PREDICTING THE SEVERITY OF ACCIDENTS IN SEATTLE

AADIT KHANDELWAL

SEPTEMBER' 2020

1. INTRODUCTION

1.1 PROBLEM

There are a lot of accidents which can be avoided by making people aware of certain accident-prone areas. So, to reduce the frequency of vehicle collisions, an algorithm or a model must be developed to predict the severity of an accident given the current weather, road conditions, street lights conditions. By doing so we can alert the drivers beforehand so as to be more careful and drive accordingly if the model predicts high severity. Hence, reducing the frequency of accidents in the community and saving the lives of people. This model targets all the people in the community.

1.2 INTEREST

The people and the government would be most likely interested in a model that predicts the severity of an accident in particular areas. This would help reduce the accidents and save lives.

2. DATA ACQUISITION AND CLEANING

2.1 Data Sources

The dataset that we have used has been provided by IBM in their course named Applied Data science. It has 194673 rows and 37 columns. This means that it has 194672 samples and 37 features.

2.2 Data Cleaning and feature selection

There are a lot of missing and unknown values in the dataset which had to be removed. 'SPEEDING' feature was dropped as a whole because it had around 180000 missing values.

I built the model on the features ROADCOND, LIGHTCOND and WEATHER. I believed that these are the conditions on which a large number of road accidents would depend upon and then we could predict the severity of an accident in a particular area and alert the drivers beforehand.

Then the data was made categorical to predict the severity code using machine learning models. All the text values in the features that we have used were replaced by numerical values.

So, first we have a plot of the count of ‘SEVERITYCODE’. We infer from the Fig 2.1 that the data is largely imbalanced and it has more samples for ‘SEVERITYCODE = 1’.

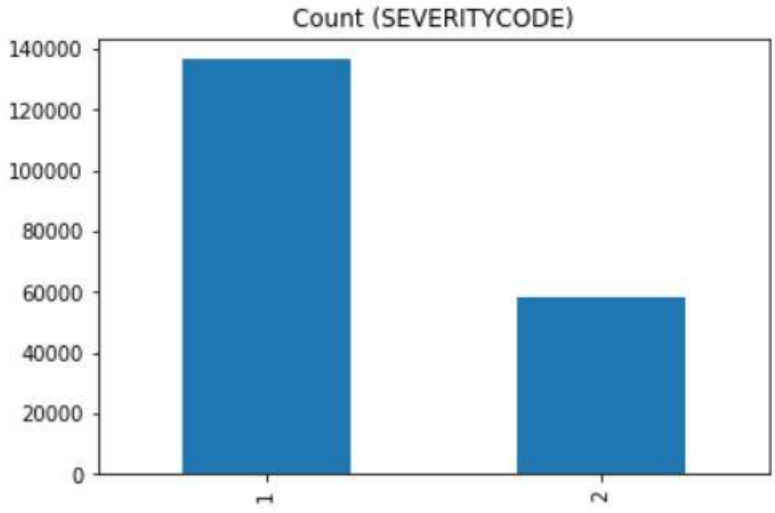

Fig 2.1

Then we use Random under sampling to sample the data and make it balanced. In under sampling we have a reduced set of samples to work with as can be seen by the count in the Fig 2.2.

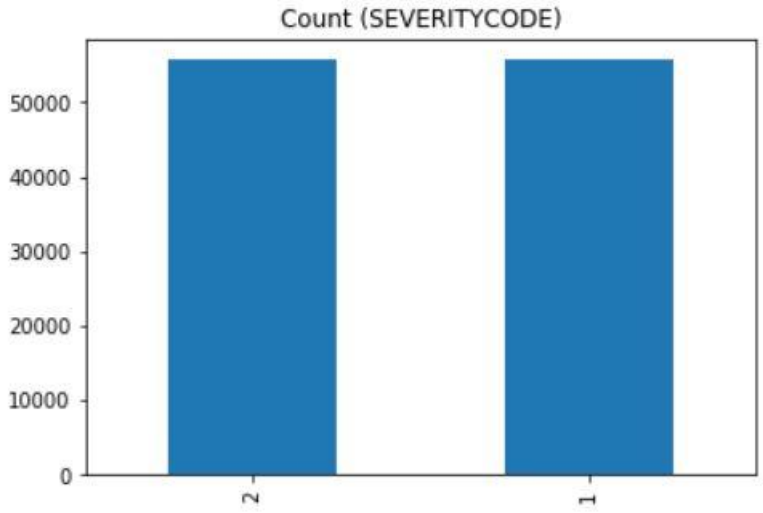

Fig 2.2 (Random Under Sampling)

After working with under sampled data, we performed random over sampling to increase the samples of the class that are very low in comparison to the other. So, in this case, samples of class 2 were increased and hence the dataset was made balanced.

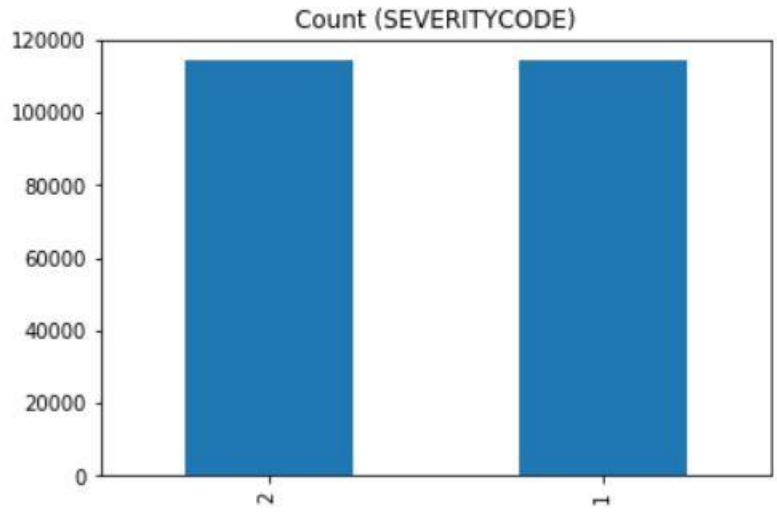

Fig 2.3 (Random Over Sampling)

3. METHODOLOGY

After the data was made suitable for our study and the application of machine learning models, we split the data into training and testing set. After splitting the data, pre-processing of the data was done.

The application of machine learning models was done on both imbalanced and balanced data so as to see the difference in accuracy between both and what are the reasons behind these levels of accuracy.

Correlation is a statistical technique that can show whether and how strongly pairs of variables are related. Finding the correlation among the features of the dataset helps understand the data better. For example, in the below figure (correlation plot using matplotlib), it can be observed that some features have a strong positive/negative correlation while most of them have weak/ no correlation.
So, we plotted the correlation plot of our dataset to see the relationship between our features among each other and with the target that is 'SEVERITYCODE'. We see that the values on the diagonal are having values as 1 and that means strong positive correlation. This has to happen because on the diagonals we are finding the correlation of a feature with the feature itself. Further, we see that WEATHER and ROADCOND have as strong positive correlation with a value of 0.81. This is how we can see and understand from the correlation plot.

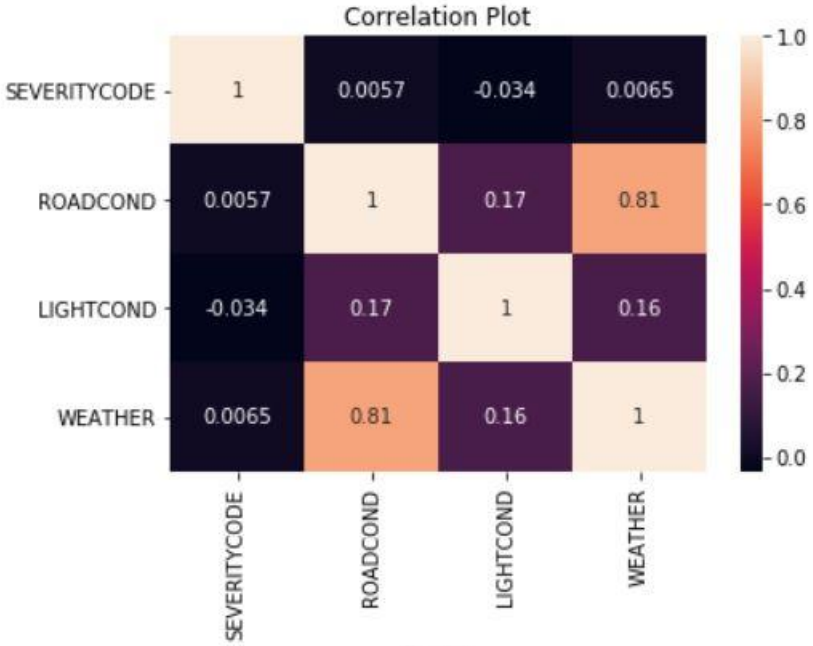

Fig 3.1

We have built models using different classifiers of which one of them is KNN. So, to build that we first find the best suitable value of 'K'. This is done by running the model for different values of K and then finding the accuracy for each value.

So, below are the plots of accuracy of model for different values of k in imbalanced, random under sampling and random over sampling situations. These plots tell us which vale of K then we used to build our model to have the greatest accuracy.

Imbalanced dataset

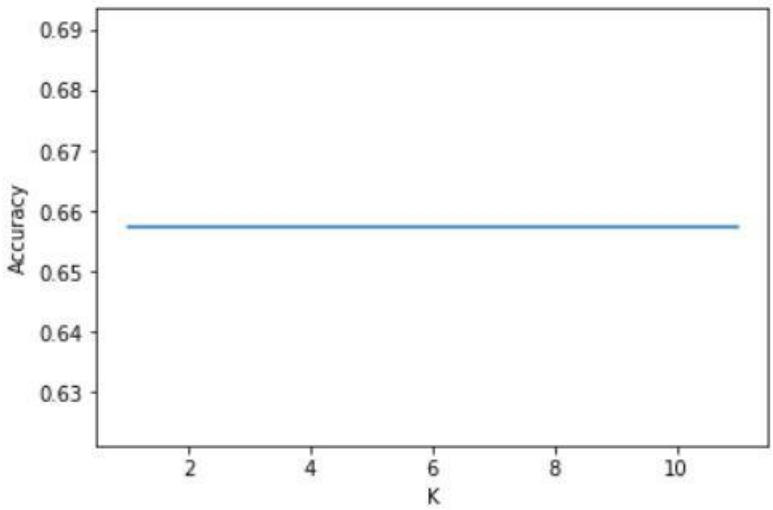

Fig 3.2

We can observe that for all values of K, it is predicting with the same accuracy. This is because the data is heavily in favour of one class and later in the results section, we can see from the classification report and confusion matrix that it is predicting only class 1.

Random Under Sampling

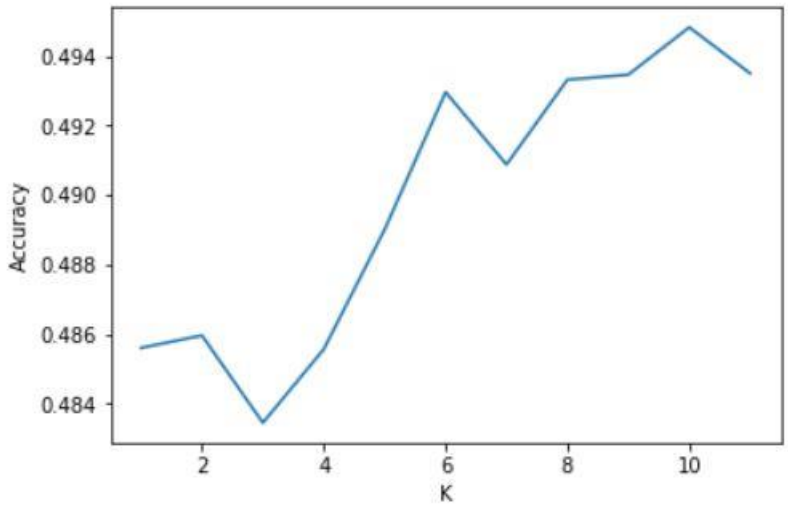

Fig 3.3

Here we can see the best value of K is obtained at K=10 and then the model is built on K=10 to find the classification report, confusion matrix and accuracy.

Random Over Sampling

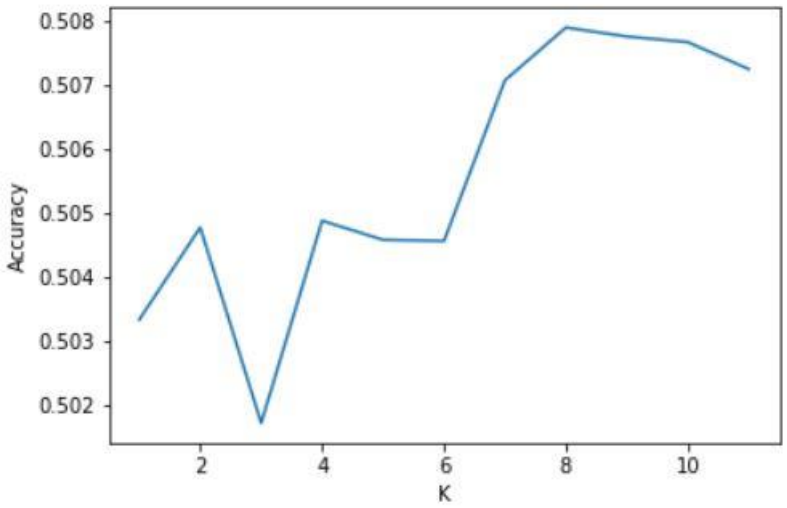

Fig 3.4

In this case the best value of K is at K=8

4. RESULTS

We have built models on balanced as well as imbalanced data but we could not get accuracy greater than 70% in any case. Number of different models have been used to make our predictions.

When working with imbalanced data, there were a huge amount of skewness towards the class of greater data, that is, 'SEVERITYCODE'=1. So, the models were very partial towards that class and the results were not good. We can see it from the classification report and confusion matrix results that we obtained.

Imbalanced data confusion matrix, Classification report and accuracy results with different classifiers are as follows:

Decision Tree

```
[[28869     0]
 [13621     0]]
              precision    recall  f1-score   support

           1       0.68      1.00      0.81     28869
           2       0.00      0.00      0.00     13621

   micro avg       0.68      0.68      0.68     42490
   macro avg       0.34      0.50      0.40     42490
weighted avg       0.46      0.68      0.55     42490

0.6794304542245234
```

Fig 4.1

Random Forest Classifier

```
[[28869     0]
 [13621     0]]
              precision    recall  f1-score   support

           1       0.68      1.00      0.81     28869
           2       0.00      0.00      0.00     13621

   micro avg       0.68      0.68      0.68     42490
   macro avg       0.34      0.50      0.40     42490
weighted avg       0.46      0.68      0.55     42490

0.6794304542245234
```

Fig 4.2

Logistic Regression

```
[[28869     0]
 [13621     0]]
              precision    recall  f1-score   support

           1       0.68      1.00      0.81     28869
           2       0.00      0.00      0.00     13621

   micro avg       0.68      0.68      0.68     42490
   macro avg       0.34      0.50      0.40     42490
weighted avg       0.46      0.68      0.55     42490

0.6794304542245234
```

Fig 4.3

K Nearest Neighbor's Classifier

```
[[28869     0]
 [13621     0]]
              precision    recall  f1-score   support

           1       0.67      0.82      0.74     28869
           2       0.29      0.15      0.20     13621

   micro avg       0.61      0.61      0.61     42490
   macro avg       0.48      0.49      0.47     42490
weighted avg       0.55      0.61      0.57     42490

0.6064485761355614
```

Fig 4.4

Support Vector Machine

```
[[28869     0]
 [13621     0]]
              precision    recall  f1-score   support

           1       0.68      1.00      0.81     28869
           2       0.00      0.00      0.00     13621

   micro avg       0.68      0.68      0.68     42490
   macro avg       0.34      0.50      0.40     42490
weighted avg       0.46      0.68      0.55     42490

0.6794304542245234
```

Fig 4.5

We observe from confusion matrix that all the models are predicting in favour of 'SEVERITYCODE=1'. This is the reason why we will now proceed with balanced data.

To make the data balanced, we did it in two different ways: Under Sampling and Over Sampling.

But after making the data balanced and building our models with the balanced data, still the accuracy of prediction was not above 55%.

UNDER SAMPLING

Decision Tree

```
[[ 3597 10149]
 [ 3123 10973]]
              precision    recall  f1-score   support

           1       0.54      0.26      0.35     13746
           2       0.52      0.78      0.62     14096

   micro avg       0.52      0.52      0.52     27842
   macro avg       0.53      0.52      0.49     27842
weighted avg       0.53      0.52      0.49     27842

0.5233101070325408
```

Fig 4.5

Random Forest

```
[[11965  1781]
 [12284  1812]]
              precision    recall  f1-score   support

           1       0.49      0.87      0.63     13746
           2       0.50      0.13      0.20     14096

   micro avg       0.49      0.49      0.49     27842
   macro avg       0.50      0.50      0.42     27842
weighted avg       0.50      0.49      0.41     27842

0.49482795776165506
```

Fig 4.6

K Nearest Neighbor's

```
[[11965  1781]
 [12284  1812]]
              precision    recall  f1-score   support

           1       0.49      0.87      0.63     13746
           2       0.50      0.13      0.20     14096

   micro avg       0.49      0.49      0.49     27842
   macro avg       0.50      0.50      0.42     27842
weighted avg       0.50      0.49      0.41     27842

0.49482795776165506
```

Fig 4.7

Support Vector Machine

```
[[ 3623 10123]
 [ 3151 10945]]
              precision    recall  f1-score   support

           1       0.53      0.26      0.35     13746
           2       0.52      0.78      0.62     14096

   micro avg       0.52      0.52      0.52     27842
   macro avg       0.53      0.52      0.49     27842
weighted avg       0.53      0.52      0.49     27842

0.5232382731125638
```

Fig 4.8

After performing under sampling, we can observe that the highest accuracy is from the decision tree and support vector machine. It is about 52.3%.

But we can see from the confusion matrix that now it does not blindly predict one class only since the data is balanced.

OVER SAMPLING

Decision Tree

```
[[ 7627 20951]
 [ 6469 22090]]
              precision    recall  f1-score   support

           1       0.54      0.27      0.36     28578
           2       0.51      0.77      0.62     28559

   micro avg       0.52      0.52      0.52     57137
   macro avg       0.53      0.52      0.49     57137
weighted avg       0.53      0.52      0.49     57137

0.5201008103330591
```

Fig 4.9

Random Forest

```
[[11965  1781]
 [12284  1812]]
              precision    recall  f1-score   support

           1       0.49      0.87      0.63     13746
           2       0.50      0.13      0.20     14096

   micro avg       0.49      0.49      0.49     27842
   macro avg       0.50      0.50      0.42     27842
weighted avg       0.50      0.49      0.41     27842

0.49482795776165506
```

Fig 4.10

Logistic Regression

```
[[10065 18513]
 [ 9071 19488]]
              precision    recall  f1-score   support

           1       0.53      0.35      0.42     28578
           2       0.51      0.68      0.59     28559

   micro avg       0.52      0.52      0.52     57137
   macro avg       0.52      0.52      0.50     57137
weighted avg       0.52      0.52      0.50     57137

0.5172305161279032
```

Fig 4.11

K Nearest Neighbors

```
[[21808  6770]
 [21348  7211]]
              precision    recall  f1-score   support

           1       0.51      0.76      0.61     28578
           2       0.52      0.25      0.34     28559

   micro avg       0.51      0.51      0.51     57137
   macro avg       0.51      0.51      0.47     57137
weighted avg       0.51      0.51      0.47     57137

0.5078845581672121
```

Fig 4.12

So, the highest accuracy obtained after over sampling was with the decision tree model and it was about 52%.

But we can see from the confusion matrix that now it does not blindly predict one class only since the data is balanced.

5. DISCUSSION

Although the weather, road conditions and lighting conditions are initially expected to have a significant impact on accident severity, these low performances indicate that the WEATHER, ROADCOND and LIGHTCOND attributes are not sufficient. Useful attributes such as SPEEDING and INATTENTIONIND may have been very useful for these models as these are expected to be significant in more severe accidents, but too many null values rendered them unusable. Certain categories could have been grouped together such as the varieties in LIGHTCOND with variations of Dark. Additionally, location data could have been used to assist in narrowing the areas where accidents most occur to help clean some outlying data and allow drivers to be more aware of certain areas.
What could not be considered, due to the limitations of this dataset, was the contributions of these external conditions to the likelihood of an accident being caused rather than its severity. It can be stipulated that attributes such as the speed of the vehicle and the quantified inattention of the driver may have significantly contributed to the severity of accidents.

6. CONCLUSION

The accuracy of models was not satisfactory in any of our case. We expected that some insight could be taken out of the data by using the features that we selected but the accuracy of models is required to be high in order to predict something with assurance.
If a dataset is heavily imbalanced, we need to balance it as in the imbalanced data, we saw that the predictions were bound to be partial to one class.