Predicting Traffic Accident Severity

Amirtha Rajan PKS

October 10, 2020

Table of Contents

[1. INTRODUCTION 2](#_Toc53483984)

[1.1 Background 2](#_Toc53483985)

[1.2 Business Problem 2](#_Toc53483986)

[1.3 Interest 2](#_Toc53483987)

[2. Data Acquisition and Cleaning 3](#_Toc53483988)

[2.1 Data Source 3](#_Toc53483989)

[2.2 Feature Selection 3](#_Toc53483990)

[2.3 Data Cleaning 5](#_Toc53483991)

[3. Exploratory Data Analysis 6](#_Toc53483992)

[3.1 Overview of Accidents 6](#_Toc53483993)

[3.2 Accident Histogram 8](#_Toc53483994)

[3.3 Relationship with Weather Condition 11](#_Toc53483995)

[3.4 Relationship with POIs 12](#_Toc53483996)

[4. Predictive Model 13](#_Toc53483997)

[4.1 Logistics Regression 14](#_Toc53483998)

[4.2 K-Nearest Neighbour 15](#_Toc53483999)

[4.3 Decision Tree 15](#_Toc53484000)

[4.4 Random Forest 17](#_Toc53484001)

[4.5 Results 18](#_Toc53484002)

[5 Conclusion 19](#_Toc53484003)

[6 Future Directions 19](#_Toc53484004)

# INTRODUCTION

## 1.1 Background

The first automobile crash in the United States occurred in New York City in 1896, when a motor vehicle collided with a pedal cycle rider. Later every year the accident count has been increasing but the number of death and number of drivers has never been lower. Now more than 38,000 people die every year in crashes on U.S. roadways. The U.S. traffic fatality rate is 12.4 deaths per 100,000 inhabitants. An additional 4.4 million are injured seriously enough to require medical attention.

The development has led to safer roads, quick emergency response, imposed traffic rules and of course faster cars nevertheless the rules of the road every drivers obeys are the laws of physics. Still nothing in the course of human revolution has prepared us for the forces involved during a reckless frontal car crash which happens over a tenth of a second and the reason for crash are limitless but some poses are more prevalent than others. Most occurred are the speeding and the cases including alcohol.

## 1.2 Business Problem

There are ways in which this can be significantly reduced by prediction measures on analysing various range of factors, including weather conditions, special events, roadworks, traffic jams and bumpy and unsafety roads. Thus we can analyse on the patterns and events of occurrence from history to predict the severity of accidents that could occur at duration of time at similar locations. This could be the key to provide well informed accurate prediction for the individual who travel though that road and provide better emergency response, which could reduce significant amount of accidents per year.

## 1.3 Interest

The importance and actions plans for this model will be visible to government to take all the measures to develop safer roads. Government can also provide human resources for rescue in critical roads which could save significate amount of people each year. Adding up to this Interested Companies that are aiming for road safety can invest in technology on prediction models on predicting severities for an accident that could occur and could also focus on preventing accident. This analysis and accuracy measure will also provide the drivers/people of critical areas well cautioned and plan their route accordingly

# Data Acquisition and Cleaning

## 2.1 Data Source

US Accident data can be found in Kaggle dataset [here](https://www.kaggle.com/sobhanmoosavi/us-accidents). The dataset however, lack data for certain months. For example, for the year 2016 the records for the month January is missing. Some information of the state name and county details were not available in the data set has I had to scrap it from the web for geo analysis.

Kaggle’ s Accident dataset were based on three sources Bing, MapQuest and MapQuest-Bing. MapQuest had larger dataset followed by Bing and MapQest-Bing with lowest dataset. However, each data source had categorized severity based on different mode of accidents.

## 2.2 Feature Selection

The Dataset has 49 features which is describes in the following table:

| **#** | **Attribute** | **Description** |
| --- | --- | --- |
| 1 | ID | This is a unique identifier of the accident record. |
| 2 | Source | Indicates source of the accident report (i.e. the API which reported the accident). |
| 3 | TMC | A traffic accident may have a Traffic Message Channel (TMC) code which provides more detailed description of the event. |
| 4 | Severity | Shows the severity of the accident, a number between 1 and 4, where 1 indicates the least impact on traffic (i.e., short delay as a result of the accident) and 4 indicates a significant impact on traffic (i.e., long delay). |
| 5 | Start\_Time | Shows start time of the accident in local time zone. |
| 6 | End\_Time | Shows end time of the accident in local time zone. End time here refers to when the impact of accident on traffic flow was dismissed. |
| 7 | Start\_Lat | Shows latitude in GPS coordinate of the start point. |
| 8 | Start\_Lng | Shows longitude in GPS coordinate of the start point. |
| 9 | End\_Lat | Shows latitude in GPS coordinate of the end point. |
| 10 | End\_Lng | Shows longitude in GPS coordinate of the end point. |
| 11 | Distance(mi) | The length of the road extent affected by the accident. |
| 12 | Description | Shows natural language description of the accident. |
| 13 | Number | Shows the street number in address field. |
| 14 | Street | Shows the street name in address field. |
| 15 | Side | Shows the relative side of the street (Right/Left) in address field. |
| 16 | City | Shows the city in address field. |
| 17 | County | Shows the county in address field. |
| 18 | State | Shows the state in address field. |
| 19 | Zipcode | Shows the zipcode in address field. |
| 20 | Country | Shows the country in address field. |
| 21 | Timezone | Shows timezone based on the location of the accident (eastern, central, etc.). |
| 22 | Airport\_Code | Denotes an airport-based weather station which is the closest one to location of the accident. |
| 23 | Weather\_Timestamp | Shows the time-stamp of weather observation record (in local time). |
| 24 | Temperature(F) | Shows the temperature (in Fahrenheit). |
| 25 | Wind\_Chill(F) | Shows the wind chill (in Fahrenheit). |
| 26 | Humidity(%) | Shows the humidity (in percentage). |
| 27 | Pressure(in) | Shows the air pressure (in inches). |
| 28 | Visibility(mi) | Shows visibility (in miles). |
| 29 | Wind\_Direction | Shows wind direction. |
| 30 | Wind\_Speed(mph) | Shows wind speed (in miles per hour). |
| 31 | Precipitation(in) | Shows precipitation amount in inches, if there is any. |
| 32 | Weather\_Condition | Shows the weather condition (rain, snow, thunderstorm, fog, etc.) |
| 33 | Amenity | A POI annotation which indicates presence of amenity in a nearby location. |
| 34 | Bump | A POI annotation which indicates presence of speed bump or hump in a nearby location. |
| 35 | Crossing | A POI annotation which indicates presence of crossing in a nearby location. |
| 36 | Give\_Way | A POI annotation which indicates presence of give\_way in a nearby location. |
| 37 | Junction | A POI annotation which indicates presence of junction in a nearby location. |
| 38 | No\_Exit | A POI annotation which indicates presence of no\_exit in a nearby location. |
| 39 | Railway | A POI annotation which indicates presence of railway in a nearby location. |
| 40 | Roundabout | A POI annotation which indicates presence of roundabout in a nearby location. |
| 41 | Station | A POI annotation which indicates presence of station in a nearby location. |
| 42 | Stop | A POI annotation which indicates presence of stop in a nearby location. |
| 43 | Traffic\_Calming | A POI annotation which indicates presence of traffic\_calming in a nearby location. |
| 44 | Traffic\_Signal | A POI annotation which indicates presence of traffic\_signal in a nearby location. |
| 45 | Turning\_Loop | A POI annotation which indicates presence of turning\_loop in a nearby location. |
| 46 | Sunrise\_Sunset | Shows the period of day (i.e. day or night) based on sunrise/sunset. |
| 47 | Civil\_Twilight | Shows the period of day (i.e. day or night) based on civil twilight. |
| 48 | Nautical\_Twilight | Shows the period of day (i.e. day or night) based on nautical twilight. |
| 49 | Astronomical\_Twilight | Shows the period of day (i.e. day or night) based on astronomical twilight. |

Upon examining the meaning of each feature from the description provided by Kaggle source I was able to differentiate the features into 5 categories Traffic attributes, Address attributes, Weather attributes, POI attributes and Daytime attributes. These characteristics determine the time, city, weather condition, accident type, light condition and poi of the collision.

Upon looking into the feature data some had similar information, for example on comparing Weather Timestamp and Start Time it was just alike and seemed like a duplicate column, so to avoid keeping redundant data I had to remove one of these features. On further analysis few feature had no value for collision prediction analysis, some features were ID, TMC and Source so I had to drop these features as well.

Other feature such as Description, Street Number, Country was in the part invaluable dataset as these had no value to the dataset, because country had only single value as we are talking about US and description had details about the incident which we can’t be using for prediction but thinking of it could be made useful in geographic visualization as a hover data.

Weather Condition is the key feature in our dataset that had the type of weather condition at the time of collision this feature data was converted to categorical feature columns. Similarly, some Address attribute features were useful in predicting the frequency of occurrence at location hence I converted these feature columns by frequency encoding.

There were few other features providing similar information which include End Time, End Latitude and End Longitude so being almost a redundant feature these features were also dropped.

## 2.3 Data Cleaning

As the Dataset were scraped from multiple sources which includes Bing, MapQuest and MapQuest-Bing and combined into one table, there were a lot of missing values. There were also many other problems in the dataset.

The First thing is to handle the feature which has more than 60% of its data missing. Naming these would be End Latitude, End Longitude and Street Number so I dropped it from the dataset as it would be hard to estimate the right values to fill-in. There were other features which had more than 50% of its data missing/nan which were Precipitation and Wind Chill, among these two features I preferred to keep Precipitation as it could be a feature with valuable correlation and so I just dropped Wind Chill.

The Next part was I took categorical features in hand and analysed on separating these features into categorical columns namely Weather Condition and Address attributes. Talking about Weather Condition, it wasn’t easy to split it to categorical features as the data in weather condition was a bit messy so I had to use regex to pick out specific keywords and placed the feature to the category that it would fit in. And next was about the Address attributes which was split based on frequency distribution into features columns.

The second part was fill the missing nan values with algorithms. The Precipitation was recovered by using log-normal distribution function using the mean and standard deviation of the grouped weather type but since we had outliers I choose to go with median and standard deviation and filled the precipitation records that had weather condition.

Then upon looking into the list of missing nan features the weather attributes were the ones I decided to handle which include Temperature, Humidity, Pressure, Visibility and Wind Speed. Since these Features had less than 2% of its values missing it was easy to just fill in with median with some logical conditions by grouping data based on location and time.

The left out feature is Wind Direction with 1.6% of nan values. Since Wind Direction is a categorical feature I used counter to get the most common values for a grouped dataset based on location and time with three iterations.

The remaining dataset had few features with less than 1% of nan which was not possible recoverable so I had to drop these records.

# Exploratory Data Analysis

## 3.1 Overview of Accidents

First, to understand on Accidents in US, data was plotted on with different visualization on to the Map to have an overview of accidents happening across the Nation.

**3.1.1 HeatMap Visualization:**

I used HeatMap in folium to plot the latitudes and longitudes to view the Accident density on the US locations.

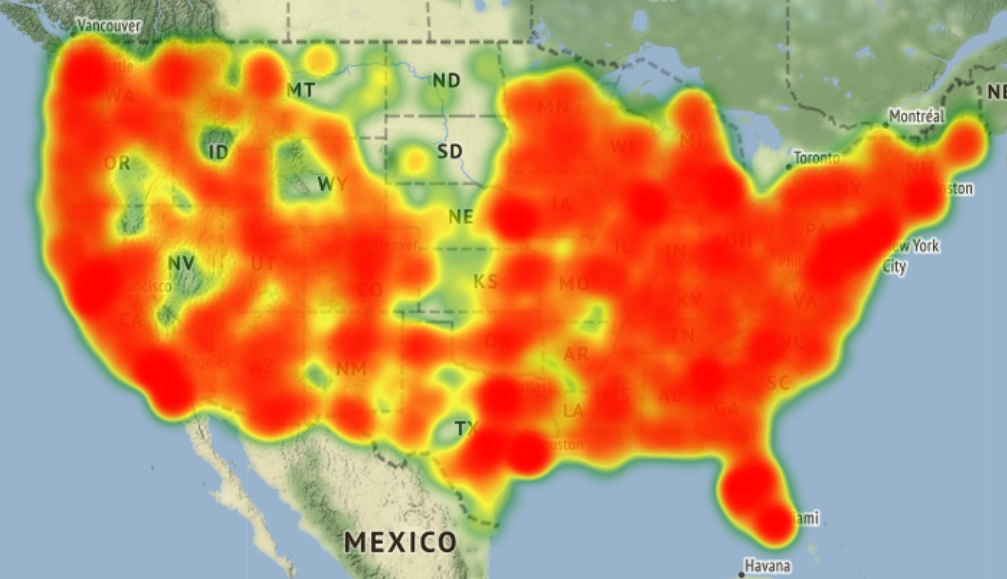


Figure . Accident Density Distribution HeatMap year 2016

**3.1.2 ChrolopethMap Visualization:**

The second Map plot was chrolopeth based on States but state name was required to plot but were missing in our dataset but fortunately we had the state code so it I had to scrap for US state data and map the state code with state name from my acquired dataset. Now the data had to be grouped based on states and its frequency to plot on the ChrolopethMap.

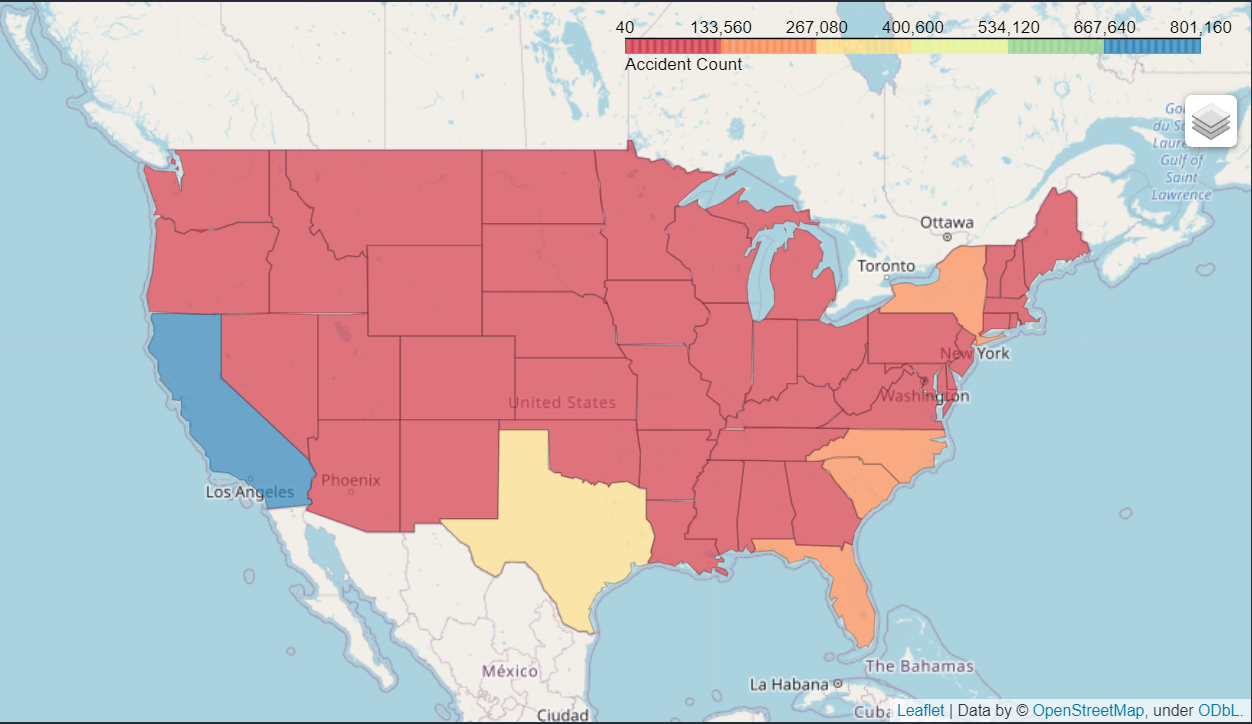


Figure Frequency Distribution of US-Accidents by State (2016-2020)

Final Map I used was to visualize the distribution of severities across counties. But some data had to be downloaded for mapping county names with its geojson data. The final output of the distribution is visualized as below.

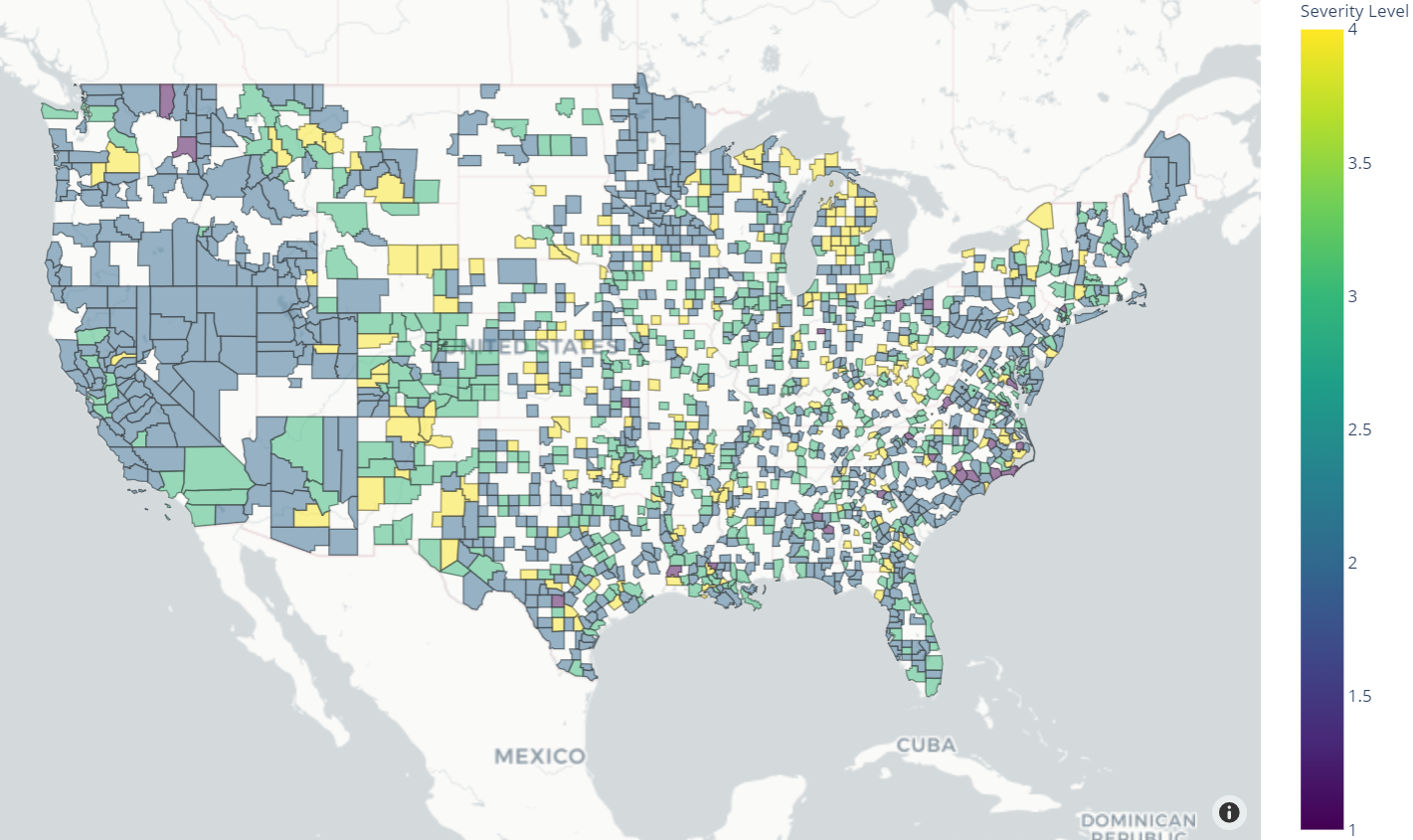


Figure Frequency Distribution of US-Accidents by County (2016-2020)

## 3.2 Accident Histogram

Even though we have seen better construction of roads over years, number of accidents has always been increasing year over year as we develop fast moving cars. The history shows us the truth that it has never been controlled in Time.

Figure 4 shows the accident counts in Time for the year 2016 to 2020 but there are missing data for the month on Jan in 2016 as data was gathered from Feb 2016 on wards for this dataset from Kaggle.

Information for the year 2016 has been increasing gradually as the source on the information gathered was increasing lately to get counts across the nation by year end.

The Year 2018 looks stabilized with accident count for the months, however in 2017 they had it in control till July end and later it peeked with 30K counts on the month of August. This was because millions of people were travelling across to safer place on an event of weather forecast warning for an upcoming Hurricane Harvey.

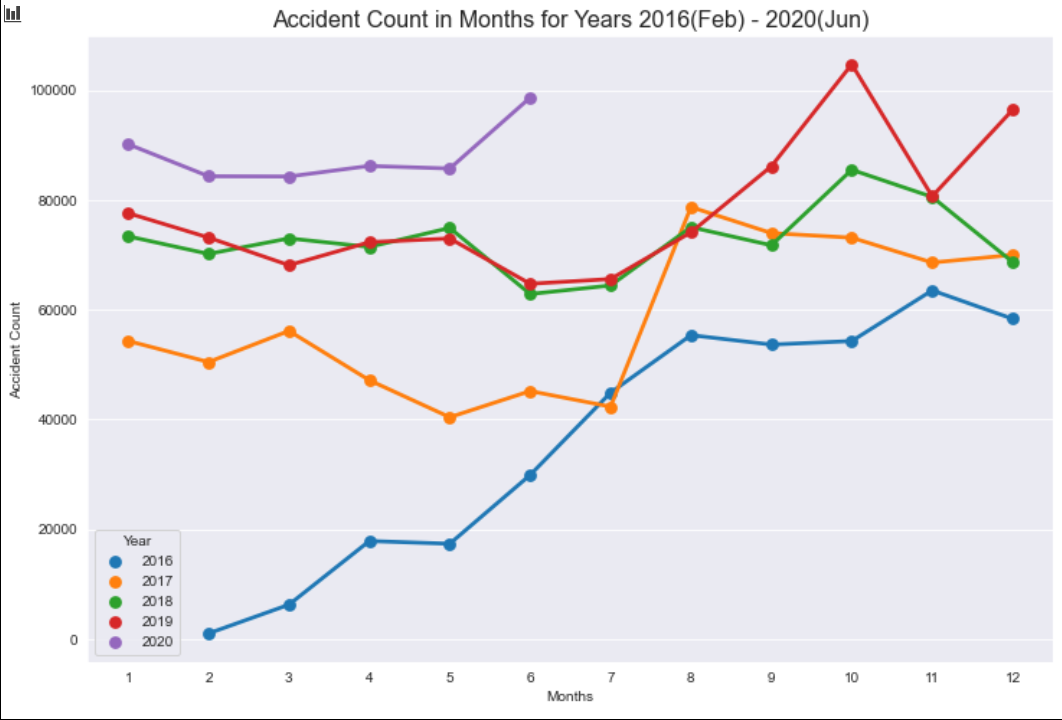


Figure Number of Accidents in months for years 2016(Feb) to 2020(Jun)

To get more insight on the Severity level on the accidents with time I had to plot severity counts in bar plots to compare them. The first bar plot on severity counts was in years, the next bar plot was in months and followed by weekdays count.

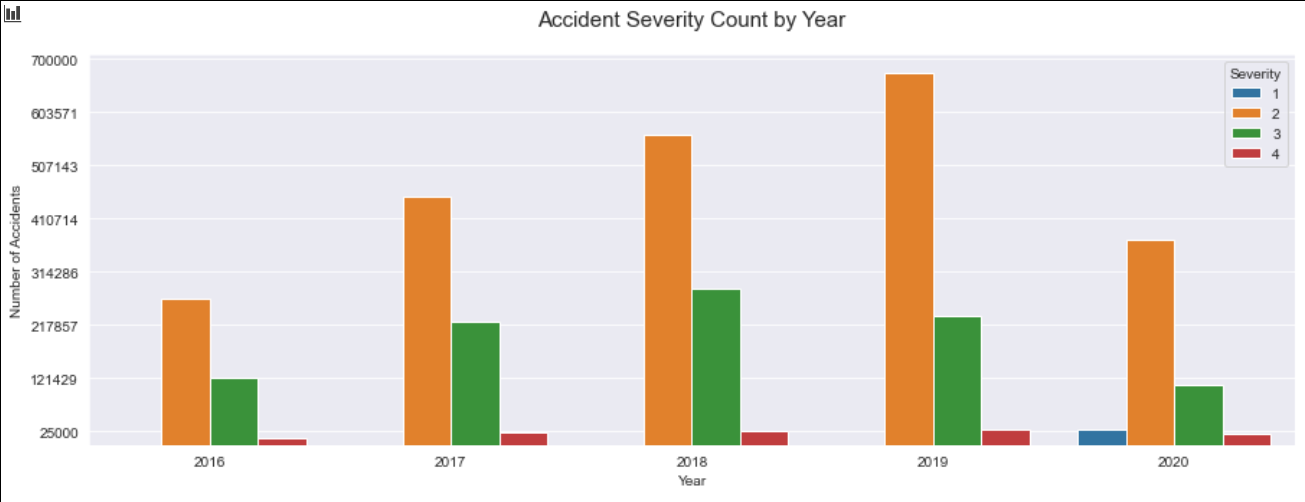


Figure Severities Count in Years

From the above bar plot the severity 1 is not visible in the bar plot as the count is too low except for the year 2020.

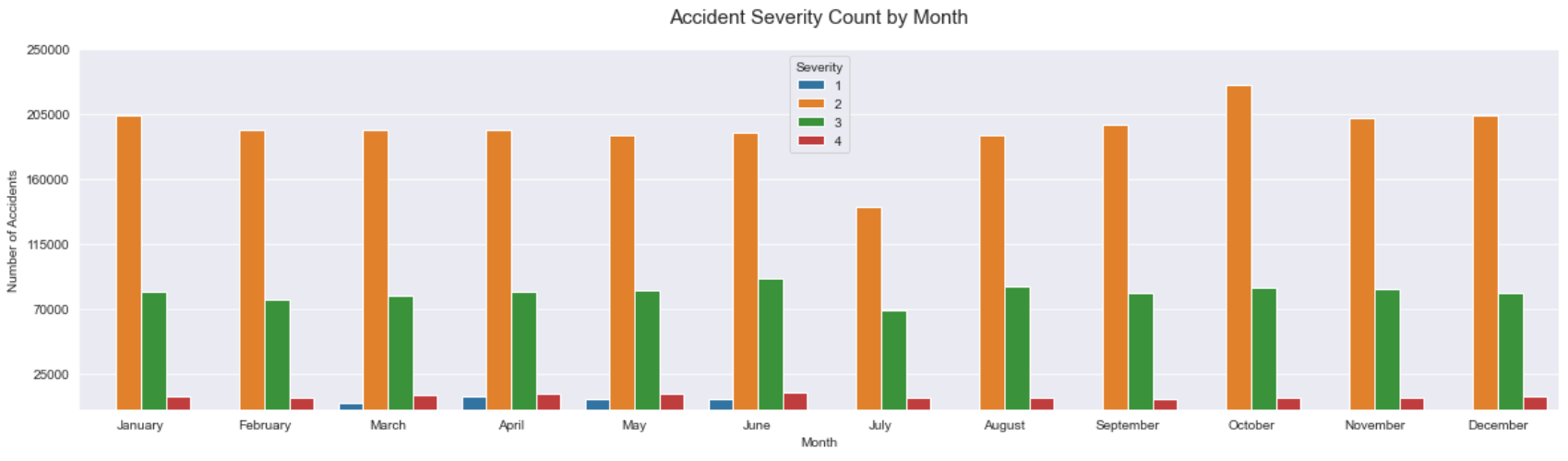


Figure Severities Count in Months

The Figure 6 shows how low the accident count is on the month of July. This could be because it is the hottest month of the year in US so travellers would probably stay home and the workers would be the one transiting round.

October month has recorded more number of accident severity counts as this month falls in autumn there are probability that there could be more travellers to see the autumn fall in various regions. But other regions on the south east part of US experienced heavy rain and thunder storms all over the month which could be the reason for more accident counts.

On further analysis the weather condition was the key feature on the accident counts on several parts of the country which experienced rainy weather throughout a season.

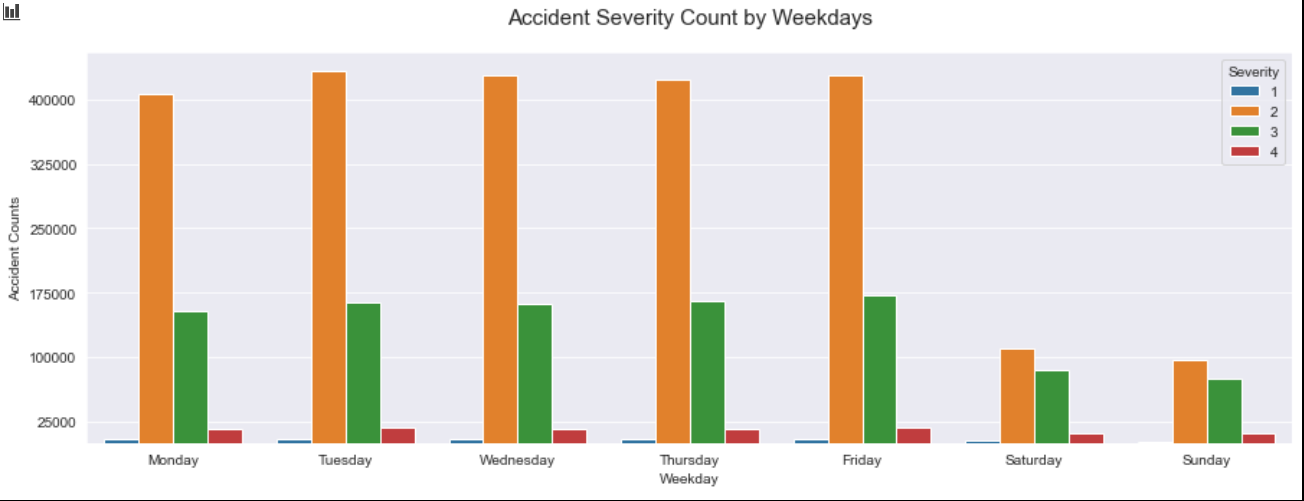


Figure Accident Severity Count for Weekdays

Weekends are widely known to have less traffic in cities, Sunday recording with the less number of accident. The weekday has significant increase and decrease in counts but doesn’t show much difference.

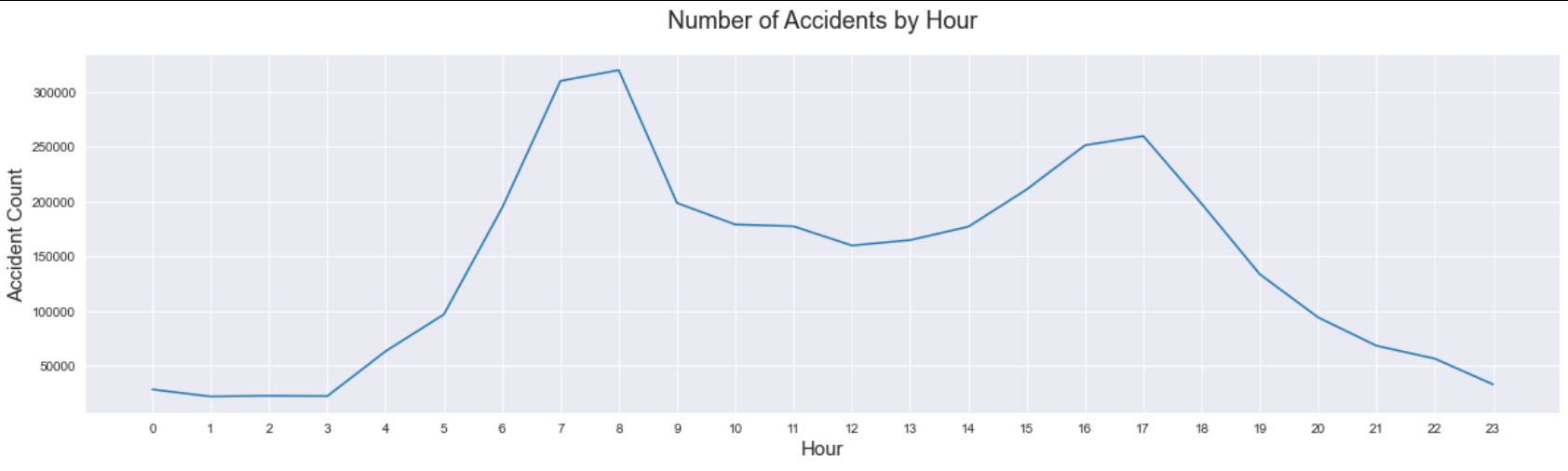


Figure Accident counts per hour

The last part of time series plot is to visualize hourly graph. The graph describes the critical hours of the day with higher accident. There are peaks around the hours 7 to 8 am, the time when people go to work and on the other side the peak is at 6 pm, the time when people return from work.

The number of accidents are less between these two peak hours, its nothing unusual as these are moderate traffic hours.

The remaining hours before dawn and after dusk are silent hours comparatively less people traveling around. This could be a pattern for hourly scale on daily basis.

## 3.3 Relationship with Weather Condition

US experiences different weather conditions from dry, tropical, polar etc. Most common or rather a normal day would be a clear or a cloudy weather which accumulates to have higher range of collision.

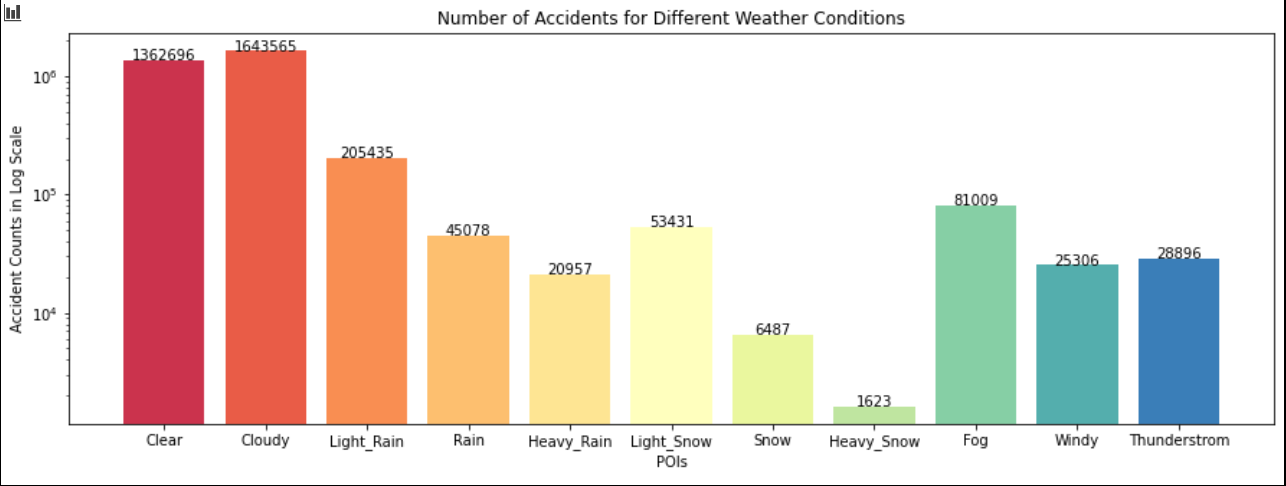


Figure Number of collisions for different weather conditions

As on my research, for the past decade, on the months of October and September US has been facing many Hurricanes and Cyclones which keeps the climate rainy throughout a month. This period of time causes light rain/drizzles over a month of time and few regions experiencing heavy rains.

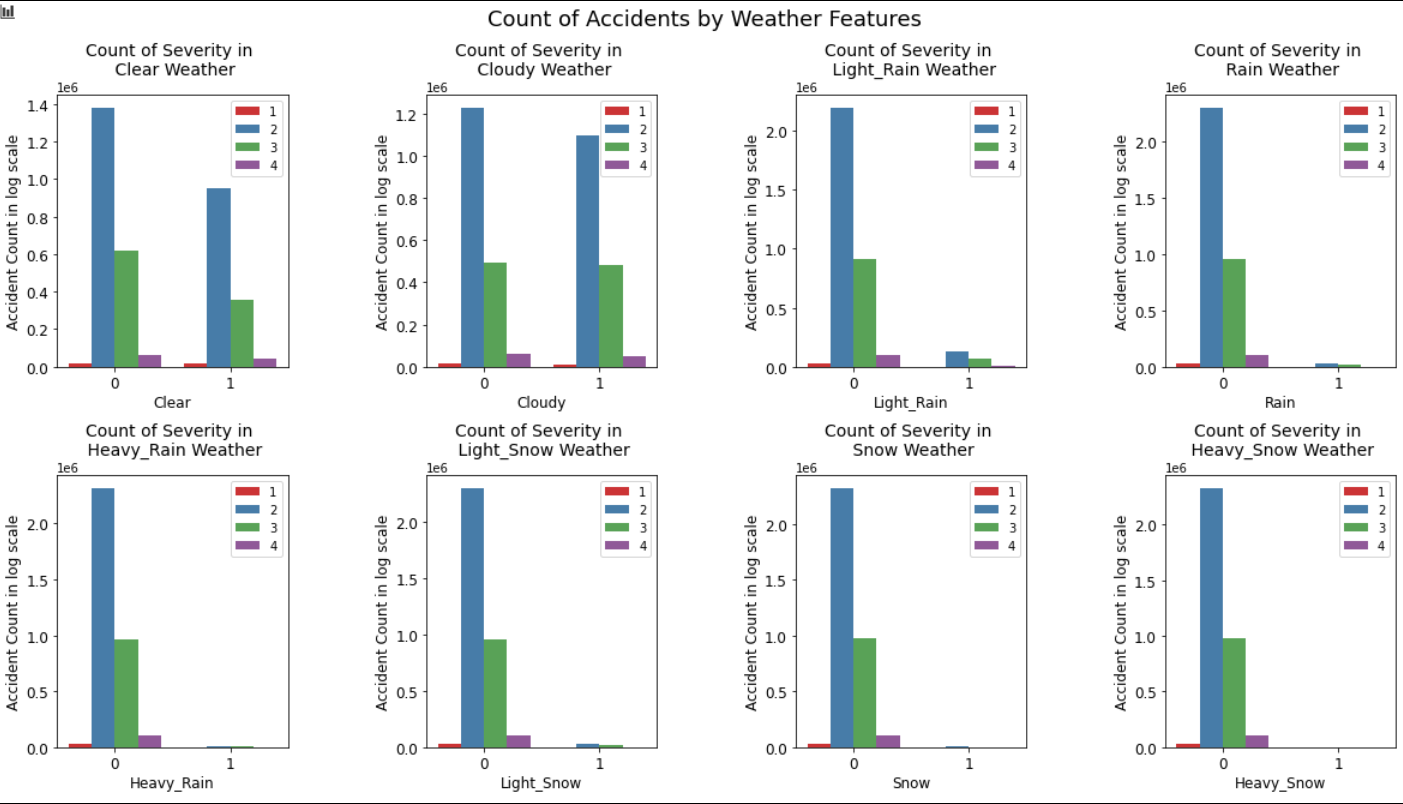


Figure Count of Accidents based on weather features

## 3.4 Relationship with POIs

POI describes the location where the accidents took place. Most common place we could see an accident is in the traffic signals, because many workers go in a hurry and tend to disobey the traffic rules.

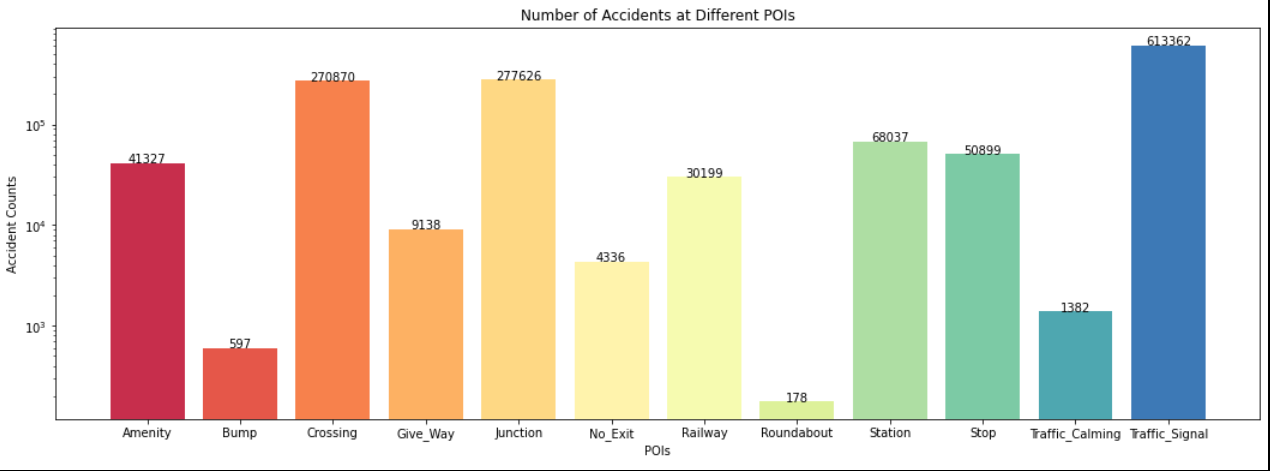


Figure Number of Accidents at Different POIs

Other common place for collisions are the junctions and Crossing. Some drives might not pay attention on the junctions and opposite or the vehicles from the side roads collides. And there are cases were when a speeding vehicle comes by a crossing and get to see someone crossing in a hurry they hit the brakes which would cause the vehicles behind to collide in these unfortunate events.

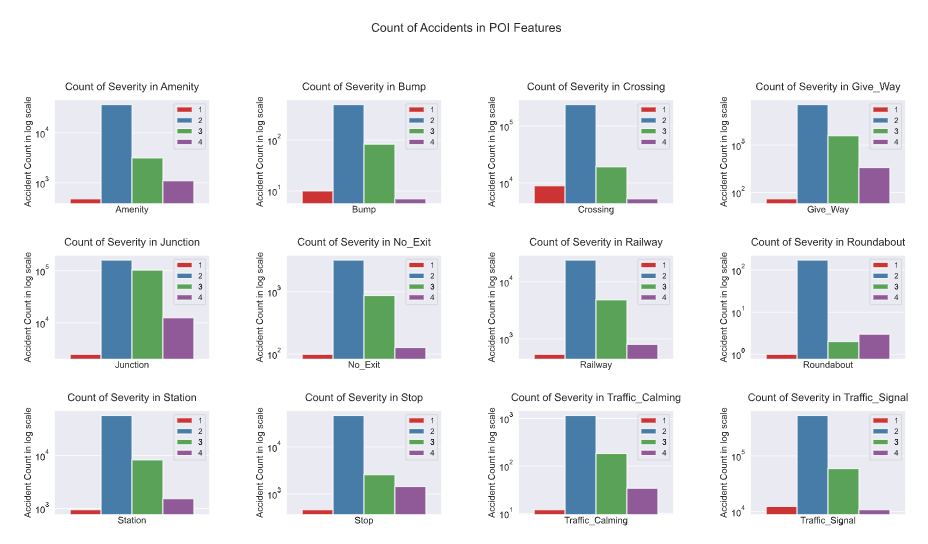


Figure Collision counts at POIs

# Predictive Model

Prediction can be based on two types of models, classification and regression, that can be used to predict the target value. For Accident severity classification algorithms can be built to predict accident severity. In my model I have classified Severity to 2 types high and low, severity level 1, 2 and 3 will be low and severity 4 will be classified as High.

This Model was trained with algorithms in a supervised learning approach to predict at good accuracy and analysing time. To find the best suited algorithm for this prediction model we compared these properties.

To feed the model with supervised dataset the actual data 3.5 million records was entirely used as it was time consuming and memory usage wasn’t enough to compute the full dataset but that will be a part of future plan for my model. For my base model, I have sampled 75000 records for low severity and 75000 records for high severity and combined it to get a data frame of 150000 samples with equal high and low severity.

Before the applying algorithm on our data, there were few steps to be done: First all the True/False values were converted to 1’s and 0’s. The next step was to apply one-hot encoding on the all categorical features. Furthermore, for constructing the data for model the target feature was separated which would be the severity level in our case. The final step was standardizing the dataset but to standardize some of the feature columns had to be dropped which I had used just for visualization purpose.

To start with our model, the first step was to split the resampled dataset into 80/20 as training and test sets.

The Predictive Algorithms that were used are:

* Logistics Regression
* K-Nearest Neighbour
* Decision Tree
* Random Forest

## 4.1 Logistics Regression

Logistics Regression is one of the classic models for predicting the target values, in our case the prediction will be done for the severity. The regression model was built for 10000 iterations. The training data were fit and prediction was done, but to visualize the accuracy the predicted values and the test values were put into the confusion matrix as shown below:

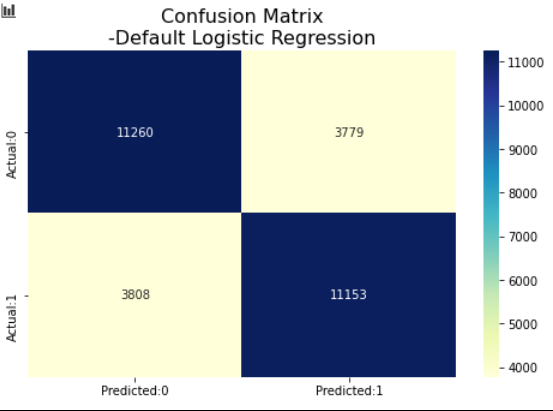


Figure Confusion Matrix for Logistic Regression

This parameter had an accuracy of 74% approximate, but this would be too low and not suitable for prediction. I build a GridSearchCV to find the best parameters for logistic regression. Even with the best params C = 0.001 , max\_iter = 100 and penalty as none the accuracy score was the same.

## 4.2 K-Nearest Neighbour

K-Nearest neighbour model was built with param k = 6. The accuracy score was not as expected which had lower score than logistic regression with 73%.

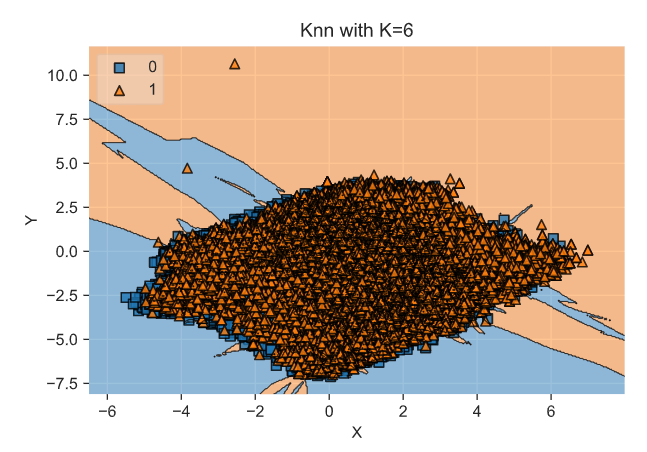


Figure Visualization of KNN model

## 4.3 Decision Tree

A GridSearchCV was performed with decision tree classifier to find the best param for the model. The best parameters were max\_features = None, min\_samples\_split = 5. The accuracy score was better than the previous models showing a result of 98% train accuracy and 79% test accuracy.

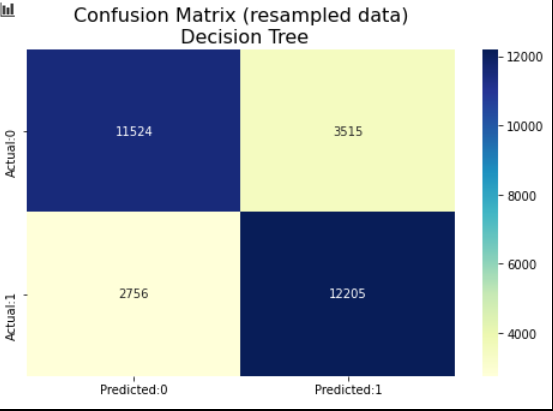


Figure Confusion Matrix for Decision tree

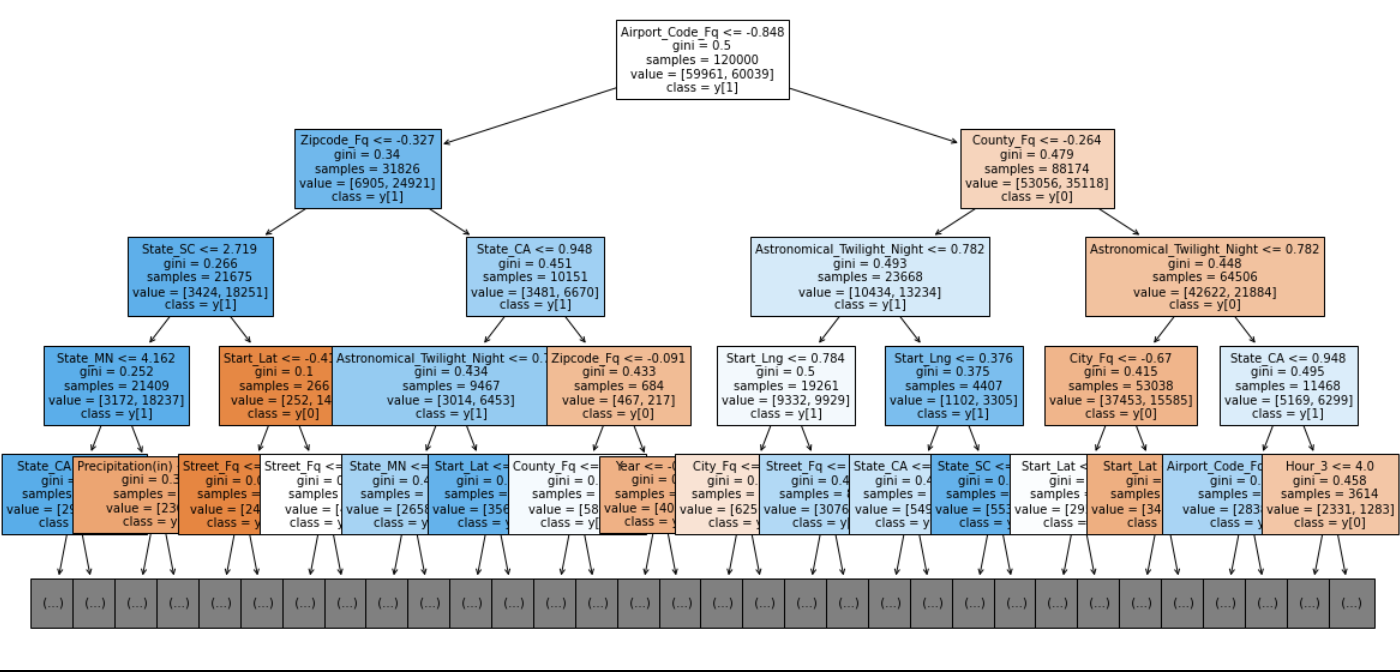


Figure Preview of decision tree model

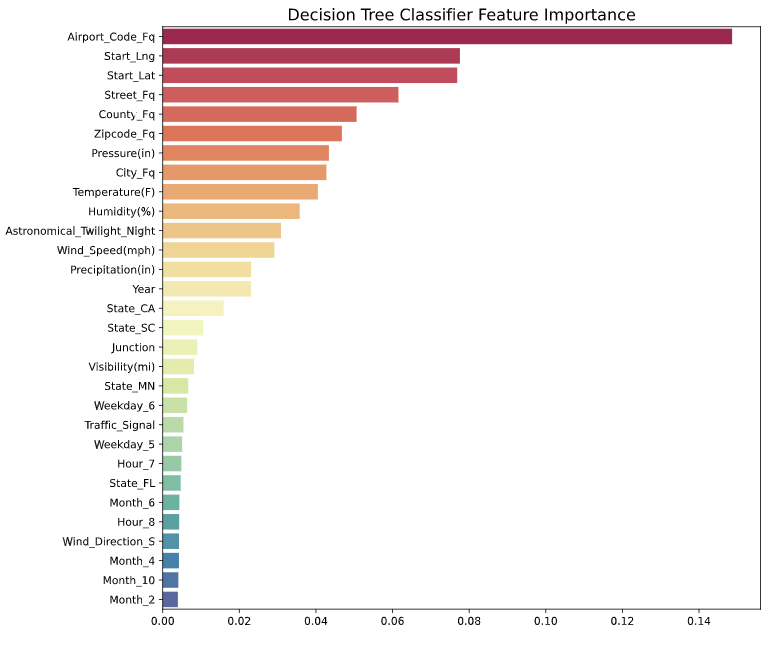


Figure Decision tree's import features

## 4.4 Random Forest

Finally, Random forest classifier was employed. The grid search was performed over choices of n\_estimators of 30, 40 and 50 and max\_depth of 20, 30 and 40 as parameters. The best param were chosen and deployed the model, the model achieved 100% train accuracy and 85.3% test accuracy.

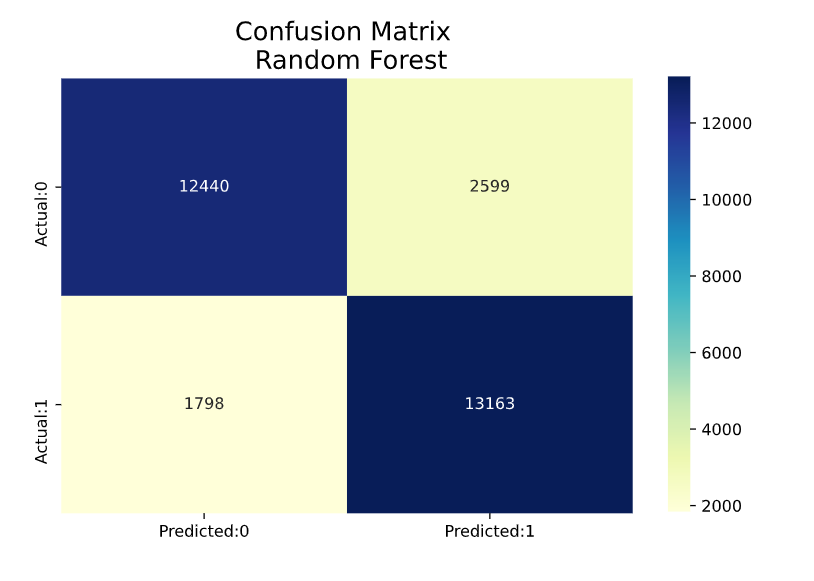


Figure Confusion matrix for Rando forest

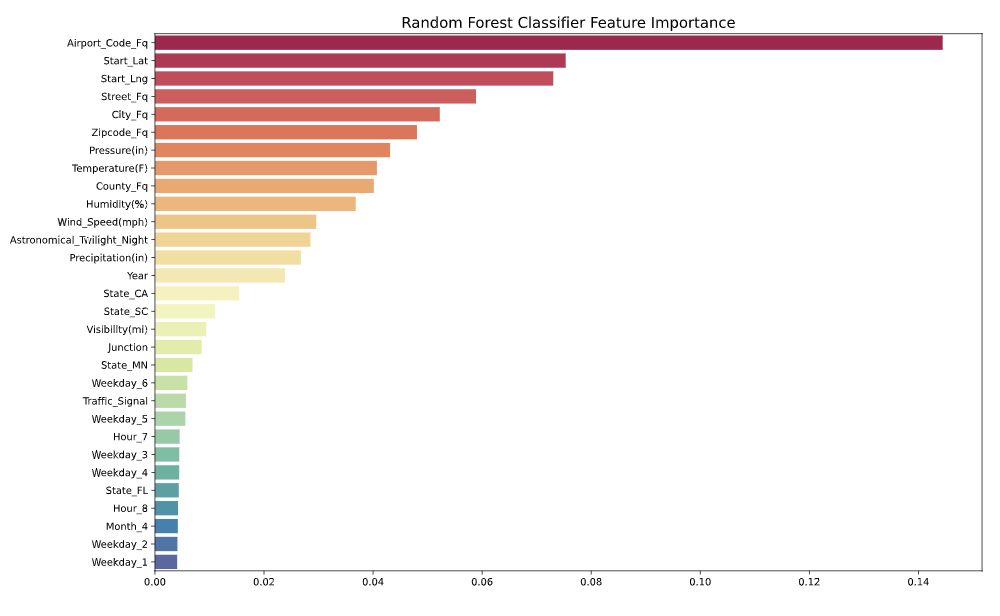


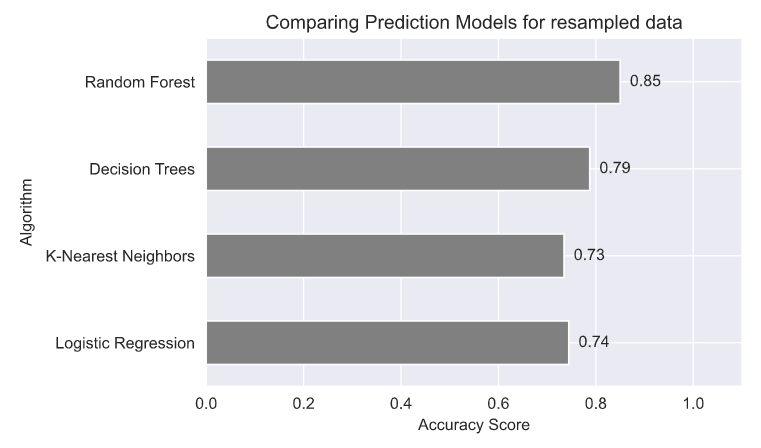
Figure Random forest classifier feature importance

## 4.5 Results

The metrics used to compare the accuracy of the models are Jaccard Score, F1 Score and accuracy Score.

For this specific problem precision means the % of predicted severe accidents that were truly severe. The recall instead, is the % of truly severe accidents that were properly predicted. For this specific problem, the recall is more important than the precision as a high recall will favour that all required resources will be equipped up to the severity of the accident. Logistic regression and KNN, have closer accuracy score, however the computational time from the regression is far better than but not close enough to Random forest which has higher scores overall. With no doubt the Random Forest is the best model. it improves the accuracy of 0.85.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Algorithm** | **Accuracy\_Score** | **F1\_Score** | **Jaccard\_Score** | **Precision\_Scores** | **Recall\_Scores** | **Time Taken** |
| 0 | Logistic Regression | 0.744367 | 0.744361 | 0.593997 | 0.740692 | 0.749950 | 330.033843 |
| 1 | K-Nearest Neighbors | 0.734633 | 0.733700 | 0.559509 | 0.764670 | 0.675891 | 1700.307617 |
| 2 | Decision Trees | 0.787367 | 0.787236 | 0.655970 | 0.772548 | 0.812980 | 75.295472 |
| 3 | Random Forest | 0.849867 | 0.849786 | 0.743814 | 0.833089 | 0.874073 | 3423.417729 |



# 5 Conclusion

When comparing all the models by their f1-score, Jaccard, Precision and Recall, we have a clear picture in terms of accuracy of the four models individually as a whole and how they will perform for each output of target variable. When comparing these scores, we can see that the f1-score is highest for Random forest with 0.85. When looking at the other models, we can see that the Decision Tree has the second highest value. Whereas, the logistic Regression is more balanced. Furthermore, the average f1-score of the two models are very close. It can be concluded that both models can be used to get the best of its performance.

When comparing to the benchmark standards, it can be seen that they perform well but not as good as for the benchmarks. These models could have performed better if few features were present with better correlation.

# 6 Future Directions

I was able to achieve a significant improvement in the benchmark for the regression problem and better accuracy in classification problem. However, there was still significant variance that could not be predicted by the models in this study. More data, especially data of different types, would help improve model performances significantly