Accident Severity Prediction

Pradiv Gnanaraj

28/10/2020

1. Introduction

1.1 Background

Wheels getting invented during the 3,500 B.C from caveman technology to 1886 when Carl Benz applied for vehicle powered by gas engine until today with Self driven cars, travel becomes unavoidable. With the increased number of vehicles used for road transport, travelling has become more frequent. Resulting in accidents from time to time. The number of Fatal cases seems to increase in number every year. The road safey team from every country have passed laws and rules to reduce death by road accident. However, this seems to be still a problem.

1.2 Problem

Data that might include another point of view and possibly a solution to reduce the number of deaths is what required to be created and following by every government. Data which includes the road, weather condition, light conditions can be learnt which you be possible method to reduce the accidents. This project aims to predict the natural conditions to reduce the accident severity based on these data.

1.3 Interest

How well this model could be if we can know the possibility in predicting the accident before we could travel. This will safe life and definitely will be implemented by governments.

2. Data acquisition and Cleaning

2.1 Data Sources

In order to work with data which has already collected these information, I found and good dataset which was uploaded to Kaggle as two files. First file itself for the dataset and the second file was a key to the true dataset, explaining the attributes and values more in detail.

Primarily Captures Road Accidents in UK between 1979 and 2015 and has 70 features/columns and about 250K rows.

2.2 Data Cleaning

There were number of items to be checked with the uploaded dataset. I started with finding the attributes and get a statistical summary of all numeric-typed attributes. Then, I wanted to ensure that the data is converted to a format which is best understand during analysis. Prior goal during this process is to : handle all the missing values, correct the data format, standardize and normalize the data.

I created a small script – a python for loop to find the total number of missing data. The missing data was quite large nearing 14,06,982 belonging to various attributes. All of the missing data in the dataset were of dtype:int64 this lead to dealing with the missing data.

I could split the missing data into following types,

- Data to be dropped by Row.
- Data to be dropped by whole column.
- Replace the data with mean.
- Replace the data with frequency.
- Replace the data based on its other functions.

Dropped Values	Replaced Values	Reasons
location_easting_osgr, location_northing_osgr, longitude, latitude, time		For the following set of columns the missing rows are very less, therefore we will drop the following rows
	casualty_class, sex_of_casualty, age_of_casualty, age_band_of_casualty, casualty_severity, pedestrian_location, pedestrian_movement,	Replaced the missing values with the mean
	casualty_class, bus_or_coach_passenger, pedestrian_road_maintenance_worker , casualty_type, casualty_home_area_type	Replaced the missing values with the mean
	casualty_imd_decile	Replaced the missing values with the mean
casualty_reference		drop this column as we have

	much clearer data from the casualty_severity column
lsoa_of_accident_location	drop this column as we are focusing on the accident serverity.

3. Indicator Variable(s)

To use categorical variables for regression analysis in the later part of model development

Selecting only the required columns for the serveirty analysis.

Splitting the variables in to three variables categories

- Accident Circumstances
- Vehicle type
- Casualty type Casualty

Before starting to understand the (linear) relationship between an individual variable and the target variable, I worked with descriptive statistical analysis. And also examining the value counts for these attributes could be a good predictor.

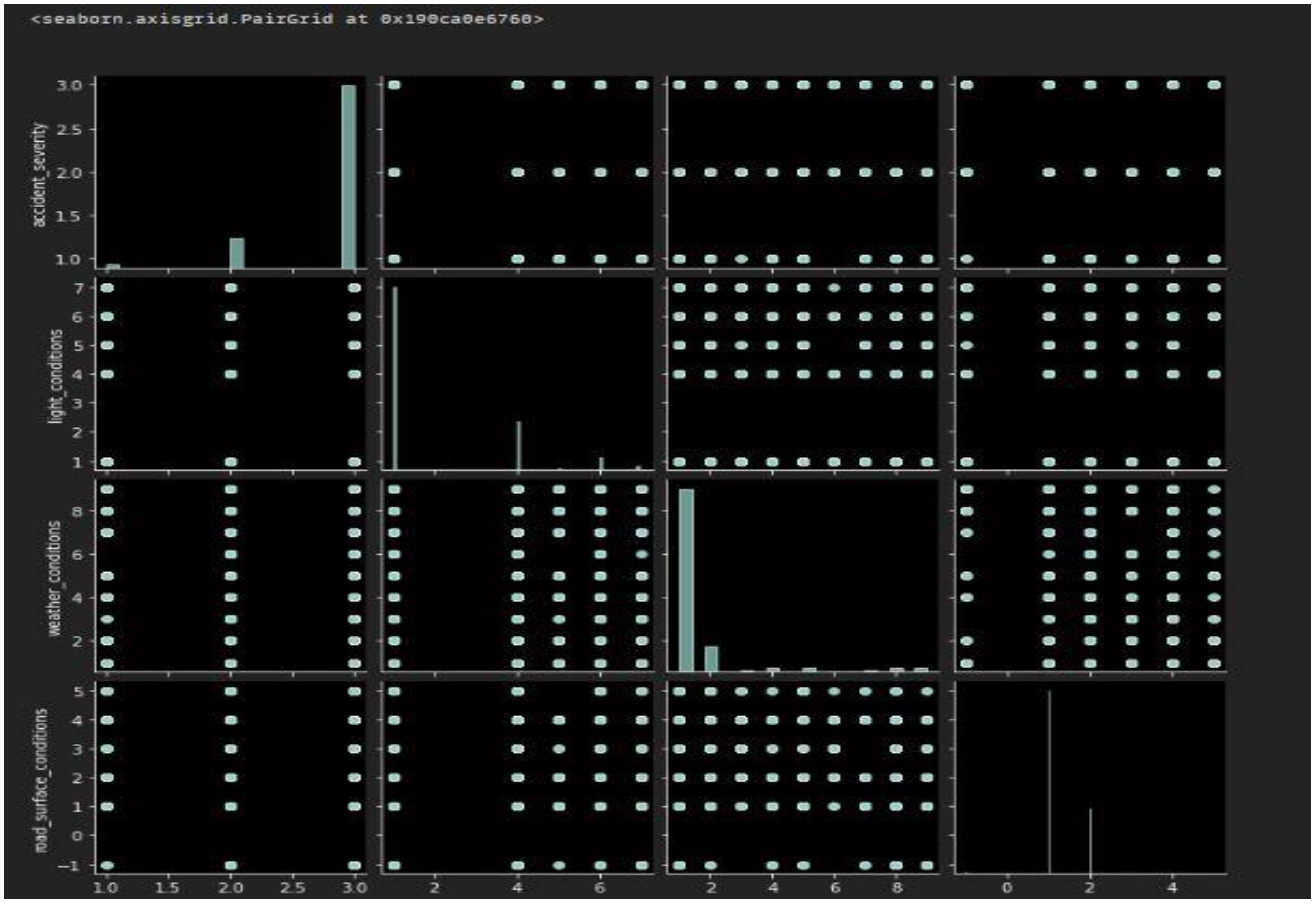
Before starting to feed the data to the model (to be developed). Finding the Pearson Correlation Coefficient and the P-value between attribute(s) and the target to find the linear distribution.

Working with the indicator variable and the dataset, the number of dropped attributes had to be increased in order to fulfill the course requirements. As a result, predictive value for the context of this project are already given, therefore the attributes with no predictive dataset were dropped.

3.1 Data Preparation:

The data preparation includes all the required activities to construct the final dataset which will be fed into the modeling tools. Data preparation can be performed multiple times and it includes balancing the labeled data, transformation, filling missing data, and cleaning the dataset.

The final dataset was prepared using the attributes such as light condition, weather condition and road surface condition. As mentioned earlier the null values were checked using a python statement to fulfill the dataset requirements.



4. Modelling

We now have a basic idea about the data. We need to extend that with visualizations. In this phase, various algorithms and methods can be selected and applied to build the model including supervised machine learning techniques. I have selected decision tree, KNN , Linear Regression and . At this phase, stepping back to the data preparation phase was often required.

Methodology

Our data is now ready to be fed into machine learning models.

We will use the following models:

K-Nearest Neighbor (KNN) : KNN will help us predict the severity code of an outcome by finding the most similar to data point within k distance.

Decision Tree : A decision tree model gives us a layout of all possible outcomes so we can fully analyze the consequences of a decision. It context, the decision tree observes all possible outcomes of different weather conditions.

Logistic Regression: Because our dataset only provides us with three severity code outcomes, our model will only predict one of those two classes. This makes our data binary, which is perfect to use with logistic regression.

5. **Evaluation**:

Before proceeding to the deployment stage, the model needs to be evaluated thoroughly to ensure that the business or the applications' objectives are achieved. Certain metrics can be used for the model evaluation such as accuracy, recall, F1-score, precision, and others.

```
0.28604651162790695
0.5530594098058867
0.32684824902723736
0.3037518037518038
```

	Algorithmn	Jaccard	F1-Score
0	KNN	0.286047	0.553059
1	Decision Tree	0.326848	0.584531
2	Logistics Regression	0.303752	0.545313
3	Random Forest	0.327132	0.678183

With no doubt the Random Forest is the best model, in the same time as the log. res. It improves the accuracy from 0.28 to 0.32 and the f1-score from 0.545 to 0.678.

Random forests consist of multiple single trees each based on a random sample of the training data.

They are typically more accurate than single decision trees.

The above result shows the decision boundary becomes more accurate and stable as more trees are added.

6. Discussion :

In the beginning of this notebook, we had categorical data that was of type 'object'. This is not a data type that we could have fed through an algorithm, so label encoding was used to created new classes that were of type int; a numerical data type.

After solving that issue we were presented with another - imbalanced data. As mentioned earlier, class 1 was nearly eighty times larger than class 3. The solution to this was down sampling the majority class with sklearn's resample tool. We down sampled to match the minority class exactly with each other.

Once we analyzed and cleaned the data, it was then fed through three ML models; K-Nearest Neighbor, Decision Tree and Logistic Regression. Although the first two are ideal for this project, logistic regression made the most sense because of its binary nature.

Evaluation metrics used to test the accuracy of our models were jaccard index, f-1 score and logloss for logistic regression. Choosing different k, max depth and hyperamater C values helped to improve our accuracy to be the best possible

7. Conclusion:

After building and comparing various classification models to predict the severity of the accident. I analyzed the relationship between severity of an accident and characteristics which involved conditions like road, light and weather to see the gravity of the accident.

Severity of an accident can be predicted in real time by using above data when an accident is reported and from there measures can be taken.