Coursera IBM Data Science Capstone Project

Accident Severity Prediction Report

1) Introduction / Business Problem

Car accidents are one of the leading causes of death in the world, especially among people between the ages of 20 and 65. Unfortunately, many of them are caused by high speed driving, road problems, drug or alcohol use, or weather conditions. Analyzing the factors involved in past accidents provided you can make an accurate prediction of the severity of future accidents.

These predictions could be used by emergency services, hospitals to have emergency personnel and material available to adequately attend. It can also be used by financial services such as auto and medical insurance to analyze their demand for future services and establish marketing strategies.

In addition, this serious accident situation can be notified to the nearby hospitals that can have all the equipment prepared for a severe intervention in advance.

2) Data Understand and Preprocessing

To carry out this project, the data set provided by the course has been selected, which corresponds to accidents that occurred in the city of Seattle that include automobile, bicycle and motorcycle accidents.

Only 10,000 records will be selected out of the 194,674 randomly, as some algorithms take time and are not suitable for such a large data set. severitycode will serve as a predictor field while the others will be the variables including status, addrtype, severitydesc, collisiontype, personcount, vehcount, junctiontype sdot_coldesc, weather, roadcond, lightcond.

After of clean and select the random record the count of severity accidents is :

Severity Code	Count
1	6659
2	3341

1. Feature selection

The total of columns in the dataset is 37. The features selected for the problem are :

```
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   SEVERITYCODE    10000 non-null  int64
 1   ADDRTYPE        10000 non-null  object
 2   COLLISIONTYPE   10000 non-null  object
 3   PERSONCOUNT     10000 non-null  int64
 4   VEHCOUNT        10000 non-null  int64
 5   WEATHER         10000 non-null  object
 6   ROADCOND        10000 non-null  object
```

2. Identification and handling missing values

Null or empty data is a problem, it could produce expected results requiring a dataset cleanup. In this case, the rows containing null data will be eliminated. Values such as undefined will be changed to null and later the procedure for their elimination will be executed.

```
df.dropna(subset=['SEVERITYCODE', 'ADDRTYPE', 'COLLISIONTYPE',
'PERSONCOUNT',’VEHCOUNT', 'WEATHER', 'ROADCOND'],  axis=0,
inplace=True)
```

Later, 10,000 records will be randomly selected.

```
df = df.sample(n=10000,random_state=10)
```

3. Encoding of data

Some of the selected fields are categorical so it is necessary to convert them to numeric to apply the prediction algorithms. Using the following command it is possible to convert the categorical fields into numeric

```
X = pd.get_dummies(data=X, columns=[ 'ADDRTYPE', 'COLLISIONTYPE',
'WEATHER', 'ROADCOND'])
```

3.) Methodology

1. Splitting into training and testing datasets

To train the data, it is necessary to partition the data set into two, one for training and one for tests. For this project a percentage of 70% was used for training and 30% for testing :

```
X_train, X_test, Y_train, Y_test =
train_test_split(X,Y,test_size=0.3,random_state=0)
```

The X dataset prefix was for variable dataset and Y prefix dataset for severitycode, the field to predict.

2. Normalizing and scaling of data

The selected characteristics must be normalized or scaled, so that the data have the same ranges, have the same weights and are as objective as possible. It also helps improve the performance of Machine Learning algorithms.

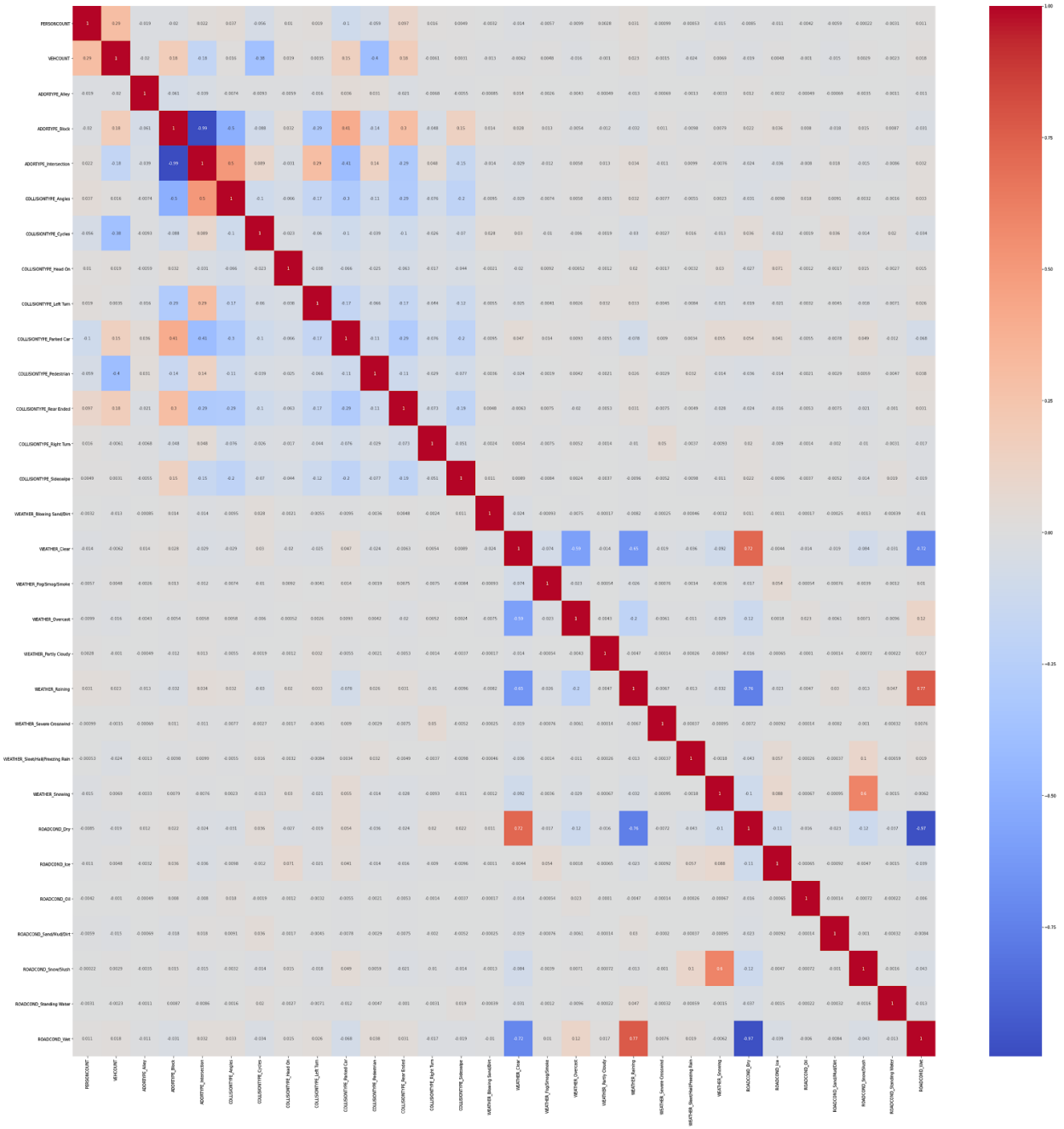
```
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)

scaler = StandardScaler()
X_test = scaler.fit_transform(X_test)
```

3. Understanding Correlation in Dataset

The correlation of the data is important since it allows to show which variables are related, their orientation and the degree of it. It is one of the most important statistical tools and helps to better understand the data.

In the data set of the problem, we can observe that there is a high correlation between the variables ROADCOND_Wet and ROADCOND_Snow since generally when it is snowing it is wet. Or for example entyre WEATHER_Clear and

COLLISIONTYPE_Angles a negative relationship with WEATHER_Clear since in general accidents do not happen with these characteristics. The correlation map between the data is shown below:



4. Machine Learning Algorithms

The algorithms that were used are Logistic Regression, KNN, Decision Tree, Random Forest, Naive Bayes and SVM. Each of them were tested, their performance was

obtained to determine which could be better to apply to the problem. The results were the following :

a. Logistic Regression

The Logistic Regression algorithm predicts the probability of occurrence of an event by fitting data to a logistic function that uses discrete values like 0-1, true-false given set of independent variables. The results of the confusion matrix and accuracy are :

```
[[1902  105]
 [ 687  306]]
```

	precision	recall	f1-score	support
1	0.73	0.95	0.83	2007
2	0.74	0.31	0.44	993
accuracy			0.74	3000
macro avg	0.74	0.63	0.63	3000
weighted avg	0.74	0.74	0.70	3000

```
0.736
```

b. Naive Bayes

The Naive Bayes classifies objects based on Bayes' Theorem with an assumption that the predictors (features) are independent of each other. The results of the confusion matrix and accuracy are :

```
[[2007    0]
 [ 993    0]]
```

	precision	recall	f1-score	support
1	0.67	1.00	0.80	2007
2	0.00	0.00	0.00	993
accuracy			0.67	3000
macro avg	0.33	0.50	0.40	3000
weighted avg	0.45	0.67	0.54	3000

```
0.669
```

b. K Nearest Neighbours

The K nearest neighbours algorithm can be used for both classification and regression problems. For this problem will be used for classification. This algorithm finds the K nearest neighbours based on initial points using the measured distance as Euclidean, Manhattan, Minkowski, or Hamming functions. The results of the confusion matrix and accuracy are :

```
Best Hyperparameter KNN :  {'n_neighbors': 6, 'p': 2}
[[1701  306]
 [ 569  424]]

              precision    recall  f1-score   support

           1       0.75      0.85      0.80      2007
           2       0.58      0.43      0.49       993

    accuracy                           0.71      3000
   macro avg       0.67      0.64      0.64      3000
weighted avg       0.69      0.71      0.70      3000

0.7083333333333334
```

c. Decision Tree

Decision Tree algorithm makes decisions with a tree-like model based on a differentiator or predictor that chooses from the data based on the most significant in the input variables and repeats recursively until it successfully splits the data in all leaves. The results of the confusion matrix and accuracy are :

```
Best Hyperparameter DTC :  {'criterion': 'entropy', 'random_state': 0}
[[1785  222]
 [ 622  371]]

              precision   recall  f1-score   support

           1       0.74      0.89      0.81      2007
           2       0.63      0.37      0.47       993

    accuracy                           0.72      3000
   macro avg       0.68      0.63      0.64      3000
weighted avg       0.70      0.72      0.70      3000

0.7186666666666667
```

d. Random Forest Tree

Random Forest Classifier is a tree-based learning algorithm. Is a set of decision trees from a randomly selected subset of training set that aggregates the votes from different decision trees to decide the final class of the test object. Is used for both classification and regression but in this problem will be used for classification. The results of the confusion matrix and accuracy are :

```
Best Hyperparameter RFT :  {'criterion': 'gini', 'n_estimators': 50,
'random_state': 0}
[[1776  231]
 [ 605  388]]

              precision    recall  f1-score   support

           1       0.75      0.88      0.81      2007
           2       0.63      0.39      0.48       993

    accuracy                           0.72      3000
   macro avg       0.69      0.64      0.65      3000
weighted avg       0.71      0.72      0.70      3000


0.7213333333333334
```

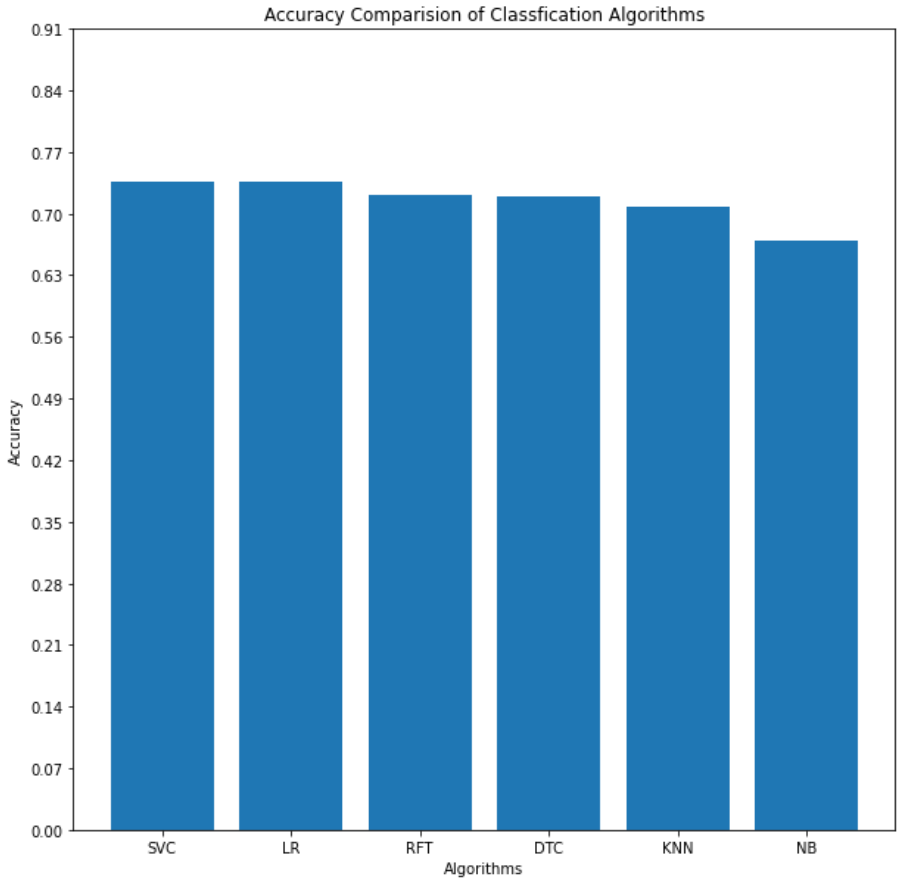
e. Support Vector Machine

Support Vector Machine is an algorithm which can be used for both classification and regression challenges. In the SVM algorithm, each data item is plotted as a point in n-dimensional space based on the number by each feature. The classification is performed by finding the hyper-plane that differentiates the two classes. Is used for both classification and regression but in this problem will be used for classification. The results of the confusion matrix and accuracy are :

```
Best Hyperparameter SVM :  {'kernel': 'rbf', 'random_state': 0}
[[1926   81]
 [ 710  283]]

              precision    recall  f1-score   support

           1       0.73      0.96      0.83      2007
           2       0.78      0.28      0.42       993

    accuracy                           0.74      3000
   macro avg       0.75      0.62      0.62      3000
weighted avg       0.75      0.74      0.69      3000


0.7363333333333333
```

4) Results

All the algorithms implemented above gave an accuracy score equal to or greater than 0.7. For show the result of the accuracy of the algorithms can observed the next diagram. The bars represent the accuracy of each model in descending order respectively.



Kernel Support Vector Machine and Logistic Regression are the best classifier based on accuracy of 73%. Too, Random Forest and Decision Tree show a god accuracy with 71-72%.

5) Discussion

The performance of the algorithms is even but it is not enough to adequately predict the severity of the accidents with the selected characteristics since at least 80% is required and the ones that had the best performance were a little lower. A study is needed to improve the performance of the algorithms.

6) Conclusion

The model requires adjustments to achieve at least 85%. This can be accomplished by selecting more data, searching for more scenarios, or experimenting with more variables.

There was a lot of data with null or bad data, some null features were removed and perhaps the 10,000 records were not enough in this case. A data science project, especially those that involve machine learning, require a repetitive process to fine tune the predictive method.