Car accident severity model for Seattle

Applied Data Science Capstone by IBM/Coursera

Marta Ferreira

1. Business Understanding:

1.1 Background of the study

The usage of car has increased, especially in past few decades since the 60's, as the price of the cars are more affordable to the open public. This factor can present a direct correlation with the increase number of car accidents. Conditions of road, light, weather and human factors (such as speeding or alcohol intake), are some of the many factors that can influence a car accident. But when we need to predict the severity of a car crash, that is not an easy job.

To try to create a predictive model, this time we will be focusing our attention to Seattle, a seaport city on the West Coast of the United States of America (USA), with around 3.98 million habitants in the Metropolitan area and 750.000 in the city. A city and metropolitan area in constant growth and development. According to the 2017 Washington State Department of Transportation (WSDOT) data, a car accident occurs every 4 minutes and a person dies due to a car crash every 20 hours. Fatal crashes went from 508 in 2016 to 525 in 2017, resulting in average in the death of 555 people. This number has stayed relatively steady for the past decade.

1.2 Problem and Interested parties

To help reduce the severity and frequency of car collision, this project wants to use the Seattle car collision data to generate insights on how modelling can help reduce accidents.

The objective of this project is to be able to predict the severity of a road accidents under certain conditions like weather, road and visibility.

It can be used for specific resources, like municipal police or emergency teams (fire fighters or hospitals), as a way to predict the flux of accidents that they can expect in a specific day. But it can also be used by the public, from the common car drivers or a to the pedestrian. In the future it can be useful to predict the location of the most common places of an accident to eld place and help the government officials to improve roads or be more watchful to certain roads. By looking into road condition factors and address types such as intersection and block, we could see if there is any possible improvement in road condition and city planning.

2. Data understanding:

2.1 Data Source

The Data used in the following project is the Car Collision dataset of the city of Seattle, from IBM Cloud which contains among others the collision data, address, and a few different conditions such as Weather, road and light. The dataset includes 38 columns and 194673 observations.

2.2 Data cleaning

The first objective of the Dataset is going to be cleaning the “noise” or remove all the columns that we are not going to require for the base of this specific study. It will be need to have in regard the missing data, which there are about 10527 records in this category.

The target variable will be 'SEVERITYCODE', as it is a measure of the severity of the accident and varies between 1 and 2. The attributes 'WEATHER', 'ROADCOND' and 'LIGHTCOND' are going to be converted from Categorical to Numerical values, as part of the normalization process.

Values for SEVERITYCODE represent the following consequences of the accident: Property Damage Only Collision and Injury Collision

Lastly, we need to have in consideration that this is unbalanced labelled dataset, so we are going to need to normalize the data first of all, before they being used.

2.3 Feature selection

The objective is going to try to considered as well the factor 'SPEEDING', to use as a dependent regarding the Weather and Road conditions. The attributes 'WEATHER', 'ROADCOND' and 'LIGHTCOND' will be used to weigh the severity of an accident. But it is going to be considered the ‘INCDTTM’ (date and time of the accident) to see if we can get more into a specific date.

3. Methodology:

To complete this project, Github was used as a repository and Jupyter Lab to process the data and build in the Machine Learning model. After getting the needed libraries, such as Pandas, NumPy, Matplotlib and Sklearn, the dataset was uploaded.

The first step was to get to know the dataset and know which attributes were the best to use in our study. As already previously mentioned, to be able to predict the severity of accidents we decided to go with the attributes Weather, Light and Road condition as the main attributes to compare. We got a few more attributes at the table as stated ('SPEEDING', ‘INCDTTM’, ‘ST_COLCODE’).

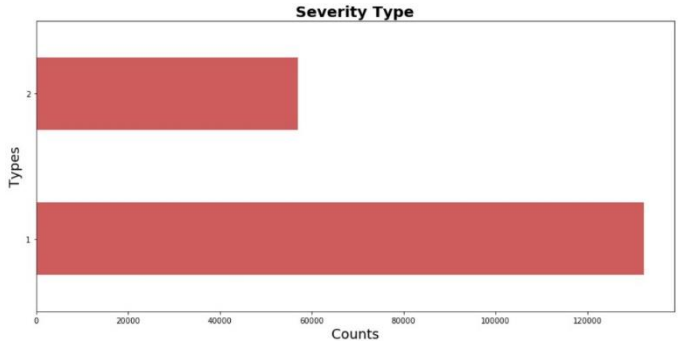
```
col = ["SEVERITYCODE", "INCDTTM", "WEATHER", "ROADCOND", "LIGHTCOND", "VEHCOUNT", "PERSONCOUNT", 'PEDCOUNT', 'PEDCYLCOUNT']
df_ac = df[col]
df_ac[:5]
```

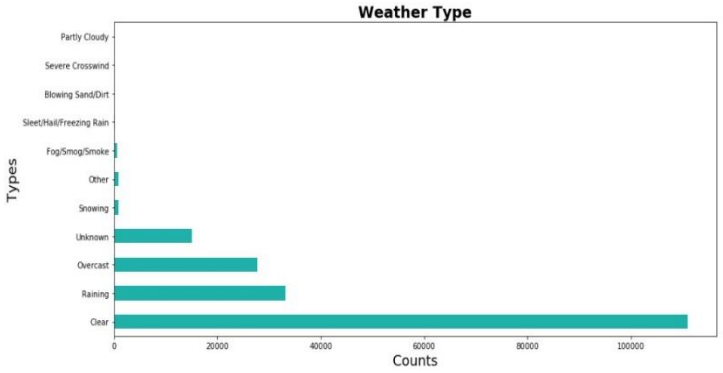
	SEVERITYCODE	INCDTTM	WEATHER	ROADCOND	LIGHTCOND	VEHCOUNT	PERSONCOUNT	PEDCOUNT	PEDCYLCOUNT
0	2	3/27/2013 2:54:00 PM	Overcast	Wet	Daylight	2	2	0	0
1	1	12/20/2006 6:55:00 PM	Raining	Wet	Dark - Street Lights On	2	2	0	0
2	1	11/18/2004 10:20:00 AM	Overcast	Dry	Daylight	3	4	0	0
3	1	3/29/2013 9:26:00 AM	Clear	Dry	Daylight	3	3	0	0
4	2	1/28/2004 8:04:00 AM	Raining	Wet	Daylight	2	2	0	0

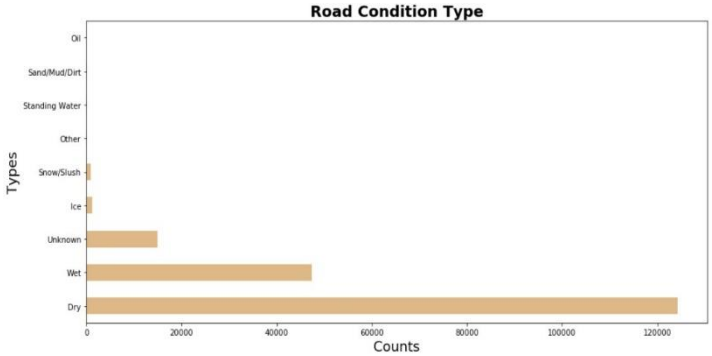
So, we got the preferred attributes into a separated Dataframe and started to work. Getting the grasp that we were dealing with an unbalanced dataset the immediately step was to normalized the dataset. Since we noticed that the attribute for Date (‘INCDTTM’) was not right, we needed to putt all the rows in the same format (month/year).

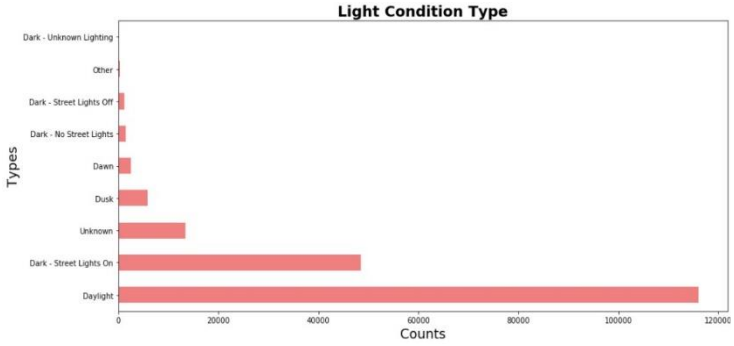
After normalizing the data and removing the Null or NaN data we were able to check a few aspects that changed the direction of the study. Regarding the Date and time of accident we noticed during the processing of this attribute that not every field had all the required attributes to this, so we decided only to represent the number of accidents per year, and this is not going to be considered to the model. We noticed as well that the data for the Speeding we didn't had a lot of information, so we discarded this attribute for now.

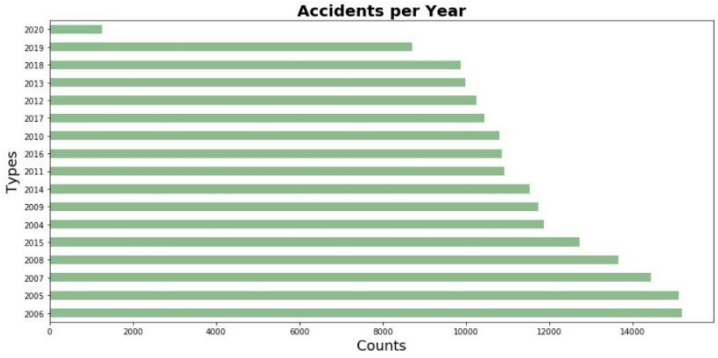
We can try to have a little scoop in the attributes that we want to use to compare with the Severity of the accident. For this aspect we will use the raw data, to represent the three main attributes on this case study Weather, Road and Light conditions.











The first graph gives a visualization of the accident severity type, when 2 represents "Injury", with more than 57.000 accidents, and 1 is the total with "Propriety damage", more than 132.000. Unfortunately, the data is not provided with a most deeply on other types, for example fatalities. For the Weather type, normally, it is expected that the worst types of weather will cause more accidents, but most of the accidents occurred with Clear

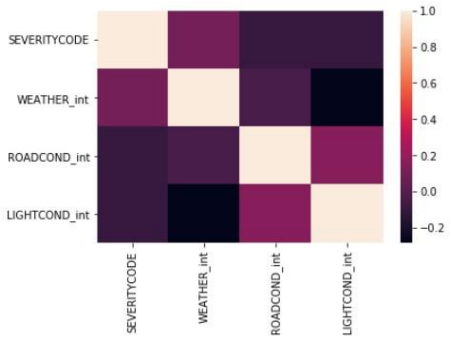
weather (111.000 accidents). Following it by 33.000 accidents in raining conditions. When we observed the "Road condition" we see that mainly the accidents happened with a Dry floor (124.000). Regarding the "Light conditions" the largest group represent that the accidents happened by Day. Ending the representation of the three main attributes in study, we can see that the accidents happened mostly during the day, with dry roads and with clear sky. Now we will need to process the rest of the information to be able to see the correlation between accident severity and the conditions represented above. Lastly, representing the total number of accidents per year, we can observe that since 2006 till 2019 (not considering 2020 since the year is not ended) it occurred a steady decrease on the number of accidents. Even so, compared with 2006 with more than 14.000 accidents, the year with less accidents (2019) presents only around 9.000 accidents.

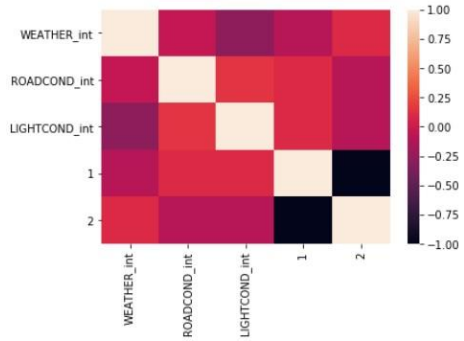
3.1 Analysis:

We will start now the analyses of the elements. We will start with saving the attributes that we want to study in a specific Dataframe and from there we will create the Machine Learning model. So first we went with the One Hot Encoding process, that made possible to put all attributes in a numerical value to be easier to work with.

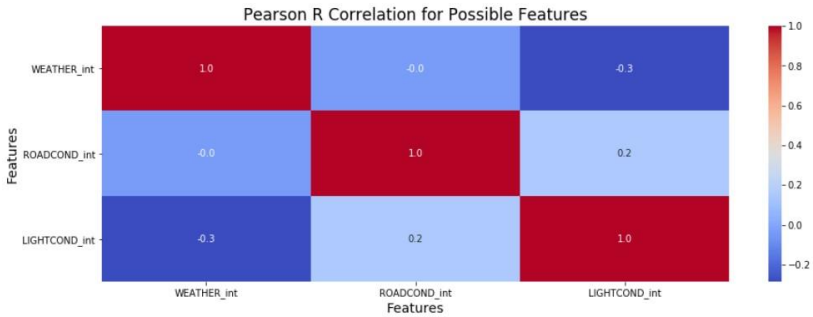
	WEATHER_int	ROADCOND_int	LIGHTCOND_int	1	2
0	1	1	1	0	1
1	2	1	2	1	0
2	1	2	1	1	0
3	3	2	1	1	0
4	2	1	1	0	1

Just for fun, decided to see the correlation between each attribute. First with Severity code all together, and then the Severity code separated in the two types of accident severity, using the One Hot Encoding technique.





Well, it seems that Road and Light conditions have the greater correlation with the accident severity variable and surprisingly looks like the same attributes are important for both types of accident severity, Road and Light conditions. Other variable that have great correlation in both graphs is the Light and Weather condition.



As a last form of representation, decided to use the Pearson R correlation to represent the correlation between the variables to study face the Severity attribute.

3.2 Prediction

First step was to get the Panda Dataframe into a Numpy array, to be able to get the model running. For this we have to consider the X and Y variable.

X - feature vector ("WEATHER_int", "ROADCOND_int" and "LIGHTCOND_int")
y - predicted variable ("SEVERITYCODE")

We decided to use four types of models to run, to see which one provides a better answer to our issue: Decision Tree, K Nearest Neighbour (KNN), Logistic Regression and Support Vector Machine (SVM).

Some of the aspects to have in consideration regarding the models in use:

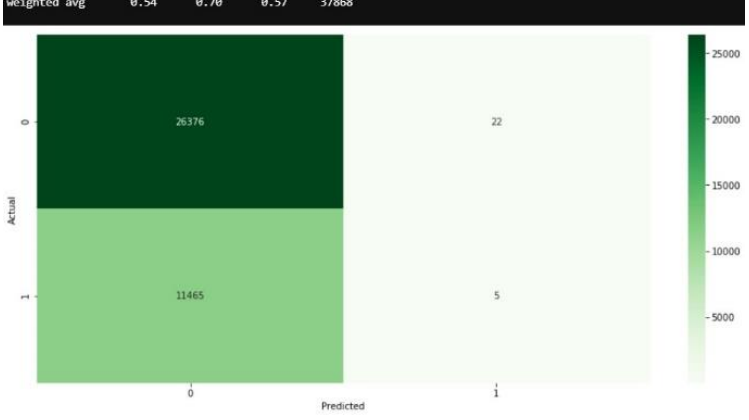
- Precision quantifies the number of positive class predictions that actually belong to the positive class.

- Recall quantifies the number of positive class predictions made out of all positive examples in the dataset.

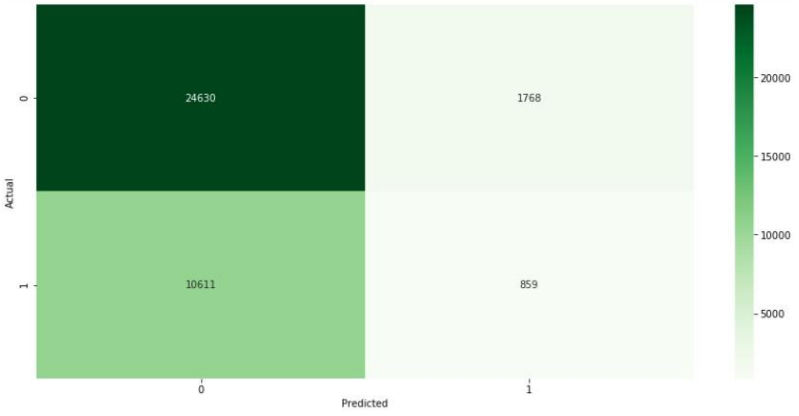
- F-Measure provides a single score that balances both the concerns of precision and recall in one number.

- Jaccard Index compares members for two sets to see which members are shared and which are distinct. It's a measure of similarity for the two sets of data, with a range from 0% to 100%. The higher the percentage, the more similar the two populations.

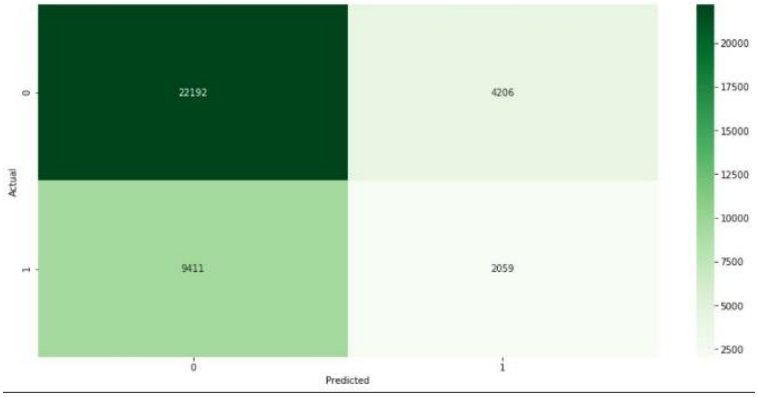
```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=None,
            splitter='best')
[[26376    22]
 [11465     5]]
              precision    recall  f1-score   support

           1       0.70      1.00      0.82     26398
           2       0.19      0.00      0.00     11470

   micro avg       0.70      0.70      0.70     37868
   macro avg       0.44      0.50      0.41     37868
weighted avg       0.54      0.70      0.57     37868
```




```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
           metric_params=None, n_jobs=None, n_neighbors=4, p=2,
           weights='uniform')
[1 1 1 1 1]
              precision    recall  f1-score   support

           1       0.70      0.93      0.80     26398
           2       0.33      0.07      0.12     11470

   micro avg       0.67      0.67      0.67     37868
   macro avg       0.51      0.50      0.46     37868
weighted avg       0.59      0.67      0.59     37868
```



```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='warn',
          n_jobs=None, penalty='l2', random_state=None, solver='warn',
          tol=0.0001, verbose=0, warm_start=False)
              precision    recall  f1-score   support

           1       0.70      0.84      0.77     26398
           2       0.33      0.18      0.23     11470

   micro avg       0.64      0.64      0.64     37868
   macro avg       0.52      0.51      0.50     37868
weighted avg       0.59      0.64      0.60     37868
```




```
SVM Jaccard index: 0.70
SVM F1-score: 0.57
```

We can define the used models in the following form:

Decision Tree: According to the website TowardsDataScience.com "a Decision Tree is a simple representation for classifying examples. It is a Supervised Machine Learning where the data is continuously split according to a certain parameter. Decision Tree consists of: Nodes, Edges/ Branch, Leaf nodes".

K Nearest Neighbor (KNN): The KNN model is a supervised ml, where the data are 'trained' with points that correspond to their classification. Once a point is predicted, the model takes the nearest 'Kth' points to determine classification.

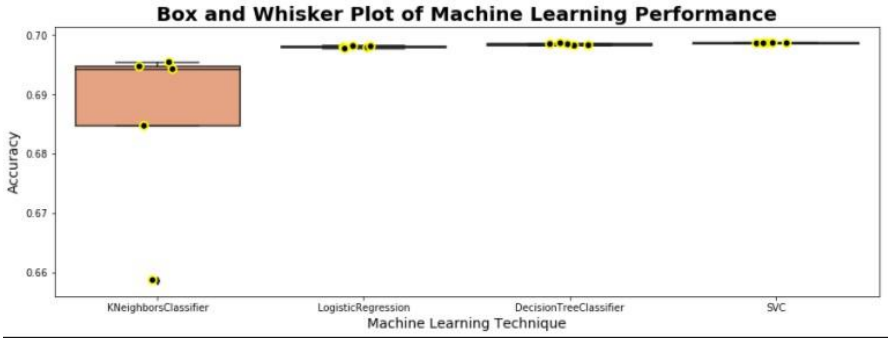
Logistic Regression: In the Statistics Solutions website, we can observe Logistic Regression as "the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables."

4. Results:

After running the model with the four type of processes we were able to get these results, with respective visual help:

```
Jaccard and F1 Scores
        Decision Trees Scores:
                f1 score: 0.5727151085624445
                jaccard score: 0.6966568078588782

        KNN Scores:
                f1 score: 0.5940192367374898
                jaccard score: 0.6731012992500264

        Logistic Regression Scores:
                f1 score: 0.5940192367374898
                jaccard score: 0.6731012992500264

        Support Vector Machine:
                f1 score: 0.572688426582199
                jaccard score: 0.6971057357135312
```

```
model_name
DecisionTreeClassifier    0.698437
KNeighborsClassifier      0.685582
LogisticRegression        0.697983
SVC                       0.698648
Name: accuracy, dtype: float64
```



In the results above, we used Decision Tree, KNN, Logistic Regression and Support Vector Machine to attempt to solve our question for the local government officials of Seattle, Washington. After testing the K-Neighbor that gave us the value of 4, the accuracy off the model resulted in 68.5% predicting the severity level. However, Support Vector Machine had a of 70% accuracy, compared to the Decision Tree which accuracy was at 69.8% compared to Logistic Regression, which was at 69.7% accuracy of predicting traffic severity levels in Seattle.

We could see that the SVM model and the Decision Tree model were the ones with the higher scores and, for that, one of them should be considered to be use as a form of predictable model in the current case study.

5. Discussion:

The most important observation has to be made with the dependent variable like SEVERITYCODE. Which indicates that there is no serious injury or fatality occurred. The SEVERITYCODE data signal that either somehow the data has been altered at the time of dataset creation or the sample data is incomplete and other important information has been missed from the report. Once the data for SEVERITYCODE has been corrected then different Machine Learning models can be trained to predict the severity of the accident.

The results from the different models, as we can see from the results presented, are not very exceptional, one can say that are very mediocre. Based on this we have to build and train the Machine Learning models for different conditions. For example, speed of vehicle, state of driver (ex. intake of alcohol or emotional state), etc. to predict the severity of the accident in a more accurate way.

One of the aspects that could be study in a near future is the location of the accidents. It will be interesting to know if we have any correlation between the location and the severity of the accident. This could help the local government in the decision of road improvement and / or better road signalization, light conditions, among other variables.

6. Conclusion:

The main objective of this project was to alert the stakeholders about the severity of an accident so that they can take preventive measures accordingly. And even if the prediction of car accident severity is not completely finished, we have a small grasp on the subject. To have a complete work and a strong model, we will need to collect more data and train more the model and we definitely need more information on the type of severity of the accident, since the two presented factors are not enough to base decisions.

We create a Machine Learning model based on classification algorithms to predict traffic accident severity in Seattle. The recommended model to be use varies between the Support Vector Machine or the Decision Tree model, for prediction of severity levels, since they presented the highest accuracy levels.

We can conclude that particular conditions (example weather, road, light) have a somewhat impact on whether or not travel could result in different accident severities.

The final decision of the factor to have in consideration on the subject of car accident severity will be made by stakeholders based on the aspect they want to base their action.

7. Some References:

Jaccard Index - Statisticshowto.com (2020) - https://www.statisticshowto.com/jaccard-index/

Decision Tree Classification - TowardsDataScience.com (2020) - https://towardsdatascience.com/decision-tree-classification-de64fc4d5aac

Linear Regression - Statisticssiolutions.com (2020) - https://www.statisticssolutions.com/what-is-logistic-regression/

https://medium.com/predict-car-accident-severity-in-seattle-with/predict-car-accident-severity-in-seattle-with-machine-learning-826806840ee7

https://www.programmersought.com/article/35945378649/