CAR ACCIDENT SEVERITY

INSIGHTS AND PREDICTIONS ON SEATTLE CITY DATA

THANVEER AHAMED BADREALAM 18 SEPTEMBER 2020

1. Introduction

The Seattle Department of Transportation's annual traffic report illustrates the constant challenge to the city posed by car accidents. In 2017, Seattle police reported 10,959 motor vehicle collisions on city streets. According to the report, in 2017 there were a total of 187 fatal and serious injury collisions on Seattle streets. Data available from the Washington State Department of Transportation (WSDOT) reflect an even worse tally in 2018, with 212 crashes that resulted in serious injury or wrongful death.

Society as a whole - the accident victims and their families, their employers, insurance firms, emergency and health care personal and many others are affected by motor vehicle crashes in many ways. It would be great if real-time conditions can be provided to estimate how safe each trip is. In this way, it can be decided beforehand if the driver will take the risk, based on reliable information.

1.1 Business Problem

The Seattle government is going to prevent avoidable car accidents by employing methods that alert drivers, health system, and police to remind them to be more careful in critical situations.

In most cases, not paying enough attention during driving, abusing drugs and alcohol or driving at very high speed are the main causes of occurring accidents that can be prevented by enacting harsher regulations. Besides the aforementioned reasons, weather, visibility, or road conditions are the major uncontrollable factors that can be prevented by revealing hidden patterns in the data and announcing warning to the local government, police and drivers on the targeted roads.

1.2 Target Audience

The target audience of the project are drivers, local Seattle government, police, rescue groups, and last but not least, car insurance institutes. The model and its results are going to provide some advice for the target audience to make insightful decisions for reducing the number of accidents and injuries for the city.

2. Data

The data utilised for the analysis is provided by the Traffic Records Group in the SDOT Traffic Management Division. The department ensures the safe and efficient running of the transportation system to provide safe and affordable access to places and opportunities. The data includes all collisions provided by the Seattle Police Department and recorded by the Traffic Record that occurred at an intersection or mid-block of a segment, from 2004 to the present.

The attributes we will utilise are described in the following table:

Feature	Description
OBJECTID	ESRI unique identifier
INCKEY	A unique key for the incident
ADDRTYPE	Collision address type: - Alley - Block - Intersection
COLLISIONTYPE	Collision type
PERSONCOUNT	The total number of people involved in the collision
PEDCOUNT	The number of pedestrians involved in the collision
PEDCYCLOUNT	The number of bicycles involved in the collision
VEHCOUNT	The number of vehicles involved in the collision
JUNCTIONTYPE	Category of junction at which collision took place
WEATHER	A description of the weather conditions during the time of the collision
ROADCOND	The condition of the road during the collision
LIGHTCOND	The light conditions during the collision
SEVERITYCODE	A code to describe the severity of the collision

Our target attribute is the severity of the collision which is categorised as 1 if there is only property damage and 2 if it includes personal injury.

2.1 Feature Selection

The dataset is not ready for data analysis as we need to first remove the irrelevant columns from the dataset. We drop all columns except OBJECTID, INCKEY, STATUS, ADDRTYPE, COLLISIONTYPE, PERSONCOUNT, PEDCOUNT, PEDCYCLOUNT, VEHCOUNT, JUNCTIONTYPE, WEATHER, ROADCOND, LIGHTCOND.

2.2 Data Cleaning

Many of the features mentioned above contain incomplete information such as 'NaN' values. We are going to fill the missing values in the dataset with the value 'others'.

Additionally, the frequency of the property damage accidents (Class 1) are more than double the ones involving injuries (Class 2). Therefore, the data cleaning process must involve balancing the data to prevent a biased model.

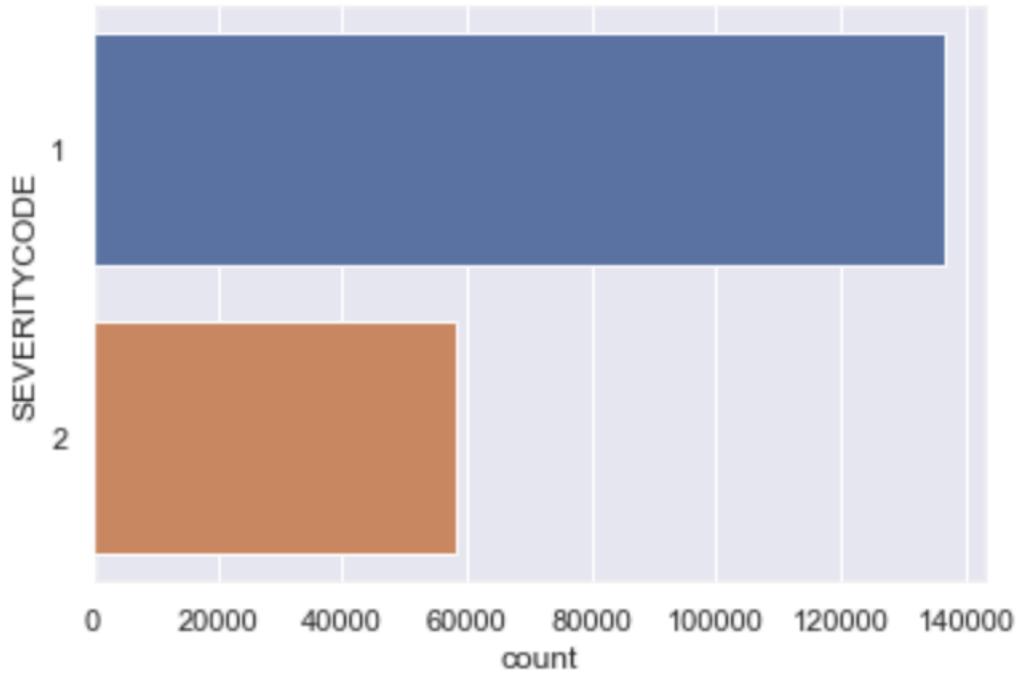

Figure 1. Severity Classification before cleaning and balancing the data

We do this by downsampling Class 1. Down-sampling involves randomly removing observations from the majority class to prevent its signal from dominating the learning algorithm.The most common heuristic for doing so is resampling without replacement. Here are the steps:

1. First, we'll separate observations from each class into different DataFrames.

2. Next, we'll resample the majority class **without replacement**, setting the number of samples to match that of the minority class.
3. Finally, we'll combine the down-sampled majority class DataFrame with the original minority class DataFrame.

Once the data is balanced, the number of entries corresponding to severity Class 1 and Class 2 are equal.

Figure 2. Severity Classification After downsampling Class 1

3. Methodology

3.1 Exploratory Data Analysis

Exploratory Data Analysis or (EDA) is understanding the data sets by summarizing their main characteristics often plotting them visually. This step is very important especially when we arrive at modelling the data in order to apply Machine learning. Plotting in EDA consists of Histograms, Box plot, Scatter plot and many more. It often takes much time to explore the data. Through the process of EDA, we can ask to define the problem statement or definition on our data set which is very important.

3.1.1 Type of Collision

One important aspect revealed by the data is related to the severity of accidents based on the collision type. This feature has different characteristics based on the area of impact, such as: angles, parked car, rear end, right turn, sideswipe, head on, left turn, pedestrian and cycles.

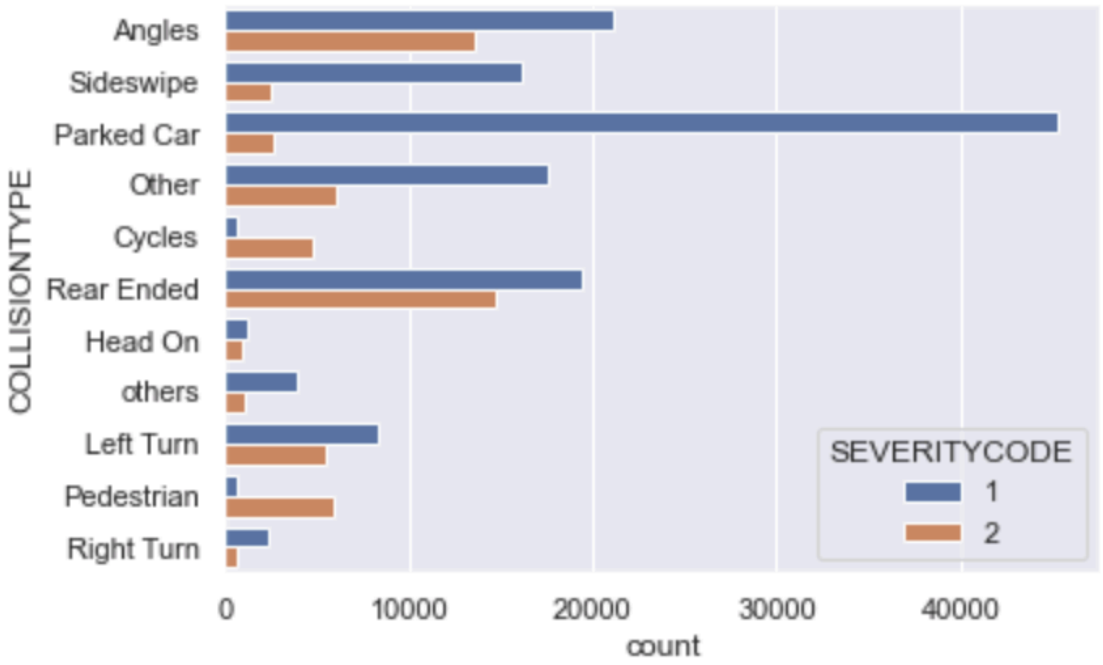

Figure 3. Collision categories, frequency and severity

3.1.2 Place of Collision

It is clearly visible that there is a higher frequency of severe accidents at intersections rather than in the middle of the block. Mid-block collisions are not severe, mostly involving car damage.

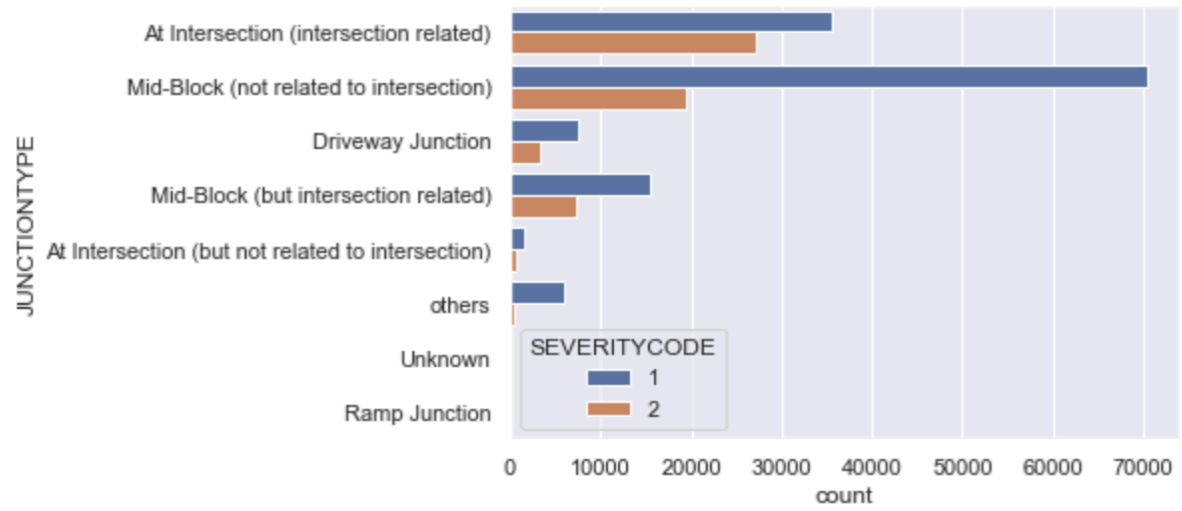

Figure 4. Junction types, frequency and severity

3.1.3 Weather, Light and Road conditions

Looking at the weather data, severe accidents are not frequent during rainy or overcast weather conditions. So this might not add too much information to our model.

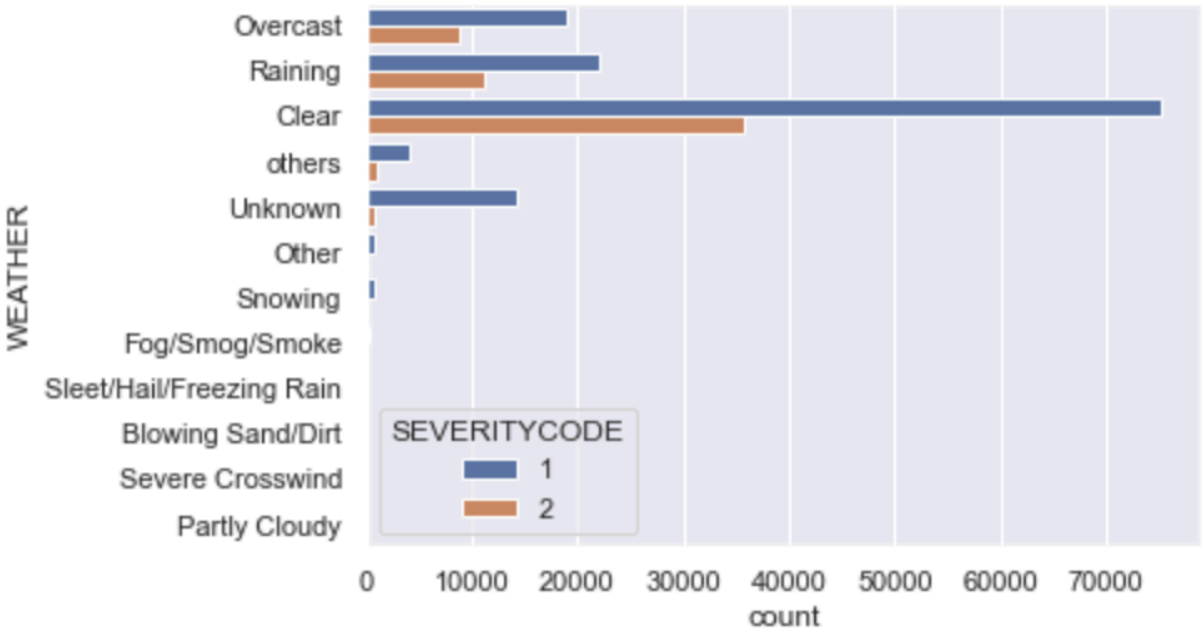

Figure 5. Weather conditions, frequency and severity

Another aspect related to severity is the road condition. This feature has different characteristics like wet, dry, snow, ice etc. Light conditions reveal some insight into the data. There are more occurrences of severe collisions during daylight compared to night. One reason for this maybe that the driver is more conscious driving during the night hours. Dusk and dawn tend to be more related to severe collisions due to the position of the sun relative to the driver.

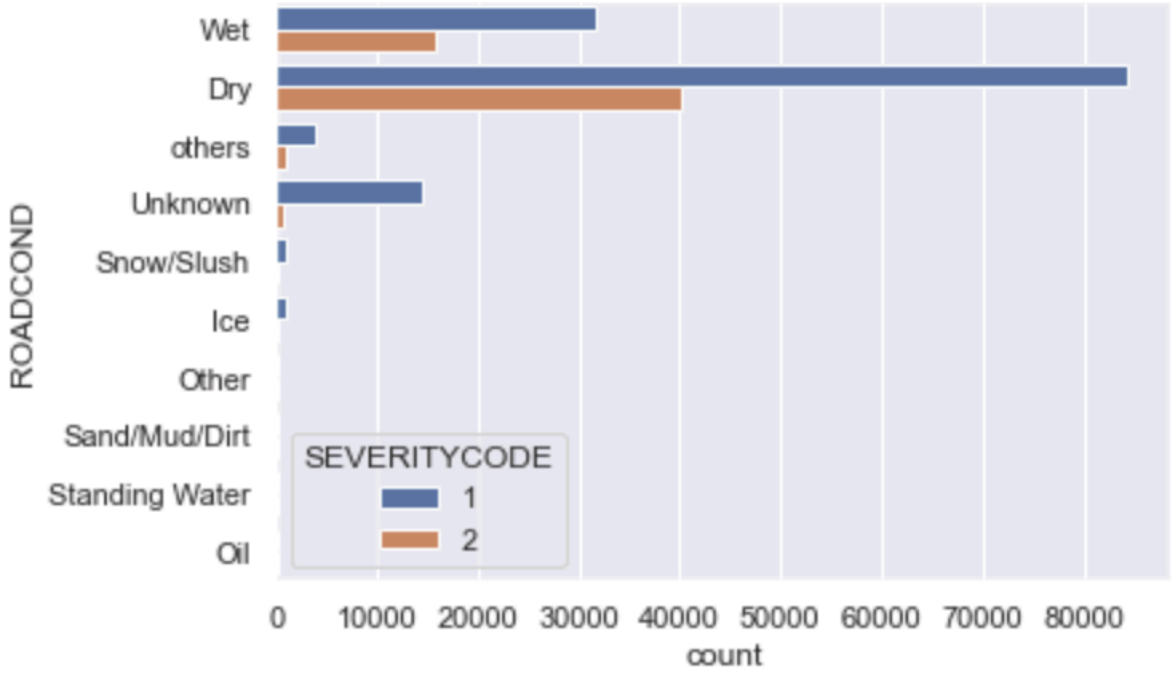

Figure 6. Road conditions, frequency and severity

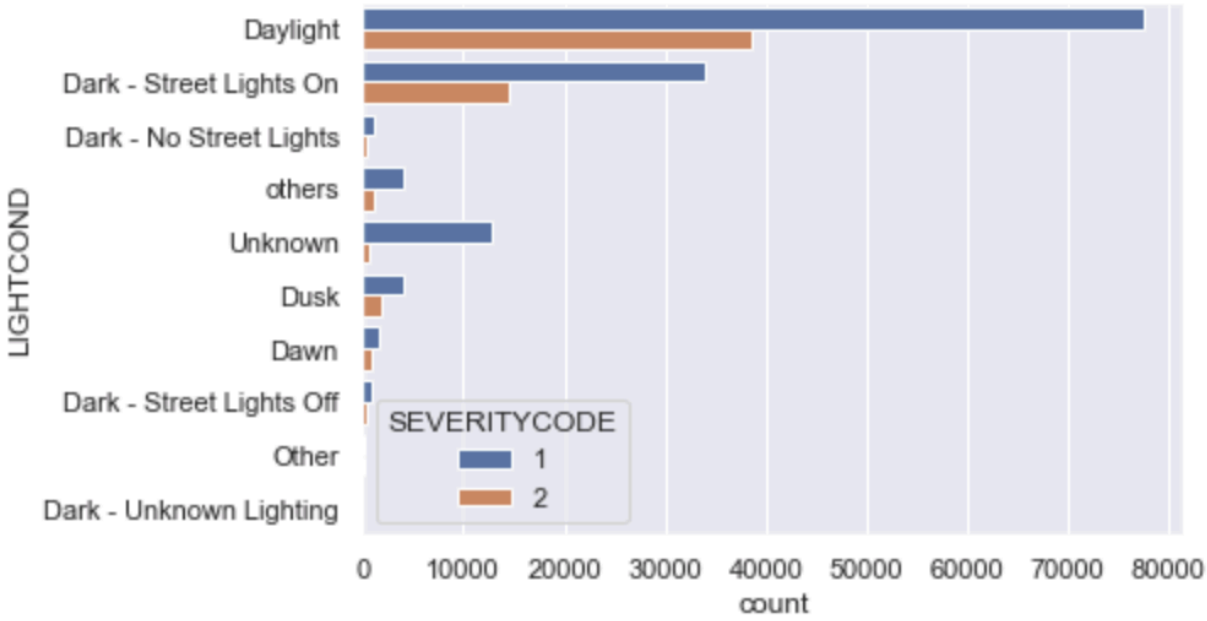

Figure 7. Light conditions, frequency and severity

3.1.4 Clustering collisions by geographic area

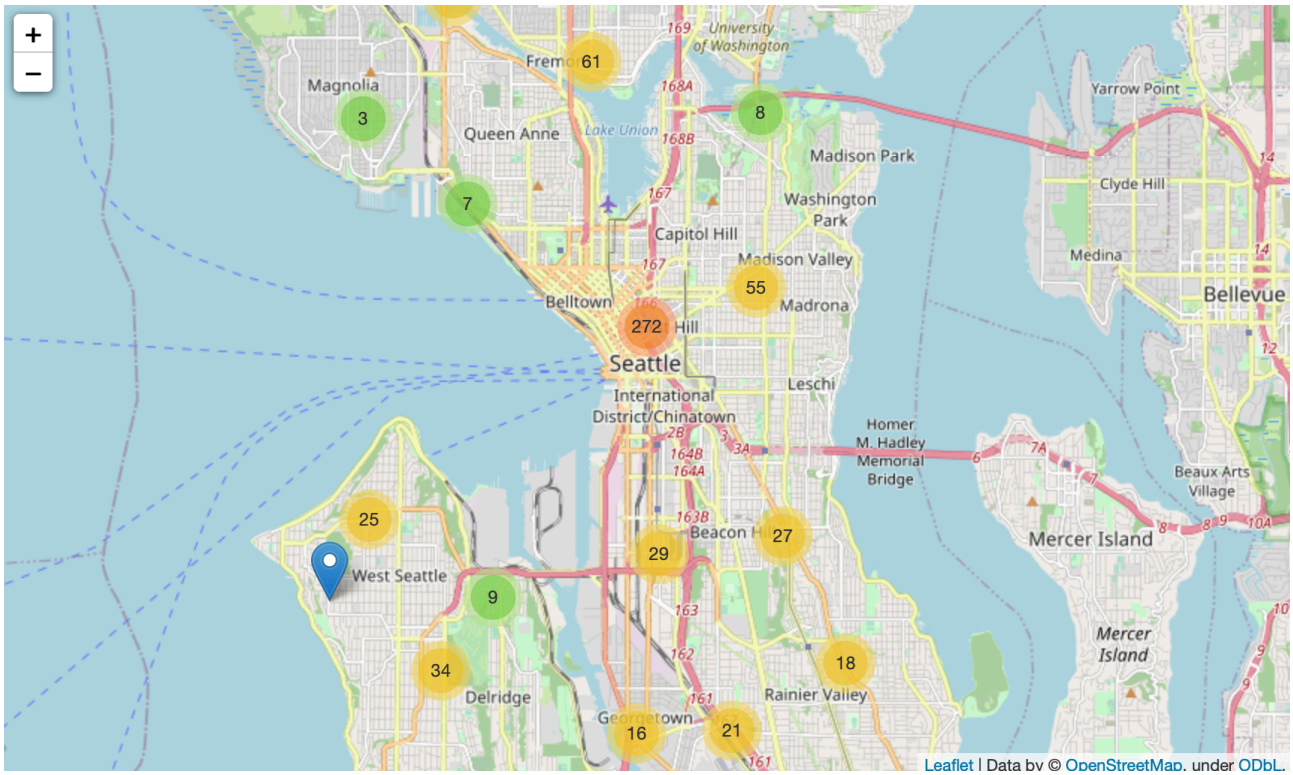

Figure 8. Clustered collisions by area

From the above map it is clear that most of the accidents occur in the downtown area. The downtown area local government should further evaluate and increase infrastructure to reduce collision incidences.

3.2 Predictive Modelling

The machine learning models used are K-Nearest Neighbours (KNN), Decision Trees, Logistic Regression and Random Forest Classifier.

3.2.1 K-Nearest Neighbour

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on the Supervised Learning technique. The KNN algorithm assumes the similarity between the new case/data and available cases and puts the new case into the category that is most similar to the available categories. The KNN algorithm stores all the available data and classifies a new data point based on the similarity. This means that when new data appears, it can be easily classified into a well suited category by using the K-NN algorithm. KNN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data. It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. The KNN algorithm at the training phase just stores the dataset and when it gets new

data, then it classifies that data into a category that is much similar to the new data.

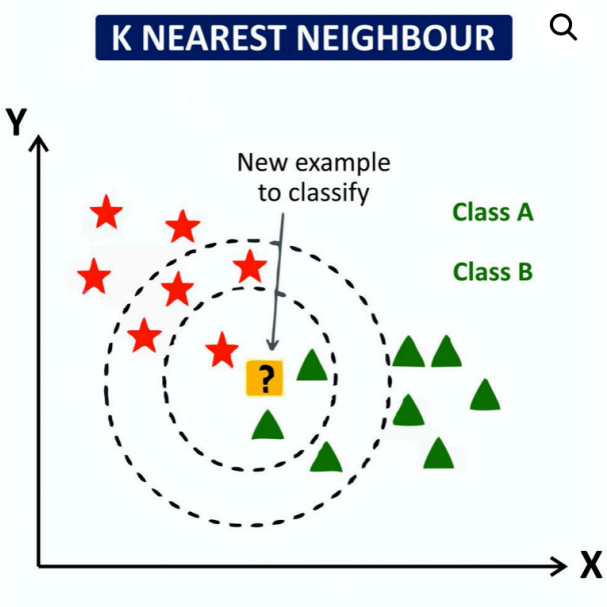

Figure 9. KNN algorithm

3.2.2 Logistic Regression

Logistic Regression is a Machine Learning algorithm which is used for classification problems. It is a predictive analysis algorithm and based on the concept of probability. We can call a Logistic Regression model a Linear Regression model but the Logistic Regression uses a more complex cost function, this cost function can be defined as the '**Sigmoid function**' or also known as the 'logistic function' instead of a linear function. The hypothesis of logistic regression tends it to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

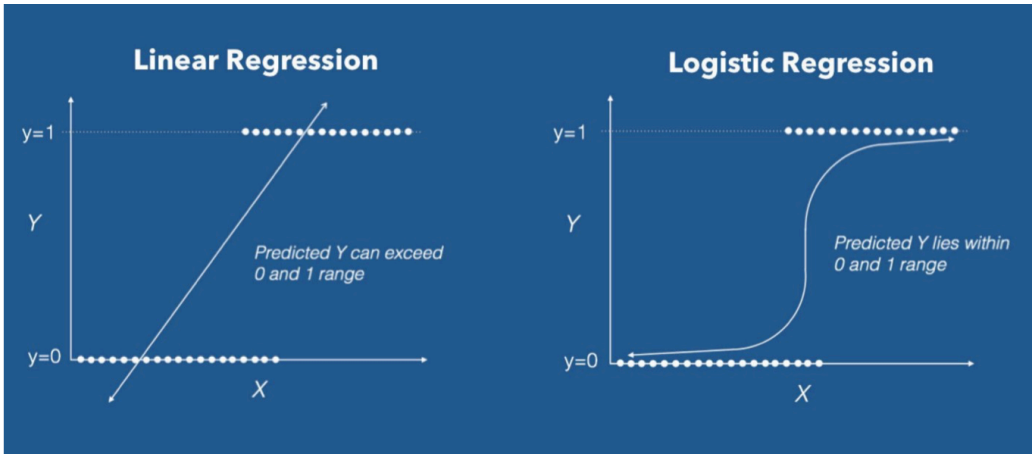

Figure 10. Comparison of Linear and Logistic regression

3.2.3 Decision Tree

Decision Tree is a **Supervised learning technique** that can be used for both classification and regression problems, but mostly it is preferred for solving classification problems. It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.** In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node.** Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions. It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

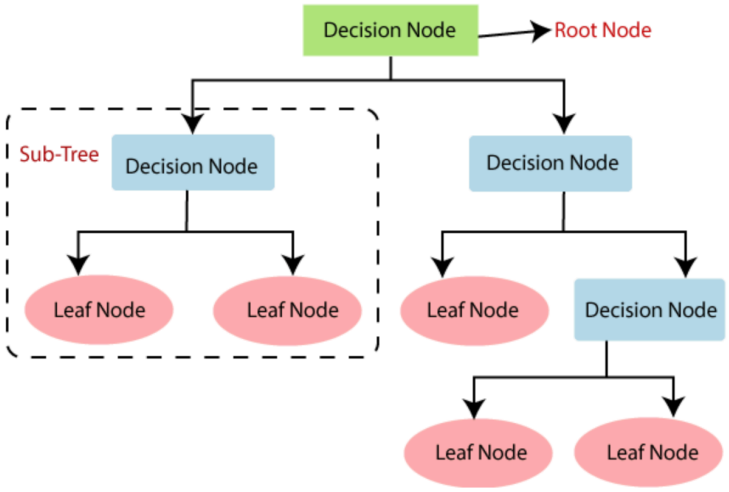

Figure 11. Decision tree algorithm

3.2.4 Random Forest

Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. As we know that a forest is made up of trees and the more the trees the more robust the forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.

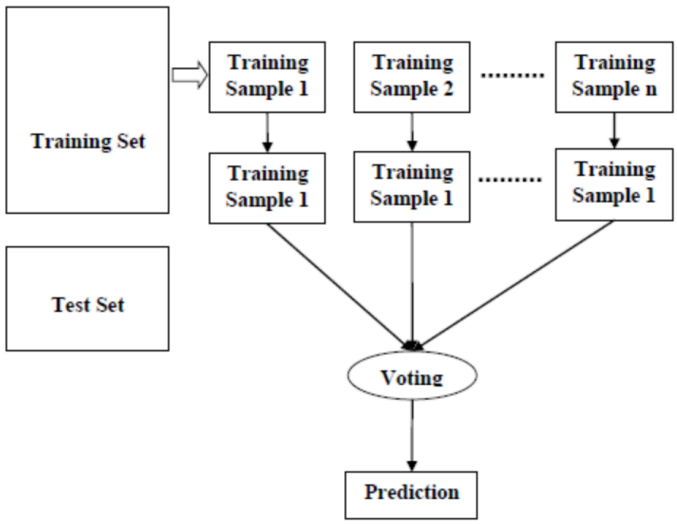

Figure 12. Random forest algorithm

3.3 Encoding Categorical data

Typically, any structured dataset includes multiple columns – a combination of numerical as well as categorical variables. A machine can only understand numbers. It cannot understand text. That's essentially the case with Machine Learning algorithms too. That's primarily the reason we need to convert categorical columns to numerical columns so that a machine learning algorithm understands it. This process is called categorical encoding. Label Encoding is a popular encoding technique for handling categorical variables. In this technique, each label is assigned a unique integer based on alphabetical ordering. Few columns in our dataset are categorical, so we apply Label Encoding for ADDRTYPE, STATUS, COLLISIONTYPE, JUNCTIONTYPE, WEATHER, ROADCOND, LIGHTCOND columns.

4. Results

4.1 KNN Modeling

In this model, the first thing to do is to find out the best k parameter. To do so, it can be iterated over a set of K values and find the one with best accuracy score.

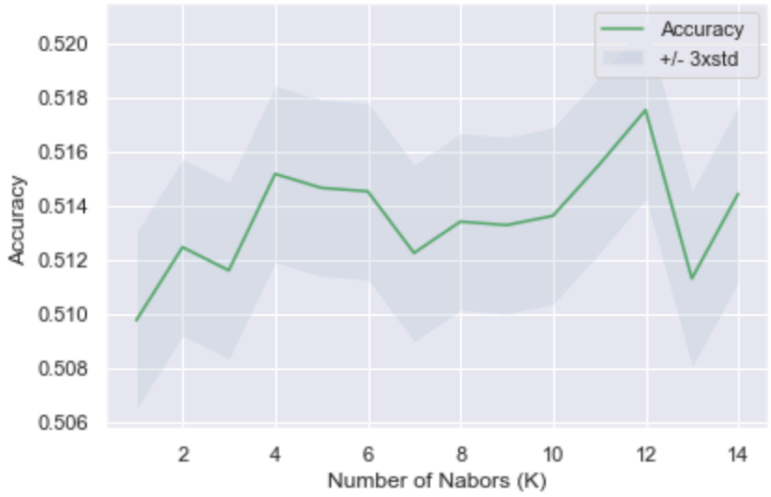

Figure 13. Iteration of accuracy scores for different k values

From the above graph, we choose K=12 as the best k value and run the knn model and find the evaluation metrics. The classification report and the confusion matrix are provided in the following table and image respectively.

	Precision	Recall	F1 Score	Support
1-property damage	0.52	0.61	0.56	11754
2-personal injuries	0.52	0.42	0.46	11522
Accuracy			0.52	23276
Macro Avg	0.52	0.52	0.51	23276
Weighted Avg	0.52	0.52	0.51	23276

Table 1. Classification report for KNN (k=12)

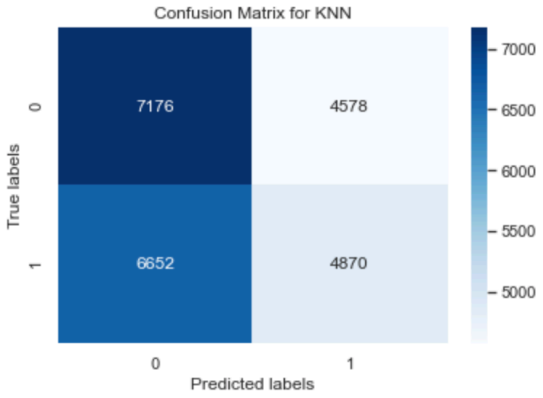

Figure 14. Confusion matrix for KNN

4.2 Decision Tree

For the Decision Tree model we have specified the parameters for criterion as 'gini' and max_depth as 4 and all the other parameters are set to default. The classification report for this model can be found in the following table.

	Precision	Recall	F1 Score	Support
1-property damage	0.74	0.62	0.68	11754
2-personal injuries	0.67	0.78	0.72	11522
Accuracy			0.70	23276
Macro Avg	0.71	0.70	0.70	23276
Weighted Avg	0.71	0.70	0.70	23276

Table 2. Classification report for Decision Tree

We have also generated a confusion matrix of this model which is is given in the following image.

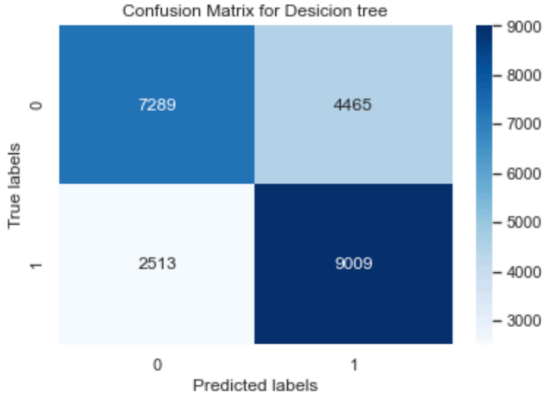

Figure 15. Confusion matrix for Decision tree

4.3 Logistic Regression

For Logistic Regression, we have set all the parameters to default. The classification report for this model is in the following table. The confusion matrix generated for this model is in the following image.

	Precision	Recall	F1 Score	Support
1-property damage	0.63	0.65	0.64	11754
2-personal injuries	0.64	0.61	0.62	11522
Accuracy			0.63	23276
Macro Avg	0.63	0.63	0.63	23276
Weighted Avg	0.63	0.63	0.63	23276

Table 3. Classification Report for Logistic regression

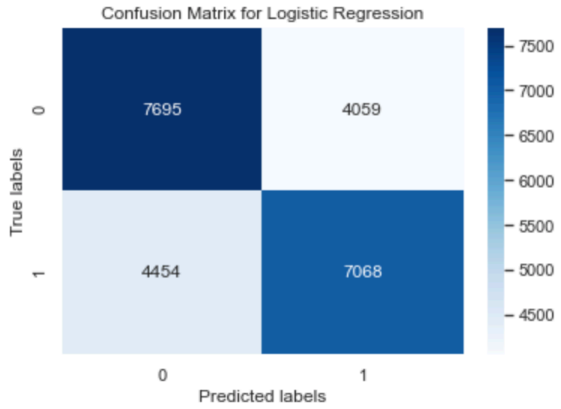

Figure 16. Confusion matrix for Logistic regression

4.4 Random Forest

In this Random Forest Classifier the parameters were set to default. the classification report for this model is given in the following table.

	Precision	Recall	F1 Score	Support
1-property damage	0.67	0.68	0.67	11754
2-personal injuries	0.67	0.66	0.66	11522
Accuracy			0.67	23276
Macro Avg	0.67	0.67	0.67	23276
Weighted Avg	0.67	0.67	0.67	23276

Table 4. Classification Report for Random forest

The Confusion Matrix generated for this model is in the image below.

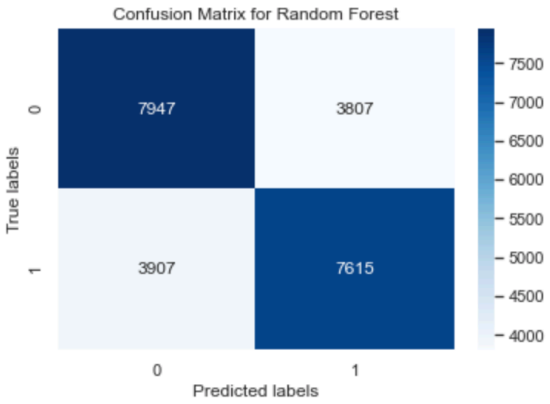

Figure 16. Confusion matrix for Random forest

4.5 Summary

Algorithm	Accuracy	F1-score	Jaccard
KNN	0.51	0.56	0.38
Decision Tree	0.70	0.67	0.51
Logistic Regression	0.63	0.64	0.47
Random Forest	0.66	0.67	0.50

Table 5. Performance summary for the Machine learning models

Based on the results obtained from different models, we charted out a table to compare the accuracy score, F1-score and Jaccard score. The table is given above.

Classification Accuracy is what we usually mean, when we use the term accuracy. It is the ratio of number of correct predictions to the total number of input samples. The accuracy score is slightly more for Decision Tree than the other models.

F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1]. It tells you how precise your classifier is (how many instances it classifies correctly), as well as how robust it is (it does not miss a significant number of instances). High precision but lower recall, gives you an extremely accurate result, but it then misses a large number of instances that are difficult to classify. The greater the F1 Score, the better is the performance of our model.

The F1 Score is almost the same across all the models.

Jaccard index is a measure for examining the similarity (or dissimilarity) between two sample data objects. It is defined as the proportion of the intersection size to the union size of the two data samples. It provides a very simple and intuitive measure of similarity between data samples.The Jaccard score for KNN is much lower compared to the other models.

Based on the results table, the Decision tree classifier can be considered as a better model.

5. Discussion

The original dataset had a lot of features of type object, that we converted from categorical to numeric using Label Encoding.

After conversion, when we checked for the frequency of the values in the target variable, the data was imbalanced. So we downsampled the dataset in order to avoid bias in the model.

The target variable (SEVERITYCODE) is categorical with Class 1 and Class 2 as the attributes. So we used four classification models- K-Nearest Neighbours, Decision Tree, Logistic Regression and Random Forest to predict the target variable. We evaluated the models using Accuracy score, F1 score , Jaccard score, a classification report and Confusion matrix. From the results, we concluded that the Decision Tree model was the most accurate.

We can still improve the above models, by better tuning of the hyper-parameters like the “k” in KNN, the “max_depth” in the Decision Tree, the “C” parameter in the Logistic Regression and the “max_features”, “n_estimators” parameters in the Random Forest.

6. Conclusion

The data analyzed revealed some important information about car accidents. Severe accidents with resulting personal injury tend to happen at intersections, during rear end collisions and either involve pedestrians or bicycles. Left turns are also considered risky due to their high volume of severe accidents.

Dangerous weather and road conditions such as snow and ice don’t seem to produce a significant rise in the frequency of accidents. However, caution must be taken on rainy days with wet roads.

Finally, the results of the machine learning algorithms using predictors such as the weather, road and light conditions produces mediocre results. Other factors have to be considered to improve the prediction rate of the models being utilised.