Applied Data Science Coursera Project Report

Introduction/Business Problem

Accidents in the US and around the world are a significant problem that increases the fatality rate of a country and also leads to economic loss. With the improvement in machine learning over recent years, it is essential that we study the effects of various attributes on roads and can effectively model them allowing us to predict accident severity. The input to such a model would be the characteristics of previous accidents such as driver behavior, road conditions, road type, environmental conditions, and the output vector would be the class of accident severity. By building such a model, we would enable the drivers to have information about the seriousness of getting into a road accident if they kept driving on the road that they are on. This extra piece of information could make sure that the driver changes their style of driving on a particular route or change their course to avoid accidents. Therefore, it is essential to invest in building a road accident severity prediction model.

For the most part, the distributions of these attributes are more or less equally divided; therefore, these can act as good attributes for applying to our machine learning models.

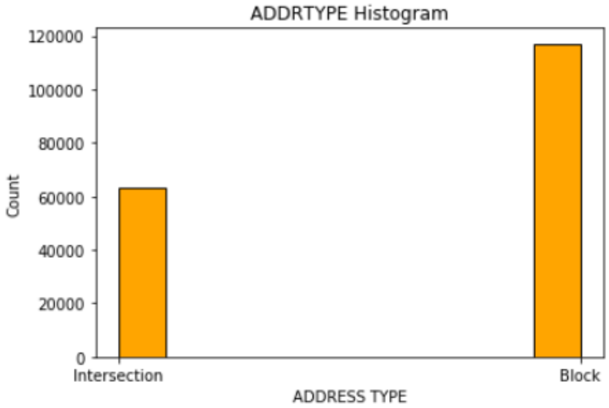
Data

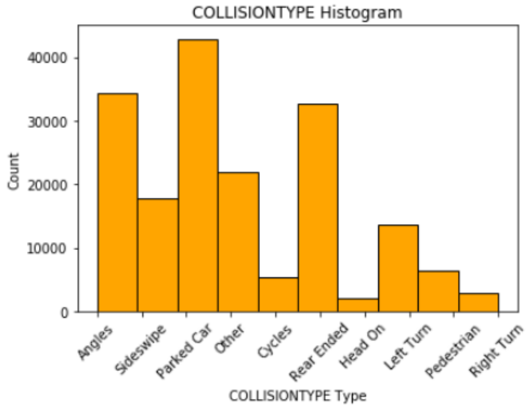
The data set that I will be using for this project is obtained from coursera.org for Seattle city. This dataset has 194,673 rows and 37 attributes about the road. Attributes such as Location, Road condition, weather condition, junction junction, car speeding, number of people involved, light conditions, number of vehicles involved in are some of the attributes that can be used to create the model. The label of the data is present in the column Severity. There are unbalanced labels present in the dataset which will need to be balanced in order to reduce the bias in the model.

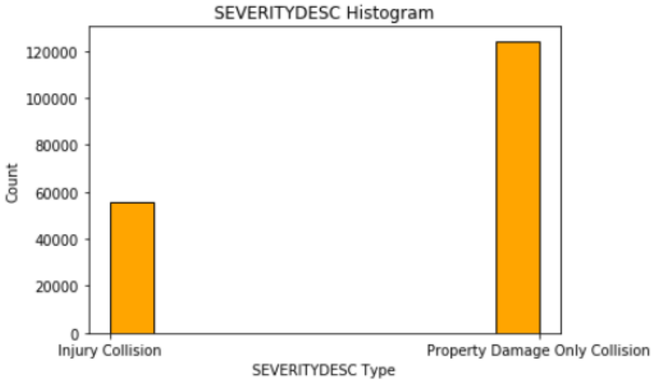
Visualization

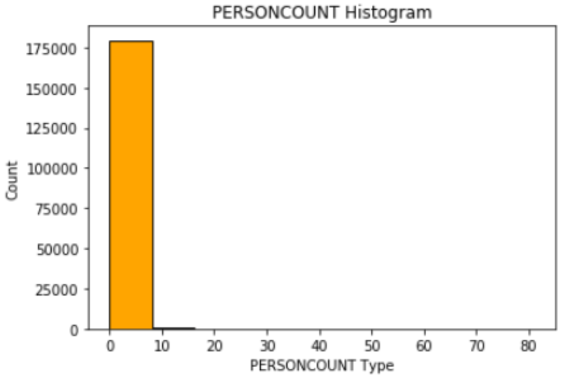
In this section, we will be visualizing all of the attributes by plotting their histograms, and at the end of this visualization section, we will display 10,000 location points of the accidents on the map of Seattle in order to better understand the distribution of these accidents around the city.

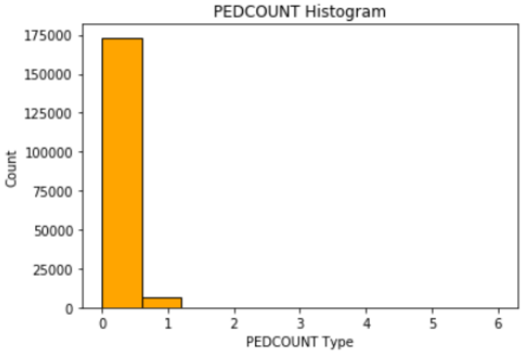
Histograms of the various attributes are as follows:

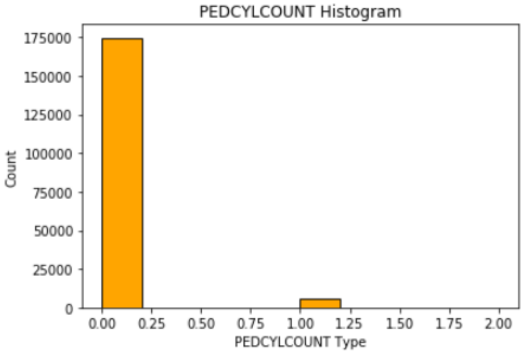


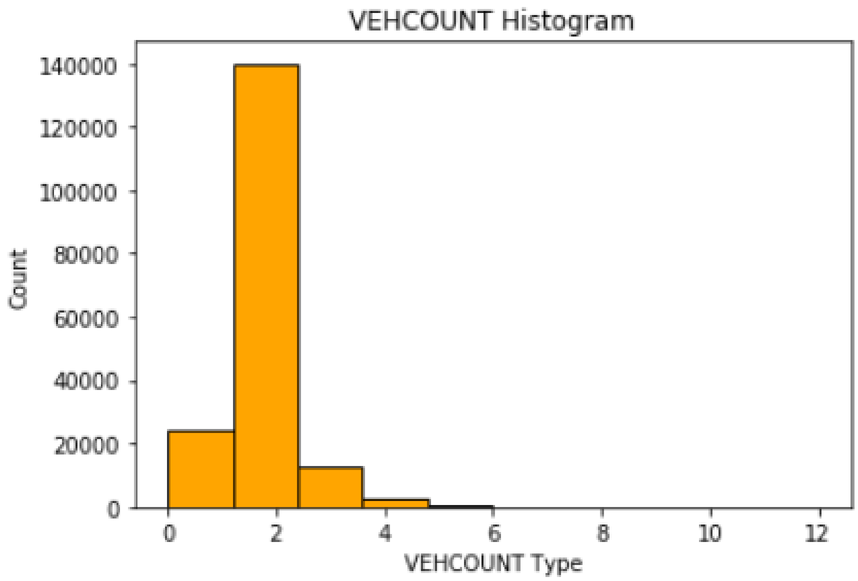




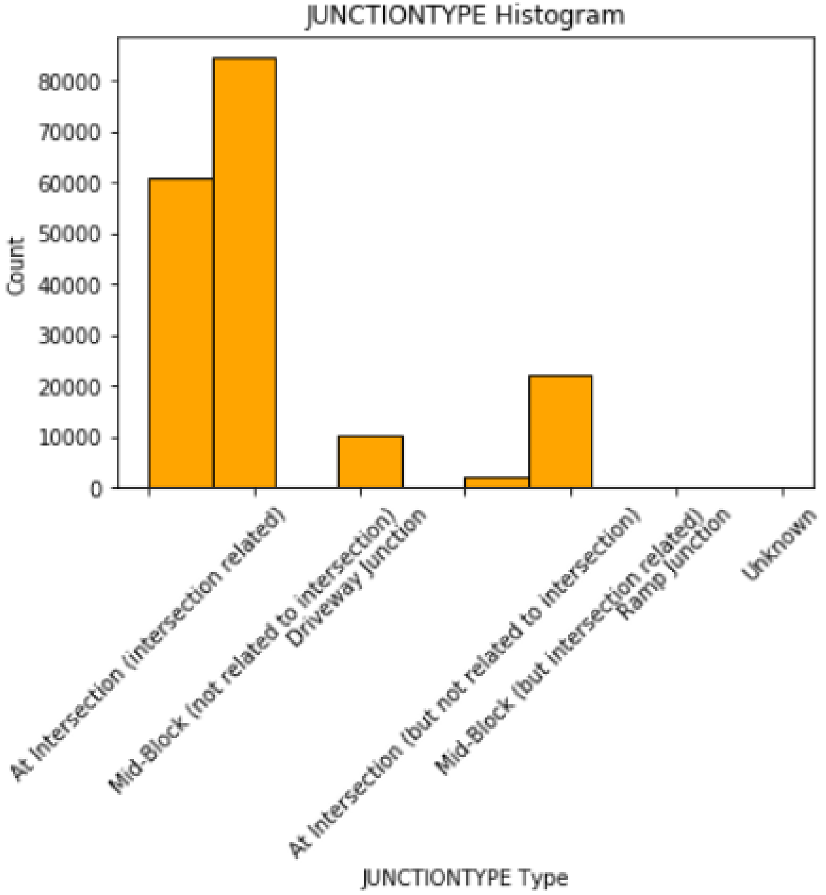




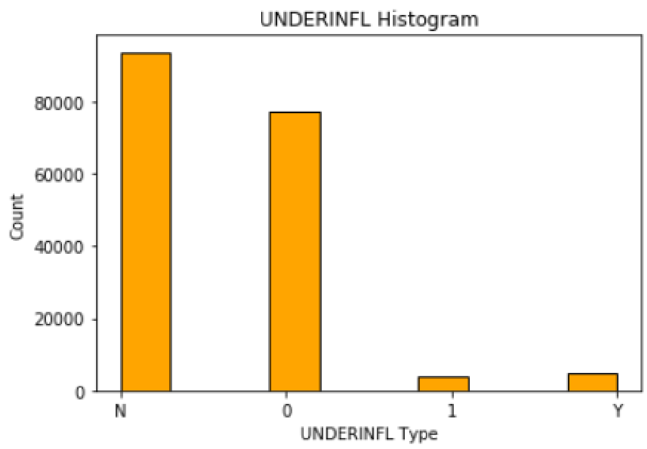


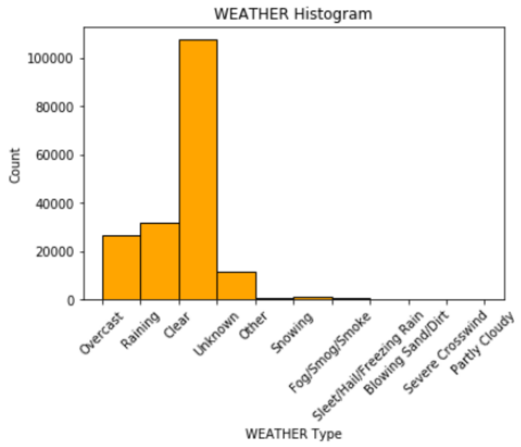


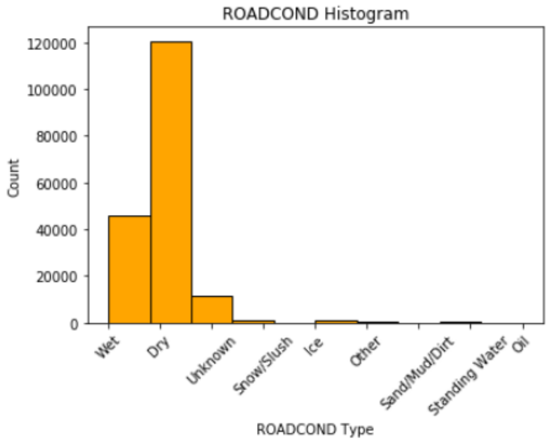
VEHCOUNT Histogram
Count
140000
120000
100000
80000
60000
40000
20000
0
0
2
4
6
8
10
12
VEHCOUNT Type

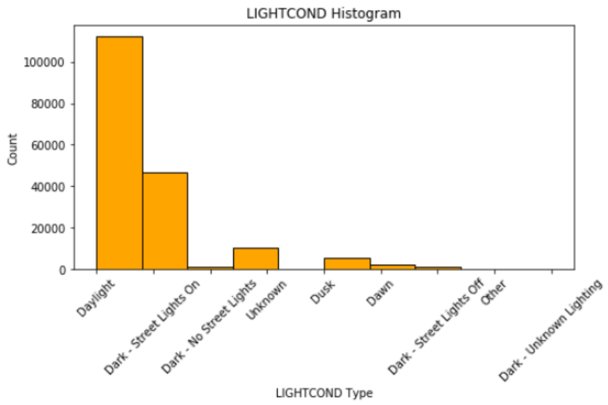


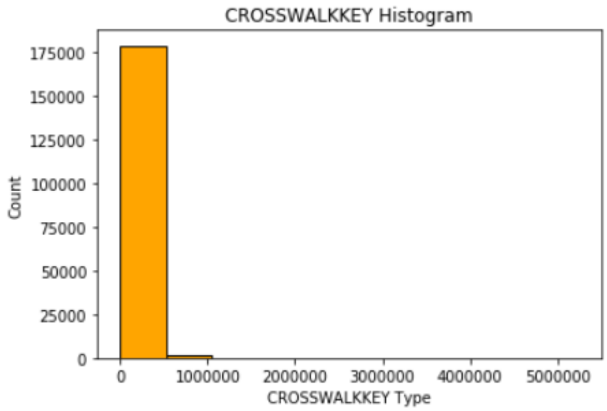
JUNCTIONTYPE Histogram
Count
80000
70000
60000
50000
40000
30000
20000
10000
0
At Intersection (intersection related)
Mid-Block (not related to intersection)
Driveway Junction
At Intersection (but not related to intersection)
Mid-Block (but intersection related)
Ramp Junction
Unknown
JUNCTIONTYPE Type

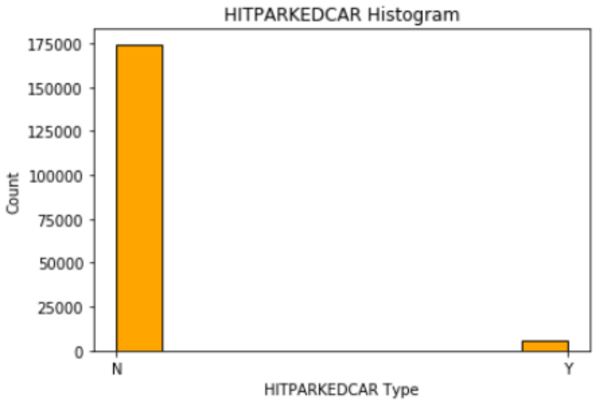




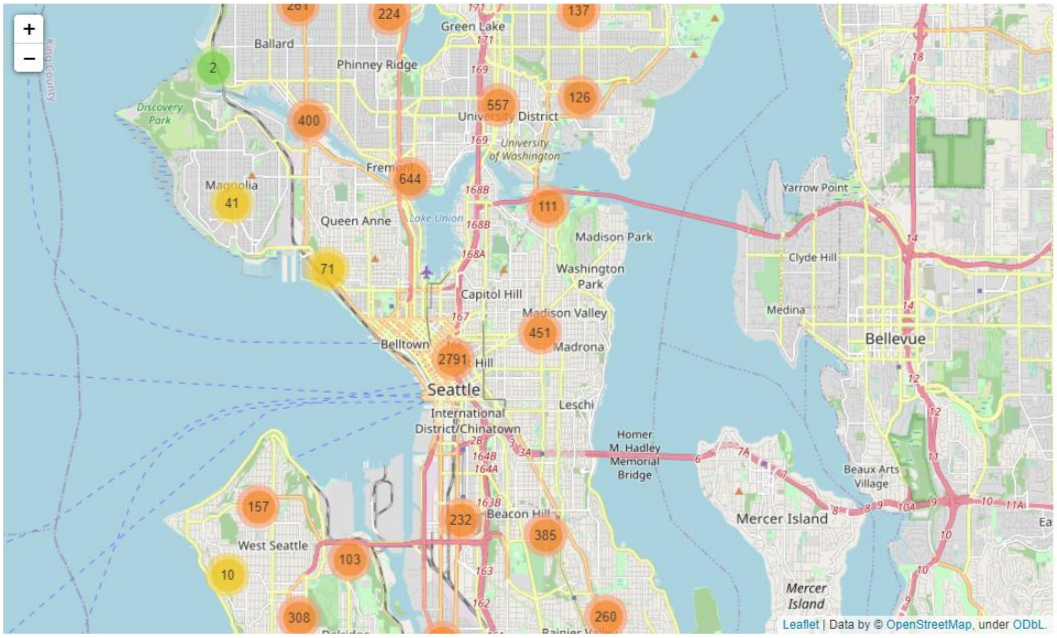








Next, we plot the different locations of the accidents in the form of a cluster on the map of Seattle



The cluster map of the data points shows us that the most number of accidents occur near the **International District/Chinatown** area and close to the **Georgetown** area where the **airport** resides; this makes sense since the most traffic in a city is expected to be around these areas.

Methodology

In order to store the dataset, I used my local machine. The master data consists of all the attributes present in the dataset, and we aim to use the 20 attributes that we have assembled in our data frame, as discussed above.

We use the **folium** library in order to visualize the geographic details of the accidents around the city of Seattle. I created a map of the accidents around the city of Seattle to understand the areas most affected by accidents.

We use **pandas** and **matplotlib** libraries to analyze our dataset and see for cases of:

- *Missing values*
- *NA values*
- *Types of values present in each dataset*
- *Transformation of these attributes into our desired format for the machine learning process*

We were able to find that there is a class imbalance present in our dataset, which should hopefully not affect our models severely. So we decide to proceed further.

In this project, I plan to use four models for the dataset they are:

- *K Nearest Neighbors*
- *Decision Tree*
- *Logistic Regression*
- *Support Vector Machine*

The results obtained from running these models on the dataset will be summarized at the end of this notebook.

Analysis

The Analysis part is the section of this notebook where we perform **transformations** on our dataset and enable it for proper use with our **machine learning models**. We will go through the following steps during the Analysis phase:

- Data Preprocessing

  In this section, we focus on the attributes that have the type object present in them and try to convert them to a numeric format by using **Label Encoder** present in the **scikit-learn package.**

- Data Normalization

  Since the range of values present in our dataset is high, we need to normalize this data before it can be used in machine learning models. We use **Standard Scalar** from the **scikit-learn** library in order to normalize our dataset.

- Train and Testing Data Split

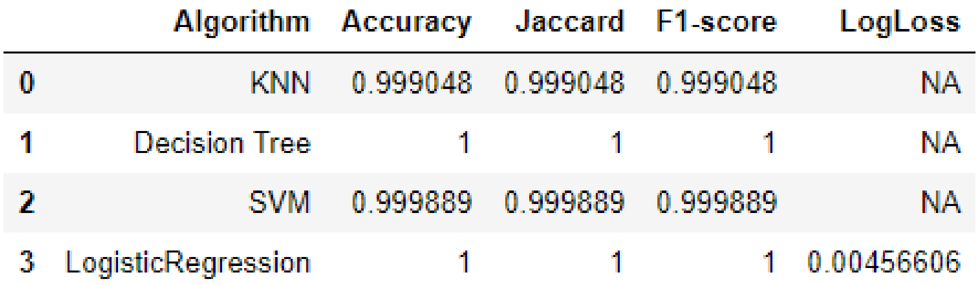
  This is the last part of the analysis in this part, we **split our dataset into training or testing dataset**, and we divide the dataset such that we have **65%** of the dataset for **training** and we use the rest **35%** for **testing.**

Results and Conclusion

After successfully transforming our dataset in the above sections, we can now finally use it in our machine learning models, as discussed above, and we will be using four algorithms for our analysis.

For discussing the results and conclusion of this project, we will go through 4 main metrics calculated over the models these metrics are as follows:

- *Accuracy*
- *Jaccard Similarity Score*
- *F1 Score*
- *Log_loss (Only for Logistic Regression)



	Algorithm	Accuracy	Jaccard	F1-score	LogLoss
0	KNN	0.999048	0.999048	0.999048	NA
1	Decision Tree	1	1	1	NA
2	SVM	0.999889	0.999889	0.999889	NA
3	LogisticRegression	1	1	1	0.00456606

From the above-given evaluation table it is easy to see that although all of our algorithms performed substantially well on the dataset, the two best algorithms for predicting accident severity are **Decision Tree classifier** and **Logistic Regression** model