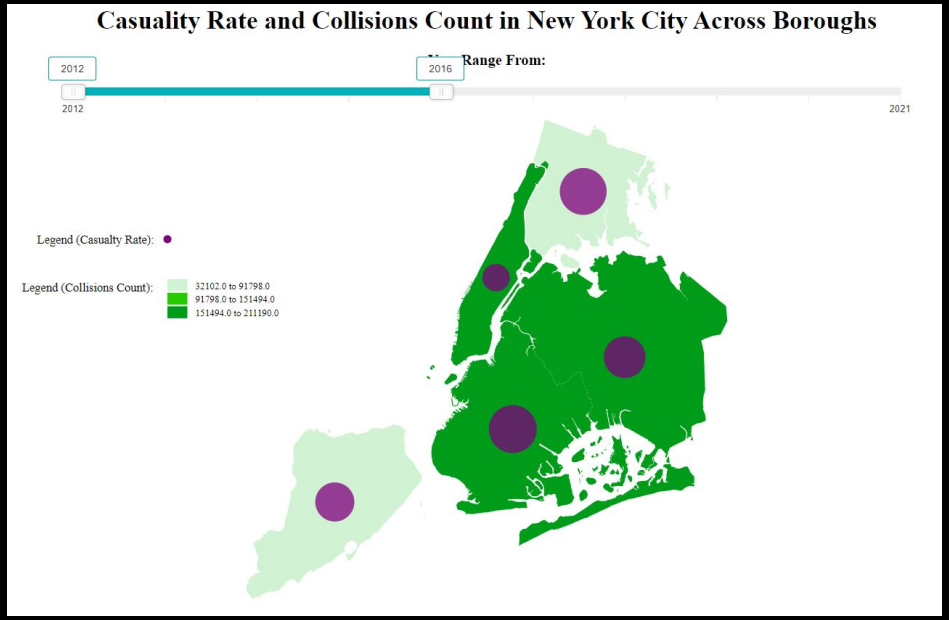
Casualty Rate and Collisions Count in NYC

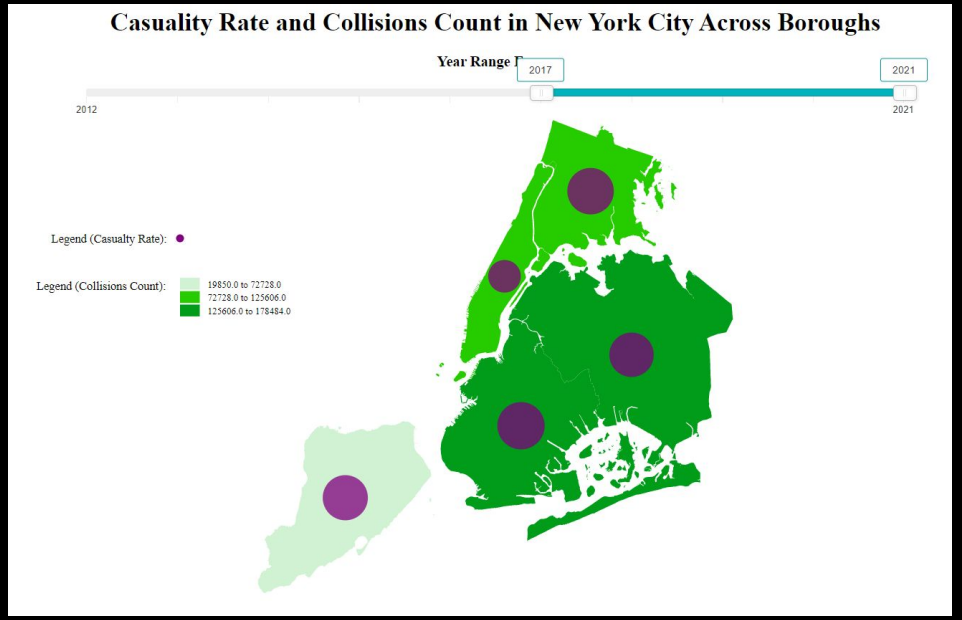
Bronx and Staten Island have similar casualty rate at around 30%.

For Manhattan, despite having higher collision count of 282,554, it only has a casualty rate of 18.8%

Task 5	Examine casualty rate and number of collisions across different boroughs

RESULTS AND DISCUSSION





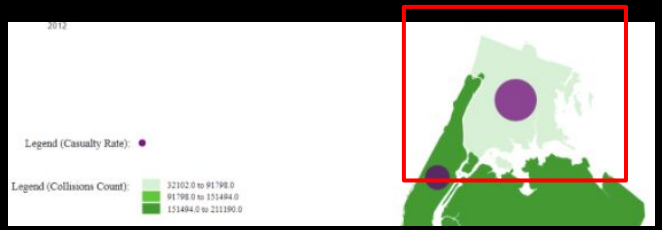
Comparison of Casualty Rate and Collisions Count in NYC before and after 2016

Task 5	Examine casualty rate and number of collisions across different boroughs

RESULTS AND DISCUSSION

2012-2016 vs 2016-2021

- The number of collision has been decreasing when we look at 5 years spans
- Bronx decrease at a decreasing rate compared to the other borough
  → can see based on the legend of the map



Bronx, 2012-2016

Bronx, 2016-2021

Task 5	Examine casualty rate and number of collisions across different boroughs

RESULTS AND DISCUSSION

Casualty Rate

- Among each borough has **INCREASED**!!! (worrying sign in NYC)
- Although we might think that the casualty rate will usually be low if number of collisions is low as we assume that they should have proper and effective measures to cope with the traffic condition
- Yet, we can't really tell that there is a relationship between number of collisions and casualty rate. The inverse correlation is really worrying as it just means that there is lack of measures and policy to prevent casualties during a collision.

Task 5	Examine casualty rate and number of collisions across different boroughs

04

CONCLUSION

MILESTONES & SCHEDULE

A summary of the team's timelines and milestones

CONCLUSION

- Team met problem statement objectives and managed to produce meaningful visualisations that aided in identifying critical collisions dedicated to improving traffic safety

- Insights derived from visualizations can definitely aid in potential solutions that help in reducing traffic fatalities and casualties

FUTURE WORK

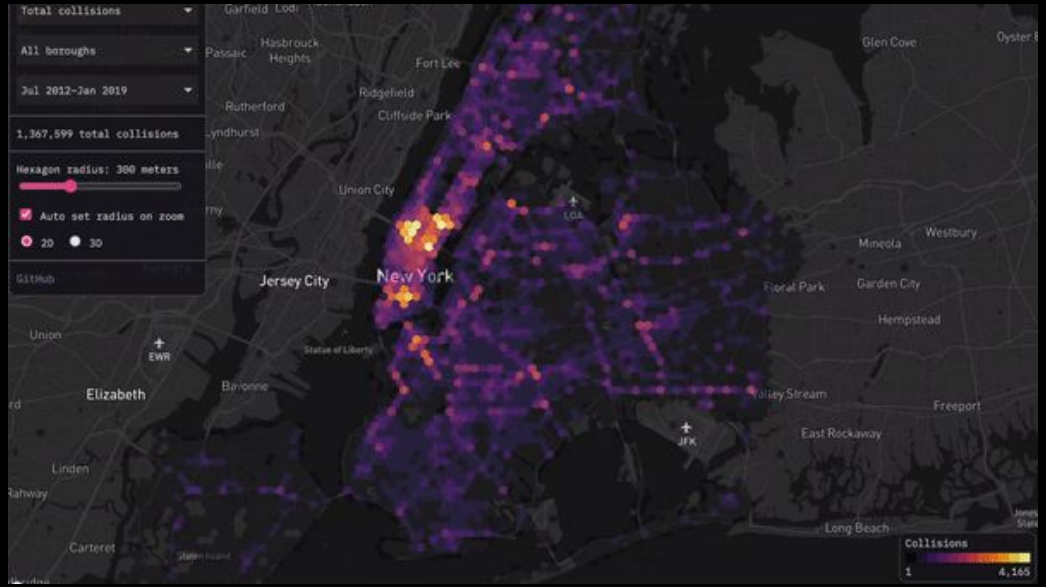
There is still plenty of opportunities and area of improvement in NYC.

Given enough time, we can explore more map varieties e.g. contour map (value based on target measure)

Create a more robust visualisation platform as more time is needed to load the data.

Current visualizations could be better integrated with more interactive options provided.

Justify why NYC claimed first for the worst traffic in 2021



DEMO OF VISUAL ANALYTIC SYSTEM


THANK YOU

REFERENCE

Carlin, D. (2021, June 30). New York City Overtakes Los Angeles To Top List Of Worst Traffic In The Country, Survey Says. Retrieved from https://newyork.cbslocal.com/2021/06/30/new-york-city-worst-traffic-in-the-country/

Driverknowledge. (n.d.). Car Accident Statistics in the U.S. Retrieved from https://www.driverknowledge.com/car-accident-statistics/

Grossman, E. (2021, May 27). New York City anticipates a summer tourism season unlike any other. Retrieved from https://www.marketwatch.com/story/new-york-city-anticipates-a-summer-tourism-season-unlike-any-other-11622135445

Injury_doctors. (2020, February 20). NEW YORK CAR ACCIDENT STATISTICS: WHICH NYC BOROUGH IS THE MOST DANGEROUS? Retrieved from https://www.injuredcalltoday.com/new-york-car-accident-statistics-by-nyc-borough/

New York City Car Accident Statistics – 2020 Update. (n.d.). Retrieved from https://jknylaw.com/new-york-car-accident-lawyer/statistics/

NYC Open Data. (2021, October 8). Motor Vehicle Collisions - Crashes. Retrieved from https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95

Ng, W., & Sun, D. (2021, September 11). 5 killed in Tanjong Pagar car crash: What happened. Retrieved from https://www.straitstimes.com/singapore/timeline-of-tanjong-pagar-fatal-crash

Road Accident Statistics. (2021, January). Retrieved from https://www.budgetdirect.com.sg/car-insurance/research/road-accident-statistics-in-singapore

White, M. (2021, June 20). The Top 10 Largest U.S. Cities by Population. Retrieved from https://www.moving.com/tips/the-top-10-largest-us-cities-by-population/

METHOD: VehicleNet

Visual Encodings:

- Node-link diagram: Force Directed Graph
- Connectivity through vertex/node with one or more edges connecting it to another node
- Text Labels for Vehicles
- Legends for Boroughs
- Size encoding for node attributes (frequency of vehicle).
- Size encoding for link attributes (severity of collision).
- Colour encoding for node attribute (boroughs)
- Simulation of physical forces
- Drag & Pull on nodes to interact with individual graphs