Predicting the Severity of Car Accidents

Dimitris Manolakis

October 13, 2020

Table of Contents

1. Introduction 3
1.1 Background 3
1.2 Problem 3
1.3 Interest 3
2. Data acquisition and cleaning 4
2.1 Data sources 4
2.2 Data cleaning 4
2.3 Feature selection 5
3. Exploratory Data Analysis 7
3.1 Relation of Accidents with Day of the Week 7
3.2 Relation of Accidents with Hour of the Day 8
3.4 Relation of Accidents with Road Type 9
3.3 Relation of Accidents with Light Conditions 10
3.5 Relation of Accidents with Weather Conditions 13
3.6 Relation of Accidents with Road Surface Conditions 14
3.7 Relation of Accidents with Area Type 15
3.8 Relation of Accidents with Vehicle Type 17
3.9 Relation of Accidents with Driver's Age 18
3.10 Relation of Accidents with Vehicle Age 19
3.11 Association of each Feature with Accident's Severity 20
4. Predictive Modeling 21
4.1 Classification models 21
4.2 Applying standard algorithms and their problems 21
4.3 Solution to the problem 23
4.3.1 Resampling 23
4.4 Performances of different models 24
5. Conclusions 24
6. Future directions 25
7. References 25

1. Introduction

1.1 Background

The effective treatment of road accidents and thus the enhancement of road safety is a major concern to societies due to the losses in human lives and the economic and social costs. In USA for 2016, NHTSA[1] data shows 37,461 people were killed in 34,436 motor vehicle crashes, an average of 102 per day. Over the years, tremendous efforts have been made by governments, transportation researchers and practitioners in order to improve road safety. Creating a model that predicts accidents and their severity can be advantageous for many reasons, as road design can be even more optimized, drivers can be informed on the danger they are facing based on their personal information and the current environmental conditions.

1.2 Problem

A number of factors contribute to the risk of collisions, including vehicle design and type, road design and environment, driving skills, as well as weather and light conditions. This project aims to predict the severity of a possible accident based on factors that are set before the accident happen like the driver's age, the area the driver is currently at, including the light conditions and the time (hour) of the day.

1.3 Interest

Government organizations that administrate road safety would be interested in accurate prediction of accidents and its severity, like the

[1] National Highway Traffic Safety Administration (NHTSA)

National Highway Traffic Safety Administration. The project can be expanded into an application that uses 'live' data on a driver's information, location and environmental conditions and warn him/her to slow down or even use other safer paths to his/her destination.

2. Data acquisition and cleaning

2.1 Data sources

The data used on the project comes from a dataset called UK Car Accidents. It is collected from the UK Department for Transport and can be found on Kaggle. The dataset starts at 2005, ends at 2015 and is composed by three different csv files (Accidents, Causalities, and Vehicles). It was selected because it offered a great amount of variables (features) for each accident like personal information on the driver, environmental conditions and vehicle information. Also, the accidents were classified into three types of severity (1. Fatal, 2. Serious, 3. Slight) which gave promise for a model with more specific predictions than most datasets that offered binary classification for the accidents severity.

2.2 Data cleaning

The 3 different csv files were joined into one data frame using the common column 'Accident Index'. The data was in a good condition as most of the variables were already categorical variables (integers) and there were few null values (172 null values for the 'Time' column only).

Multiple columns such as 'Weather Conditions', 'Road Surface Conditions' and others, included Unknown or Other values marked as -1 or 9 accordingly. All the rows that had either value was removed from the data.

Additionally, the 'Age Band of Driver' column included values for extremely under aged drivers that were under 15 years old. The rows containing these values were removed from the data.

Finally, a number of functions were created that converted categorical variables to the corresponding string which described the variable's value (e.g. for 'Day' column: 1 converted to 'Monday', 2 converted to 'Tuesday' etc.). These functions were needed in order to better explore the relations between the data and create easy to read statistical plots.

2.3 Feature selection

In total, 11 features were selected in order to be further analyzed as predictors for the model. The following table contains these features:

Table 1: The 10 features selected from the dataset

Features	Day of Week, Time, Road Type, Light Conditions, Weather Conditions, Road Surface Conditions, Urban or Rural Area, Vehicle Type, Age Band of Driver, Age of Vehicle

The criterion I used to select these features was based on finding the factors that were determined before the accident happened. For instance, I did not want to use information on the amount of vehicles that ended up in the accident or personal information on the casualties. Instead I wanted information that can be known before the accident happens like driver's age or the vehicle's type.

Analyzing the relation between each one of the features with the Accident Severity was a difficult task for 2 reasons. Firstly, I needed a statistic that measures correlation between categorical variables. After some research on the topic, I decided that the optimal metric for my case is Cramer's V. It is based on a nominal variation of Pearson's Chi-Square Test, and comes built-in with some great benefits:

1. Similarly to correlation, the output is in the range of [0,1], where 0 means no association and 1 is full association. (Unlike correlation, there are no negative values, as there's no such thing as a negative association. Either there is, or there isn't)

2. Like correlation, Cramer's V is symmetrical

After testing the association between the variables and the severity, I found them highly unassociated. And this takes us to the second reason that analyzing the correlation between the severity and the features is a difficult task. The 84% of the accident have a Slight severity which makes the dataset very imbalanced since this is the feature we try to predict. The solution to this problem, at least for measuring association between the variables and the accidents, was to test them against the total amount of accidents each category was involved. For example, 83% of the accidents happened during Fine weather which makes the feature not a good predictor for the model. In some cases I further explored the frequency of Fatal or Serious accidents against a particular feature in order to gain deeper insights. The following table contains the features which was selected as predictors for the model.

Table 2: The 4 features selected as predictors

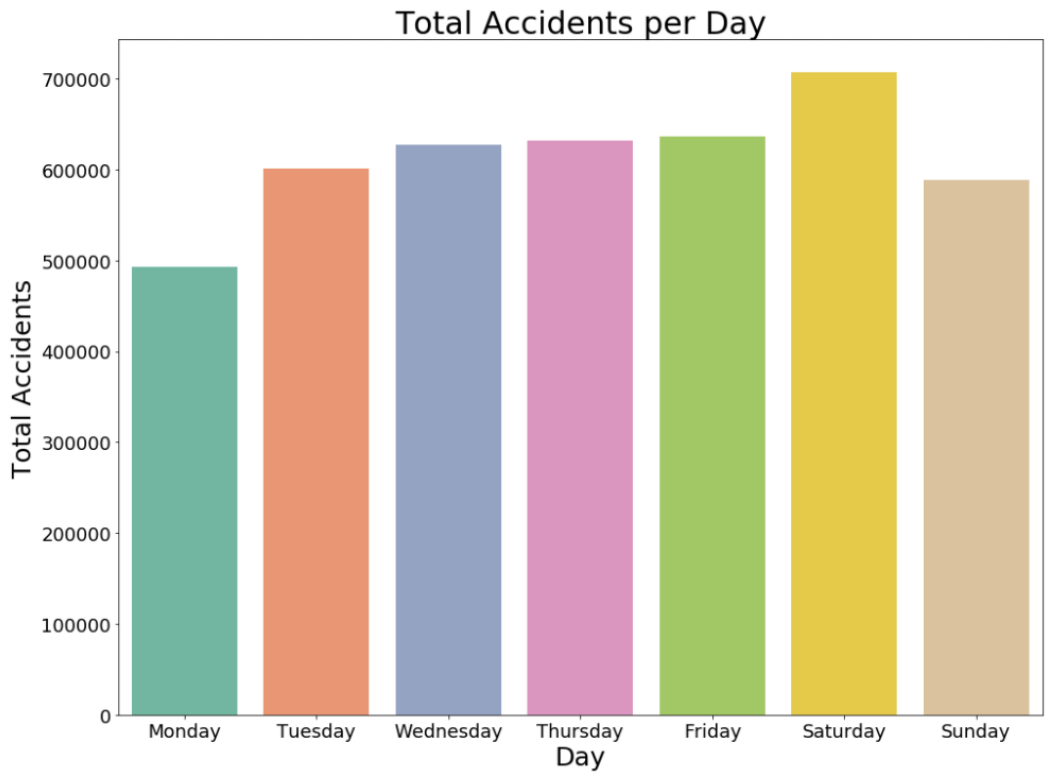
Features	Light Conditions, Urban or Rural Area, Age Band of Driver, Hour

3. Exploratory Data Analysis

3.1 Relation of Accidents with Day of the Week

Someone could say that the “busier” a day is, there more traffic will be at the road. Hence, we expect business days from Monday to Friday to be more dangerous for drivers.

Bar plot 1: Accidents vs Day of the Week



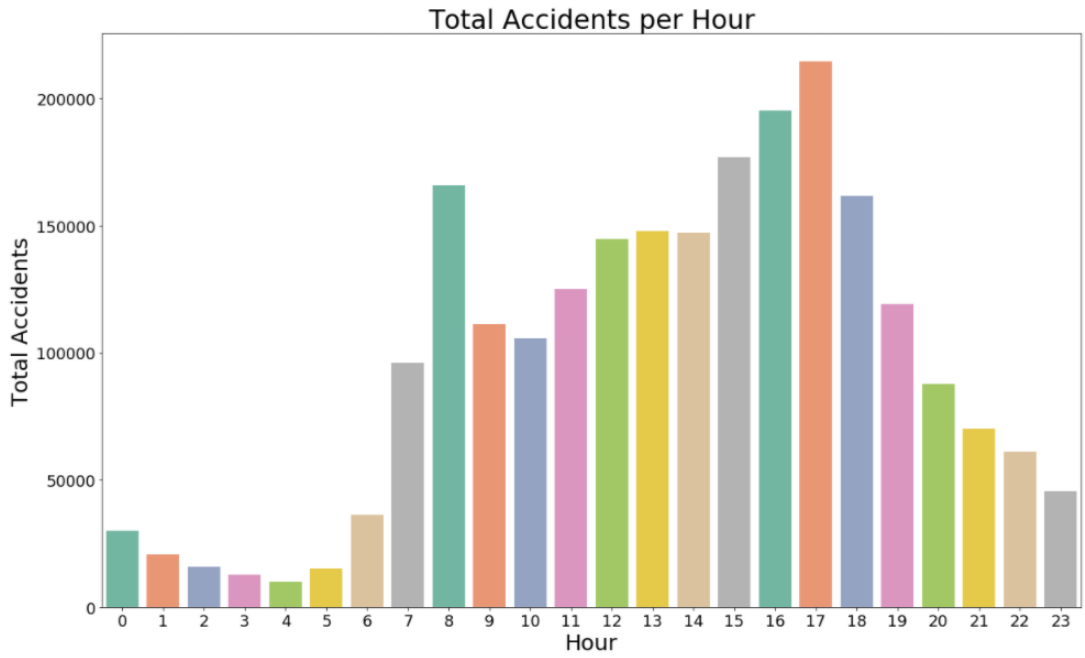
Looking at the bar plot we can see that accidents are distributed in a similar way among each day of the week, 5 of the 7 days have a total of accidents in the range of 600.000 – 650.000. Monday has the least amount of accidents while most accidents happen on Saturday (not a business day), a day dedicated usually for amusement activities (clubbing, traveling, alcohol etc.) which increases traffic and as it seems, the chance of a car accident. In conclusion, since the accidents are

similarly distributed among the days of the week, we will not use this variable as a predictor.

3.2 Relation of Accidents with Hour of the Day

As was the case when examining the Days that accidents happen, we expect most of the accidents to take place during the 'busy' hours of the day when people are driving to or from their jobs.

Bar plot 2: Accidents vs Hour of the Day

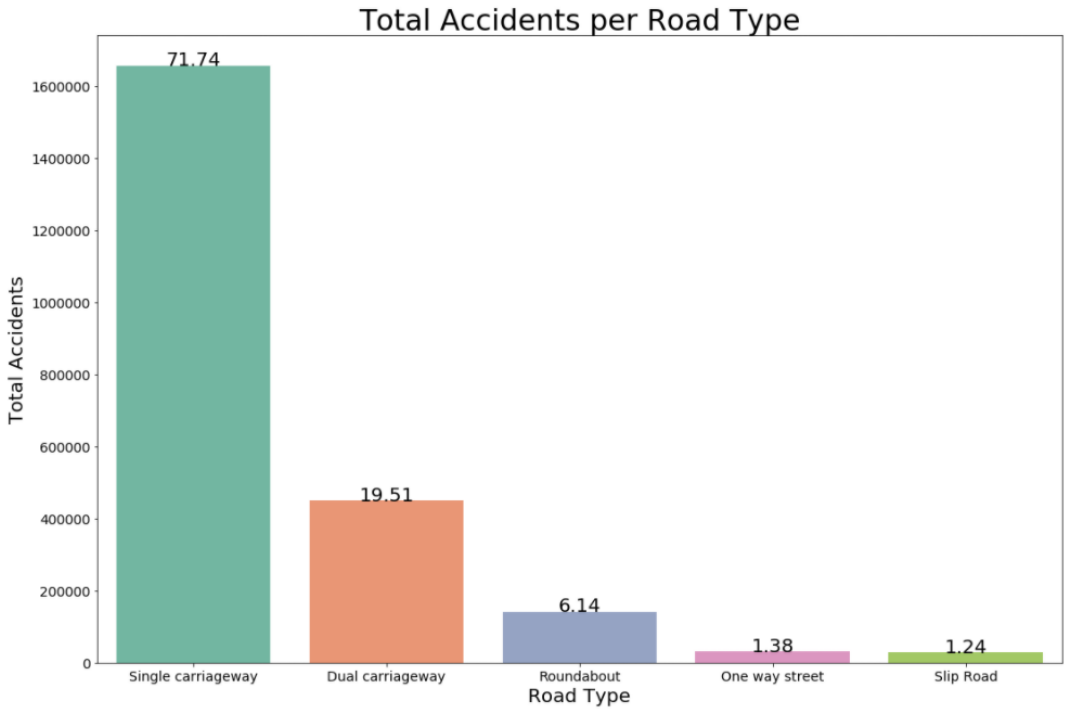


The bar plot shows the distribution of accidents among the hours of a day. We can notice a high variance on the amount of accidents, the early hours of a day have a small amount of accidents which is normal considering the reduced traffic at these hours. Later on, the amount of accidents rises during busy hours (8, 16-18). This feature of the data can be used as a predictor for the model.

3.3 Relation of Accidents with Road Type

One could assume that the most dangerous road types are the ones that allow high speeds, such as interstates or freeways.

Bar plot 3: Accidents vs Road Types

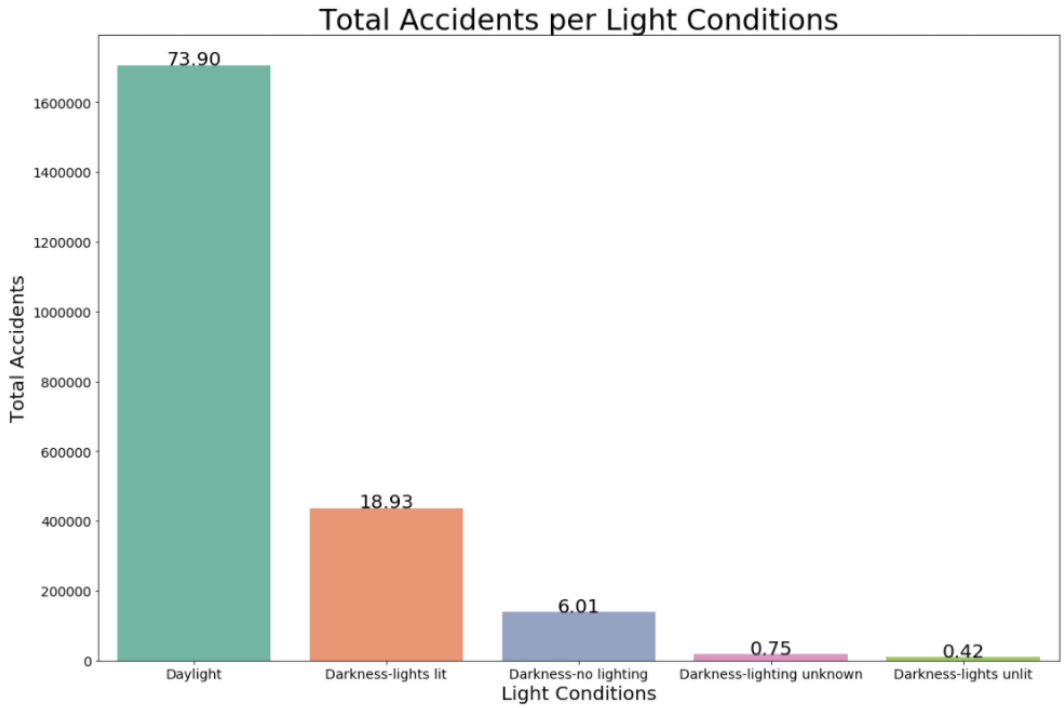


The bar plot clearly shows an imbalance in the Road Type feature, 71.74% of the accidents happened in a Single carriageway. This makes the feature not useful for the model.

3.4 Relation of Accidents with Light Conditions

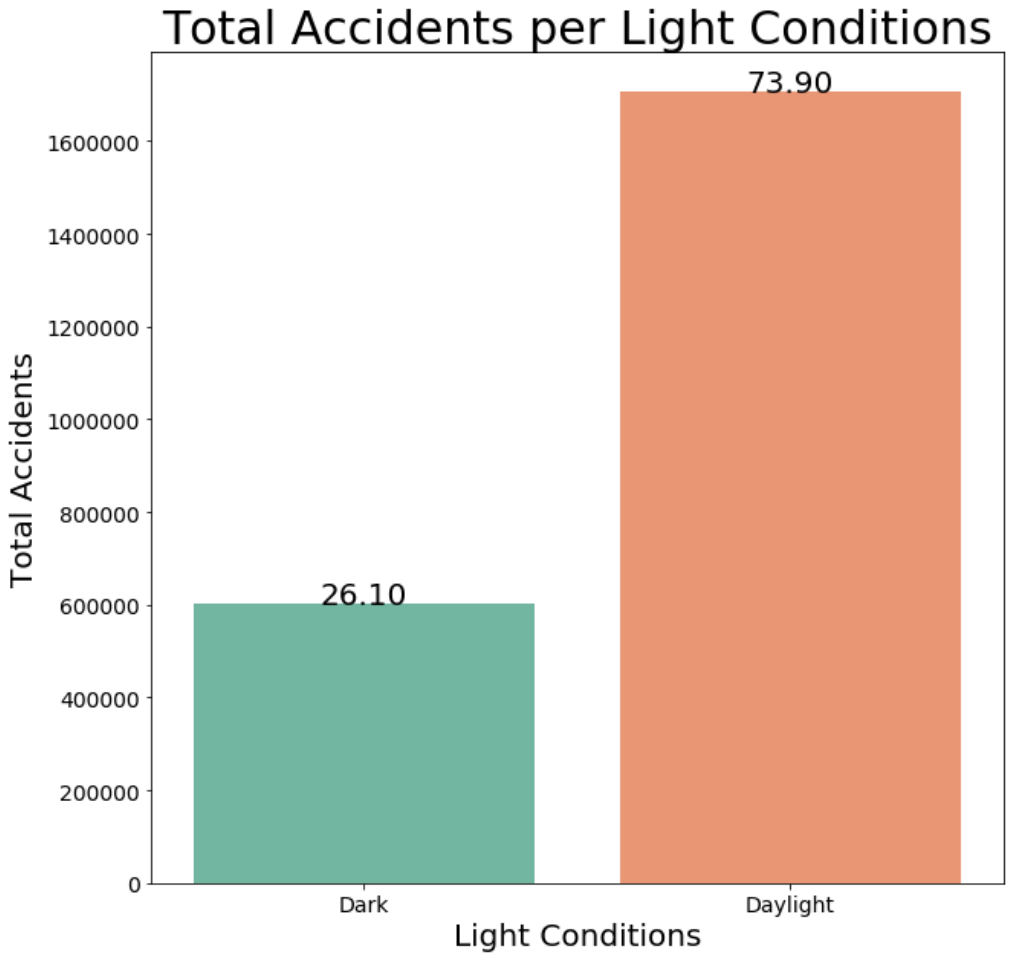
Most people would assume that driving in darkness is the most dangerous light conditions you can get but looking at the bar plot:

Bar plot 4.1: Accidents vs Light Conditions



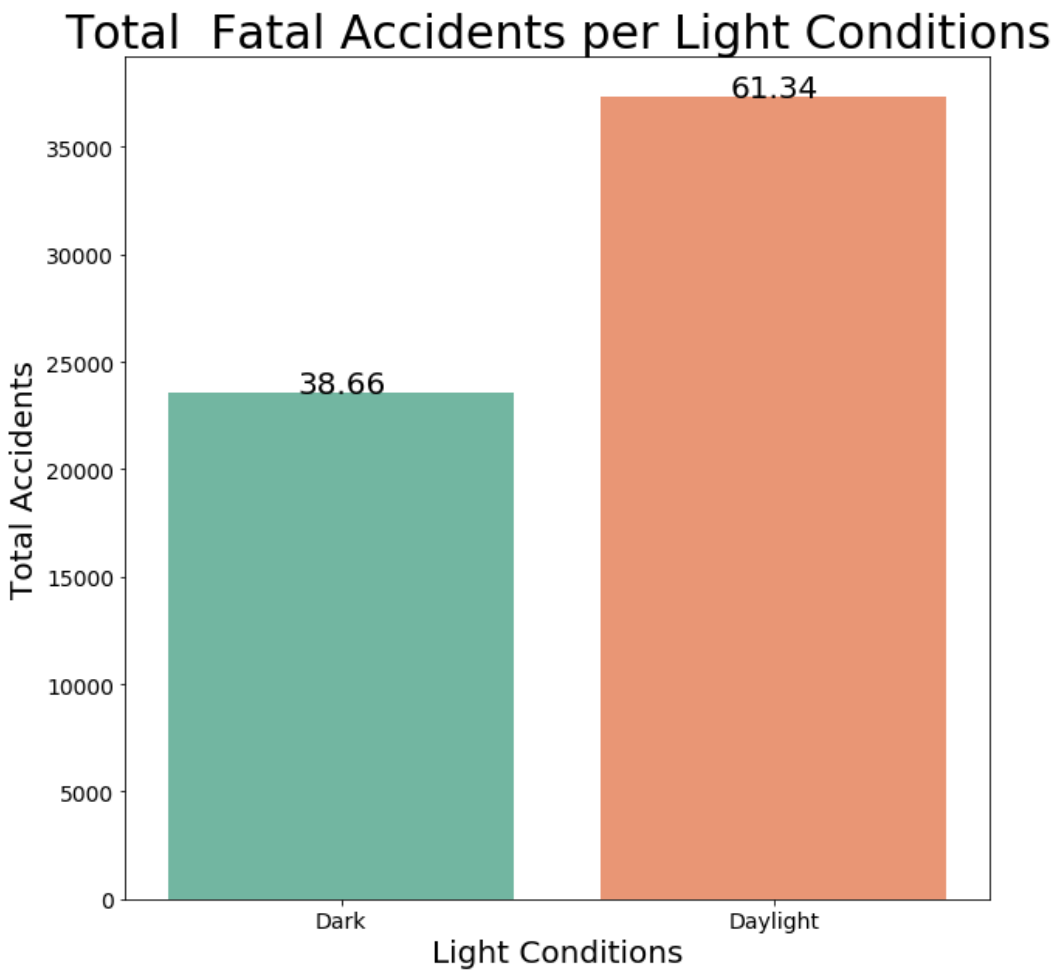
We can notice that 74% of the accidents happen in daylight. This fact is also noticed in the Hour bar plot above, where most of the accidents happened at morning or early afternoon hours. It is normal for the drivers to be more careful when driving in the dark. Furthermore traffic tends to be at lower levels at night or very early morning (when it is mostly dark). The light conditions are in a way represented from the Time column but let's take the time to dig a little bit deeper in the data in order to make sure there is no information to be gained from this feature.

Bar plot 4.2: Accidents vs Light Conditions (values combined)



This bar plot is the same with the last one but all the darkness categories are now combined into one (Dark). Having combined the data we can now move on 'digging deeper' in the data and explore the relation of light conditions with fatal only accidents.

Bar plot 4.3: Fatal Accidents vs Light Conditions

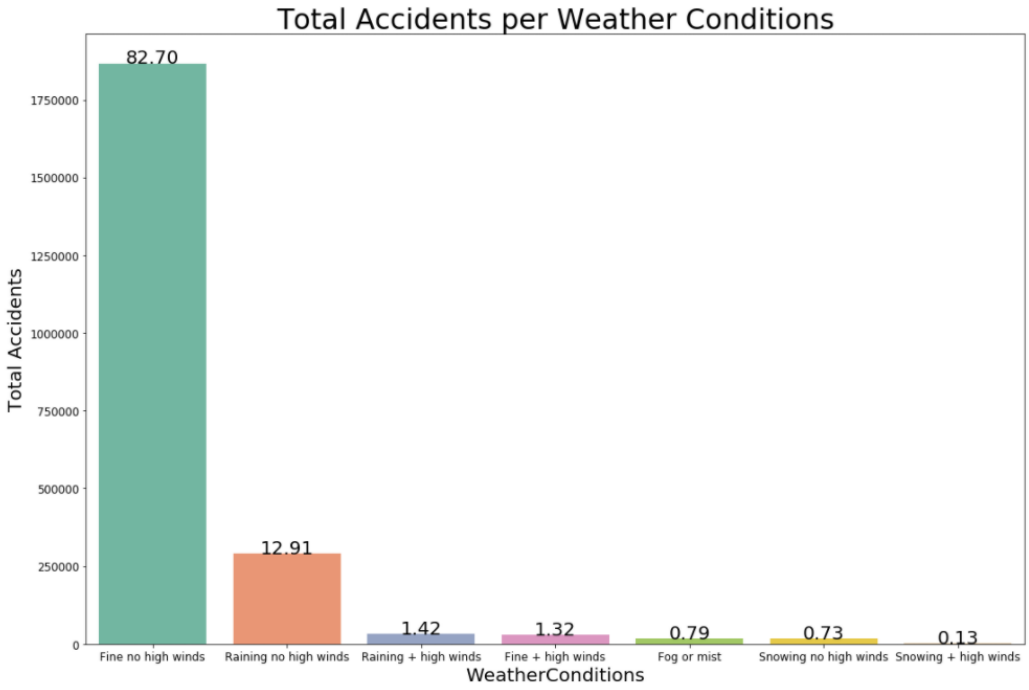


This is what we were looking for! While only 26% of the accidents happen when dark, the percentage goes up to almost 40% when examining only fatal accidents. The bar plot shows an association between fatal accidents and dark light conditions, hence we will include the column “Light Conditions” in the model.

3.5 Relation of Accidents with Weather Conditions

Even though assuming that driving in extreme Weather Conditions is more dangerous than when the Weather is fine, looking at the bar plot:

Bar plot 5: Accidents vs Weather Conditions

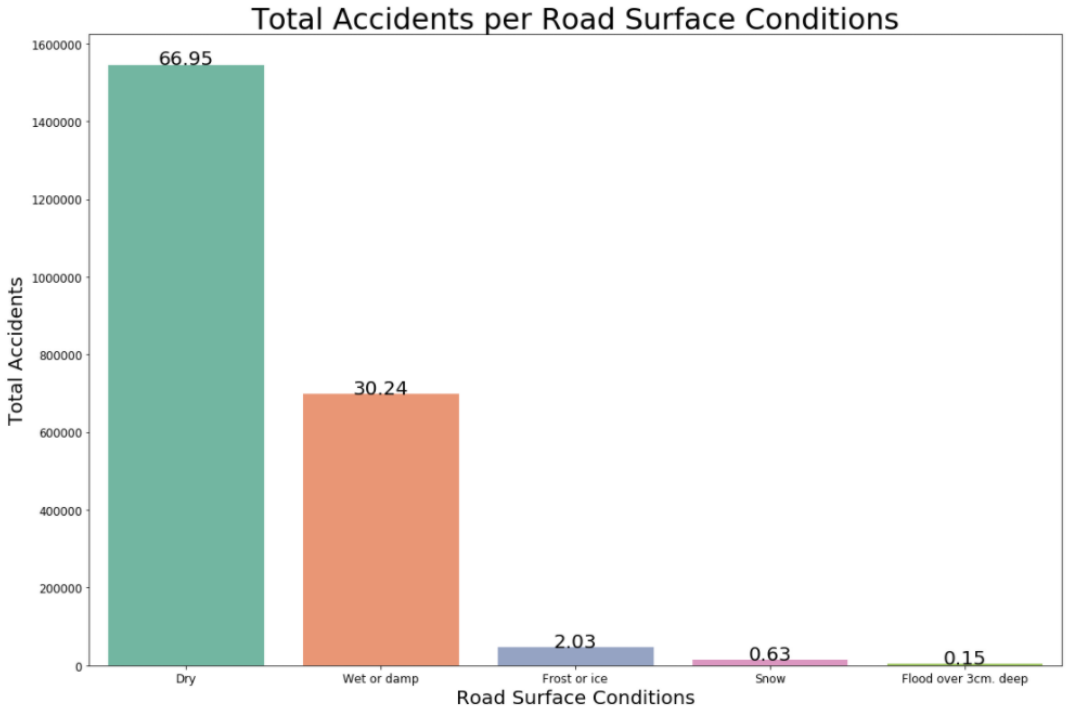


We can notice that 82.70% of the accidents happen while the weather is fine with no high winds. Even though it seems weird that bad weather like rain which traditionally makes driving harder, does not affect accidents, an argument can be made that drivers are more focused and careful when driving in bad weather conditions. The weather feature will not be used in our model.

3.6 Relation of Accidents with Road Surface Conditions

In the same way we expected extreme weather to cause more accidents, we expect extreme Road Surface Conditions such as wet roads or roads covered in ice to also make driving dangerous. The bar plot tells another story:

Bar plot 6: Accidents vs Road Surface Conditions

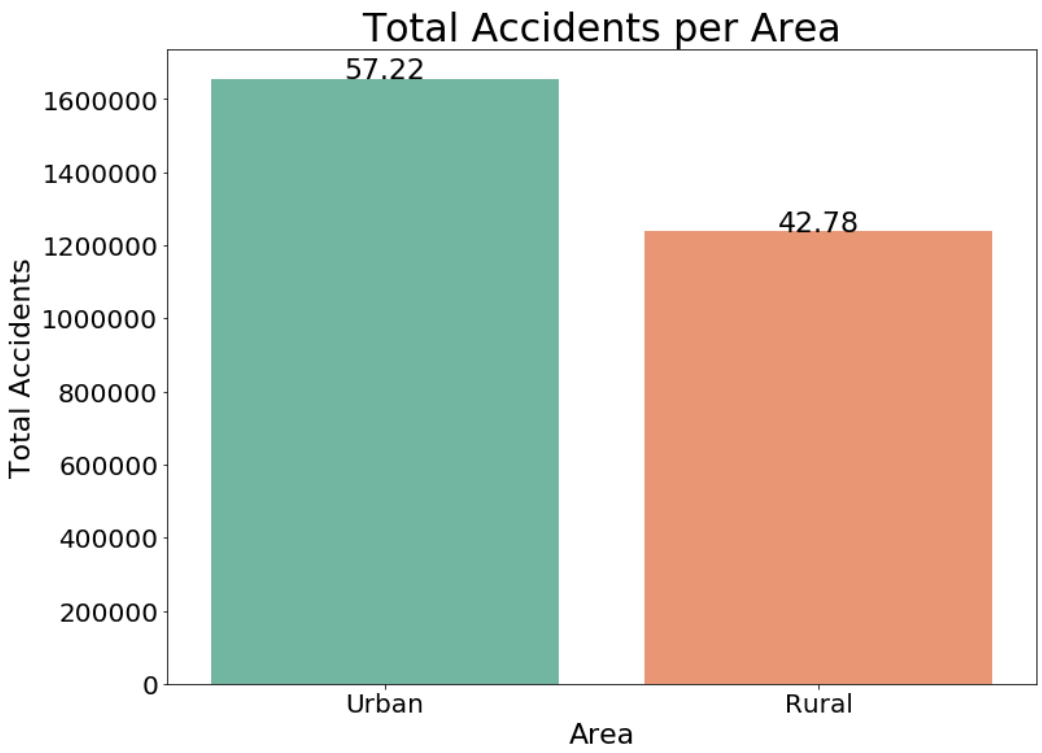


Almost 70% of the accidents happen on Dry Roads. There is a chance that drivers pay more attention and are more focused when there are extreme road surface conditions. This is imbalance makes the feature not a good predictor for the model.

3.7 Relation of Accidents with Area Type

Looking at NHTSA statistics[2] we can see that even though rural roads tend to be less crowded, they accounted for 50 percent of all traffic fatalities in 2016, despite accounting for just 30 percent of all vehicle-miles traveled that year.

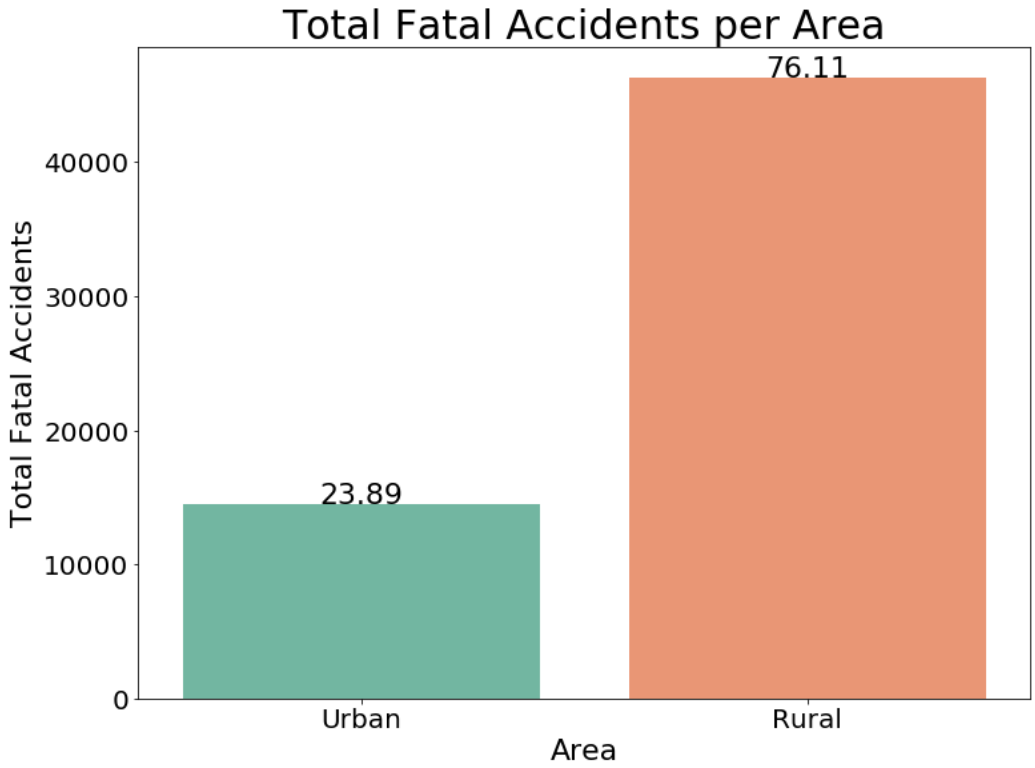
Bar plot 7.1: Accidents vs Area Type



Contrary to the NHTSA statistics the bar plot shows that most accidents happen in Urban Areas but the difference is not very big (16.18). Let's what is the difference only for the fatal accidents, which are rare in the dataset.

[2] "The Understated Dangers of Driving On Rural Roads: Minimizing Driving Risks." *EPermittTest*, www.epermittest.com/drivers-education/dangers-rural-roads.

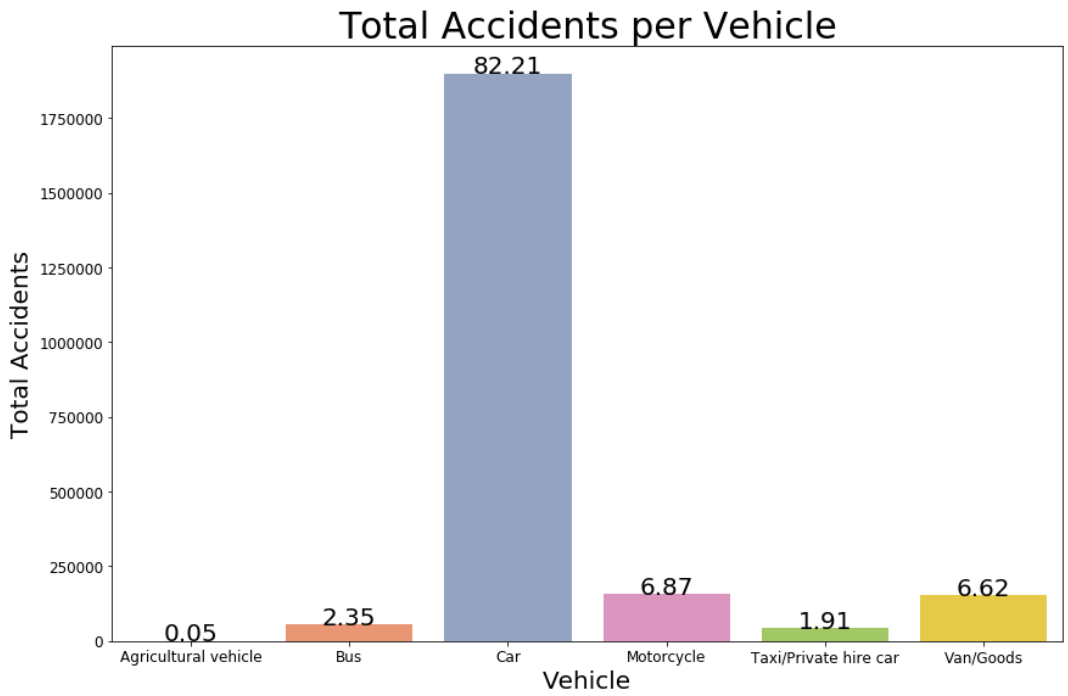
Bar plot 7.2: Fatal Accidents vs Area Type



The bar plot shows how things are changed when we examine just the fatal accidents. While accidents in our dataset are almost equally distributed among rural and urban areas, when it comes to fatal accidents, 75% of them happen at rural areas, showcasing a high association of fatal accidents with rural areas. We will include this feature in the model.

3.8 Relation of Accidents with Vehicle Type

Bar plot 8: Accidents vs Vehicle Type

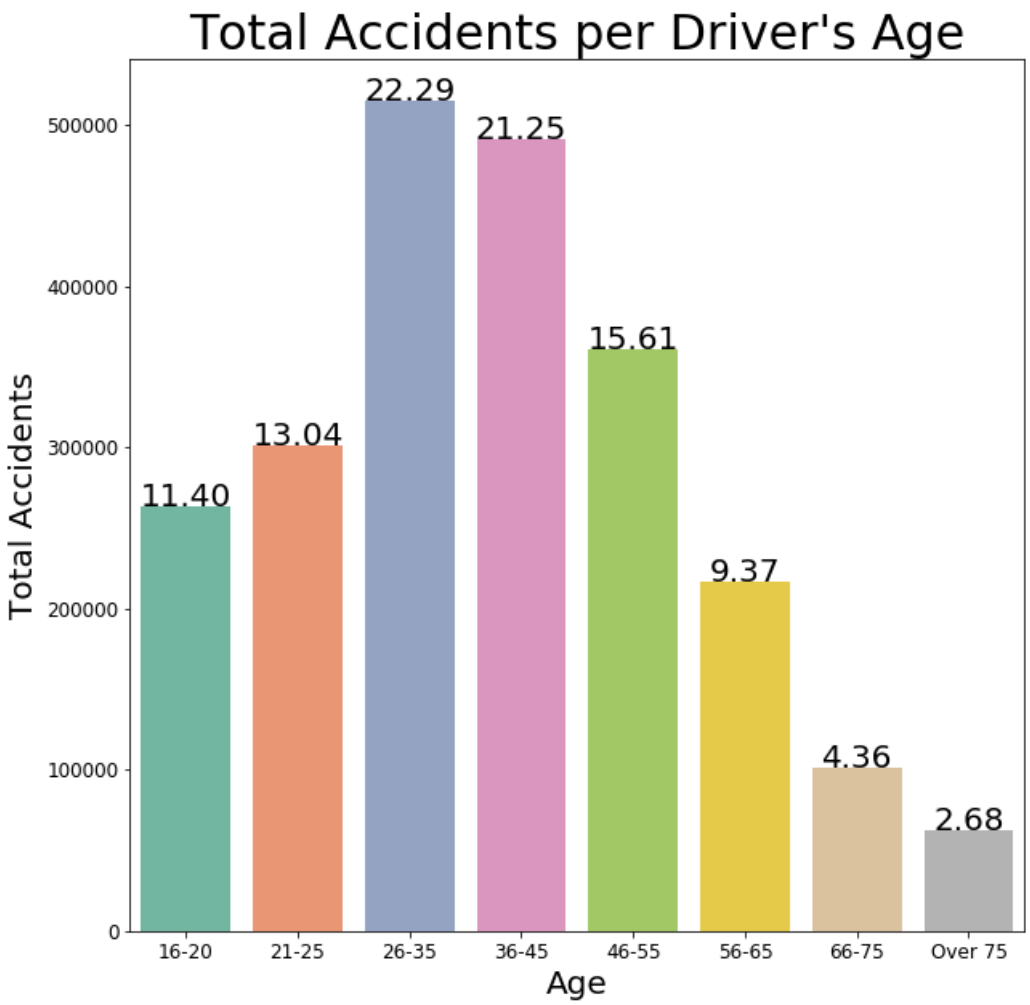


We can notice that 82.21% of the accidents are caused by cars. This probably happens because cars are the most common vehicle on public roads. An imbalance such this make the feature not a bad predictor for our model.

3.9 Relation of Accidents with Driver's Age

A logical assumption would be that younger drivers which lack driving experience or older ones that lack the reflexes, would be mostly associated with accidents.

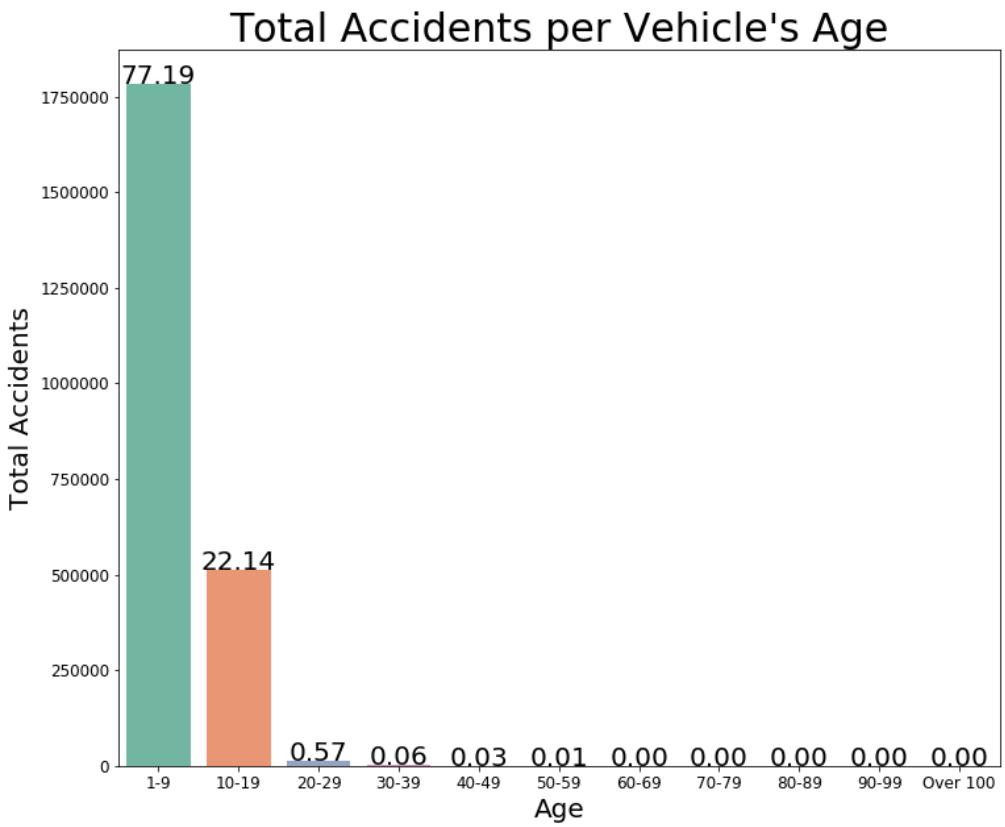
Bar plot 9: Accidents vs Vehicle Type



The bar plot shows another case, where accidents are caused mostly from drivers aged 26-55. Maybe overconfidence is a big factor in this case and causes more accidents without the driver's experience or reflexes being important. This feature is one we should use in our model given the important information it contains.

3.10 Relation of Accidents with Vehicle Age

Bar plot 10: Accidents vs Vehicle Age



The vehicle's age does not seems to play an important role on the frequency of accidents. The most logical reason would be that most of the vehicles on the road are less than 10 years old, creating this imbalance in the dataset.

3.11 Association of each Feature with Accident's Severity

Figure 1: Cramer's V statistic, a measure of association

```
cramers_v(df['Day_of_Week'],df['Accident_Severity'])
```

```
0.032381111406731986
```

```
cramers_v(df['Accident_Severity'],df['Day_of_Week'])
```

```
0.032381111406731986
```

Cramer's V is indeed symmetrical, it gives the same value and the order of the input features does not matter.

```
cramers_v(df['Accident_Severity'],df['Time'])
```

```
0.11954514766863902
```

```
cramers_v(df['Accident_Severity'],df['Road_Type'])
```

```
0.04786839404913596
```

```
cramers_v(df['Accident_Severity'],df['Light_Conditions'])
```

```
0.08277667961628699
```

```
cramers_v(df['Accident_Severity'],df['Weather_Conditions'])
```

```
0.03684736395293936
```

```
cramers_v(df['Accident_Severity'],df['Road_Surface_Conditions'])
```

```
0.009163015415411712
```

```
cramers_v(df['Accident_Severity'],df['Urban_or_Rural_Area'])
```

```
0.144498400346582
```

```
cramers_v(df['Accident_Severity'],df['Vehicle_Type'])
```

```
0.10280761620992851
```

```
cramers_v(df['Accident_Severity'],df['Age_Band_of_Driver'])
```

```
0.030983211629291886
```

```
cramers_v(df['Accident_Severity'],df['Age_of_Vehicle'])
```

```
0.02540612571603147
```

The above figure shows the results of measuring the features association with the Accident's Severity (from the project's Jupyter Notebook). The

highest value is 0.14 for the “Urban or Rural Area” feature which is still very low (0 means no association, 1 means perfect association).

4. Predictive Modeling

4.1 Classification models

In this study, I carried out classification modeling since the target variable is a categorical variable which makes it ideal to use this type of models.

4.2 Applying standard algorithms and their problems

I applied Logistic Regression, K-Nearest Neighbors, Support Vector Machine and Decision Tree models to the dataset, using Macro Average f1 score as the tuning and evaluation metric. This metric was chosen in order to evaluate always the model’s biggest problem which was predicting all 3 of the classes, in a similar level of accuracy. Macro Average f1[3] score is an excellent metric for this cause as when macro-averaging, all classes contribute equally regardless of how often they appear in the dataset. The confusion matrix was also a great tool in order to gain insights on how the models predicted each one of the 3 classes.

The results had all the same problem. There was a great imbalance in the target variable “Accident Severity” where 84% of the accidents had a Slight Severity. This led to a model that only predicted accidents of Slight Severity with an “illusory” accuracy of 84%, since the test set I evaluated the model on followed the same pattern as the training set with a lot of

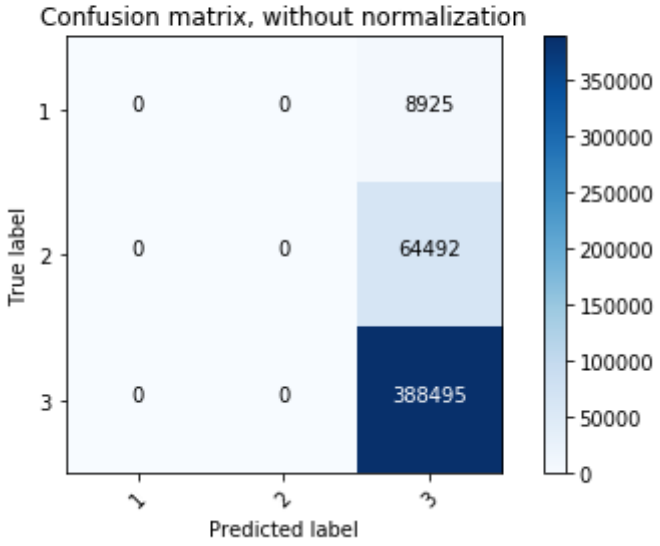
[3] “Macro F1-Score.” *Peltarion.com*, peltarion.com/knowledge-center/documentation/evaluation-view/classification-loss-metrics/macro-f1-score.

slight accidents. Even though the model predicted only one class, it was the majority class of the test set giving as a result a very high accuracy.

Looking at the confusion matrix and the classification report it was very easy to spot the problem with the model. The confusion matrix clearly shows that the only Predicted Labels of the model were of class 3. The Macro Average f1 score gives a value of 0.30 as well.

Figure 2: Confusion matrix and Classification Report
(results of modeling on an imbalanced dataset)

```
Confusion matrix, without normalization
[[     0      0   8925]
 [     0      0  64492]
 [     0      0 388495]]
```



Classification Report

	precision	recall	f1-score	support
Class 1	0.00	0.00	0.00	8925
Class 2	0.00	0.00	0.00	64492
Class 3	0.84	1.00	0.91	388495
accuracy			0.84	461912
macro avg	0.28	0.33	0.30	461912
weighted avg	0.71	0.84	0.77	461912

4.3 Solution to the problem

What we have in our hands is a very imbalanced dataset but this is a realistic problem when it comes to 'real-life' data. There are ways to engineer the data in order to improve the model. After some research on the topic I came across Python's library imbalanced-learn, which is fully compatible with scikit-learn. The library offers resampling methods which are going to be used in order to bring some balance in the dataset.

4.3.1 Resampling

A widely adopted technique for dealing with highly unbalanced datasets is called resampling. It consists of removing samples from the majority class (under-sampling) and / or adding more examples from the minority class (over-sampling). Despite the advantage of balancing classes, these techniques also have their weaknesses (there is no free lunch). The simplest implementation of over-sampling is to duplicate random records from the minority class, which can cause overfitting. In under-sampling, the simplest technique involves removing random records from the majority class, which can cause loss of information.

A very important step in this procedure is to first split the data into train and test datasets and then apply Resampling only on the training set, and not the testing set, in order to test the model on original data.

The following figure shows the way I resampled the training set:

Figure 3: Resampling

Training set before resampling:
Unique values -> Class 1 (fatal): 36.366, Class 2 (serious): 257.786, Class 3 (slight): 1.553.495

Training set after resampling:
Unique values -> Class 1 (fatal): 200.000, Class 2 (serious): 340.000, Class 3 (slight): 360.000

The resampling process happened in 2 steps:

1. I under-sampled the majority class (3) from 1.553.495 samples to 360.000 samples in order to balance the classes and minimize loss of information.

2. I over-sampled the minority classes (1, 2) to 200.000 and 340.000 samples accordingly, in order to balance the classes and minimize overfitting.

4.4 Performances of different models

Using the new approach of oversampling the training set, I built Logistic Regression, K-Nearest Neighbors, Support Vector Machine and Decision Tree models using Macro Average f1 score as the evaluation metric. Resampling did make the models predict other classes than 3 (Slight) and improved the Macro f1 score to 0.36. Support Vector Machine had the best performance of all models.

Table 3: Performance of the classification models.

	Logistic Regression	K-Nearest Neighbors	Support Vector Machine	Decision Tree
Macro Average f1 score	0.35	0.35	0.36	0.35
Accuracy (%)	0.57	0.57	0.60	0.54

5. Conclusions

In this study, I analyzed the relationship between Accidents Severity and Environmental factors that preceded the accidents, the Drivers personal information and the Timing of the accidents. I identified Light Conditions, the Drivers Age, the Hour of the Day and the Area Type as the most important features that affect Accident frequency and severity. I built classification models to predict the severity of a possible accident.

These models can help government organizations to pinpoint dangerous areas or roads and improve road design. Another possibility is the development of an application that is ‘fed’ live data on the environment’s conditions the driver is driving through, his/her personal information and warn them on the danger they are currently facing, in

the same way people that smoke are given facts about how smoking hinders their health, on top of the packets. In our case though this could happen through connected cars that have internet access and notify the driver when the danger levels are high or even through mobile phones where a driver can check information on his/her journey's danger before they start it.

6. Future directions

I was able to achieve a 60% accuracy in the classification problem and improve the way the models predicted classes other than Slight accidents. However, the models still struggle to find patterns for Fatal and Serious accidents and poorly predict these classes. There is definitely a need for more data on Fatal and Serious accidents in order to minimize the use of Resampling in our data and improve the models performance.

Furthermore, even more features can be collected on the accidents like information on prior to the accident alcohol consumption or drug usage from the driver's side, on the usage of mobile phones, the frequency and the type of mechanical failures and the amount of passengers inside the vehicle. These data are obviously more difficult to extract but can bring significant improvements to the model.

7. References

- George, Y., Athanasios, T., & George, P. (2017, June 08). Investigation of road accident severity per vehicle type. Retrieved October 16, 2020, from https://www.sciencedirect.com/science/article/pii/S2352146517307081

- Janiobachmann. (2019, July 03). Credit Fraud || Dealing with Imbalanced Datasets. Retrieved October 16, 2020, from https://www.kaggle.com/janiobachmann/credit-fraud-dealing-with-imbalanced-datasets

- Alencar, R. (2017, November 15). Resampling strategies for imbalanced datasets. Retrieved October 16, 2020, from https://www.kaggle.com/rafjaa/resampling-strategies-for-imbalanced-datasets

- Zychlinski, S. (2019, December 26). The Search for Categorical Correlation. Retrieved October 16, 2020, from https://towardsdatascience.com/the-search-for-categorical-correlation-a1cf7f1888c9

- Lemaitre, G., Nogueira, F., & Aridas, C. K. (2016, September 21). Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. Retrieved October 16, 2020, from https://arxiv.org/abs/1609.06570.

- Brownlee, J. (2020, August 27). Random Oversampling and Undersampling for Imbalanced Classification. Retrieved October 16, 2020, from https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/