Car Accident Severity Prediction

Introduction

As per a report from USA's CDC, 1.35 million people die each year due to road accidents, apart from the damage to property. If we can understand why road accidents happen, if we can identify the important factors behind accidents, we can create a curative plan to reduce the number of road accidents. This report aims to identify the factors behind accidents. This analysis can be used in a plethora of industries and scenarios. For e.g. Home deliveries have been increasing in the last 10 years, and has recently shot up due to COVID-19. All the delivery agent traversing the last mile (from Pick up point to a customer's address) can benefit from this analysis. If our model states that a given stretch of road has a high risk of accident, routing algorithms (like Google maps) can direct the riders towards a different road.

This report will identify the important factors behind road accidents and predict the severity of road accidents in Seattle Area. This will be useful for police departments as they can identify the accident hotspots for better manpower allocation. This can also be utilized by last mile delivery agents.

Data

I am using the data collected by Seattle City's Police Department from 2004 to Feb 2020. It has around 194k+ observations regarding the reported road accidents during the past 15+ years. Each observation can have 38 attributes. Some of the attributes are SEVERITYCODE, LOCATION, STATUS, VEHCOUNT, etc.

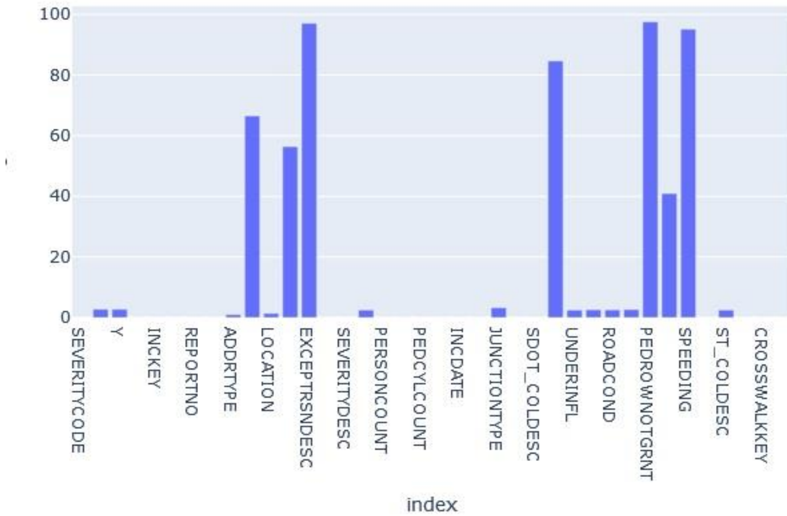
For e.g. SEVERITYCODE corresponds to the severity of the collision. '3' represents a fatal accident, whereas '1' represents property damage. Similarly, ADDRTYPE lists out the type of address (Alley, lock or intersection) where the accident has occurred. A detailed description of attributes can be found at link.

Most of the attributes are categorical in nature and will have to be encoded for analysis. For e.g. JUNCTIONTYPE explains the category of junctions where the accident has taken place. It has the following distinct values - At Intersection (intersection related)'; 'Mid-Block (not related to intersection)'; 'Driveway Junction'; 'Mid-Block (but intersection related)'; 'At Intersection (but not related to intersection)'; NAN; 'Unknown' and 'Ramp Junction'. Post encoding it was changed to values 0, 1, 2, 3, 4, 5 and 6.

I used the 'ADDRTYPE', 'COLLISIONTYPE', 'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'SDOT_COLCODE', 'UNDERINFL', 'ROADCOND', 'LIGHTCOND' and 'HITPARKEDCAR' columns as independent variables to build the model. 'SEVERITYCODE' is the dependent variable for the model.

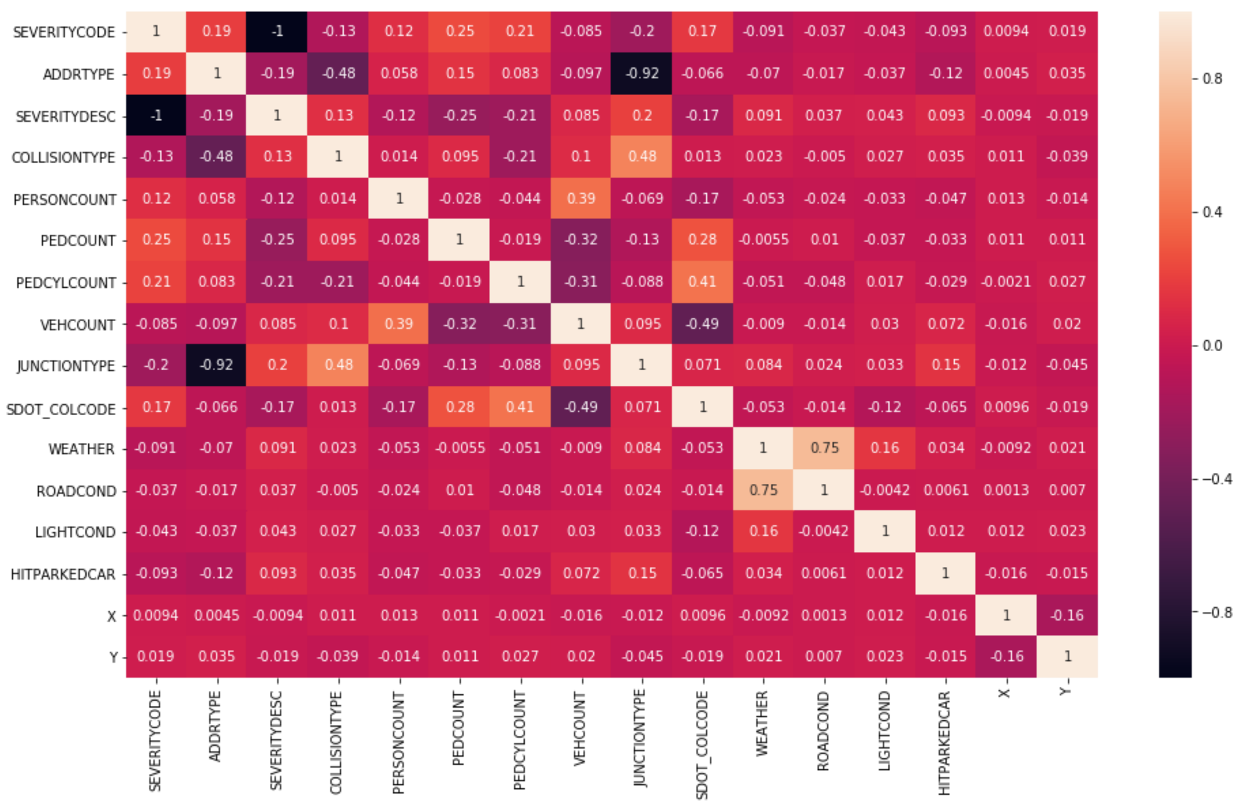
Methodology

I used Anaconda's Python Notebook for the data cleaning, data analysis, model training and testing. I imported multiple libraries for my use, such as numpy, pandas, plotly, seaborn, etc. for analysis and visualization.
I imported the data set using panda's read function, and then used the describe function to get a basic understanding. Upon review, I found out that there are lots of missing values in different attributes. I then plotted a graph to find the percentage of missing values in each attribute.



I dropped the following columns from the data frame due to large no missing values - 'SEVERITYCODE', 'ADDRTYPE', 'SEVERITYDESC', 'COLLISIONTYPE', 'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'JUNCTIONTYPE', 'SDOT_COLCODE' ,'UNDERINFL', 'WEATHER', 'ROADCOND', 'LIGHTCOND', 'HITPARKEDCAR'. I also dropped the columns 'X' and 'Y', which represent longitude and latitude, as they are not relevant to our model.

I then converted the categorical values to Boolean values using the 'LabelEncoder' function for correlation analysis. I then created a heat map of correlation values amongst the different attributes.



From the heat map, I was able to identify attributes with a high correlation, and hence removed them from the data set. I removed the following columns - 'SEVERITYCODE', 'ADDRTYPE', 'COLLISIONTYPE', 'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'SDOT_COLCODE', 'UNDERINFL', 'WEATHER', 'ROADCOND', 'LIGHTCOND', 'HITPARKEDCAR'.

I also removed the observations which had outlier values for the PERSONCOUNT attribute, using a box plot.

Results

Training and Testing ML Model

I used a Logistic Regression machine learning model for our solution. I divided the existing dataset into training and testing datasets.

Post the training, the model had F1 score for Severity 1 is 0.838, whereas for Severity 2, it is only 0.381. It is happening due to the class imbalance in the observations.

Class Imbalance Correction

I corrected the class imbalance problem by having equal entries for both Severity 1 and 2.

Final Result

I tried 4 different models to predict the severity of accidents in Seattle Area.

1. Logistic Regression – After class imbalance correction, the model had F1 score for Severity 1 of 0.68 and 0.64 for Severity 2.

2. Logistic Regression – It used the ‘balanced’ criteria for class_weight attribute. It automatically balances the observations for different categories. The model had F1 score for Severity 1 of 0.68 and 0.64 for Severity 2, same as earlier one.

3. Random Forest - The model had F1 score for Severity 1 of 0.68 and 0.73 for Severity 2.

4. KNN - The model had F1 score for Severity 1 of 0.69 and 0.54 for Severity 2.

I will use the Random Forrest model (model #3) to predict the road accident severity, as it has a better F1 score. The logistic regression model should also work.

Discussion

Initially, I was trying to identify the relevant factors (independent variables) which will impact the severity of road accidents. The major factors can be grouped into weather conditions, road conditions and location of the accident.

We started off with 194k+ observations and 38 attributes. But we had to drop multiple attributes due to missing data, which might have lowered the accuracy of our models. The data set also lacked the observations for Severity 3 (fatal accidents). We need to automate the data collection for accidents, so that we will have richer data to build models on. A good training data is an important ingredient for a good prediction model. Automation of data collection can be easily scaled up nowadays by collecting data from car sensors.

This model can be also be used for live applications such as map routing for general users. For e.g. Point A to B has 4 different viable paths. Maps (Google, Apple, etc.) can also show the probability of accident on the 4 different routes along with the estimated travel time.

Conclusion

This analysis has identified factors due to which road accidents happen. But, by no means is this an exhaustive set.

Seattle Police Department can utilize this model to place officers at hot spots, which have a higher chance of accident. Drivers can also utilize this model take counter-active measures to avoid accidents.