



**Northeastern University**  
*College of Professional Studies*

**DATA MANAGEMENT & BIG DATA**

**ALY 6110, CRN 70352**

**PROFESSOR DAYA RUDHRAMOORTHY**

**MODULE 6 FINAL PROJECT**

**FINAL PROJECT REPORT**

**SUBMITTED BY: GROUP 1**

**Richa Umesh Rambhia**

**Vaidehi Chauhan**

## Table of Contents

Introduction .....	3
Exploratory Data Analysis .....	4
Data Cleaning.....	6
Data Visualization.....	8
Dashboard .....	17
Pre Modeling Steps .....	19
Model Building .....	21
Recommendations & Conclusion .....	26
Tools & Packages.....	28
References .....	29

## Introduction

[The Motor Vehicle Collisions – Crashes](#) dataset contains information and details about the various crashes that took place having various attributes describing each incident. The dataset has data collected from the police department that reported these incidents in the New York city. It helps in analyzing the crashes and understanding whether someone was injured or killed and what kind of injury did they face. It also analyzes the various factors due to which the crash took place, such that it can be prevented by determining the features that caused the crash.

The data has been collected from [data.cityofnewyork.us](http://data.cityofnewyork.us) and has about **1931867 row values of data** and **29 variables**, which consists of both numerical and categorical data. The features of the dataset consists of variables such as *Crash Date*, *Crash Time*, *Location*, *Street Name*, *Number of persons injured*, *Number of persons killed*, *Number of Cyclist injured*, *Number of Cyclist killed*, *Contributing Factor Vehicle 1*, etc. which helps in analyzing the dataset to understand the motor vehicle collisions that took place and what effect did it have. It contains the different locations where the accident took place along with the date and crash time. In many accidents, person get injured or killed or the pedestrians, cyclist and motorist are also injured or killed, and so this dataset contains information about that. The data contains information about the exact location of the event, including the number of people either injured or killed, type of the vehicle, and one of the most important feature being the factors which contributed to this event.

The data for the vehicle collisions and accidents have been collected which helps in analysis of this data such that the incidents can be prevented, and safety measures can be taken accordingly. The goal here is to identify the factors that are causing the vehicle collisions and recommend which factor would lead to a severe incident or a non-severe incident such that appropriate measures can be taken to order to prevent the collisions and accidents and ensure safety of the public. The variable that will help with the prediction or classification of the vehicle collisions would be the contributing factor vehicle and the number of persons killed or injured.

The objective of the project is to predict the number of persons killed based on the contributing factors such that necessary precautions and actions can be taken in order to avoid the accidents and reduce the death rates and injuries of the person.

## Exploratory Data Analysis

The idea to use this data is that we might be able to work around the explanatory analysis to figure out the most frequently occurring contributing factors which lead to such horrific events. This could be used to come up with a statistical combined with authority values solution to reduce the accidents. We can also have a deep look at the type of vehicles that are persuaded in the accidents, the company's manufacturing such type of vehicles can look into preventive measures such as introducing safety systems to minimize the risk.

The analysis of the Motor Vehicle Collisions – Crashes dataset will include the **data analytical process** and steps such as the *exploratory data analysis, data cleaning, data visualization, pre-modeling steps, and model building*. These steps will help in analyzing the data and predicting the target variable in order to be able to control the vehicle crashes and avoid accidents by taking precautions and safety measures being implemented. Data Visualization plays an important role in the data analytical process which helps in getting a clear understanding of the data points and also presenting the findings to the target users in an effective way.

The **exploratory data analysis** is performed on the motor vehicle collisions data which contains **1931867 row values of data** and **29 variables** having categorical, numerical, and boolean type data. The different parameters of the dataset are Crash Date, Crash Time, Location, Street Name, Number of persons injured, Number of persons killed, Number of Cyclist injured, Number of Cyclist killed, Contributing Factor Vehicle 1, etc. The various steps of EDA such as *descriptive analysis, statistical analysis, and data profiling* are implemented in order to get a better understanding of the dataset which is as follows.

```
#displaying the number of rows and columns of the dataset  
print("Total number of Rows and Columns:",collision_data.shape)
```

```
Total number of Rows and Columns: (1931867, 29)
```

*Figure 1. Dataset row and column values*

Column Names:

```
Index(['CRASH DATE', 'CRASH TIME', 'BOROUGH', 'ZIP CODE', 'LATITUDE',
      'LONGITUDE', 'LOCATION', 'ON STREET NAME', 'CROSS STREET NAME',
      'OFF STREET NAME', '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',
      'NUMBER OF MOTORIST KILLED', 'CONTRIBUTING FACTOR VEHICLE 1',
      'CONTRIBUTING FACTOR VEHICLE 2', 'CONTRIBUTING FACTOR VEHICLE 3',
      'CONTRIBUTING FACTOR VEHICLE 4', 'CONTRIBUTING FACTOR VEHICLE 5',
      'COLLISION_ID', 'VEHICLE TYPE CODE 1', 'VEHICLE TYPE CODE 2',
      'VEHICLE TYPE CODE 3', 'VEHICLE TYPE CODE 4', 'VEHICLE TYPE CODE 5'],
      dtype='object')
```

*Figure 2. Field values of the dataset*

	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	NUMBER OF MOTORIST KILLED	COLLISION_ID
count	1708825.00	1708825.00	1931849.00	1931836.00	1931867.00	1931867.00	1931867.00	1931867.00	1931867.00	1931867.00	1931867.00
mean	40.64	-73.77	0.29	0.00	0.05	0.00	0.03	0.00	0.21	0.00	3049602.52
std	1.88	3.56	0.68	0.04	0.24	0.03	0.16	0.01	0.65	0.03	1502895.11
min	0.00	-201.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	22.00
25%	40.67	-73.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3096853.50
50%	40.72	-73.93	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3602131.00
75%	40.77	-73.87	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4085358.50
max	43.34	0.00	43.00	8.00	27.00	6.00	4.00	2.00	43.00	5.00	4568716.00

*Figure 3. Statistical Analysis*

As observed in the statistical analysis, the average count of the various parameters in the dataset are computed which is helpful in understanding the total count of the number of persons killed based on the contributing factors. If we consider the parameters of the number of persons killed and injured, the average count is as mentioned in the figure above. Also, the minimum and maximum values of the variables are computed which is 0.00 and 43.00 respectively for the number of persons killed injured which helps us understand the total number of persons injured on an average. Thus, the statistical analysis helps understand the statistics of the parameters of the dataset.

## Data Cleaning

**Data Cleaning** is an important step to be performed in the analysis process and from the below dataset statistics we observe that there are many missing values for each of the parameters and variables of the dataset which we need to consider for cleaning. In order to determine which field values consists of the missing values, the count of missing values for each column is displayed. Apart from this, the distribution plot for the parameters gives an overview of the data points being normally distributed or whether the data is left skewed or right skewed.

It is observed that the various location and street columns have maximum of null values of the entire dataset and thus those columns can be dropped out as it is not an important feature that needs to be considered within the analysis of the data. Apart from that, the remaining columns that have maximum null values and that cannot be dropped are explored further in order to replace the null values with some value suitable for the data points. Lastly, the number of persons killed and injured have some null values within the field value and thus the rows of data can be dropped as it is a minimal amount of data as compared to the entire dataset. Thus, the following methods are performed to clean the data.

CRASH DATE	0
CRASH TIME	0
BOROUGH	599304
ZIP CODE	599538
LATITUDE	223042
LONGITUDE	223042
LOCATION	223042
ON STREET NAME	401257
CROSS STREET NAME	710430
OFF STREET NAME	1622980
NUMBER OF PERSONS INJURED	18
NUMBER OF PERSONS KILLED	31
NUMBER OF PEDESTRIANS INJURED	0
NUMBER OF PEDESTRIANS KILLED	0
NUMBER OF CYCLIST INJURED	0
NUMBER OF CYCLIST KILLED	0
NUMBER OF MOTORIST INJURED	0
NUMBER OF MOTORIST KILLED	0
CONTRIBUTING FACTOR VEHICLE 1	5814
CONTRIBUTING FACTOR VEHICLE 2	287618
CONTRIBUTING FACTOR VEHICLE 3	1796783
CONTRIBUTING FACTOR VEHICLE 4	1901886
CONTRIBUTING FACTOR VEHICLE 5	1923845
COLLISION_ID	0
VEHICLE TYPE CODE 1	11325
VEHICLE TYPE CODE 2	347740
VEHICLE TYPE CODE 3	1801212
VEHICLE TYPE CODE 4	1902849
VEHICLE TYPE CODE 5	1924070

Figure 4. Count of missing values

```
#changing datatype of the variables

collision_data['CRASH DATE'] = collision_data['CRASH DATE'].astype('datetime64[ns]')
collision_data['CRASH TIME'] = collision_data['CRASH TIME'].astype('datetime64[ns]')
collision_data['BOROUGH'] = collision_data['BOROUGH'].astype('str')
#collision_data['ZIP CODE'] = collision_data['ZIP CODE'].astype('int')
collision_data['ON STREET NAME'] = collision_data['ON STREET NAME'].astype('str')
collision_data['CROSS STREET NAME'] = collision_data['CROSS STREET NAME'].astype('str')
collision_data['OFF STREET NAME'] = collision_data['OFF STREET NAME'].astype('str')
collision_data['CONTRIBUTING FACTOR VEHICLE 1'] = collision_data['CONTRIBUTING FACTOR VEHICLE 1'].astype('str')
collision_data['CONTRIBUTING FACTOR VEHICLE 2'] = collision_data['CONTRIBUTING FACTOR VEHICLE 2'].astype('str')
collision_data['CONTRIBUTING FACTOR VEHICLE 3'] = collision_data['CONTRIBUTING FACTOR VEHICLE 3'].astype('str')
collision_data['CONTRIBUTING FACTOR VEHICLE 4'] = collision_data['CONTRIBUTING FACTOR VEHICLE 4'].astype('str')
collision_data['CONTRIBUTING FACTOR VEHICLE 5'] = collision_data['CONTRIBUTING FACTOR VEHICLE 5'].astype('str')
collision_data['VEHICLE TYPE CODE 1'] = collision_data['VEHICLE TYPE CODE 1'].astype('str')
collision_data['VEHICLE TYPE CODE 2'] = collision_data['VEHICLE TYPE CODE 2'].astype('str')
collision_data['VEHICLE TYPE CODE 3'] = collision_data['VEHICLE TYPE CODE 3'].astype('str')
collision_data['VEHICLE TYPE CODE 4'] = collision_data['VEHICLE TYPE CODE 4'].astype('str')
collision_data['VEHICLE TYPE CODE 5'] = collision_data['VEHICLE TYPE CODE 5'].astype('str')

#filling null values with 0 and then converting the datatype
collision_data['NUMBER OF PERSONS KILLED'] = collision_data['NUMBER OF PERSONS KILLED'].fillna(0)
collision_data['NUMBER OF PERSONS INJURED'] = collision_data['NUMBER OF PERSONS INJURED'].fillna(0)

collision_data['NUMBER OF PERSONS KILLED'] = collision_data['NUMBER OF PERSONS KILLED'].astype('int')
collision_data['NUMBER OF PERSONS INJURED'] = collision_data['NUMBER OF PERSONS INJURED'].astype('int')

print("Datatype conversion completed!")
```

✓ 4.7s

Datatype conversion completed!

Figure 5. Datatype conversion

```

#replacing null values/blanks of Borough column with 'No Value'
collision_data['BOROUGH'] = collision_data['BOROUGH'].replace('NaN', 'No Value')
print("Replace Successful")
[7] ✓ 0.1s Python
... Replace Successful

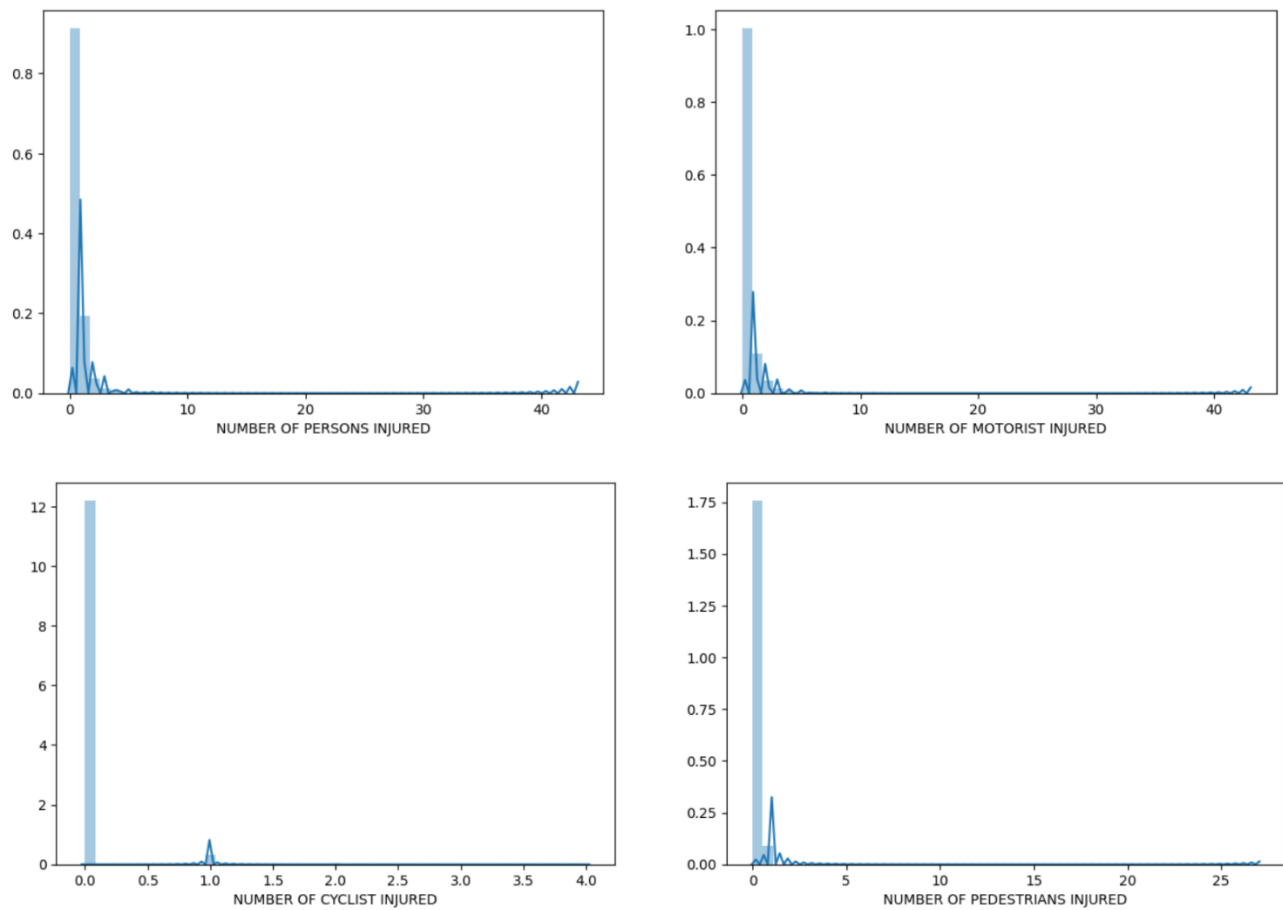
#replacing null values/blanks of Contributing Factor 1 column with 'No Value'
collision_data['CONTRIBUTING FACTOR VEHICLE 1'] = collision_data['CONTRIBUTING FACTOR VEHICLE 1'].replace('NaN', 'No Value')
print("Replace Successful")
[8] ✓ 0.9s Python
... Replace Successful

#replacing null values/blanks of Vehicle Type Code 1 column with 'No Value'
collision_data['VEHICLE TYPE CODE 1'] = collision_data['VEHICLE TYPE CODE 1'].replace('NaN', 'No Value')
print("Replace Successful")
[9] ✓ 0.1s Python
... Replace Successful

#dropping columns which have maximum null values and are not required for analysis
collision_data = collision_data.drop(['ZIP CODE', 'LATITUDE', 'LONGITUDE', 'LOCATION', 'ON STREET NAME', 'CROSS STREET NAME', 'OFF STREET NAME', 'CONTRIBUTING FACTOR VEHICLE 2',
'CONTRIBUTING FACTOR VEHICLE 3', 'CONTRIBUTING FACTOR VEHICLE 4', 'CONTRIBUTING FACTOR VEHICLE 5', 'VEHICLE TYPE CODE 2', 'VEHICLE TYPE CODE 3', 'VEHICLE TYPE CODE 4',
'VEHICLE TYPE CODE 5'], axis=1)
print("Variables dropped!")
[10] ✓ 1.1s Python
... Variables dropped!

```

*Figure 6. Replacing and dropping values of the dataset*



*Figure 7. Distribution Plot*

## Data Visualization

Based on the visualizations created, which are presented below, and the dataset analyzed, it is observed that the data contains information about the people killed and injured due to the crash which has more details such as contributing factors, crash date and time, location of the crash, etc. which helps to visualize where the death rate or injury to the person was high and what factors contribute to the crash, which can be then avoided by implementing various safety measures. The objective here is to understand the reasons and factors for which the crash took place, such that preventive measures can be recommended and implemented to reduce the crash collision in the future.

The dataset is loaded in the data source of Tableau where Data Interpreter is used in order to clean the data and bring the data in a format which can be accepted for visualizations by Tableau. The data cleaning part and null values are taken into consideration below the visualization and necessary filters are applied for the same. The attributes of the dataset such as persons killed and injured, cyclist killed and injured, motorist killed and injured, etc. are combined together by a calculation in Tableau to get an overall number of people killed and injured due to the crash collisions, and it is observed that, total number of *people killed* due to the crash collision are **5321** and the total number of *people injured* due to the crash collision are **1,123,478**. The below graphs and visuals along with the analysis of the data give an overall view of the dataset which can help us in providing further recommendations that can be used to avoid and reduce the accidents.

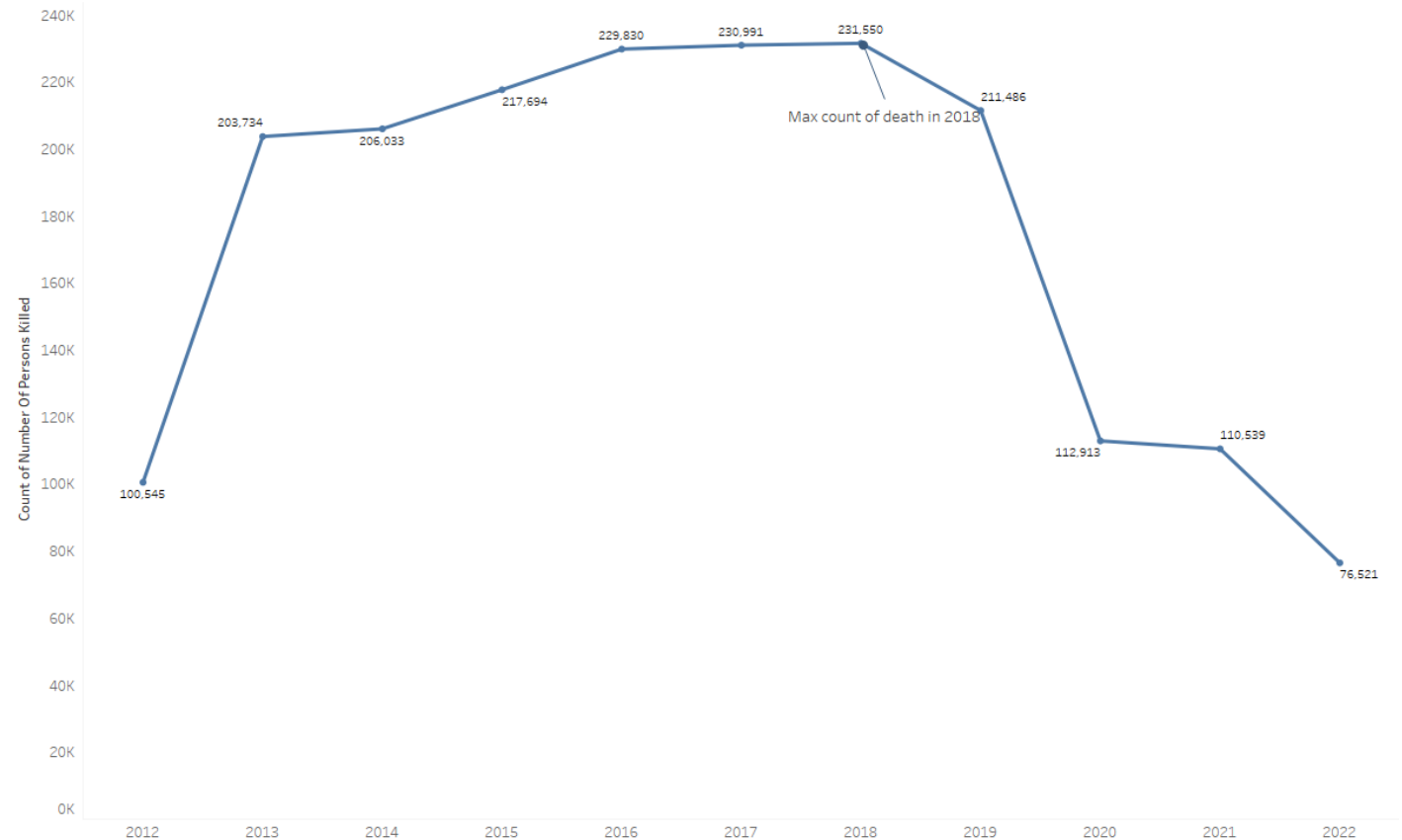
### Graph 1: Count of number of persons killed reported in the year

The graph below represents the count of number of persons killed which were reported in the year from 2012 to 2022. As it can be observed, the count of persons killed increased rapidly after 2012, and kept increasing for the rest of the years. In the year **2018**, the count of persons killed is the highest as compared to the other years which is **231,550**, after which the count decreased from the year 2019 to 2022. The year **2022** encountered minimum of death rates, i.e., **76,521**, as compared to others and thus it is observed that appropriate measures were taken. This helps in keeping a track of the persons killed within the year and based on the past analysis, an analysis or prediction



can be made for the coming years which could help the police department to keep record of the accidents and ensure safety measures.

Count of No. of persons killed reported in the year



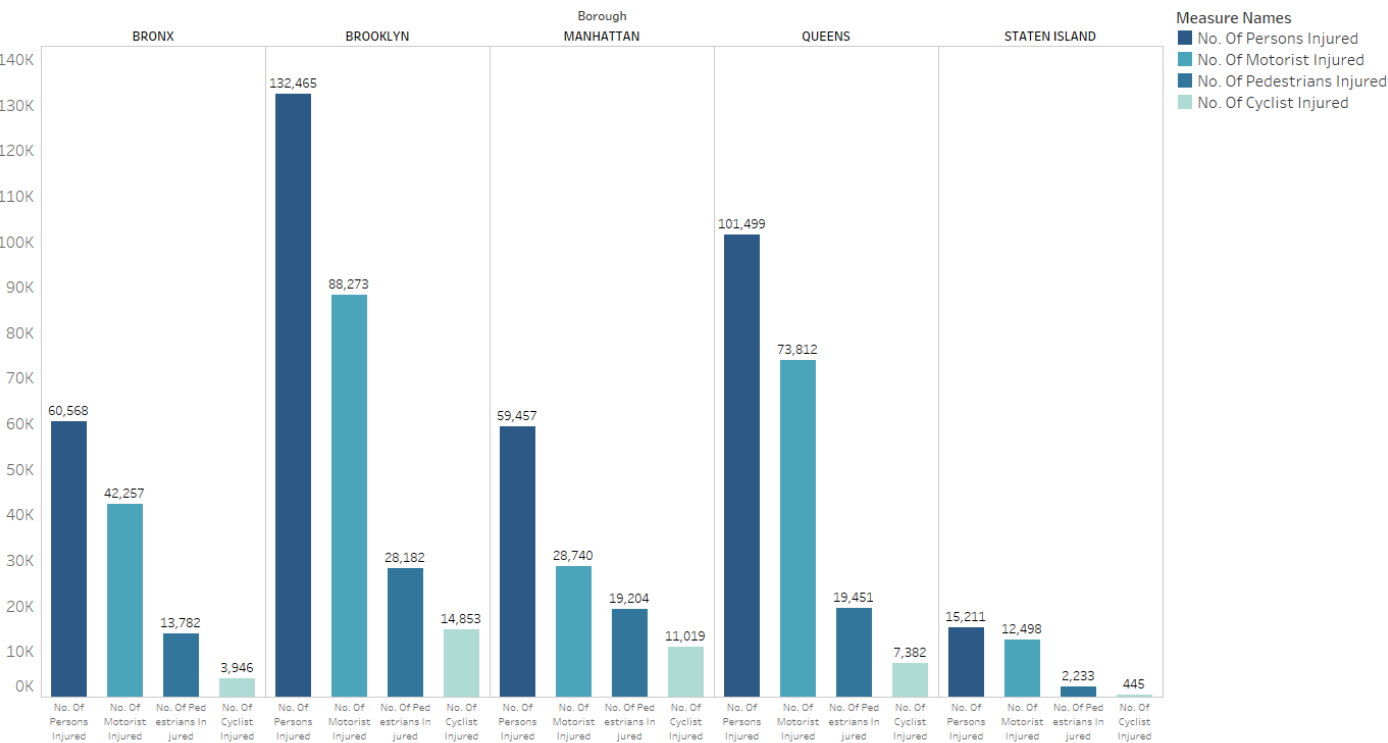
The trend of count of Number Of Persons Killed for Crash Date Year.

## Graph 2: Analysis of people injured based on Borough

Graph 2, i.e., analysis of people injured based on Borough gives an idea about the persons injured, the motorist injured, the cyclist injured, and the pedestrians injured which is filtered based on the Borough. This visual representation helps understanding the injury rate of various categories filtered by borough, such that for the maximum number of injury encountered in a borough, necessary actions and precautions can be taken. Thus, it is observed that **Brooklyn** has the highest number of persons injured which is **132,465** whereas **Staten Island** has the least number of persons injured, i.e., **15,211**. Overall analysis of the graph shows that Brooklyn encounters the highest

number of persons, motorist, cyclist, and pedestrians being injured as compared to others, and thus necessary measures are required in Brooklyn in order to reduce the injury number and avoid crash collisions. Followed by Brooklyn is **Queens**, which has the next highest count of injuries and thus needs attention on the same.

Analysis of people injured based on Borough

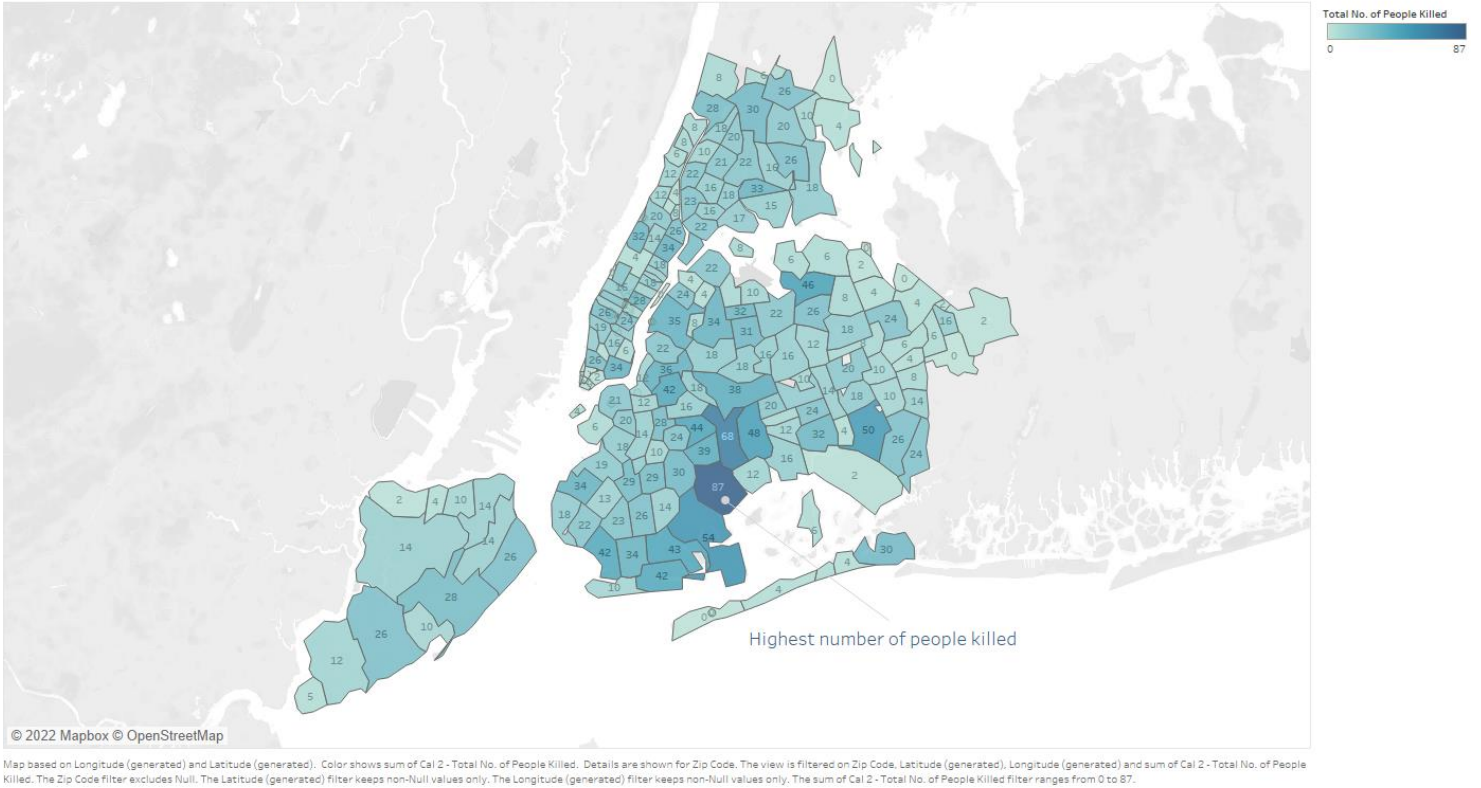


No. Of Cyclist Injured, No. Of Motorist Injured, No. Of Pedestrians Injured and No. Of Persons Injured for each Borough. Color shows details about No. Of Cyclist Injured, No. Of Motorist Injured, No. Of Pedestrians Injured and No. Of Persons Injured. The view is filtered on Borough, which keeps BRONX, BROOKLYN, MANHATTAN, QUEENS and STATEN ISLAND.

**Graph 3: Total number of people killed in the area**

The below graph represents the total number of people killed in the area, based on the Zip Code. Here, the sum of number of persons killed, number of motorist killed, number of cyclist killed, and number of pedestrians killed are taken into consideration for an overall analysis to understand the average of number of people killed in the New York state, by area. The highest number of people killed is in the area with zip code **11236**, which is **87**. For this particular graph, a slider is enabled which would help analyze the data in an effective way for a particular area.

Total No. of People killed in the area  
(Move the slider for lowest and highest count in the city)

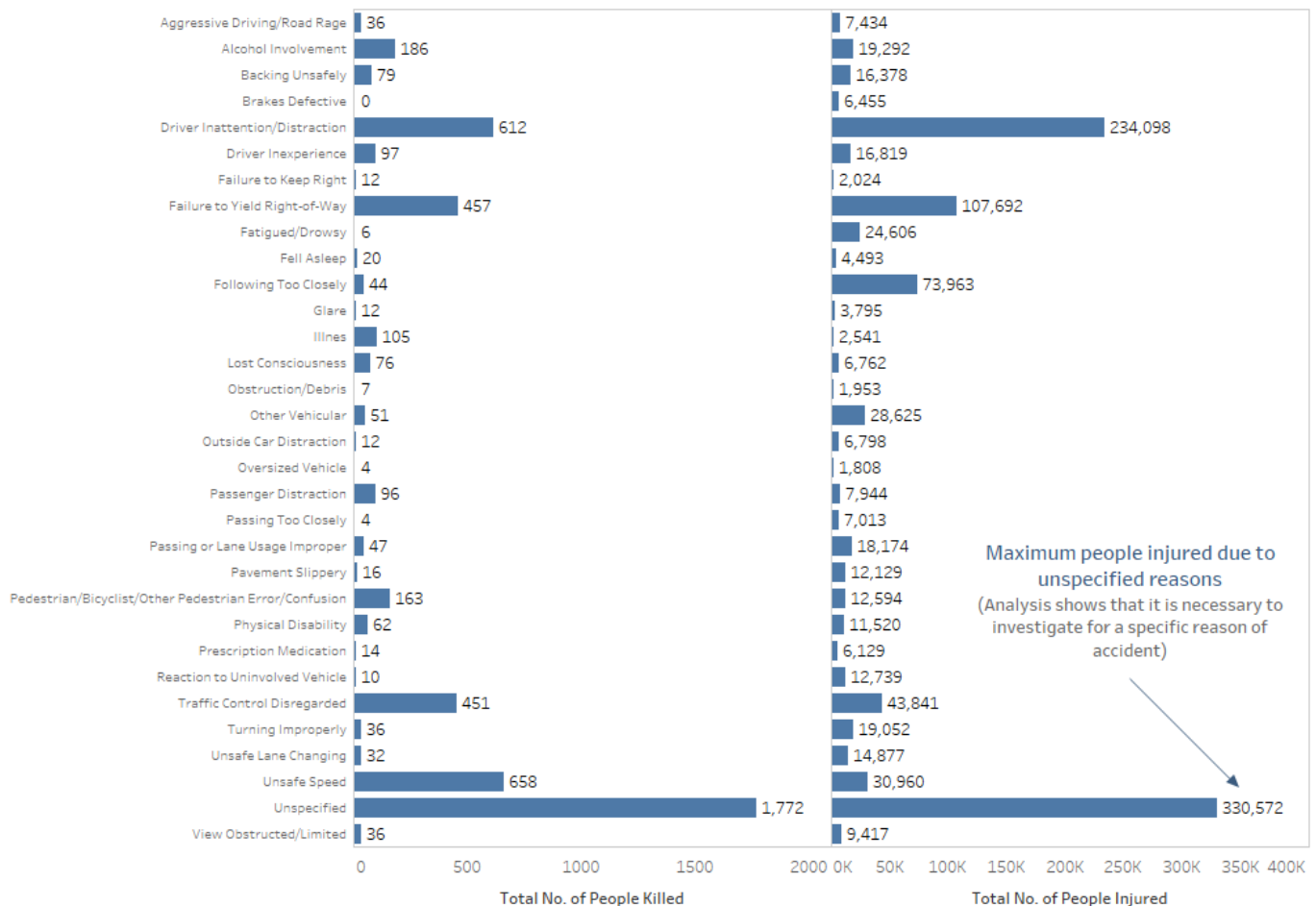


#### Graph 4: Number of persons killed & injured based on contributing factor

The next graph represents the total number of persons killed and injured based on the contributing factor. Contributing factors are the features that help understand why the accident or the crash collision happened in the first place. The dropdown function here would help in analyzing the number of persons killed and injured based on the specific contributing factor. But the observation here is that the highest number of persons killed and injured are due to **unspecified reasons**, and thus it is recommended to investigate these incidents in order to have a specific reason of accident.

## No. of Persons Killed & Injured based on the Contributing Factor

(For specific analysis, choose CF value from dropdown)

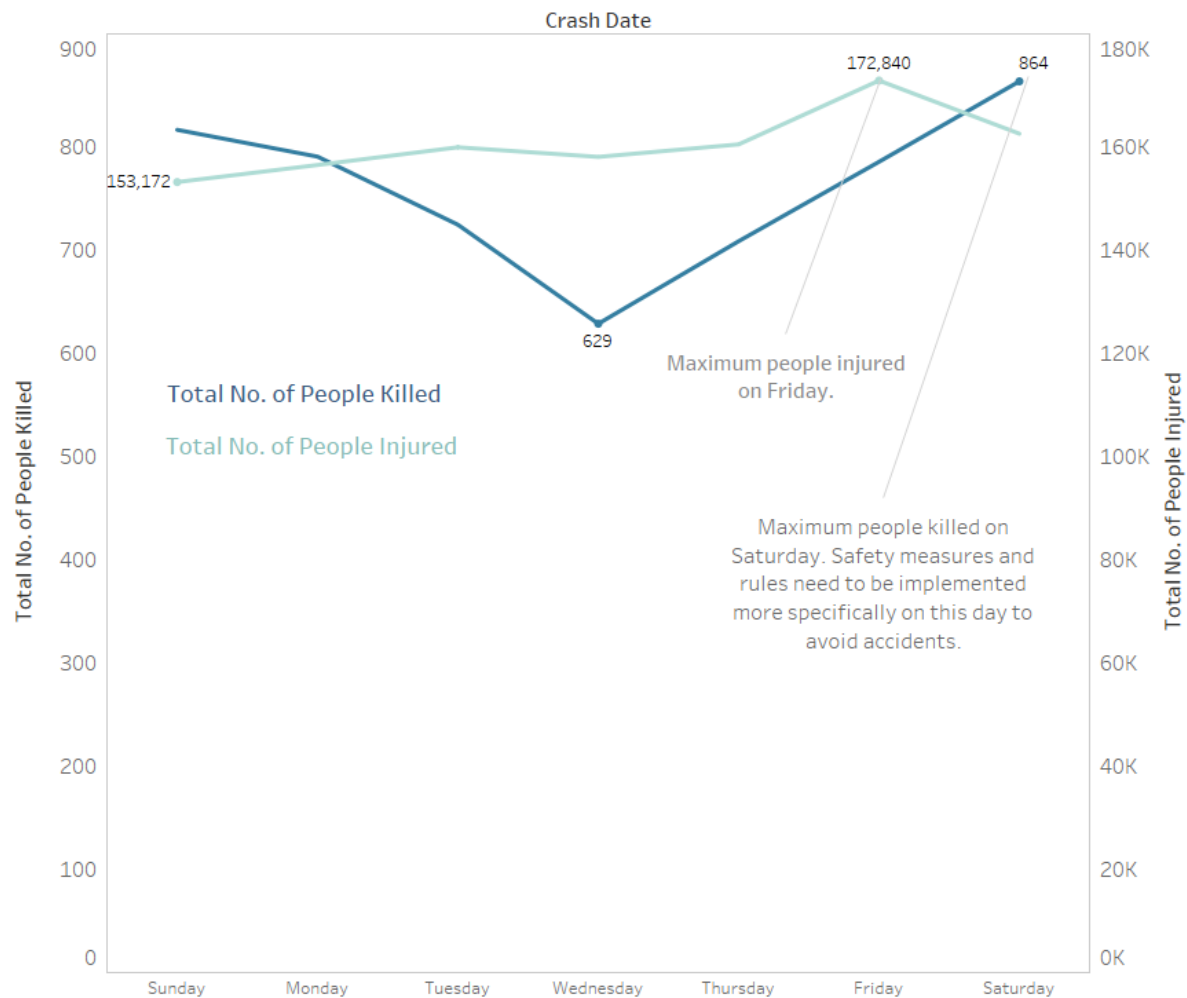


Sum of Cal 2 - Total No. of People Killed and sum of Cal 3 - Total No. of People Injured for each Contributing Factor Vehicle 1. The data is filtered on sum of Number Of Persons Injured and Calculation2 - CF1. The sum of Number Of Persons Injured filter ranges from 900 to 165,669. The Calculation2 - CF1 filter keeps 62 of 62 members. The view is filtered on Contributing Factor Vehicle 1, which keeps 32 of 62 members.

### Graph 5: Day wise Analysis of number of people killed & injured

The day wise analysis of number of people killed and injured help in analyzing the crash collisions on a regular basis, specifically analyzing in a day wise manner. The graph thus represents both the total number of persons killed and injured during each day and we can see that *maximum people meet with an accident on Saturday* and are *highest number of people are injured on Friday*. This recommends implementing strict actions and rules specifically on these particular days which will help avoid the crash collisions and reduce the deaths and injuries.

## Day wise Analysis of No. of People Killed & Injured

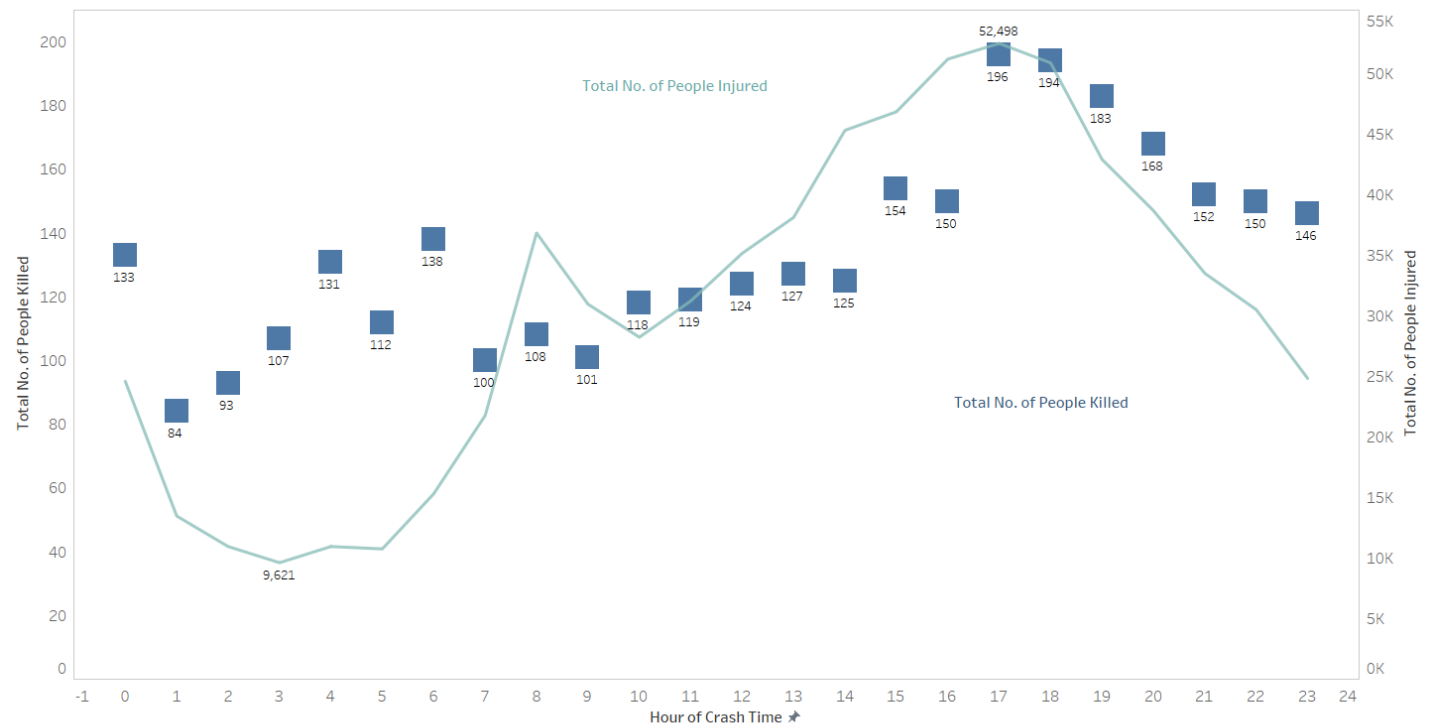


The trends of Cal 2 - Total No. of People Killed and Cal 3 - Total No. of People Injured for Crash Date Weekday. Color shows details about Cal 2 - Total No. of People Killed and Cal 3 - Total No. of People Injured.

## Graph 6: Hourly Analysis of people killed & injured

Similar to the day wise analysis of the number of people killed and injured, an hourly analysis of the same is implemented in order to understand the graph with respect to the hourly manner and understand the trend of the highest and lowest accidents in each hour of the day. The square shapes indicate the total number of people killed and the line graph indicates the total number of people injured, thus the visual is a *combination graph*. The highest & lowest number of people killed & injured are indicated in the graph as shown below.

Hourly Analysis of People Killed & Injured

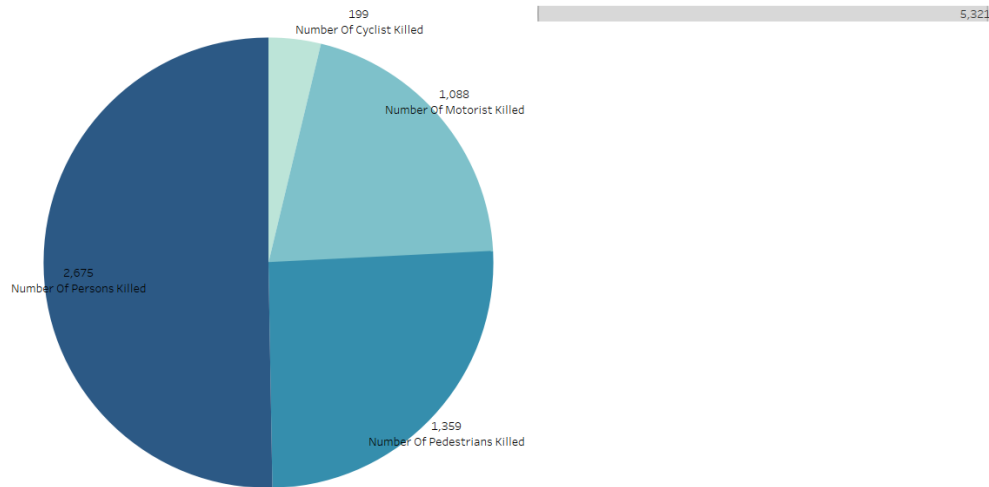


The trends of Cal 2 - Total No. of People Killed and Cal 3 - Total No. of People Injured for Crash Time Hour. Details are shown for Cal 2 - Total No. of People Killed and Cal 3 - Total No. of People Injured. The data is filtered on Borough, which keeps BRONX, BROOKLYN, MANHATTAN, QUEENS and STATEN ISLAND.

## Graph 7: Overall Analysis of People Killed

The overall analysis of people killed graph represented in a pie chart helps us understand the maximum and minimum or the highest and lowest number of people killed, thus helping to analyze the category of people on the road that highly meet with the accident. As observed in the graph, highest number of pedestrians, i.e., **1359 pedestrians** are killed as compared to others due to the collision crash and thus based on this analysis it is recommended that necessary precautions and actions to be taken for the safety of the pedestrians.

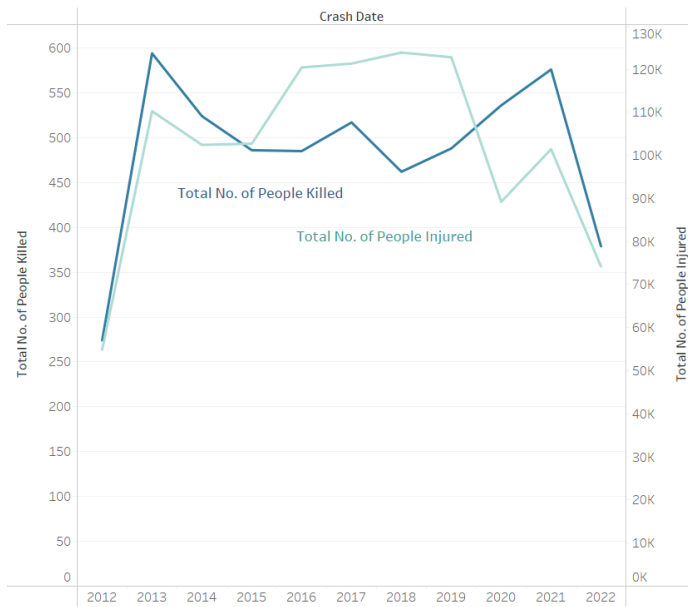
## Overall Analysis of People Killed



Number Of Cyclist Killed, Number Of Motorist Killed, Number Of Pedestrians Killed, Number Of Persons Killed, Number Of Cyclist Killed, Number Of Motorist Killed, Number Of Pedestrians Killed and Number Of Persons Killed. Color shows details about Number Of Cyclist Killed, Number Of Motorist Killed, Number Of Pedestrians Killed and Number Of Persons Killed. Size shows Number Of Cyclist Killed, Number Of Motorist Killed, Number Of Pedestrians Killed and Number Of Persons Killed. The marks are labeled by Number Of Cyclist Killed, Number Of Motorist Killed, Number Of Pedestrians Killed, Number Of Persons Killed, Number Of Cyclist Killed, Number Of Motorist Killed, Number Of Pedestrians Killed and Number Of Persons Killed.

## Graph 8: Overall Analysis of People Killed & Injured in the year

### Overall Analysis of People Killed & Injured in the year



The trends of Cal 2 - Total No. of People Killed and Cal 3 - Total No. of People Injured for Crash Date Year. Color shows details about Cal 2 - Total No. of People Killed and Cal 3 - Total No. of People Injured.

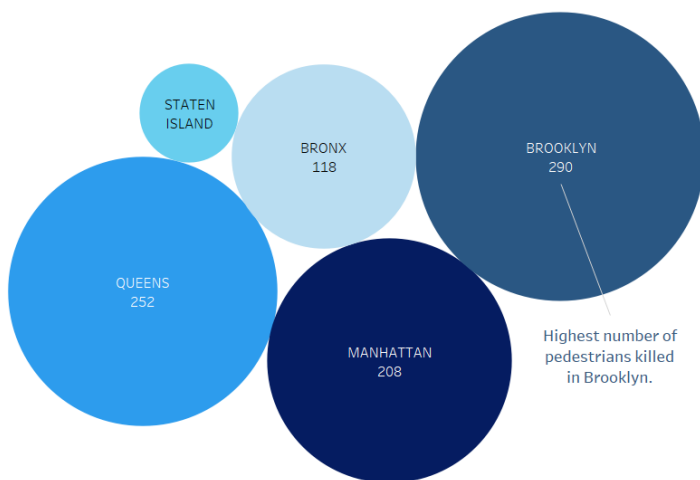
This graph represents the overall analysis of people killed & injured within the year. The sum of persons killed & injured, along with the motorist, cyclist, and pedestrians are taken into consideration for an overall analysis to distinguish between the people killed and people injured within the years. The visual thus helps in understanding the trend and pattern of the crashes that have taken place within the year along with the total count of people who died or have been injured due to the crash. A line graph is helpful when we have a time series

pattern of data and thus it illustrates the importance of time on a yearly basis for the analysis of persons killed and injured within the year due to the collision crash.

### Graph 9: Number of Pedestrians killed in Borough

The next graph talks specifically about the analysis of pedestrians killed in different borough of New York city. Pedestrians killed in borough is specifically considered in order to understand the total number of kills happened to people who walked on the road. This would imply that if a borough in the city has maximum number of pedestrians killed, it requires more attention in order to avoid and reduce the death rate. Thus, through the bubble chart it is observed that **Brooklyn** has the highest number of pedestrians killed due to the crash collision.

No. of Pedestrians killed in Borough



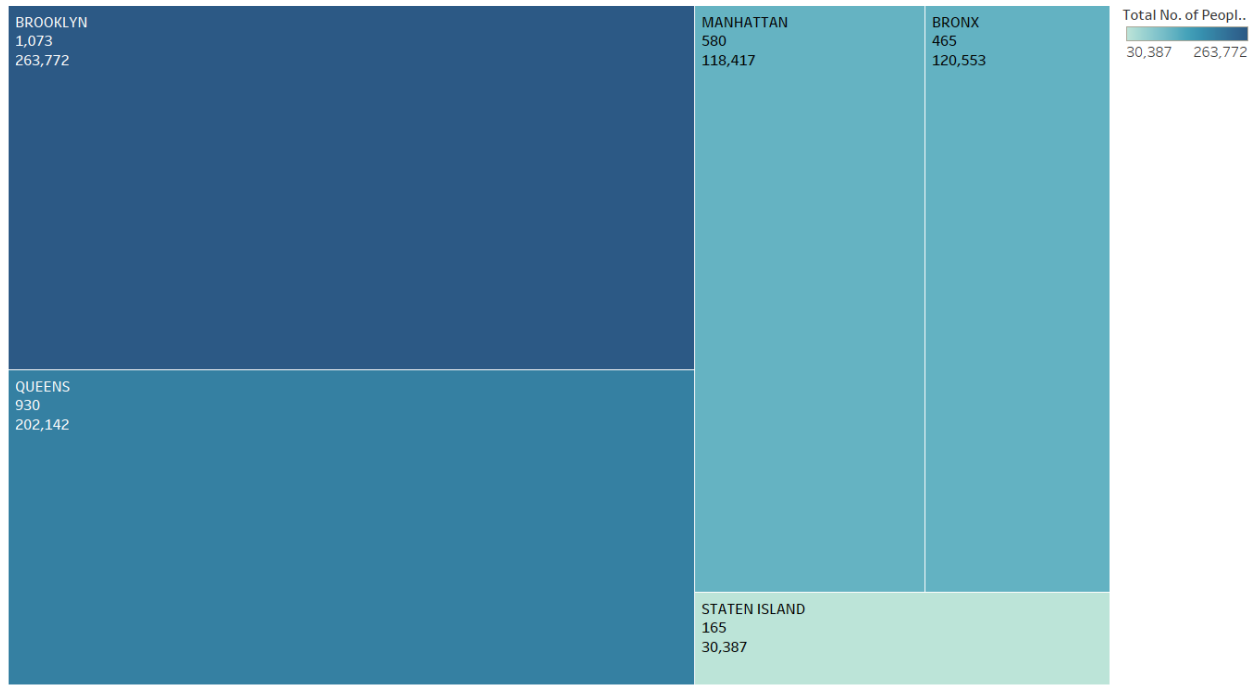
Borough and sum of Number Of Pedestrians Killed. Color shows details about Borough. Size shows sum of Number Of Pedestrians Killed. The marks are labeled by Borough and sum of Number Of Pedestrians Killed. The data is filtered on Contributing Factor Vehicle 1, which excludes Null, 1 and 80. The view is filtered on Borough, which keeps BRONX, BROOKLYN, MANHATTAN, QUEENS and STATEN ISLAND.

### Graph 10: Total Number of people killed & injured in Borough

Lastly, the heatmap represents the total number of people killed and injured in the borough. After digging a little further to see which borough had the most fatalities with respect to the number of persons killed and number of persons injured, we discovered that Brooklyn is the most populated borough in New York and has the most fatal pedestrian and bike accidents.



Total No. of People Killed & Injured in Borough



Borough, sum of Cal 2 - Total No. of People Killed and sum of Cal 3 - Total No. of People Injured. Color shows sum of Cal 3 - Total No. of People Injured. Size shows sum of Cal 2 - Total No. of People Killed. The marks are labeled by Borough, sum of Cal 2 - Total No. of People Killed and sum of Cal 3 - Total No. of People Injured. The view is filtered on Borough, which keeps BRONX, BROOKLYN, MANHATTAN, QUEENS and STATEN ISLAND.

## Dashboard

The dashboard of **Motor Vehicle Collision Crash Analysis** shown below gives an overall analysis of the crashes and collisions taken place in the New York city. This dashboard is an effective representation of the analysis and to identify the various trends and patterns of the dataset. The aim of building the dashboard is to present the analysis to the users in an effective manner and help recommend solutions based on the findings from the visuals. It summaries the dataset, its purpose, the findings from each of its visual and helps in building an effective solution from the complete analysis. The analysis of the contributing factor helps to understand the root cause of the problem and the crash and thus helps in building a solution by implementing certain rules for the same. The representation of the dashboard for the Motor Vehicle Collision Crash dataset is as shown below.

*(The tableau dashboard, .twbx file is attached along with this report, for effective view of the visuals)*

# Motor Vehicle Collision Crash Analysis

The Motor Vehicle Collisions Crash Analysis talks about the details on the crash event. The Motor Vehicle Collisions data tables contain information from all police reported motor vehicle collisions in NYC.

Navigation - Hover over each data points on the dashboard for exact values.

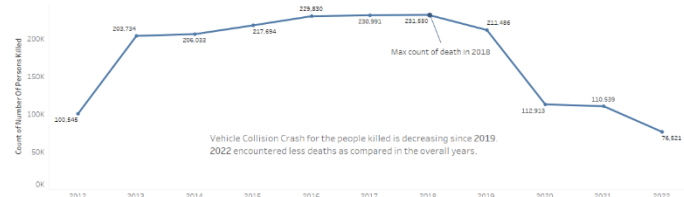
Total No. of People Killed in NYC due to Motor Vehicle Collision Crash

5,321

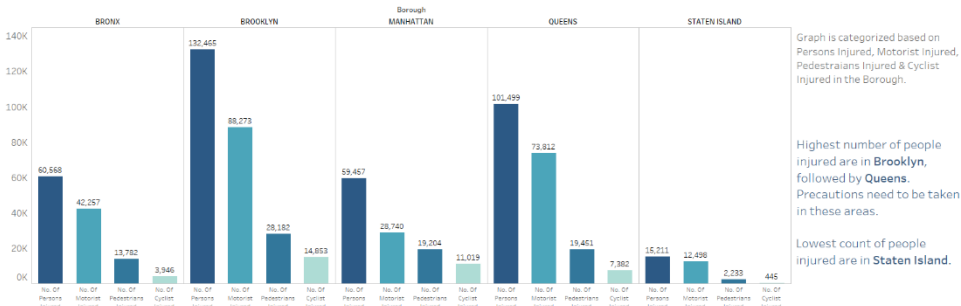
Total No. of People Injured in NYC due to Motor Vehicle Collision Crash

1,123,478

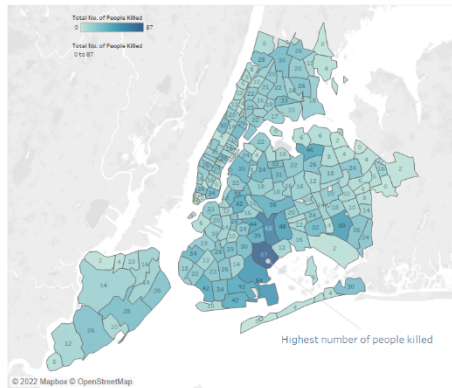
Count of No. of persons killed reported in the year



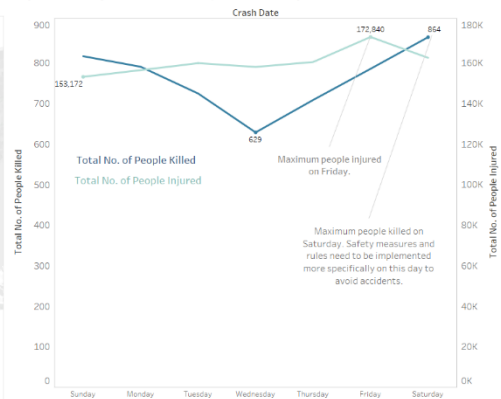
Analysis of people injured based on Borough



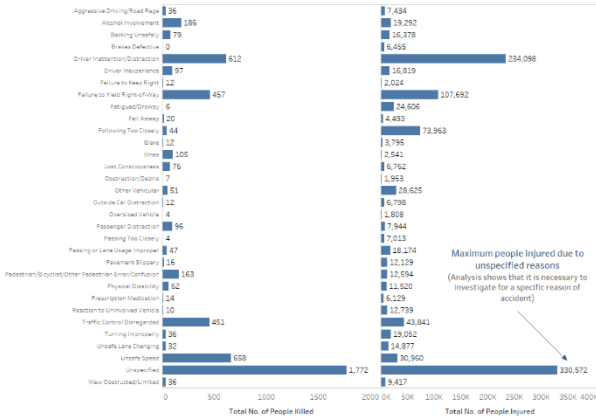
Total No. of People killed in the area (Move the slider for lowest and highest count in the city)



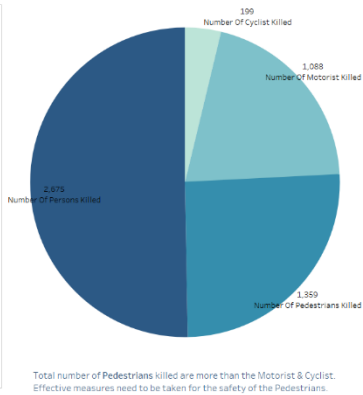
Day wise Analysis of No. of People Killed & Injured



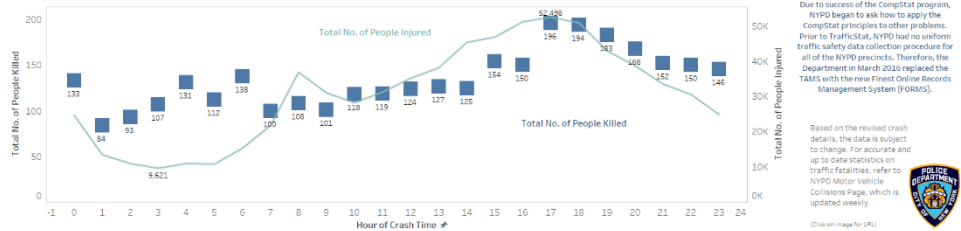
No. of Persons Killed & Injured based on the Contributing Factor (For specific analysis, choose CF value from dropdown)



Overall Analysis of People Killed



Hourly Analysis of People Killed & Injured



## Pre Modeling Steps

The pre modeling steps include various process and implementation of steps in order to understand the independent variables and the relationship between each of these parameters and features in order to build the machine learning model where these features would be given for the training of the model. The pre modeling steps consists of feature selection and extraction, correlation plot, and label encoding. The automatic feature selection and extraction will automatically identify the important and significant parameters and variables which will help to predict the dependent variable. The correlation plot determines the correlation between each of the variables and if there is a high collinearity that exists between the variables, either of the variables are dropped to avoid multicollinearity and overfitting of the model. Lastly, label encoding is performed on the categorical variables which are considered for the feature extraction.

### Label Encoding

Since the dataset consists of categorical data which needs to be considered as the features for model building, the independent variables need to be label encoded in order to have numerical data passed to the model. Thus, the features such as ‘Borough’, ‘Contributing Factor Vehicle’, and ‘Vehicle Type Code’ are label encoded to numerical form which can be considered as features to training of the model.

BOROUGH	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	NUMBER OF MOTORIST KILLED	CONTRIBUTING FACTOR VEHICLE 1	COLLISION_ID	VEHICLE TYPE CODE 1	Borough Labels	Contributing Factor Vehicle 1 Labels	Vehicle Type Code 1 Labels
nan	2	0	0	0	0	0	2	0	Aggressive Driving/Road Rage	4455765	Sedan	5	3	893
nan	1	0	0	0	0	0	1	0	Pavement Slippery	4513547	Sedan	5	39	893
nan	0	0	0	0	0	0	0	0	Following Too Closely	4541903	Sedan	5	21	893
BROOKLYN	0	0	0	0	0	0	0	0	Unspecified	4456314	Sedan	1	56	893
BROOKLYN	0	0	0	0	0	0	0	0	nan	4486609	nan	1	61	1287

*Figure 8. Label Encoding*

## Correlation Plot

The correlation plot helps to understand the correlation between each of the independent variables of the dataset and to plot the correlation values of the parameters in order to know the collinearity between the variables such that the variables with high collinearity value can be dropped in order to avoid multicollinearity and overfitting of the model. From the correlation plot it is observed that since there are different variables and parameters which have a **high collinearity** that is existing with a correlation value of **0.9** and **0.7**. This would result in multicollinearity and the model would outperform resulting in inaccurate model and results. Thus, some of the features having high collinearity are dropped before considering them for training of the model based on which features are important for the model. The correlation plot for the motor vehicle collision crash dataset is as shown below.

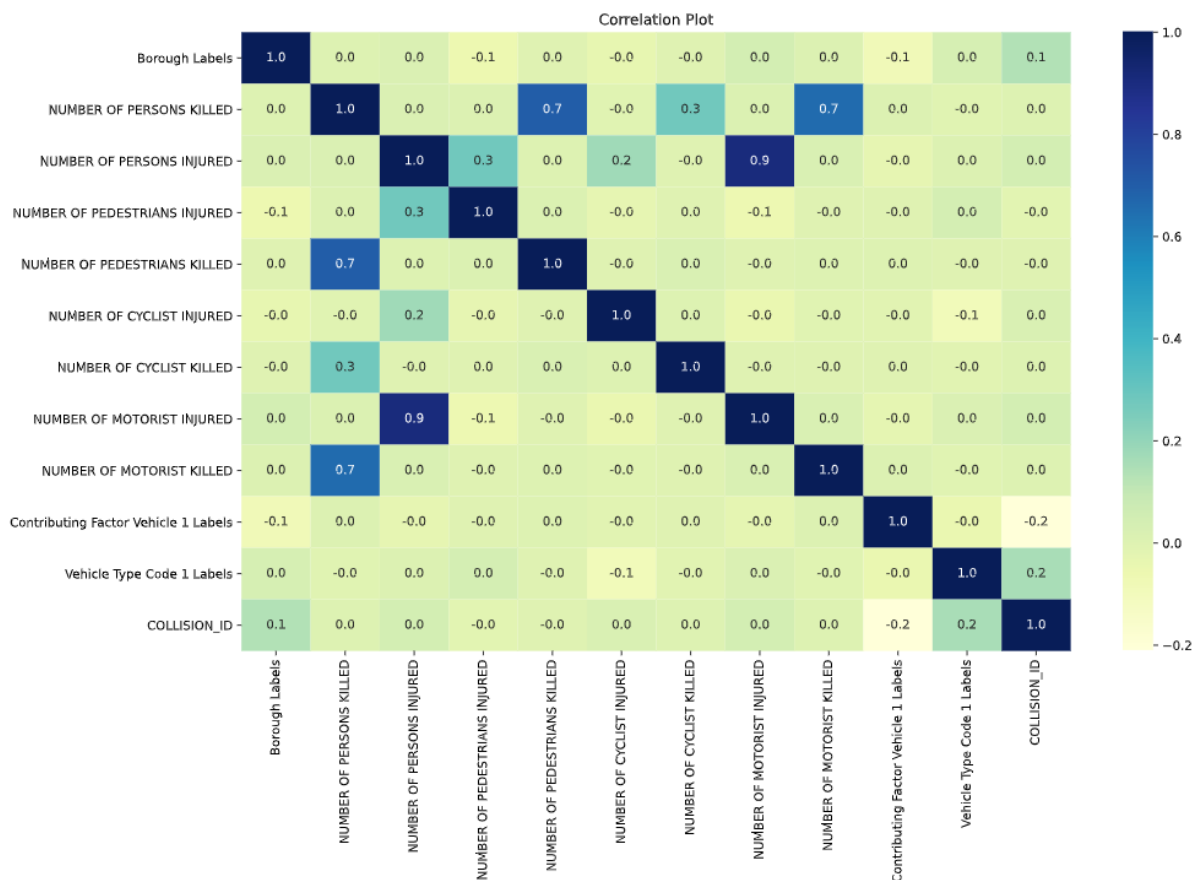
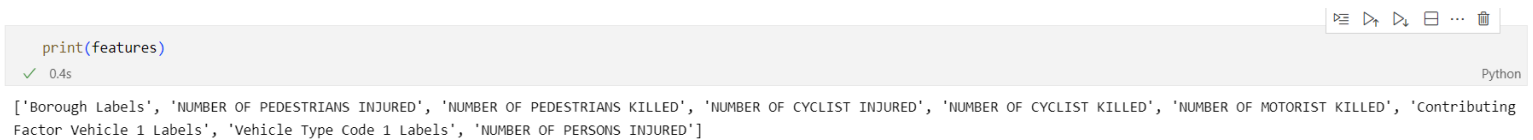


Figure 9. Correlation Plot

## Feature Selection & Extraction

The automatic feature selection and extraction is performed in order to extract the important and significant features from the dataset which would help in the classification and prediction of the number of persons killed based on the contributing factors. The features that were selected are *'Borough Labels', 'NUMBER OF PEDESTRIANS INJURED', 'NUMBER OF PEDESTRIANS KILLED', 'NUMBER OF CYCLIST INJURED', 'NUMBER OF CYCLIST KILLED', 'NUMBER OF MOTORIST KILLED', 'Contributing Factor Vehicle 1 Labels', 'Vehicle Type Code 1 Labels', 'NUMBER OF PERSONS INJURED'*. These features would help in the prediction of total number of persons killed based on various features selected for training of the model and also to understand the various contributing factors that are affecting the death rates and injuries of the person.



```
print(features)
```

✓ 0.4s Python

```
['Borough Labels', 'NUMBER OF PEDESTRIANS INJURED', 'NUMBER OF PEDESTRIANS KILLED', 'NUMBER OF CYCLIST INJURED', 'NUMBER OF CYCLIST KILLED', 'NUMBER OF MOTORIST KILLED', 'Contributing Factor Vehicle 1 Labels', 'Vehicle Type Code 1 Labels', 'NUMBER OF PERSONS INJURED']
```

*Figure 10. Features selected for Modeling*

## Model Building

The model building is performed after the implementation of the pre modeling steps in order to predict the number of persons killed based on contributing factors. Since the objective of the project is to predict the numbers of persons killed and determine the factors affecting the death rate, classification machine learning models cannot be implemented and thus regressor models are trained and implemented which would predict the number of persons killed. The different regressor models implemented for the prediction of the number of persons killed are **Linear Regressor Model, Decision Tree Regressor, and Random Forest Regressor** models. For the implementation of these machine learning models and based on the feature extraction and correlation plot, the significant features that are considered are as mentioned in the pre modeling steps. Since the variables of the collisions data gave a high collinearity, the number of motorist injured is not considered due to its high collinearity with number of persons injured parameter.

## Linear Regressor Model

Linear regression is a basic predictive analytics approach that predicts an output variable using historical data. The core concept is that if we can fit a linear regression model to observed data, we can use it to predict future values. The implementation of the Linear Regressor Model consist of various features such as borough, number of persons injured, number of pedestrians injured, number of pedestrians killed, number of cyclist killed, number of cyclist injured, contributing factors and vehicle type code for the prediction of the number of persons killed based on contributing factors. The dataset is split into training and testing data, i.e., **80% training** and **20% for testing** the model.

### Linear Regression Model

```
[49] #linear regression model

linear_regressionmodel = LinearRegression()
linear_regressionmodel.fit(X_train, y_train)

... ▾ LinearRegression
LinearRegression()

▷ ▾ #predict the result for the model

predicted_value_LR = linear_regressionmodel.predict(X_test)

[50]
```

The accuracy obtained for both the training and testing set of the Linear Regression model is **98%** and **98.26%** respectively. Since this is a regressor type of model, the model evaluation is based on the **MAE, MSE, RMSE and R-Squared** values in order to determine the prediction error of the model implemented.

*Figure 11. Linear Regressor Model*

```
Accuracy of Linear Regressor model on training set: 0.98
Accuracy of Linear Regressor model on test set:      0.98
0.9826
```

```
Model Evaluation of Linear Regression.
Mean Absolute Error: 0.0
Mean Squared Error: 0.0
Root Mean Squared Error: 0.0
R-Squared value: 0.9826460733537029
```

*Figure 12. Accuracy & Model Evaluation*

## Decision Tree Regressor Model

Decision trees are frequently used in operations research, particularly in decision analysis, to aid in the identification of the most likely method to achieve a goal. One of the advantages of a decision tree model is that the predictor and target variables are straightforward to grasp.

The independent variables considered for the implementation of the Decision Tree Regressor are borough, number of persons injured, number of pedestrians injured, number of pedestrians killed, number of cyclist killed, number of cyclist injured, contributing factors and vehicle type code. The decision tree model is implemented with a train and test data **split of 80-20** with a **maximum branch depth of 5**.

## Decision Tree Regressor Model

```
▷ ~  
#decision tree regressor  
  
decisiontree_model = DecisionTreeRegressor(max_depth=5)  
decisiontree_model.fit(X_train, y_train)  
[54]  
...  
▼ DecisionTreeRegressor  
DecisionTreeRegressor(max_depth=5)  
  
#predict the result for the model  
  
predicted_value_dt = decisiontree_model.predict(X_test)  
[55]
```

The accuracy of the Decision Tree model obtained for the prediction of number of persons killed is **99%** and **98.1%** for both the **training and testing set** respectively, which is a good accuracy overall, but machine learning models having a 99% accuracy is not effective. This is because the model is overfitted due to the multicollinearity of the parameters.

*Figure 13. Decision Tree Regressor Model*

The feature importance, and model evaluation for the Decision Tree Regressor model is as follows.

```
Model Evaluation of Decision Tree Regressor.  
Mean Absolute Error: 0.0  
Mean Squared Error: 0.0  
Root Mean Squared Error: 0.0  
R-Squared value: 0.9808530602314666
```

*Figure 14. Model Evaluation*

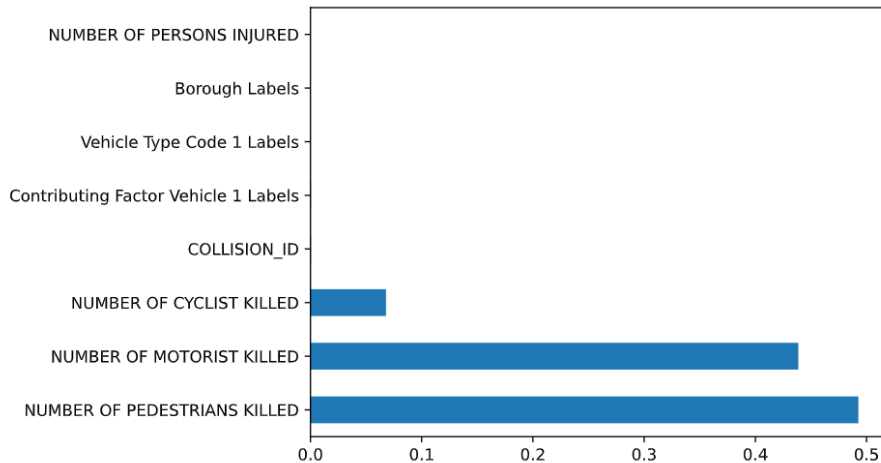


Figure 15. Feature Importance graph for Decision Tree Regressor Model

## Random Forest Regressor Model

To generate a more precise and reliable forecast, Random Forest creates many decision trees and blends them together. It has the benefit of being able to solve both classification and regression issues. The random forest is a classification system that uses numerous decision trees to make judgments. When creating each individual tree, it employs bagging and feature randomization to generate an uncorrelated forest of trees whose committee prediction is more accurate than that of any one tree.

### Random Forest Regressor Model

```
[60] #random forest regressor

randomforest_model = RandomForestRegressor(n_estimators = 500, max_depth=3)
randomforest_model.fit(X_train, y_train)

...
RandomForestRegressor
RandomForestRegressor(max_depth=3, n_estimators=500)

[61] #predict the result for the model

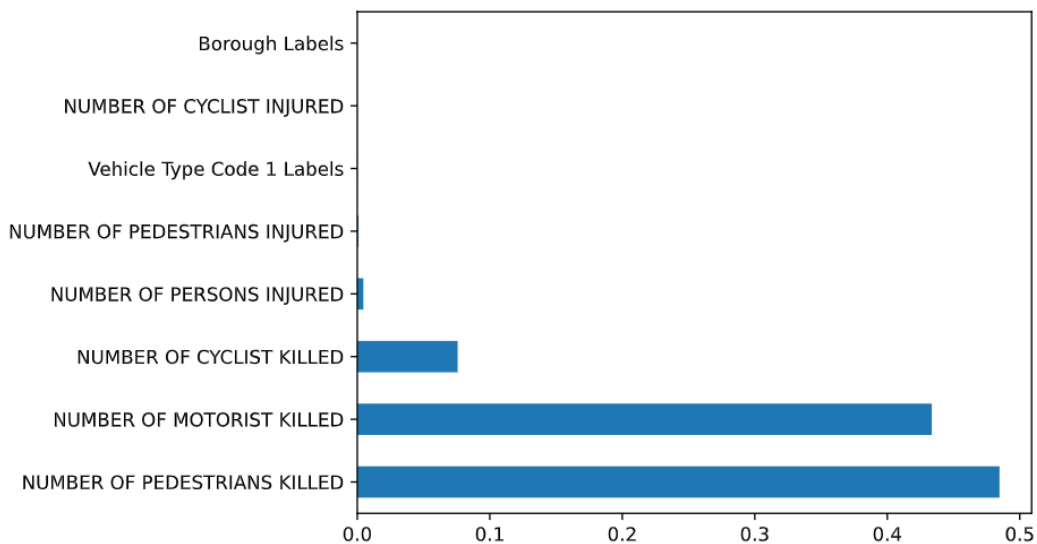
predicted_value_rf = randomforest_model.predict(X_test)
```

Figure 16. Random Forest Regressor Model



For the prediction of the number of persons killed, the features selected for the training of the model is same as that selected for the Linear Regression model and Decision Tree model. The data is split into **80-20 ratio** for training and testing of the model where the Random Forest Regressor is implemented with a minimum of **500 tress** and **maximum depth branch of 3**.

The accuracy of the model obtained is **97% for training data** and **97.7% for test dataset**. The feature importance and model evaluation for the random forest regressor model is as follows.



*Figure 17. Feature Importance graph for Random Forest Regressor Model*

```

Model Evaluation of Random Forest Regressor.
Mean Absolute Error: 0.0
Mean Squared Error: 0.0
Root Mean Squared Error: 0.0
R-Squared value: 0.9768055430330816

```

*Figure 18. Model Evaluation*

## Recommendations & Conclusion

The Motor Vehicle Collision Crash dataset gives an overview of the crashes that took place in New York, which helps in analyzing the factors due to which the accident took place and the number of people killed and injured due to the crash. One of the major causes of death in the US is car crashes. *The number of pedestrian and bicycle fatalities is increasing*, which is causing more individuals to die unexpectedly. The data and visualization shows that the statistics separated by years, with **2013** *having the greatest number of fatalities*, followed by **2021**. If we compare the number of wounded persons, 2018 and 2019 were the years with the largest numbers. The two of them share the fact that 2012 had the lowest number of fatalities and injuries. After digging a little further to see which Borough had the most fatalities, we discovered that while cyclists had the fewest there, drivers on Staten Island had the most fatalities. **Brooklyn** is the most populated Borough in New York and has the *most fatal pedestrian and bike accidents*.

The map analysis and visualization helps understand the total number of persons killed and injured in each area which is based on the *zip code* in the city. This gives the area wise analysis of the crashes such that the area having the highest accidents can be given attention and necessary precautions and safety measures can be implemented. Also, based on the day wise or hour wise analysis of the crashes that have taken place, it is observed that **majorly on weekends** the crash rate increases, which could possibly be due to the week off, and thus people spend time on outings which could lead to traffic and has a high possibility for crashes to happen. Hence, more care should be taken on weekends by implementing strict rules for safety of the people.

Another feature or parameter that helps in analysis and recommendations is the **contributing factor parameter** of the crash. This variable helps understand the reason of the crash and with the help of the visual representations it is observed that the maximum number of persons killed and injured are due to **unspecified reasons**, and thus it is recommended that the reason of the crash needs to be identified in order to help understand the reason of the crash which can help take necessary actions on the same and reduce the crash and accident rate, ensuring safety of the people.

The evaluation of the regressor models are based on the **MAE, MSE, RMSE, and R-squared** values. The accuracy obtained for the models are **98.3% for Linear Regressor model, 98.1% for Decision Tree and 97.7% for Random Forest Regressor model**. Hence, Linear Regressor model can be considered as the best fit model for the prediction of the number of persons killed. The table below compares the evaluation of the various regressor models implemented in order to determine the best fit model for the prediction of the number of persons killed.

*Table 1. Regressor Model Comparison*

	<b>Machine Learning Regressor Models</b>		
<b>Evaluation Metric</b>	Linear Regressor	Decision Tree Regressor	Random Forest Regressor
Accuracy	98.26%	98.1%	97.7%
Mean Absolute Error	0.0	0.0	0.0
Mean Squared Error	0.0	0.0	0.0
Root Mean Squared Error	0.0	0.0	0.0
R-Squared value	0.9826	0.9808	0.9768

Thus, based on the model evaluation and comparison, it is observed that the accuracy for Linear Regressor model is more as compared to other two models. The MAE and MSE values represent the difference between the actual and predicted values extracted by the mean error over the dataset. The RMSE is the error rate where the **R-Squared** represents how well the value fits compared to the original values, and the higher the value of R-Squared the better the model is. Thus, based on the evaluation metrics, it is observed that the **root mean squared error** is less in all of the three model which implies that the prediction rate error is less. Hence, based on the accuracy of the model and the r-squared value, Linear Regressor Model is recommended to use for the prediction of the number of persons killed based on the contributing factors.

## **Tools & Packages**

The various tools and packages used for the Motor Vehicle Collision Crash Analysis are as follows.

### **Tools:**

1. Microsoft Excel
2. Python Programming Language
3. Visual Studio
4. Jupyter Notebook
5. Tableau – Data Visualization & Dashboard
6. Microsoft PowerPoint

### **Packages:**

1. Pandas
2. NumPy
3. Matplotlib
4. Seaborn
5. Featurewiz
6. Sklearn
  - a. Label Encoder
  - b. Train\_Test\_Split
  - c. Accuracy\_score
  - d. Confusion\_matrix
  - e. Metrics
  - f. Linear Regressor
  - g. Decision Tree Regressor
  - h. Random Forest Regressor
  - i. Mean\_Squared\_Error
  - j. Mean\_Absolute\_Error

## References

Visualizing distributions of data — seaborn 0.11.2 documentation. (2021). Seaborn.

<https://seaborn.pydata.org/tutorial/distributions.html>

sklearn.linear\_model.LinearRegression. (2022). Scikit-Learn.

[https://scikit-learn.org/stable/modules/generated/sklearn.linear\\_model.LinearRegression.html](https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html)

Decision Tree Regression. (2022). Scikit-Learn.

[https://scikit-learn.org/stable/auto\\_examples/tree/plot\\_tree\\_regression.html](https://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression.html)

sklearn.ensemble.RandomForestRegressor. (2022). Scikit-Learn.

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>

D. (2019b, October 10). Regression Accuracy Check in Python (MAE, MSE, RMSE, R-Squared). Data Tech Notes. <https://www.datatechnotes.com/2019/10/accuracy-check-in-python-mae-mse-rmse-r.html>