Introduction to Data Science Final Project Part 2

1. Dataset Selection **10 points**

Choose a Dataset of your own choice. Explain why you have chosen this dataset and explain the features of the dataset and their importance in your analysis. Dataset must have at least 500 rows, 15 columns.

2. Import necessary libraries and load the dataset. If the dataset is in numbers format, convert it to CSV or Excel for easier handling **30 points**

a) Do all the important steps of Data Exploration check the first few rows, data types, and basic statistics.
b) Data Cleaning: Check for missing values print, Remove or fill missing values and check for duplicates
c) Exploratory Data Analysis (EDA). Create visualizations to explore relationships between variables at least 5-7.

3. Analytical Findings and Steps Taken and conclusion **10 points**

Write a summary of your findings, focusing on the most influential factors affecting in your dataset.

4. Submission Files

a) A Jupyter Notebook or Python script containing the above code.
b) Visualizations that effectively illustrate your findings.
c) A short report summarizing insights gained from the analysis.

⇒ **File format:** [firstName_lastName_DS621.pdf]

Good luck!

Report

Dataset name: Motor_Vehicle_Collisions_-_Crashes Number of features/columns (atleast 15): 29 Columns Number of Rows: Around 2 Million Source: NYC Open Data Student Name: Sonali Gupta Student ID: 1056612

1. The dataset I am using for this project is Motor Vehicle Collisions crash table where each row represents a crash event from all police reported motor vehicle collisions in NYC. The latest data is from October 15th, 2024 when I downloaded the dataset. I chose this dataset as it is a real-world live data and gives an opportunity to work on huge data that might be unclean and is complex. I was looking forward to work on this data as part of a community of transportation data science learners that I joined this trimester and thought it'll be nice to explore and answer questions on this data as part of this project as it'll help me start in a structured way and supplement my learnings.

   Few interesting columns of this dataset that we will be using:

Column name	Description
CRASH DATE	Occurrence date of collision
LATITUDE	Latitude coordinate of collision
LONGITUDE	Longitude coordinate of collision
CONTRIBUTING FACTOR VEHICLE 1	Factors contributing to the collision for Vehicle 1
VEHICLE TYPE CODE 1	Type of vehicle 1 in collision (eg. ATV, bicycle, car/suv, ebike, etc)
NUMBER OF MOTORIST KILLED	Number of vehicle occupants killed
CRASH TIME	Occurrence time of collision

a) Crash date and Crash Time : These column store the date and time when a crash occurred. This data is important for time-series analysis over the years and to find if there is a pattern when crashes occur during a day.

b) Number of Pedestrians, Cyclists, and Motorists Injured/Killed: These columns provide a breakdown of the injuries and fatalities by type of individual involved. These will help us understand the impact of crashes and the fatality.

c) Longitude and Latitude are also important data as it can help us identify if there is concentration of accidents happening in an area. Once we identify such hotspots, using it we can try and reduce accidents by taking appropriate measures.

d) We have a lot of data like Borough, ON STREET NAME, ZIP CODE that can help pinpoint the exact location of the incident.

e) There are other columns like vehicle type and contributing factor that specific vehicle that tells us about what type of vehicle was involved in the crash and what was the reason behind the incident.

2. Necessary libraries:
    a. Pandas: to create Data frames and Series
    b. Matplotlib: to create visualizations like bar charts
    c. Seaborn: to create visualizations like bar charts for categorical data

Basic Exploration:

The dataset has 29 columns and 21,26,535 rows(around 2 million)

```
# Let's begin by chekcing first few rows
data.head()
```

	CRASH DATE	CRASH TIME	BOROUGH	ZIP CODE	LATITUDE	LONGITUDE	LOCATION	ON STREET NAME	CROSS STREET NAME	OFF STREET NAME	...	CONTRIBUTING FACTOR VEHICLE 2
0	09/11/2021	2:39	NaN	NaN	NaN	NaN	NaN	WHITESTONE EXPRESSWAY	20 AVENUE	NaN	...	Unspecified
1	03/26/2022	11:45	NaN	NaN	NaN	NaN	NaN	QUEENSBORO BRIDGE UPPER	NaN	NaN	...	NaN
2	06/29/2022	6:55	NaN	NaN	NaN	NaN	NaN	THROGS NECK BRIDGE	NaN	NaN	...	Unspecified
3	09/11/2021	9:35	BROOKLYN	11208	40.667202	-73.866500	(40.667202, -73.8665)	NaN	NaN	1211 LORING AVENUE	...	NaN
4	12/14/2021	8:13	BROOKLYN	11233	40.683304	-73.917274	(40.683304, -73.917274)	SARATOGA AVENUE	DECATUR STREET	NaN	...	NaN

On loading the csv into panda dataframe, the dtypes or datatypes are distributed as below:
    a. float64 -> (4) columns
    b. int64 -> (7) columns
    c. object-> (18) columns

Statistics:

Checking basic statistics for the columns, we can see that the maximum people injured in a single incident were 43. On average, each crash has around 0.316 injuries. Data.describe() returns statistics in scientific notation so below if the code to display in readable format.

```
# Converting the above data to non-scientific representation
data.describe().apply(lambda x: x.apply('{0:.5f}'.format))
```

	LATITUDE	LONGITUDE	NUMBER OF PERSONS INJURED	NUMBER OF PERSONS KILLED	NUMBER OF PEDESTRIANS INJURED	NUMBER OF PEDESTRIANS KILLED	NUMBER OF CYCLIST INJURED	NUMBER OF CYCLIST KILLED	NUMBER OF MOTORIST INJURED
count	1873352.00000	1873352.00000	2126517.00000	2126504.00000	2126535.00000	2126535.00000	2126535.00000	2126535.00000	2126535.00000
mean	40.61987	-73.73786	0.31692	0.00153	0.05741	0.00075	0.02771	0.00012	0.22772
std	2.05880	3.85842	0.70625	0.04128	0.24578	0.02803	0.16623	0.01097	0.66750
min	0.00000	-201.35999	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
25%	40.66767	-73.97478	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
50%	40.72065	-73.92713	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
75%	40.76963	-73.86673	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
max	43.34444	0.00000	43.00000	8.00000	27.00000	6.00000	4.00000	2.00000	43.00000

Understand Missing(Null) values

Above we can see NaN values in a lot of rows. First, we can see which all columns have null values and how many null values they have. For it we will use the below code.

```
missing_values = data.isnull().sum()
```

It returns us sum of null values of each column.
We notice that a lot of columns have null values. For now, I am not updating or dropping data based on the null values. We will do that based on the requirement of column we are using further.

Analysis:

```
[30]: # Top 5 columns with the most null values
      missing_values.sort_values(ascending=False).head()
```

```
[30]: VEHICLE TYPE CODE 5               2117381
      CONTRIBUTING FACTOR VEHICLE 5     2117091
      VEHICLE TYPE CODE 4               2093062
      CONTRIBUTING FACTOR VEHICLE 4     2091824
      VEHICLE TYPE CODE 3               1979378
      dtype: int64
```

Top 5 columns with most null values are VEHICLE TYPE CODE 5, CONTRIBUTING FACTOR VEHICLE 5, VEHICLE TYPE CODE 4, CONTRIBUTING FACTOR VEHICLE 4, VEHICLE TYPE CODE 3. This is expected since not all crashes involve multiple vehicles or factors.

Check For Duplicates

It is common for huge datasets to have duplicates. We will run the below code to find the sum of duplicate rows. The duplicate function will check for rows and return True or False is a row is a duplicate occurrence, and sum will add all Trues to give the total count.

```
data.duplicated().sum()
```

There are no duplicates in this dataset.

Exploratory Data Analysis

Cause of crashes

Let's start by checking what are the top reasons for accidents. We will make use of the column "CONTRIBUTING FACTOR VEHICLE 1" which is a string column that tells us about the Factor contributing to the collision for first vehicle impacted in the accident. Since for most of the accidents, there will be data on at least one vehicle, we use this column.

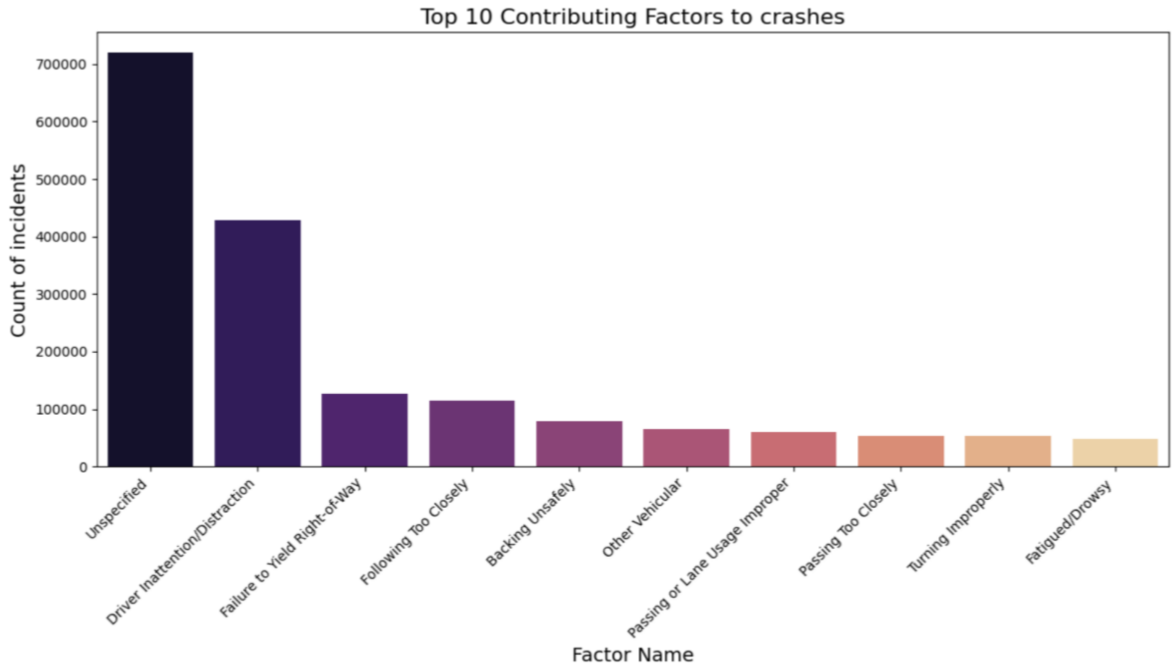
```
top_factors = data['CONTRIBUTING FACTOR VEHICLE 1'].value_counts().head(10)
top_factors
```

```
CONTRIBUTING FACTOR VEHICLE 1
Unspecified                       719321
Driver Inattention/Distraction    427752
Failure to Yield Right-of-Way     126615
Following Too Closely             114099
Backing Unsafely                   78256
Other Vehicular                    66032
Passing or Lane Usage Improper     59977
Passing Too Closely                53355
Turning Improperly                 52448
Fatigued/Drowsy                    47447
Name: count, dtype: int64
```

Besides for "Unspecified," the top 3 contributing factors that cause the most crashes are:

- Driver Inattention/Distraction
- Failure to Yield Right-of-Way
- Following Too Closely

Unspecified shows that some data is missing and this term is used as a default value. Also plotting the above data



Types of Vehicles in Crash

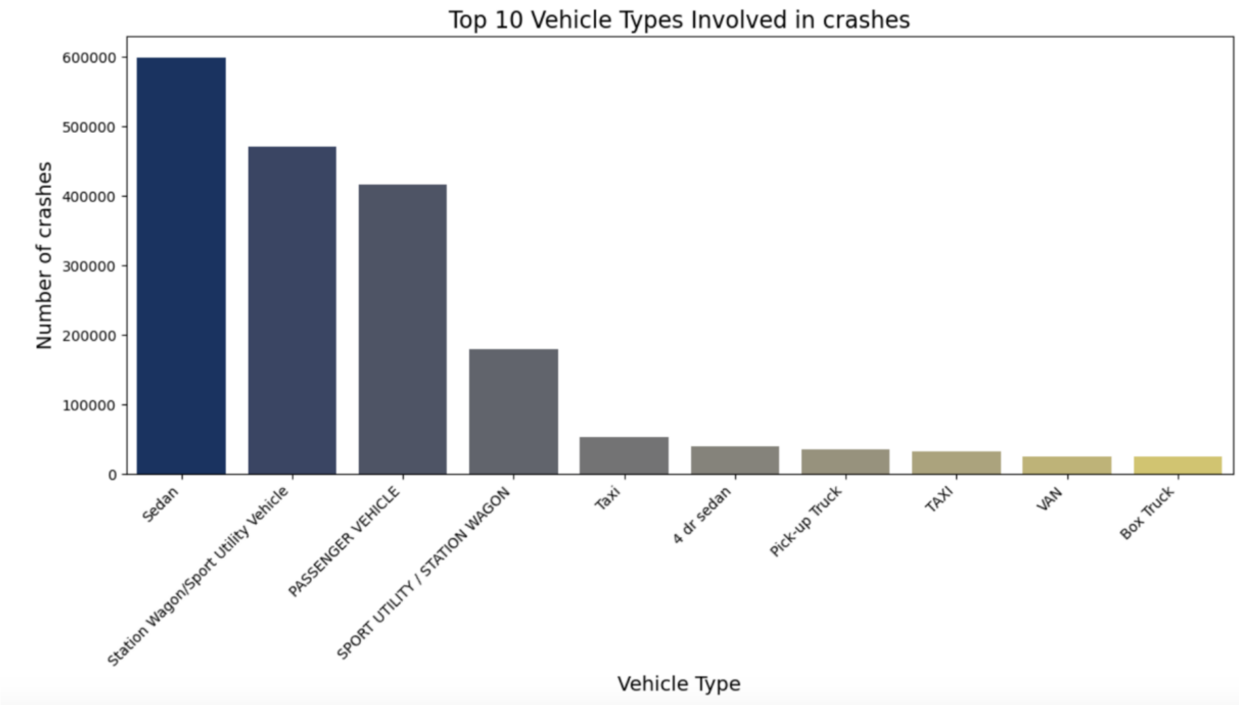
Let's study if there is a pattern or trend in the crashes and the type of vehicle. We will make use of the column "VEHICLE TYPE CODE 1" which is a string column that tells us about the type of the first vehicle impacted in the accident. Since for most of the accidents, there will be data on at least one vehicle, we use this column.

```
# Determine the top vehicle types involved in crashes
top_vehicle_types = data['VEHICLE TYPE CODE 1'].value_counts().head(10)
top_vehicle_types
```

```
VEHICLE TYPE CODE 1
Sedan                                 599241
Station Wagon/Sport Utility Vehicle   471118
PASSENGER VEHICLE                     416206
SPORT UTILITY / STATION WAGON         180291
Taxi                                   52996
4 dr sedan                             40180
Pick-up Truck                          35911
TAXI                                   31911
VAN                                    25266
Box Truck                              25134
Name: count, dtype: int64
```

The top 3 vehicles that were most involved in crashes

- Sedan
- Station Wagon/Sport Utility Vehicle
- PASSENGER VEHICLE

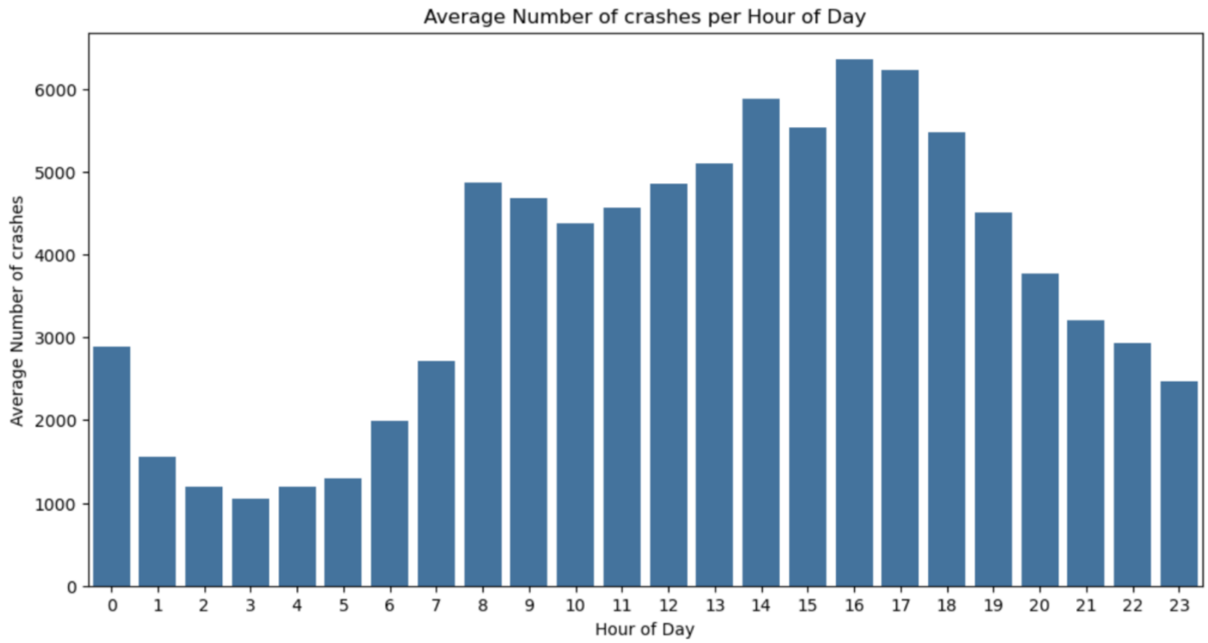


Analysis: This might be due to the fact that Sedan are in large numbers on the road as they are commonly owned by families. Also all top 3 vehicle type are larger vehicle type. This could also be a reason for frequent accidents.

Time of Crash

Let's begin by creating a chart that displays the average number of crashes per hour of the day. This will help us understand whether additional factors are contributing to crashes - i.e. rush hour, school dismissal time, night fall, etc.

For this we will make use of column CRASH TIME and convert it to Datetime. Then we will fetch the hour from the time. This code will return a Series with the hour (0-23) for each row in that column. Finally will calculate the average number of crashes per hour of the day for each hour of the day.



Analysis: The data indicates that the highest number of crashes occur between 4 PM and 5 PM. This aligns with peak rush hour, as many people are leaving work and commuting home during this time. There is also a peak at 8 AM. Which is again a time when people are leaving for work in the morning.

COVID-19 impacted the number of crashes

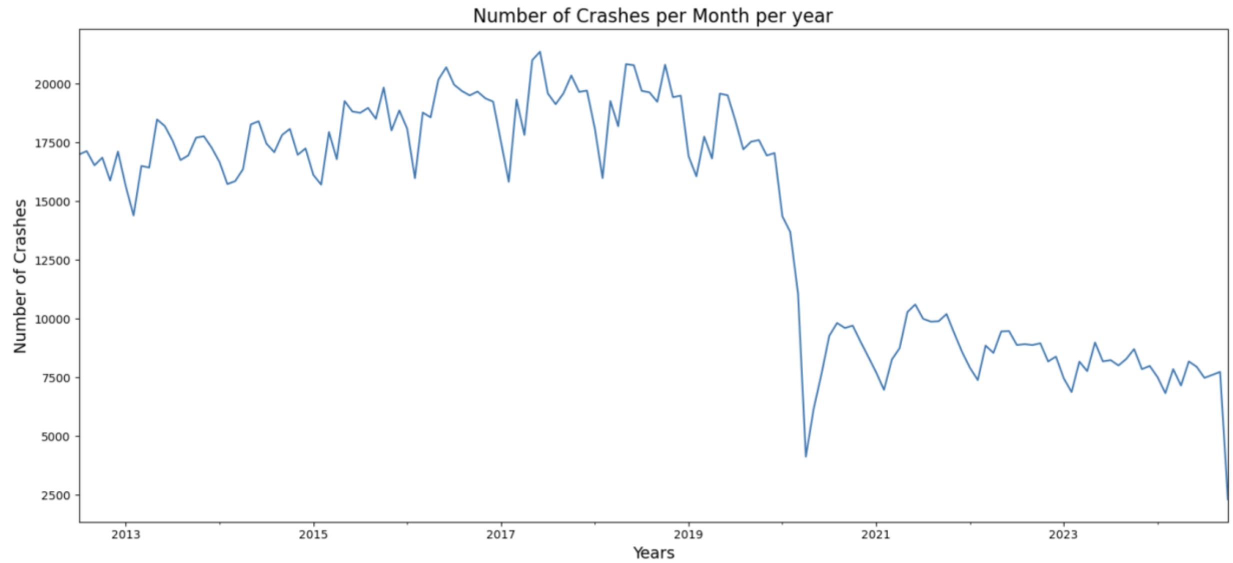
Let's study the number of crashes every month for each year. This will help up see a trend over the years, and also if there is a sudden change over the years. We will make use of CRASH DATA and group by each year-month and count the number of rows. This will show us crashes for each month across years. Below is the code:

```
[82]: # Convert 'CRASH DATE' to datetime format
      data['CRASH DATE'] = pd.to_datetime(data['CRASH DATE'])

      # Group by month and year to get the number of crashes per month
      monthly_crashes = data.groupby(data['CRASH DATE'].dt.to_period("M")).size()
      monthly_crashes
```

```
[82]: CRASH DATE
      2012-07    16992
      2012-08    17142
      2012-09    16535
      2012-10    16864
      2012-11    15889
                 ...
      2024-06     7959
      2024-07     7487
      2024-08     7615
      2024-09     7740
      2024-10     2313
      Freq: M, Length: 148, dtype: int64
```

On plotting the above data



Analysis: The trendline shows us significant change in crashes in mid 2020, a sharp decline. This is interesting as restrictions in the US due to COVID-19 were issues in March 2020 when a community within New Rochelle, New York, was declared to be a "containment area." Once 2023 rolled it we can see that there is still not a rise back to that of 2019 because most of the work continued to be hybrid, leading to a decline in travel for short and long distance.

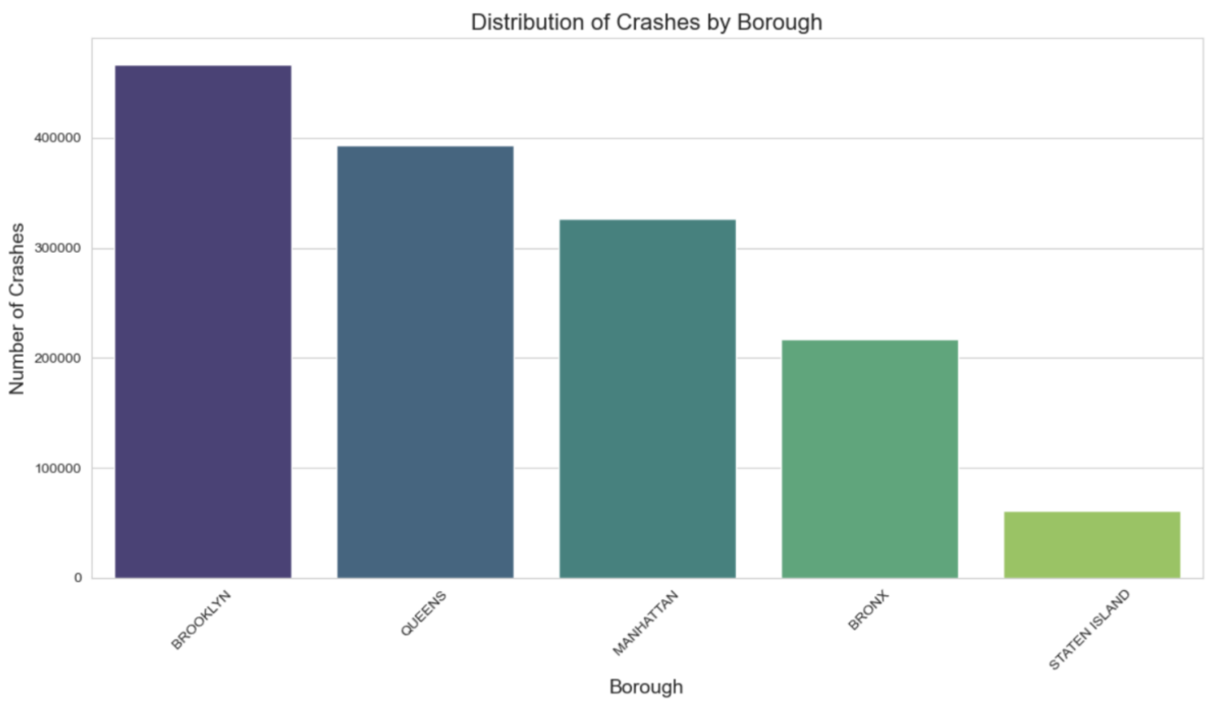
Crashes across Borough of NYC

Let's build a bar chart to compare and analyze the number of crashes across the five boroughs: Brooklyn (also known as Kings County), Queens, Manhattan, Bronx, and Staten Island. For this we will make use of the column BOROUGH and count unique rows for each of them.

```
# Find the count of unique values of BOROUGHS
borough_count = data['BOROUGH'].value_counts()
borough_count
```

```
BOROUGH
BROOKLYN         467124
QUEENS           393024
MANHATTAN        326774
BRONX            216843
STATEN ISLAND     61433
Name: count, dtype: int64
<Figure size 1200x700 with 0 Axes>
```

On plotting the above data



Analysis - The maximum crashes happen in Brooklyn. It makes sense because Brooklyn has the maximum population and Staten Island has the least population. It is interesting to see Manhattan doesn’t have the maximum accidents even though it is the busiest borough. Also Queens had the largest area followed by Brooklyn.

<u>Types of Crashes and their frequencies</u>

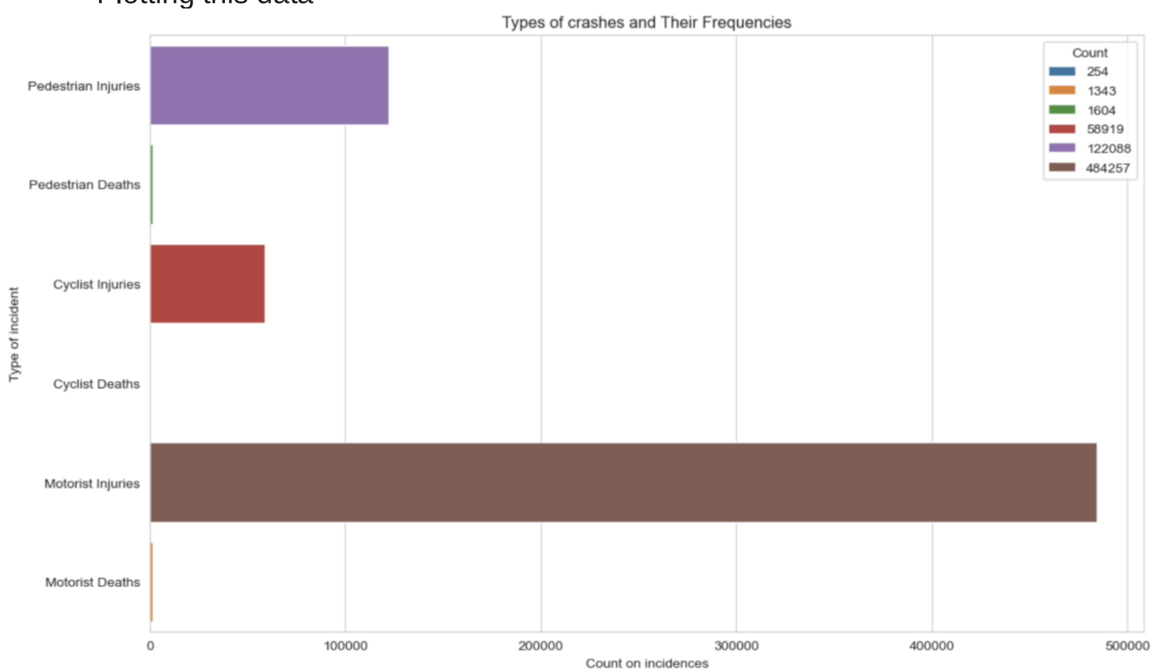
Let's graph the types of crashes within this dataset and their frequencies. Begin by aggregating your data, convert to DataFrame for simple plotting, and plot. For this we make use of different columns as there are deaths and injuries for each incident in the relative row. We will sum the values in each row for each incident, that is injury or death, for each type of causality, that is pedestrian, cyclist or motorist.

2…
```python
# Aggregating data for Pedestrians, Cyclists and Motorists
types_of_crashes = {
    'Pedestrian Injuries': data['NUMBER OF PEDESTRIANS INJURED'].sum(),
    'Pedestrian Deaths': data['NUMBER OF PEDESTRIANS KILLED'].sum(),
    'Cyclist Injuries': data['NUMBER OF CYCLIST INJURED'].sum(),
    'Cyclist Deaths': data['NUMBER OF CYCLIST KILLED'].sum(),
    'Motorist Injuries': data['NUMBER OF MOTORIST INJURED'].sum(),
    'Motorist Deaths': data['NUMBER OF MOTORIST KILLED'].sum()
}
```

4]:
```python
types_of_crashes
```

4]:
```
{'Pedestrian Injuries': 122088,
 'Pedestrian Deaths': 1604,
 'Cyclist Injuries': 58919,
 'Cyclist Deaths': 254,
 'Motorist Injuries': 484257,
 'Motorist Deaths': 1343}
```

Plotting this data



Analysis: The maximum incidents of motorists being injured. That would be people in the vehicle that had a crash. There can be multiple people in a vehicle which might contribute to the high number as well. We can see that more pedestrians were injured compared to cyclist. Also death for pedestrians is higher than Motorist deaths. This shows Pedestrians are the most vulnerable.

3. Analytical Findings and Conclusion

The dataset has a lot to offer. Based the analysis that we have done, we can summarize the findings as below.

1. The maximum crashes occur during peak rush hour of 4 to 5 PM and also there in an increased number of accidents happening during 8 to 9 AM in the morning. The count of accidents drastically drop during late hours of the night between 1 to 4 AM which is expected as the traffic is low.
2. The top reason for accidents is inattention/distraction. This reason is controllable and if the drivers are more mindful and present, the number of accidents can be brought down.
3. Among the five boroughs of NYC, Brooklyn sees the highest number of accidents. But in the heatmap we can also see Manhattan having a hotspot based on location. These places can use added safety regulation.

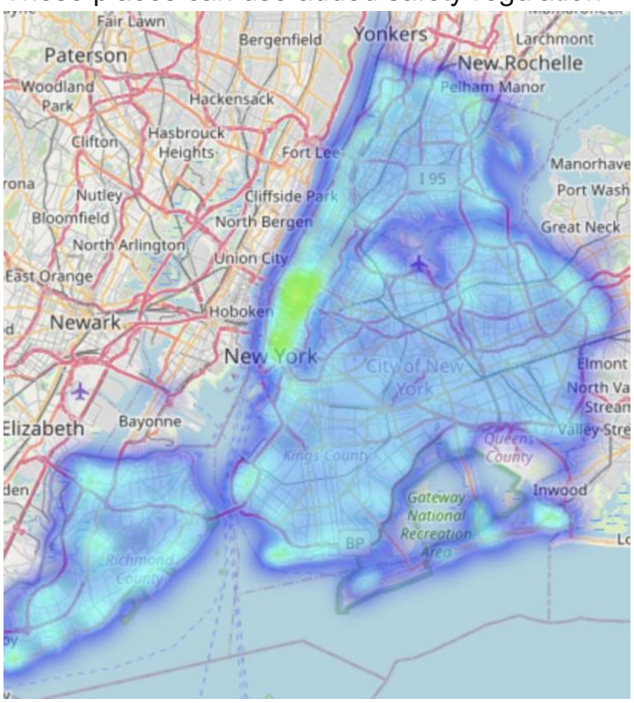


Figure 1Heatmap based on latitude and longitude

4. We see that larger size vehicles like Sedan, Sports utility van, passenger vehicle are more prone to accidents. The data is also a little unclean as there are different entries for “taxi” and “TAXI”. This could use some standardization when storing information.
5. We can also conclude how restrictions and lockdown during Covid – 19 impacted the trend of incidents. With low movement and remote working, as the traffic on road decreased, we could see a tremendous drop in the accidents.
6. In addition to the high number of motorists that occupy the vehicle getting injured, we see that pedestrians are also very vulnerable and have fatal

injuries. This shows the importance to have road safety regulations in place to prevent or at least decrease causalities.

---------------------------------------**THE END**---