# National College of Ireland

Bachelor of Science in Computing

Data Analytics

2020/2021

Alan Patterson

X18108105

X18108105@student.ncirl.ie

# Predictive Analysis of UK Road Accidents Using Machine Learning

## Technical Report

# Table of Contents

# 1. Executive Summary

Responsible for approximately 1.35 million deaths each year, road-traffic accidents are currently the 8th leading cause of death globally. For governments across the world to implement measures to reduce the number of fatalities caused by road traffic accidents, it is crucial to understand of the main causes of these accidents and the circumstances under which they occur.

The aim of this project is two-fold:

1. To gain insight into the current climate of road accidents in the United Kingdom (UK) through an exploratory analysis of multiple road accident datasets.
2. To investigate the effectiveness of various machine learning techniques in predicting severity of road accidents in the UK.

Multiple datasets were collected, collated, explored and prepared for several machine learning models which were trained and tested to evaluate and compare predictive performance.

# 2. Introduction

## 2.1. Background

Road traffic accidents remain a very serious problem worldwide; responsible for approximately 1.35 million deaths each year, an average of just under 3,700 deaths per day. Globally, road traffic accidents are the 8th leading cause of death across all age groups and the primary cause of death amongst children and young adults (aged 5 – 29), with vulnerable road users accounting for over half of these deaths; pedestrians and cyclists making up 26% and a further 28% relating to users of motorbikes and 'three-wheelers' [1].

In March 2010, the United Nations declared the years 2010 – 2020 were to be a decade of action for road safety, ambitiously aiming to halve the number of road fatalities and serious injuries by 2020. Although the UN did fall short of achieving this goal, it has declared a second decade of action for road safety, from 2021 -2030, encouraging member states to take a number of positive actions with the renewed aim of reducing road traffic deaths and serious injuries by half, by 2030 [2].

Understanding the factors that contribute to the frequency and severity of road traffic accidents is crucial to implementing effective preventative measures, and this can only be achieved through the collection, collation, and analysis of accurate data. This project will centre around the analysis of UK road accident data provided by the UK Department of Transport following an increase in demand for more updated and detailed information to be made available to the public.

Each year, the UK government provides informative road accident statistical reports. In 2019, there were a total of 1,752 road traffic deaths reported and a total of 153,158 reported casualties, of all severities, 25,945 of which suffered serious injuries. The trend in road traffic deaths in the UK has been relatively flat from 2010 to 2019, as can be seen when comparing the 2019 figures with the 1,754 road deaths occurring in 2012; indicating a slight reduction when factoring-in the increased traffic and vehicle numbers [3].

The prediction of road accident frequency and severity is an essential component in trying to minimise the impact of these accidents, as such, there is vast amount of research done in this field; government bodies, health and safety organisations, insurance and reinsurance companies, all have a vested interest in maximising the predictive performance of their analytical models. Generalised Linear Modelling (GLM) has been a popular choice by actuaries for forecasting claim frequencies and severities, but with the adaptability and flexibility of various machine learning models and their combinations, the predictive performance of machine learning models is an interesting field to study and compare. GLMs takes the traditional normal linear model and generalises it by relaxing the some of its restrictions and facilitates the analysis of non-normal data, insurance claims data being one such example. One limitation of this dataset is that it doesn't provide the exposure data required for actuarial models such as the above mentioned GLMs, as the data details how many accidents have occurred, but not the total number of vehicles there are on the roads.

With the recent, significant technological advancements in the field of autonomous vehicles, the first fully autonomous car predicted to be made available to the market in the next year or so, and as increased volumes of telematics data is collected and stored, the question of road safety and the effects of fully autonomous vehicles being in circulation will be a topic of great importance and of great interest to analysts.

## 2.2. Aims

The goal of this project is to gain insight into the current climate of road traffic accidents in the UK, to gain additional understanding of what drives severity of road accidents, and to investigate the effectiveness of various machine learning techniques in the prediction of road accident casualty fatalities.

A number of machine learning models will be trained and their performance evaluated following the exploratory analysis to be carried out with a view to addressing the following initial research questions:

1. In which areas do the most accidents, and the most severe accidents occur?
2. When do accidents occur most and how has the frequency and severity of accidents developed over time?
3. How effective is machine learning in predicting the severity of road accidents in the UK?

## 2.3. Technology

| Name | Type | Description |
|---|---|---|
| MS Excel, Power Query, Power Pivot | Data Preparation Tool | Spreadsheet application, data transformation and preparation engine, and analysis environment based on the Vertipaq engine. The Power Query uses the functional programming language M, and Power Pivot uses the DAX (Data Analysis Expressions) language. |
| MS Power BI | Data Visualisation Tool | A powerful data visualisation tool from Microsoft, which utilises M and DAX languages. |

| | | |
|---|---|---|
| Python | Programming Language | The programming language used to perform the project analysis |
| Anaconda | Distribution and Package Manager for Python and R | Provides a number of python libraries used for data preparation and analysis |
| Jupyter Lab | Python Programming IDE | Part of the Anaconda Distribution<br>Integrated Development Environment (IDE) for python |
| Spyder4 | Python Programming IDE | Part of the Anaconda Distribution<br>Integrated Development Environment (IDE) for python |
| Matplotlib | Python Library | Data visualisation python library |
| SKLearn | Python Library | Machine learning python library |
| Pickle | Python Library | Library used for saving Python workspaces, reducing need to continually read csv datasets. |

## 2.4. Structure

### Section 3 – Data

The 'Data' section of this report provides details of the source datasets used to form the basis of this project's research. This section provides detailed information on the source and structure of the datasets collected for use and details the steps taken to collect and pre-process the source data for use.

### Section 4 – Methodology

The "Methodology" section of this report provides detailed descriptions of the methods used to transform the pre-processed source data into the working datasets on which the project's analysis will be performed. This section will provide a complete walkthrough of the process

### Section 5 – Analysis

The "Analysis" section of the report will focus on discussing the various analytical techniques employed in order to gain insight from the data. Details on the selection of attributes for inclusion and exclusion will be provided as well as an outline for the next phases of analysis to be performed.

The preliminary finding of the analysis along with some initial machine learning accuracy scores will be included in this section.

### Section 5 – Analysis

This section outlines the predictive models utilised and the analysis carried out

### Section 6 – Results

This section outlines the results of predictive analyses performed

### Section 7 – Conclusions

This section provides the projects conclusions and a reflection on the project as a whole

### Section 8 – Further Research and Development

This section outlines work which could be performed on top of this project.

# 3. Data

## 3.1. Source Data

This project utilises a total of 21 datasets sourced from the UK government data portal, which can be divided into four main categories, with each category having its own uniform structure:

- Accident Datasets (x6)
- Vehicle Datasets (x6)
- Casualty Datasets (x6)
- Adjustments Datasets (x2)
- Variable Lookup Dataset (x1)

These datasets were downloaded from the UK government data portal[1] as CSV files. The Accidents, Casualties, and Vehicles datasets comprised five annual datasets containing records of road traffic accidents in the UK from 2015 to 2019 and one additional larger file containing records from 2005 to 2014.

To accompany the Accident and Casualty datasets, an additional two Adjustment datasets containing severity adjustment records spanning all years, were collected. These adjustment records link to records contained in the Accident and Casualty datasets through a shared primary key, or identifier.

### *Accident, Vehicle, and Casualty Datasets*

The Accidents dataset can be considered the superclass of the Vehicle and Casualty datasets, as each record in the Vehicles dataset should link with a record in the Accidents dataset through a shared 'accident_index' identifier. There is a one-to-many relationship between the Accident dataset and the Vehicle dataset, and one-to-many relation between the Vehicle dataset and the Casualty dataset.

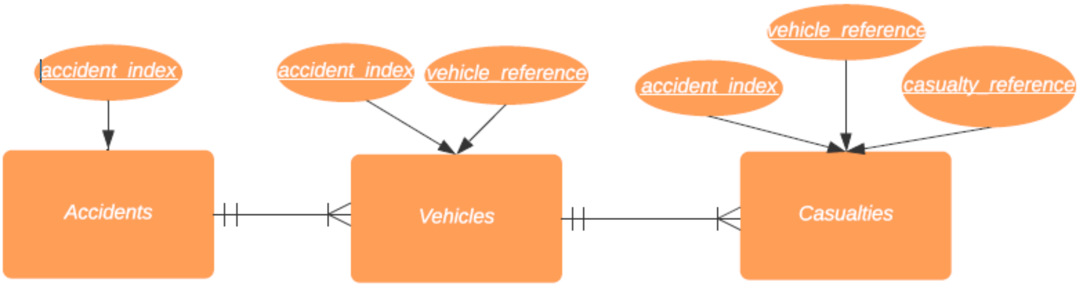


*Figure 1 – High-Level ERD for Datasets*

Similarly, each record in the Casualty datasets links to a record in the Accident dataset, through a shared 'accident_index', and to a record in the Vehicle dataset, through shared 'accident_index', 'vehicle_reference', and 'casualty_reference' identifiers.

| Collated Dataset | Number of Records | Number of Attributes |
|---|---|---|
| Accidents | 2.29m | 33 |
| Casualties | 3.07m | 17 |
| Vehicles | 4.20m | 24 |
| Accident Adjustments | 2.46m | 4 |
| Casualty Adjustments | 3.31m | 4 |

*Additional detail on each of the structure of each of the individual source datasets can be found in the appendix.*

[1] https://data.gov.uk/dataset/6efe5505-941f-45bf-b576-4c1e09b579a1/road-traffic-accidents

### *Adjustments*

The adjustment records contain an adjustment factor, with values ranging from 0 – 1, which when linked to the corresponding 'accident severity' or 'casualty severity' values in the respective datasets, provide an updated severity value. These adjustment factors were calculated in an analysis performed by the UK statistics office to account for variances in the accident reporting systems used throughout the years and various authorities throughout the years in the UK, to provide an updated and standardised severity value for each accident. This known change in reporting systems might prove to be a challenge for the predictive classification models developed, so it will be interesting to analyse the performance of predictive classification models.

### *Variable Lookup*

The Variable Lookup Dataset provides labels for the coded variable values in the datasets. With few exceptions, the variables contained in the Accident, Vehicle, and Casualty datasets are assigned a numeric value (code), as opposed to text.

The Variable lookup file is an MS Excel workbook which comprises 48 worksheets containing reference tables for attributes included in the Accident, Vehicle, and Casualty datasets; their codes and corresponding labels.

The Variable Lookup file informs that, for most attributes, missing or unknown data can be identified by a variable value of -1, although there are exceptions to this.

## 3.2. Working Dataset

An initial review of each of the individual Accident, Casualty, and Vehicle datasets was carried out to check for consistency across the structure of datasets; ensuring the datasets have the same attributes, format and positioning, naming conventions, date conventions, etc. Queries were written to append each of the individual Accident, Casualty, and Vehicle datasets to form three complete datasets.

As the most granular of the datasets, the complete Casualty dataset was used as the primary dataset to which the Accident and Vehicle records were joined. Each accident record was joined to its corresponding Casualty record, based on the shared accident_index key and following this, an additional join was used to merge each Vehicle record to the record, based on the accident, casualty, and vehicle references. The compiled working dataset contained 3,069,041 records and 67 attributes.

Iterations of data pre-processing, feature selection, engineering would be performed on this throughout the project lifecycle. These iterations will be detailed in the Methodology section.

## 3.3. Exploratory Analysis

To better understand the data and gain look to gain insight into the figures, an exploratory analysis was carried out on the collated dataset using a combination of Python and Power BI. To guide this exploratory analysis, some points of interest were considered and the following research questions were composed:

1. In which areas do the most accidents, and the most severe accidents occur?
2. When do accidents occur most and how has the frequency and severity of accidents developed over time?
3. Which type of casualties suffer are most affected by road accidents?
4. How effective is machine learning in predicting the severity of road accidents in the UK?

## In which areas do the most accidents, and the most severe accidents occur?

To investigate where most accidents occur each accident was plotted using its longitude and latitude coordinates and a heatmap was generated to visualise this. Due to the high volume of accidents the plot points were aggregated by count of accidents in the area. In addition to plotting each accident, casualty fatalities were also plotted to investigate where the most severe accidents occur. These heatmaps can be seen below in figures below:

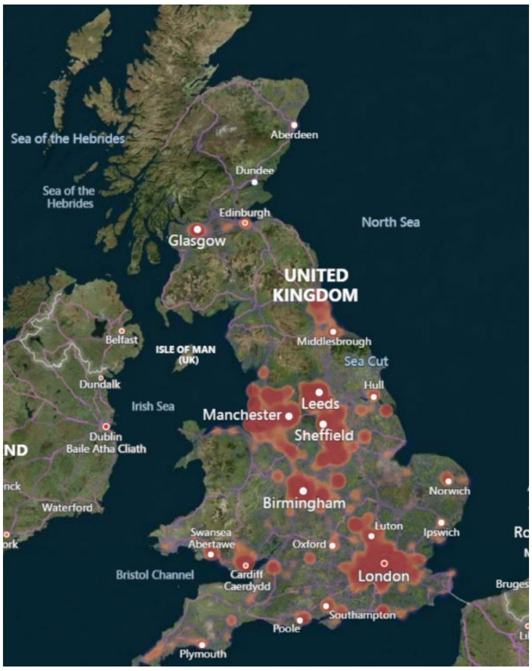


*Figure 2 – Heatmap of Road Traffic Accidents*

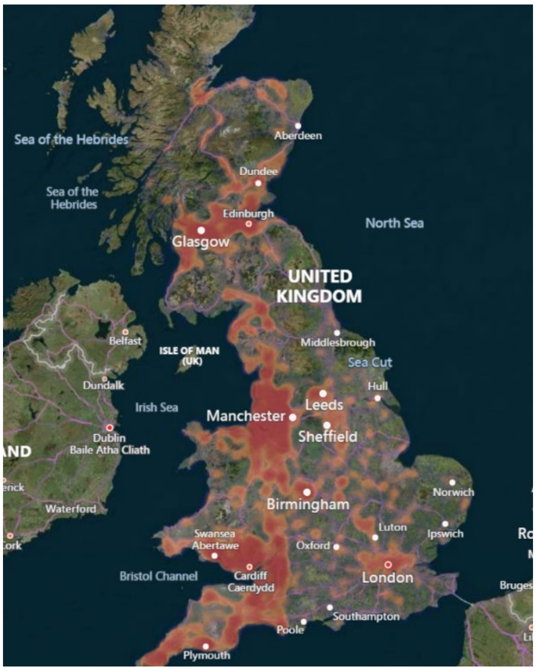


*Figure 3 – Heatmap of Road Accident Fatalities*

As might be expected, in figure 2 above, we can see that a far higher concentration of road traffic accidents occur in and around the major cities, such as London, Birmingham, Leeds, Cardiff, and Glasgow. However, it is very interesting to see the contrast between the concentration of road accidents and the concentration of road accident casualty fatalities, shown in figure 3 above. Despite accounting for significant portion of accidents we can see a disproportionately low concentration of road casualty fatalities in and around London, and we can see higher concentrations of fatalities around Manchester. Similarly, while the higher concentration of road accidents on can be seen in Glasgow and Cardiff, the concentration of fatalities is more dispersed and higher numbers can be seen in neighbouring cities such as Edinburgh and Plymouth. One could speculate on the reasons for this; traffic levels and lower speed limits within the cities, proximity to hospitals, quality of roads, etc., but it is also an interesting discovery in the context of the subject of this project, as it suggests that geographical indicators such as road numbers longitude and latitude coordinates, could be strong predictor variables.

## When do most accidents occur and how has the frequency of accidents developed over time?

To explore when the most accidents occur, the accidents were plotted using the ‘date’ attribute. Additional ‘year’, ‘month’, ‘hour’ attributes were engineered from the accident date and the accidents were grouped by these new attributes and plotted to see if any patterns could be identified.

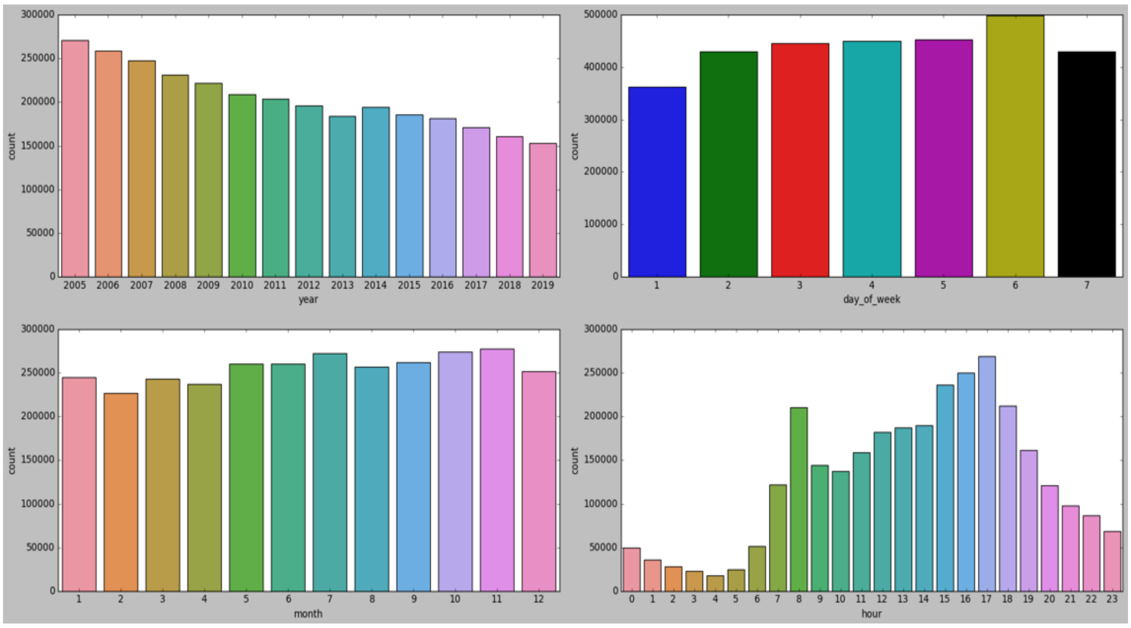


*Figure 4 - Number of Accidents Occurring by Year, Day of Week, Month, and Hour of Day (clockwise)*

A few points of interest can be observed from these visualisations: a steady decline can be seen in the number of accidents occurring each year, while according to the RAC foundation, the UK is experiencing a some of its highest ever levels of traffic on motorways; reaching its highest ever level in 2019. [4]. Interestingly, the average annual mileage per car is reported as decreasing but the number of vehicles has increased per household.

From the visualisation, it can also be seen that a lower volume of accidents can be seen as occurring over the weekends, with Sunday (reported as day #1 in the data), shows the lowest volume of accidents.  The monthly plot shows a fairly consistent spread throughout the months of the year. The smaller number of accidents occurring in February could possibly be explained by the shorter month and factoring in the leap years, and no spike identified in the winter months, which one might have anticipated with the potentially worse weather conditions. The pattern seen in the hourly plot makes logical sense, showing spikes near the usual commuting times, fewer occurring at night, and only a small number occurring in the early hours of the morning.

Interestingly, though, when exploring casual fatalities as a percentage of all accidents, by hour, we can see that, although less accidents are reported as occurring in the early hours of the morning, the accidents which do occur during these hours may prove to be more fatal.

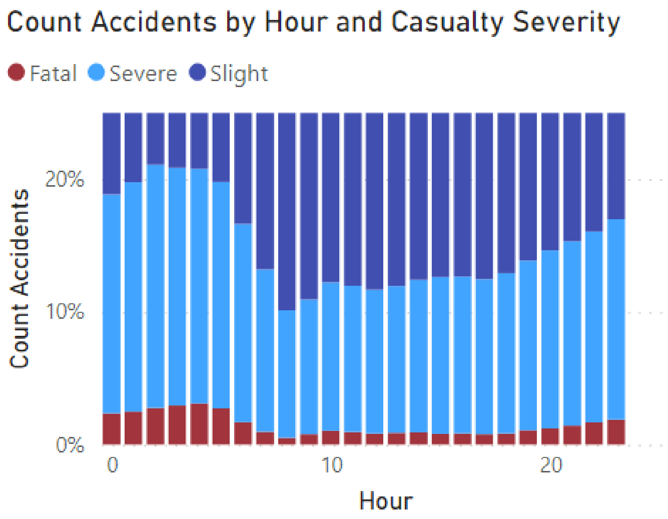


*Figure 5 - Casualty Fatalities as a % of Accidents, by Hour*

As can be seen in the figure above, a slight bump can be seen in the percentage of fatalities in accidents occurring from night-time through to the morning, a trend which remains consistent through each day of the week. Again, we could speculate the cause of this; possibly a lack of traffic leading to increased driving speeds, increased levels of driver tiredness, etc., but this insight is also useful from a predictive classification context, as it suggests that the time of day the accident occurs could be a useful in classifying an accident as fatal.

# 4. Methodology

This section details the steps taken to perform the main project analysis; describing the methods utilised to analyse, transform. An iterative analysis approach was taken in this project, phases of the analysis were revisited as required and adjustments made with the aim of improving the quality of the analysis.

## 4.1. First Iteration

Earlier iterations of the project and analysis involved the preparation of two additional working datasets which were modelled and used for training machine learning models. The methodology used for this first iteration of the analysis was ultimately discarded, as the feature selection and engineering techniques applied were not deemed to be of a sufficiently high standard. Details of the methodology for this first iteration can be found attached in the appendix.

## 4.2. Data Pre-processing

Data pre-processing involves preparing the data for analysis and can involve many different processes; cleaning the data, transforming the data by engineering features or creating new ones, selecting parts of the data to retain or discard. This section outlines the pre-processing methods utilised to prepare the collated dataset for modelling and predictive analysis. The various pre-processing tasks were wrapped in a function to allow for more efficient re-running and reuse.

### Environment Setup

The main development environment selected was Spyder4, a Python IDE popular for data science. A working directory was set up on the development machine, and directories were organised for the source data, plots, visualisation, python scripts, and results. The required libraries were installed and imported, and the collated source dataset was loaded to the environment to begin pre-processing.

A python module called "Pickle" was used throughout the project for faster loading of saved objects, such as the source dataset. Pickle involves serialising a Python objects into a byte-stream for storage and de-serialising the 'pickle' byte-stream back into the Python object when required – a particularly useful module when dealing with large datasets.

For all analysis techniques involving an element of randomness, a random seed of '1' was used, to maintain consistency through different iterations of processing or modelling and so that the analysis could be reproduced and verified.

### Data Cleaning

#### *Attribute Names*

The first pre-processing step carried out was to standardise the attribute names in the dataset for more efficient analysis and simplicity. When joining all of the source datasets to create the collated dataset, the name of original source dataset (e.g. Vehicles) was added as a prefix to the attribute names to identify the origin of the variables. This provided unnecessary noise and inconsistency, so these prefixes were removed. The attribute names were then cleaned using the Janitor Python library, a library which formats attribute names according to consistent set of rules (lower case, underscore-separated, symbols removed, etc.)

#### *Missing Values*

To maximise the number of predictor variables that could be used in the data modelling, it was decided that records containing any missing values would be dropped for the first iteration.

The general approach taken in the source datasets was for unknown data points to be assigned a value of (-1), a common approach taken in data reporting. However, there were also instances where data were missing entirely. The full dataset was analysed to identify these missing values (missing, null, n/a, None) and, where identified, replace them with a -1 to accord with the main approach taken. Attributes with too high a percentage of missing values were dropped from the analysis as these attributes could not be considered reliable.

In addition to missing values, several attributes contained a different (not -1) value, or values, which was reported as meaning "unknown" in the variable lookup dataset records. Attributes containing "unknown" class values were adjusted to have these values replaced with -1 in the dataset. Having standardised the value for all missing values, any rows containing missing values were then dropped from the dataset.

Imputing values with the most frequently occurring value, rather than dropping the records entirely, was considered, but due to the size of the dataset it was determined that dropping the records was an acceptable method to use for the first iteration and should not negatively impact the performance of the model.

## 4.3. Feature Selection

Several of the dataset's attributes were quickly selected for removal; reference attributes such as 'source name' or 'accident index' must be discarded, as these do not provide any information about the accident itself and would, at best, provide no benefit to the analysis, and, at worst, harm the performance of the models.

## Feature Transformation

A number of features underwent transformation during the pre-processing phase. The casualty severity attribute values were inverted so that the value increases with the severity of the casualty, instead of decreasing. While not strictly speaking necessary for analytic purposes, this felt like a more logical format to use throughout the project.

Additional ‘month’, and ‘hour’ attributes were created from the ‘accident date’ attribute, and the date attribute was then discarded. Similar to the index and reference columns, the date of an accident alone would not improve the predictive power of the project’s classification models. The ‘accident date’ attribute could prove useful without adjustment in linear regression models aiming to forecast future accidents, but this is not the aim of this project. The ‘month’ and ‘hour’ attributes, on the other hand, could prove to be useful predictor variables for classifying whether an accident was fatal, as was suggested by a number of the trends identified in the initial exploratory analysis and visualisation.

A grouping was applied to the longitude and latitude coordinates in preparation for machine learning. The coordinate values were rounded to the nearest whole number, a technique which groups the accidents into a larger geographical area, creating two new adjusted longitude and latitude attributes in place of the original coordinates. This created 53 unique geographical areas which could be used for classification purposes.

The final feature engineering adjustment carried out in the initial data pre-processing phase was to create the dependent variable which would be the target of the predictive analysis; the ‘casualty fatality’. This attribute was created from the ‘casualty severity’ attribute as a binary 1 or 0 value indicating whether the casualty was fatally injured in the accident. Following the creation of this attribute, the ‘casualty severity’ and ‘accident severity’ dependent variables were removed from the datasets, as these values contained a direct correlation to the target variable so their inclusion would ruin the integrity of any predictive analysis.

The creation of the target ‘casual fatality’ variable concludes the initial feature selection and feature engineering phase. The resultant prepared dataset is significantly reduced from the original collated source dataset, but still sufficiently large for classification analysis.

## Feature Encoding and Variable Types

As identified in the exploratory analysis, the vast majority of the features in the dataset contained numeric values; a combination of categorical, ordinal, and continuous values. To perform feature selection analysis on the most important features to include in the analysis, functions were written to encode attributes as required so as to prepare these attributes for a chi-squared test. From the Ski-kit Learn library, the Ordinal and Label Encoders from the pre-processing module were utilised to perform this feature analysis in conjunction with the ‘SelectKBest’ function from feature selection module, set to use the Chi2 function as its test method.

The predictor variables were encoded using the Ordinal Encoder and the target variable was encoded using the Label Encoder. The SelectKBest function accepts these two encoded datasets and measures the chi-squared statistic from each of the predictor variables to the target variable, returning a specified number of the most related features. Functions were written to fit these functions to the datasets and transform the datasets accordingly, returning the updated datasets; a training and testing dataset containing encoded variables in the case of the two encoders, and a training dataset containing only the selected features, in the case of the SelectKBest function.

Additional functions were also written to adapt the SelectKBest function to return only the attribute names, as opposed to transformed datasets; details of why the decision was made to make this adjustment can be found in the Analysis section.

## 4.4. Data Modelling

### Dataset Partitioning

Before the dataset can be modelled for predictive analysis it must be partitioned into training and testing subsets, which can be used to train the models and evaluate their performance. Because of the extreme imbalance in the target variable, two main approaches were taken to partition the data into training and testing subsets in preparation for modelling. Functions were written so that these splits could be performed efficiently.

#### *Representative Train-Test Split*

The first approach taken was to perform a stratified split of the dataset, using the target variable as the basis for the split, and forming training and testing subsets of the data that are both as representative as possible of the full prepared dataset. The 'stratified shuffle split' function, from Python's 'scikit-learn' library, was used to perform this split and accepts a dataset and a target variable and splits the dataset in such a way that it is as representative as possible of the original dataset, meaning that approximately the same ratio of fatal to non-fatal casualties should be included in the training and testing subsets.

#### *Balanced Sample Train-Test Split*

In attempt to counter the extreme imbalance in the target variable, a second, more balanced dataset was also prepared for modelling. This balanced dataset was created by taking all 'fatal' casualty observations records and retaining the same number of the abundant 'non-fatal' casualty observations, randomly selected from the dataset. This technique is known as under-sampling, a technique utilised in similar predictive analyses involving imbalanced data where there is only a small chance of an interesting event in each observation.

#### *Separation of Target Variable*

Following the partitioning of the prepared dataset into training and testing subsets, the target variable was separated from the predictor variables and stored as a separate series. The standard naming conventions were used for these partitioned subsets. This process of separating the predictor and target variables is required for training the models, as the models accept a data-frame of training predictor variables and create associations with the corresponding target variables (labels). The model is then run on the testing predictor variables and the accuracy with which the model can predict the testing target variable used as the basis for evaluation.

### Model Training

For training the selection of models to be included in the analysis, a function was created to import the required modules for each of the selected models. These modules were imported from the Ski-kit Learn library. In turn, each model was fitted, or trained, with the training dataset and was then used to make predictions using the test predictor dataset.

The desired parameters were specified for each model and, in keeping with the approach taken throughout the project, where a model involved an element of randomness, the random state was set to a value of 1, to ensure that models could be run and re-run as required while maintaining the consistency and integrity of the analysis. This process was performed firstly on the full representative datasets, before then being performed on the balanced dataset.

## 4.5. Model Analysis and Evaluation

After each model was trained, the model was used to make predictions for the test dataset's target variable  using the testing predictor variables.  The accuracy scores were recorded and were written to a model results table along with the model name, the accuracy confusion matrix, additional parameter details, and this results table was then stored and printed to CSV file for additional analysis.

Once the model scores were recorded and analysed, additional iterations of training, testing and evaluating were carried out to see if the initial results could be improved upon.  The results of the feature selection analysis, carried out to investigate the impact of reducing the number of features used as predictor variables on the accuracy scores, were recorded and visualised.

# 5. Analysis

This section details the predictive analysis carried out using selected machine learning classification models and provides an overview of each model used.  The classification models were all imported from the Sci-kit Learn module and were originally selected for their ability to deal with multi-class classifications, as opposed to a binary classification solution.  Although, focus of the analysis was ultimately placed on predicting casualty fatalities.

The process for training and running these models is covered in the previous Methodology section and the analysis results can be seen in the following Results section.

## 5.1. Decision Tree Classifier

The first model selected for analysis was the trusty decision tree, this this module was taken from Sci-kit Learn's DecisionTreeClasifier module.

Decision trees are an example of a supervised machine learning method which can be used for both regression and classification problems.  Decision trees operate by developing a set of decision rules based on the input predictor variables, which it will follow when make predictions about the target variable.  With each row of predictor variables and corresponding target label that is passed to the decision tree, the model gradually improves and fine-tunes its set of decision rules.  Decision trees are very versatile and can accept both categorical and numeric datatypes; continuous and ordinal, although it is worth noting that the Sci-kit Learn documentation notes that the Decision Tree module utilised in this analysis cannot currently accept categorical data – this did not provide much of an issue as the majority of the dataset's features were numeric.

In addition to the robustness of the Decision Tree model, one of the benefits of using decision trees is they are quite intuitive to interpret, the most important attributes in the decision tree can be seen in at the top of the tree, as these will be the first decisions made on which branches the decision will follow.

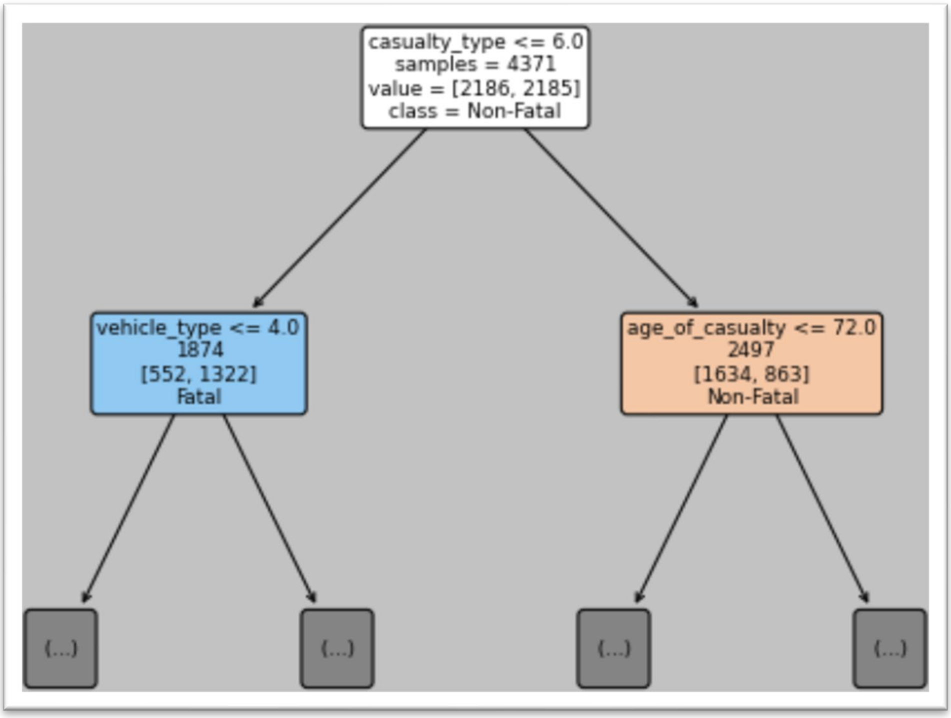


*Figure 6 - Decision Tree at depth 1, trained on balanced dataset*

As can be seen in the figure above, the primary feature used in its predictions by the Decision Tree trained on the balanced dataset, is the ‘casualty type’; the first decision rule in the model investigating whether the casualty type is below 6.

Depending on the whether or not this first decision rule is true or false, the next decision checked by the model will either be whether the casualty is under the age of 72.5, in which case the probability that the accident is non-fatal increases, or whether vehicle type is in a category below four, respectively.

The number of samples which reached each node is also indicated in the plotted tree, as well as the number of training observations which branched in either direction based on the rule. The classification result that the truthiness of each rule suggests is indicated as “Fatal” “Non-Fatal”, and the nodes are also coloured based on this value.

| Training Dataset | Max Depth | Leaves |
|---|---:|---:|
| Representative Set | 35 | 7895 |
| Balanced Set | 26 | 1065 |

The full decision tree can also be plotted, but due to the size of the tree, no information could be read from such a plot. Instead, a text representation of the decision tree can be generated for a full view of the tree and all its decision rules.

Decision trees were trained using the both the balanced dataset and the larger representative dataset using the default parameter settings. The results will be provided in the Results Section.

## 5.2. Random Forest Classifier

A Random Forest is an algorithm which operates by folding the training data into multiple training and validation sets and generating multiple decision tree models which are then trained with each of the folded subsets of the training data.  The prediction of each of the trained decision trees is then aggregated by the algorithm to inform the Random Forest's prediction. Models such as this are referred to as "ensemble models" and uses the concept of 'wisdom of the crowd' in its predictions.

As a result of this, comparing the predictive performance of a Random Forest and a Decision Tree algorithm could be considered a redundant and unfair exercise.  However, when investigating which classification models are most effective, it is a must-have model.

The Random forest classification model used in this analysis was imported from Sci-kit Learn's RandomForestClassifier module and, imported, fitted and used to make predictions in the same manner as the Decision Tree.

The results for the Random Forest Classifier Model will be provided in the Results section.

## 5.3. Gaussian Naïve Bayes

The first model used which doesn't involve decision trees is the Gaussian Naïve Bayes Classifier model.  This model is form of Naïve Bayes supervised learning model.  Sci-kit Learn's documentation notes that Naïve Bayes classifiers follow the 'naïve' assumption that the features' values are independent of one another, and the Gaussian Naïve Bayes classifier utilises the Gaussian Naïve Bayes classification Algorithm to make predictions and assumes a Gaussian distribution.

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Where σy and µy are estimated using maximum likelihood.[5]

This classifier assumes a Gaussian distribution, whereas the Categorical Naïve Bayes which describes the probability of the possible results of a random variable which come in the form of a given number of categories, might have been a more appropriate model to evaluate.

## 5.4. Multi-Layer Perceptron (MLP) Classifier

The MLP Classifier is another supervised learning algorithm which can be used for predictive classification. The MLP classifier operates by accepting a dataset of predictor features and a corresponding target variable and, using back-propagation it can develop a non-linear approximator function to make its predictions.

The MLP classifier was initially selected as one of the classification models to be included in the analysis.

## 5.5. Gradient Boosting Machine

The Gradient Boosting Machines can be used for both classification and regression problems

Feature Selection Model Accuracy Analysis

# 6. Results

This section outlines the results of the predictive analysis carried out on the prepared dataset. The scores of the models have been calculated using a variety of metrics which are based on the models confusion matrices.

Evaluation Metric

The confusion matrix is an evaluation tool used for evaluating the performance of classification models, by recording:

- Accidents which were correctly classified as non-fatal
- Accidents which were classified as being fatal, which were actually non-fatal
- Accidents which were classified as non-fatal, which were actually fatal
- Accidents which were correctly classified as non-fatal

The metric used to score the predictive performance of the various machine learning classification models analysed is the accuracy % – calculated as the correct predictions / all predictions.

## 6.1. Analysis Results

Having prepared the dataset, performed the required pre-processing and data-modelling steps, the predictive models were evaluated and the results were compiled and recorded.

The results of the machine learning classification analysis carried out to predict casualty fatalities in road accidents in the UK is summarised in the table below.

Model Scores

| Model | Training Data | Confusion Matrix | Features Used | Accuracy Score (%) |
|---|---|---|---|---|
| **Random Forest** | Balanced | [457 89]<br>[ 50 497] | ALL | **87.28** |
| **Gradient Boosting Machine** | Balanced | [447 99]<br>[ 77 470] | ALL | **83.90** |
| **Decision Tree** | Balanced | [438 108]<br>[ 88 459] | ALL | **82.07** |
| **Gaussian Naive Bayes** | Balanced | [424 122]<br>[155 392] | ALL | **74.66** |
| **MLP Neural Net Classifier** | Balanced | [435 111]<br>[176 371] | ALL | **73.74** |

As we can see from the above results table, the classification model with the highest accuracy score is the Random Forest Classification model, with an accuracy score of 87.28%. This result might come as no surprise given the nature of the Random Forest algorithm, but it was interesting to see how the performance of the various models lined up.

The Gradient Boosting Machine classifier attained the second highest accuracy score, as another form of ensemble machine learning algorithm, with the decision tree achieving the third highest scores. The performance

## 6.2. Features Selection Analysis Results

The results were documented initially with the accuracy scores achieved when evaluated on the test data with all selected attributes retained. However, as described in the Methodology Section, an additional feature selection analysis was carried out, where a chi-square test was performed on all features to determine which features most impacted the models' predictions.

This chi-squared feature selection method was performed to record the most features most related to the target variable, and store these features in an ordered table. The table was then iterated through and the models were run with the top K features, starting with the single most important feature and then gradually adding the next most important feature, as indicated by the Chi-Squared function of the "SelectKBest" function.

The results of this analysis can be seen in the graph below.

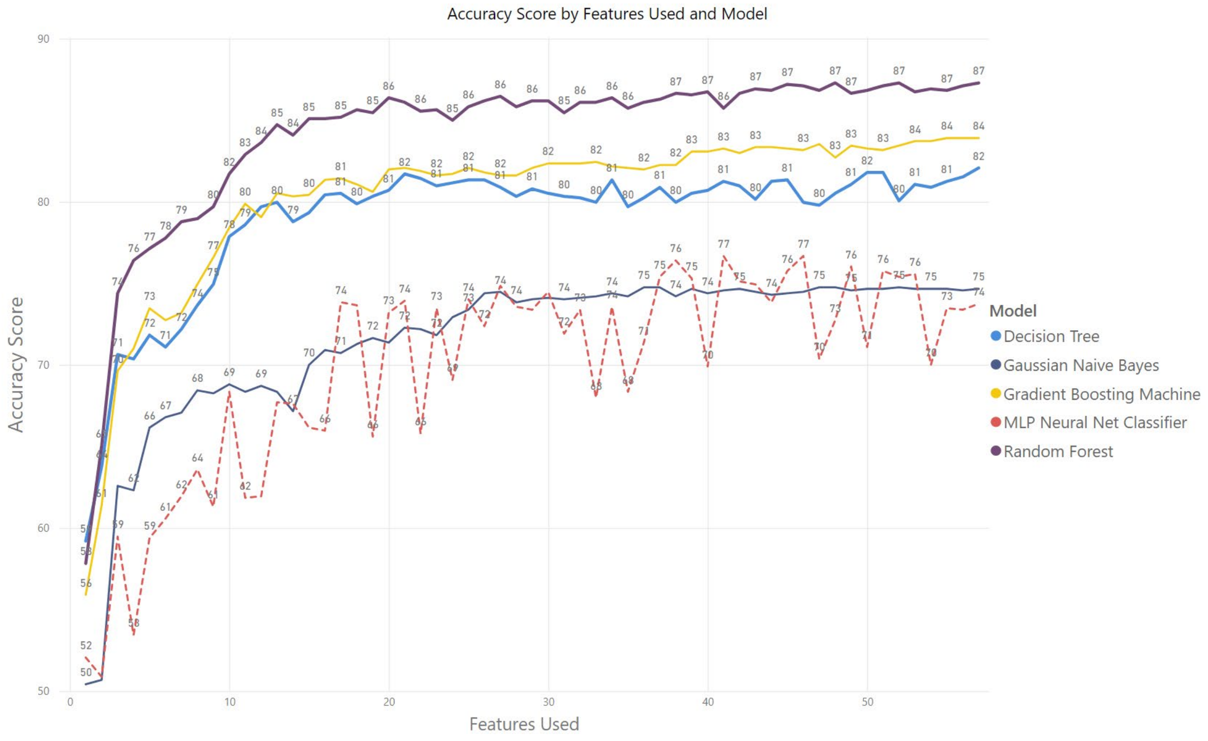
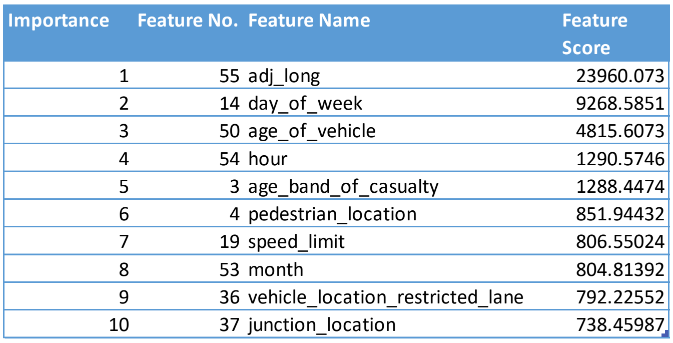


*Figure 7 - Accuracy of Models by Number of Features Used*

As we can see, the Random Forest remains the most impressive scoring model with any number of features, with the Gradient Boosting Machine maintaining its position as second highest scoring. In almost all cases, there is a steady increase as the first key features are added to the analysis, being the principal components for the prediction. From the 20th added feature, the increasing accuracy scores slows down, and in the case of the decision tree showing an occasional decrease in score, which came as somewhat of a surprise and bears further investigation.

The steady performance of the Gaussian Naïve Bayes model provides a stark contrast to the MLP Classifier which demonstrates some extremely erratic behaviour.

A summary of the top-10 most important features, as outlined by the Chi-Square SelectKBest modules can be seen in the table below:



| Importance | Feature No. | Feature Name | Feature Score |
|---|---|---|---|
| 1 | 55 | adj_long | 23960.073 |
| 2 | 14 | day_of_week | 9268.5851 |
| 3 | 50 | age_of_vehicle | 4815.6073 |
| 4 | 54 | hour | 1290.5746 |
| 5 | 3 | age_band_of_casualty | 1288.4474 |
| 6 | 4 | pedestrian_location | 851.94432 |
| 7 | 19 | speed_limit | 806.55024 |
| 8 | 53 | month | 804.81392 |
| 9 | 36 | vehicle_location_restricted_lane | 792.22552 |
| 10 | 37 | junction_location | 738.45987 |

## 7. Conclusions

This analysis proved to be a very interesting and challenging project.  The exploratory analysis carried out on the data provided great insight into the field of road accidents occurring in the UK with many interesting points of interest noted and investigated.

The exploratory analysis provided a glimpse of some of the interesting detail contained within the data, and addressed the first two research questions.

The classification analysis carried out to predict casualty fatalities in road accidents proved to be successful with some very reasonable accuracy scores attained.

The approach taken to performing the analysis was iterative and additional techniques and enhancements were constantly being applied.  The majority of development time was filled with data pre-processing and preparation, details of additional analysis and improvements to the analysis performed

The results of the analysis; the accuracy of the classification models in predicting casualty fatalities based on the attributes of the road accident, were reasonable, if not entirely unexpected.

## 8. Further Research and Development

With additional development time, the following additional analysis would be carried out by the researcher:

- Increased exploratory data analysis and visualisation
- Review of categorical feature encoding, some attributes behaved strangely at times, possibly due to incorrect attribute encoding
- Additional models would be trained and utilised
- Improved functions would be written for reusability

If I could restart the project and do things differently, I would have:

- Spent less time manually adjusting attributes and engineering features based on intuition. This cost a lost of development time for analysis which was ultimately not used.

# 9. References

[1] "Global status report on road safety 2018." [Online]. Available: https://www.who.int/publications/i/item/9789241565684.

[2] "2nd UN Decade of Action for Road Safety- 2021-2030 - EuroRAP." [Online]. Available: https://eurorap.org/2nd-un-decade-of-action-for-road-safety-2021-2030/.

[3] "Reported road casualties Great Britain, annual report: 2019 - GOV.UK." UK Department for Transport, [Online]. Available: https://www.gov.uk/government/statistics/reported-road-casualties-great-britain-annual-report-2019.

[4] R. Foundation, "https://www.racfoundation.org/motoring-faqs/mobility#a26." https://www.racfoundation.org/motoring-faqs/mobility#a26.

[5] "SKLearn GNB Classifier Documentation." https://scikit-learn.org/stable/modules/naive_bayes.html#gaussian-naive-bayes.

# 10. Appendices

# National College of Ireland

## Project Proposal

## Predictive Analysis of UK Road Accidents Using Machine Learning

26/10/2020

Bachelor of Science (Hons) in Computing

BSCHEDA

Data Analytics

2020/2021

Alan Patterson

x18108105

x18108105@student.ncirl.ie

Project Supervisor – Giovani Estrada

# Contents

## Objectives

The main objective of this project is to investigate the effectiveness of various machine learning models in predicting the frequency and accident-severity classification of road accidents in the U.K., and to measure the performance of these models.

In order to meet this objective, an exploratory study of UK road accidents data will be conducted to gain insight into a wide range of attributes to discover the key factors which contribute to the volume and severity of road accidents, on both an overall basis and on various levels of aggregation. This analysis will inform the selection and engineering of features to be included in the project's predictive analysis.

An additional motivation for this project is to gain an understanding of and familiarity with various machine learning models, as well as increasing knowledge of statistical analysis and researching techniques for use in future studies.

## Background

Road traffic accidents remain a very serious problem worldwide; responsible for approximately 1.35 million deaths each year, an average of just under 3,700 deaths per day. Globally, road traffic accidents are the 8th leading cause of death across all age groups and the primary cause of death amongst children and young adults (aged 5 – 29), with vulnerable road users accounting for over half of these deaths; pedestrians and cyclists making up 26% and a further 28% relating to users of motorbikes and 'three-wheelers' [1].

In March 2010, the United Nations declared the years 2010 – 2020 were to be a decade of action for road safety, ambitiously aiming to halve the number of road fatalities and serious injuries by 2020. Although the UN did fall short of achieving this goal, it has declared a second decade of action for road safety, from 2021 -2030, encouraging member states to take a number of positive actions with the renewed aim of reducing road traffic deaths and serious injuries by half, by 2030 [2].

Understanding the factors that contribute to the frequency and severity of road traffic accidents is crucial to implementing effective preventative measures, and this can only be achieved through the collection, collation, and analysis of accurate data. This project will centre around the analysis of UK road accident data provided by the UK Department of Transport following an increase in demand for more updated and detailed information to be made available to the public.

Each year, the UK government also provides informative road accident statistical reports. In 2019, there were a total of 1,752 road traffic deaths reported and a total of 153,158 reported casualties of all severities, 25,945 of which suffered serious injuries. The trend in road traffic deaths in the UK has been relatively flat from 2010 to 2019, as can be seen when comparing the 2019 figures with the 1,754 road deaths occurring in 2012; indicating a slight reduction when factoring-in the increased traffic and vehicle numbers [3].

The prediction of road accident frequency and severity is an essential component in trying to minimise the impact of these accidents, as such, there is vast amount of research done in this field; government bodies, health and safety organisations, insurance and reinsurance companies, all have a vested interest in maximising the predictive performance of their analytical models. Generalised Linear Modelling (GLM) has been a popular choice by actuaries for forecasting claim frequencies and severities, but with the adaptability and flexibility of various machine learning models and their combinations, the predictive performance of machine learning models is an interesting field to study and compare. GLMs takes the traditional normal linear model and generalises it by relaxing the some of its restrictions and facilitates the analysis of non-normal data, insurance claims data being one such example.

With the recent, significant technological advancements in the field of autonomous vehicles, the first fully autonomous car predicted to be made available to the market in the next year or so, and as increased volumes of telematics data is collected and stored, the question of road safety and the effects of fully autonomous vehicles being in circulation will be a topic of great importance and of great interest to analysts.

## Technical Approach

This project will follow what could be broadly considered to be an Agile CRISP-DM (Cross-Industry Standard Process for Data Mining) approach. CRISP-DM segments a project into the following six phases:

1. Business Understanding
2. Data Understanding
3. Data Preparation
4. Data Modelling
5. Evaluation
6. Deployment

Throughout the course of this project, steps 1 – 5 of the CRISP-DM approach will repeated iteratively as new insight is gained through the exploratory analysis and during the training and measuring of the performance of the selected machine learning models, before proceeding to the final phase which, in the case of this project, will entail documenting and presenting the research findings and conclusions in a final project report, and then presenting this report.

A more detailed description of this project's phases and task items can be found in the section 5 – Project Plan.

**In the context of this project:**

The Busines Understanding – refers to the objectives of the project, to investigate the effectiveness of machine learning models in predicting the frequency and severity of road accidents, and the project preparation, planning, and management. The decisions made throughout the project should align with this goal.

The Data Understanding – refers to exploratory analysis and initial review of data structure, quality, and attributes. This will be part of the iterative cycle, as additional insight is gained and areas of interest are uncovered.

The Data Preparation – refers to the iterative process of cleaning the data and preparing it for the machine learning models, by selecting, engineering, adding, and scaling features, etc. This process will be repeated throughout the project and in parallel with Data Understanding and Modelling phases.

The Data Modelling – refers to the training, fine tuning, and measurement of the machine learning models and the shortlisting of the most promising models to be measured against the test data and compared with the industry standard.

The Evaluation – will refer to measuring the performance of the selected models using the test data and comparing the results with the industry standards. The results will be documented and conclusions drawn.

The Deployment – will refer to the submission and presentation of the final project report.

## Special Resources Required

In addition to the materials listed in the References section, the special resources listed below will also be utilised and this list will be updated as required throughout the course of the project:

### Textbooks

*De Jong, P., & Heller, G. (2008). Generalized Linear Models for Insurance Data (International Series on Actuarial Science). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511755408*

### Project Management:

- Gantt chart maintained on https://app.teamgantt.com/

### System:

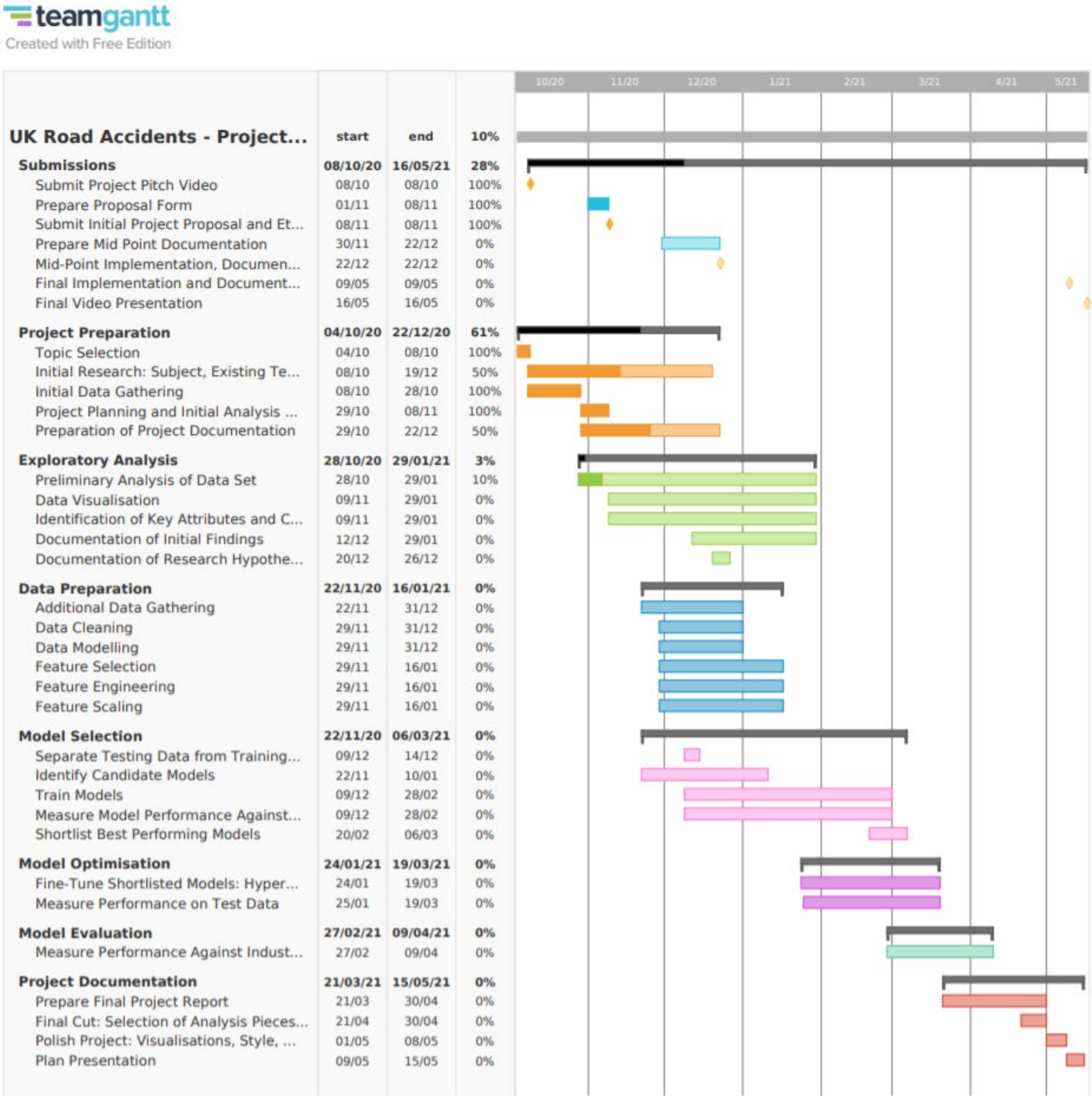
- Project analysis performed on a PC with the following specifications:

| | |
|---|---|
| Operating System: | Windows 10 Pro |
| Processor: | 3.59 GHz AMD Ryzen 5 3600 6-Core Processor |
| Installed Memory (RAM): | 16.0GB |
| System Type: | 64-bit Operating System, x64-based processor |

## Project Plan

An outline of the project’s main phases and tasks. Many aspects of the project will be performed iteratively and in parallel, as can be seen from the overlapping periods shown below.

## Project Gantt Chart



| UK Road Accidents - Project... | start | end | 10% |
|---|---|---|---|
| **Submissions** | **08/10/20** | **16/05/21** | **28%** |
| Submit Project Pitch Video | 08/10 | 08/10 | 100% |
| Prepare Proposal Form | 01/11 | 08/11 | 100% |
| Submit Initial Project Proposal and Et... | 08/11 | 08/11 | 100% |
| Prepare Mid Point Documentation | 30/11 | 22/12 | 0% |
| Mid-Point Implementation, Documen... | 22/12 | 22/12 | 0% |
| Final Implementation and Document... | 09/05 | 09/05 | 0% |
| Final Video Presentation | 16/05 | 16/05 | 0% |
| **Project Preparation** | **04/10/20** | **22/12/20** | **61%** |
| Topic Selection | 04/10 | 08/10 | 100% |
| Initial Research: Subject, Existing Te... | 08/10 | 19/12 | 50% |
| Initial Data Gathering | 08/10 | 28/10 | 100% |
| Project Planning and Initial Analysis ... | 29/10 | 08/11 | 100% |
| Preparation of Project Documentation | 29/10 | 22/12 | 50% |
| **Exploratory Analysis** | **28/10/20** | **29/01/21** | **3%** |
| Preliminary Analysis of Data Set | 28/10 | 29/01 | 10% |
| Data Visualisation | 09/11 | 29/01 | 0% |
| Identification of Key Attributes and C... | 09/11 | 29/01 | 0% |
| Documentation of Initial Findings | 12/12 | 29/01 | 0% |
| Documentation of Research Hypothe... | 20/12 | 26/12 | 0% |
| **Data Preparation** | **22/11/20** | **16/01/21** | **0%** |
| Additional Data Gathering | 22/11 | 31/12 | 0% |
| Data Cleaning | 29/11 | 31/12 | 0% |
| Data Modelling | 29/11 | 31/12 | 0% |
| Feature Selection | 29/11 | 16/01 | 0% |
| Feature Engineering | 29/11 | 16/01 | 0% |
| Feature Scaling | 29/11 | 16/01 | 0% |
| **Model Selection** | **22/11/20** | **06/03/21** | **0%** |
| Separate Testing Data from Training... | 09/12 | 14/12 | 0% |
| Identify Candidate Models | 22/11 | 10/01 | 0% |
| Train Models | 09/12 | 28/02 | 0% |
| Measure Model Performance Against... | 09/12 | 28/02 | 0% |
| Shortlist Best Performing Models | 20/02 | 06/03 | 0% |
| **Model Optimisation** | **24/01/21** | **19/03/21** | **0%** |
| Fine-Tune Shortlisted Models: Hyper... | 24/01 | 19/03 | 0% |
| Measure Performance on Test Data | 25/01 | 19/03 | 0% |
| **Model Evaluation** | **27/02/21** | **09/04/21** | **0%** |
| Measure Performance Against Indust... | 27/02 | 09/04 | 0% |
| **Project Documentation** | **21/03/21** | **15/05/21** | **0%** |
| Prepare Final Project Report | 21/03 | 30/04 | 0% |
| Final Cut: Selection of Analysis Pieces... | 21/04 | 30/04 | 0% |
| Polish Project: Visualisations, Style, ... | 01/05 | 08/05 | 0% |
| Plan Presentation | 09/05 | 15/05 | 0% |

## Project Work Items

| WBS # | Name / Title | Type | Start Date | End Date | Percent Complete |
|---|---|---|---|---|---|
| **1** | **UK Road Accidents - Project Plan** | **project** | **04/10/2020** | **16/05/2021** | **9.87** |
| **1.1** | **Submissions** | **group** | **08/10/2020** | **16/05/2021** | **27.78** |
| 1.1.1 | Submit Project Pitch Video | milestone | 08/10/2020 | 08/10/2020 | 100 |
| 1.1.2 | Prepare Proposal Form | task | 01/11/2020 | 08/11/2020 | 100 |
| 1.1.3 | Submit Initial Project Proposal and Ethics Form | milestone | 08/11/2020 | 08/11/2020 | 100 |
| 1.1.4 | Prepare Mid Point Documentation | task | 30/11/2020 | 22/12/2020 | 0 |
| 1.1.5 | Mid-Point Implementation, Documentation, Video Presentation | milestone | 22/12/2020 | 22/12/2020 | 0 |
| 1.1.6 | Final Implementation and Documentation | milestone | 09/05/2021 | 09/05/2021 | 0 |
| 1.1.7 | Final Video Presentation | milestone | 16/05/2021 | 16/05/2021 | 0 |
| **1.2** | **Project Preparation** | **group** | **04/10/2020** | **22/12/2020** | **61.21** |
| 1.2.1 | Topic Selection | task | 04/10/2020 | 08/10/2020 | 100 |
| 1.2.2 | Initial Research: Subject, Existing Techniques, Existing Studies, Existing Models | task | 08/10/2020 | 19/12/2020 | 50 |
| 1.2.3 | Initial Data Gathering | task | 08/10/2020 | 28/10/2020 | 100 |
| 1.2.4 | Project Planning and Initial Analysis Questions | task | 29/10/2020 | 08/11/2020 | 100 |
| 1.2.5 | Preparation of Project Documentation | task | 29/10/2020 | 22/12/2020 | 50 |
| **1.3** | **Exploratory Analysis** | **group** | **28/10/2020** | **29/01/2021** | **2.99** |
| 1.3.1 | Preliminary Analysis of Data Set | task | 28/10/2020 | 29/01/2021 | 10 |
| 1.3.2 | Data Visualisation | task | 09/11/2020 | 29/01/2021 | 0 |
| 1.3.3 | Identification of Key Attributes and Correlations | task | 09/11/2020 | 29/01/2021 | 0 |
| 1.3.4 | Documentation of Initial Findings | task | 12/12/2020 | 29/01/2021 | 0 |
| 1.3.5 | Documentation of Research Hypotheses | task | 20/12/2020 | 26/12/2020 | 0 |
| **1.4** | **Data Preparation** | **group** | **22/11/2020** | **16/01/2021** | **0** |
| 1.4.1 | Additional Data Gathering | task | 22/11/2020 | 31/12/2020 | 0 |
| 1.4.2 | Data Cleaning | task | 29/11/2020 | 31/12/2020 | 0 |
| 1.4.3 | Data Modelling | task | 29/11/2020 | 31/12/2020 | 0 |
| 1.4.4 | Feature Selection | task | 29/11/2020 | 16/01/2021 | 0 |
| 1.4.5 | Feature Engineering | task | 29/11/2020 | 16/01/2021 | 0 |
| 1.4.6 | Feature Scaling | task | 29/11/2020 | 16/01/2021 | 0 |
| **1.5** | **Model Selection** | **group** | **22/11/2020** | **06/03/2021** | **0** |
| 1.5.1 | Separate Testing Data from Training DataÂ | task | 09/12/2020 | 14/12/2020 | 0 |
| 1.5.2 | Identify Candidate Models | task | 22/11/2020 | 10/01/2021 | 0 |
| 1.5.3 | Train Models | task | 09/12/2020 | 28/02/2021 | 0 |
| 1.5.4 | Measure Model Performance Against Test Data | task | 09/12/2020 | 28/02/2021 | 0 |
| 1.5.5 | Shortlist Best Performing Models | task | 20/02/2021 | 06/03/2021 | 0 |
| **1.6** | **Model Optimisation** | **group** | **24/01/2021** | **19/03/2021** | **0** |
| 1.6.1 | Fine-Tune Shortlisted Models: Hyperparameters, Potential Combinations of Models | task | 24/01/2021 | 19/03/2021 | 0 |
| 1.6.2 | Measure Performance on Test Data | task | 25/01/2021 | 19/03/2021 | 0 |
| **1.7** | **Model Evaluation** | **group** | **27/02/2021** | **09/04/2021** | **0** |
| 1.7.1 | Measure Performance Against Industry Standard | task | 27/02/2021 | 09/04/2021 | 0 |
| **1.8** | **Project Documentation** | **group** | **21/03/2021** | **15/05/2021** | **0** |
| 1.8.1 | Prepare Final Project Report | task | 21/03/2021 | 30/04/2021 | 0 |
| 1.8.2 | Final Cut: Selection of Analysis Pieces and Visualisations | task | 21/04/2021 | 30/04/2021 | 0 |
| 1.8.3 | Polish Project: Visualisations, Style, Design Enhancements | task | 01/05/2021 | 08/05/2021 | 0 |
| 1.8.4 | Plan Presentation | task | 09/05/2021 | 15/05/2021 | 0 |

## Technical Details

This project will be implemented using the technologies and libraries listed below. As the project progresses, this list will be updated:

**IDEs:**

- Jupyter Lab / Notebook
- RStudio

Jupyter Lab / Notebook is a lightweight IDE which is very effective and efficient for data analytics, with Python in particular, and will be used primarily for the exploratory analysis.

RStudio will also be utilised, primarily for the generalized linear models (GLM), but might also be used for other statistical analysis.

**Python:**

- Pandas
- NumPy
- Scikit Learn
- Matplotlib's pyplot

Python will be this project's language, chosen for its versatility, simplicity, and for the abundance of useful and easily accessible analysis libraries, some of which are listed above.

Scikit is a package which includes useful machine learning models.

**R:**

- Generalized Linear Models

## Evaluation

To evaluate the predictive performance of each selected machine learning model, a portion of the data (the test data) will be set aside and will not be included in the data used to train each model (the train data). During the training of each model, a subset of the train data will be set aside for validation (validation set). Following the training of the candidate models and the optimisation of the best performing models, the test data will be used to measure the final performance of each model.

The project will investigate whether the selected machine learn models perform effectively in predicting accident frequency and severity.

## References

[1] "Global status report on road safety 2018." [Online]. Available: https://www.who.int/publications/i/item/9789241565684.

[2] "2nd UN Decade of Action for Road Safety- 2021-2030 - EuroRAP." [Online]. Available: https://eurorap.org/2nd-un-decade-of-action-for-road-safety-2021-2030/.

[3] "Reported road casualties Great Britain, annual report: 2019 - GOV.UK." UK Department for Transport, [Online]. Available: https://www.gov.uk/government/statistics/reported-road-casualties-great-britain-annual-report-2019.

Any other reference material used in the project for example evaluation surveys etc.

## 1. EARLY ITERATION METHODOLOGY

### Pre-Processing

The first phase of data preparation involved the exploration, collation, and cleaning of the source datasets. Each dataset was appended to form complete 'accident', 'vehicles' and 'casualties' datasets compiling all records for each category.

This process was carried out using both Excel Power Query for simplicity purposes, as each category of dataset maintained a uniform structure:

Each category of dataset was placed in its own folder within the project's 'data' directory. Excel's Power Query imported each of the CSV files for each category and appended these to form a complete dataset for each category, before exporting the complete dataset as a new CSV file to be used in further analysis.

While performing this initial collation of datasets, the complete datasets were also loaded into Excel's Power Pivot data model, linking each dataset based on their primary keys, to get a quick visual overview of the datasets before the main data preparation process started.

### Preliminary Data Preparation

#### *Creating the Accident Working-Dataset*

The data was read into Spyder, a Python IDE popular for data science, and the process of cleaning and preparing the data was initiated.

#### Basic Cleaning

The first step was to identify missing or unknown data in the set; all attributes were searched initially for 'null' or 'n/a' values using a combination of Pandas methods. There were null values contained within the dataset, so the next step was to analyse the number of values with a value of -1, which, the Variable Lookup File explains, means that the data point is unknown or unavailable.

Attributes, or columns, with high quantities of values reported as missing or unknown were dropped from the analysis, as they could not be considered reliable. Easting and Northing datapoints were also dropped to avoid duplication, as longitude and latitude datapoints were available. From the remaining attributes, rows which contained negative values in any attribute, other than the longitude and latitude, were dropped from the dataset. The decision to drop these rows as opposed to trying to impute or deduce the values, was made in part because the dataset is so large that the dropped records would be quite insignificant, and partly because the majority of the attributes are categorical variables which have been assigned a numeric value as their code, so due to the project's time-constraints it was unfeasible to allocate much resources into analysing which other techniques for handling these unknown values would be more effective, if one exists.

Following this, the data types ('dtypes' in Python) for each attribute was adjusted appropriately. This involved formatting the categorical variables as 'categories', updating the 'longitude' and 'latitudes' to be float types, and adjusting the date from a String to a datetime object.

The next task was to merge the "Accident Adjustments" dataset with the Accident dataset, to add the "adjusted severity" adjustment variable to the Accident dataset, to ultimately be used as the predicted 'Y' variable.

### Merging Adjusted Severity

To accomplish this, the Accident Adjustments dataset was read to the working Spyder Python script, with a 'left join' merge being performed on the dataset, meaning that all records from the Accident dataset were maintained, regardless of whether there was a corresponding Adjustment value. Initially an 'inner join' merge was performed, however, it was uncovered that accidents which had a severity of 1 (fatal) did not appear in the adjustment file, as the severity of these accidents would not be adjusted. So, the left outer merge was the selected merge technique; with records without corresponding severity adjustments receiving an adjustment of 0, achieved by replacing the missing values with zero.

Following an initial unsuccessful merge attempt, it was identified that there were consistency issues with the reading of the 'accident_index' variables, with variables being formatted differently in the Accident and Adjustments datasets, and with some instances containing values incorrectly converted to integers from Strings. This was amended by stripping out leading and trailing spaces from both datasets' identifiers.

#### *Creating the Casualty Working-Dataset*

Following the creation of the Accident dataset, the process of creating the Casualty working dataset began.

Along with the original Casualty dataset, the created previous prepared Accident dataset was imported to the working Spyder environment, as was the original Vehicle dataset.

The same initial process was carried out to select variables for the Casualty and Vehicle datasets, as was performed for the Accidents datasets; attributes with high volumes of missing information or unknown values were excluded and attributes considered to be interesting or potential to have value, were maintained.

To merge the selected attributes from the Accident dataset with the selected Casualty dataset attributes, an inner-join was performed between the two sets using the 'accident_index' variable.

This join was selected so as to exclude any records from the Casualty dataset that do not have a corresponding Accident record.

The updated Casualty dataset was then merged again with the selected attributes from the cleaned Vehicle dataset, by creating a composite key in both datasets using the ‘accident_index’, ‘casualty reference’, and ‘vehicle_reference’ variables.  Again, an inner join was selected to exclude records which were missing corresponding records in the other sets.

## Analysis

Following the initial preparation of the working datasets a more detailed analysis of each of the selected attributes was carried out on the Accident dataset.  While the initial preparatory analysis had been performed in the Spyder IDE, this next phase of analysis was performed using Jupyter Lab. The working Accident dataset was imported into the environment and a ‘pickle’ file was created for convenient storing of the environment state, as opposed to reading from CSVs.

With the goals of the project in mind; to explore the climate of road accidents in the UK and to investigate the effectiveness of machine learning techniques in the prediction of accident frequency and severity, the accident severity had initially been the prime candidate for the Y (predicted) variable in the machine learning models selected.  As well as to explore the data to gain insight, this phase of analysis aimed to determine what additional preparatory steps are required before the first machine learning model is tested on the data.

### Exploration Through Visualisation

The first action taken was to visually plot the dataset’s variables, to get a good picture of the data.  A number of different techniques were utilised to analyse the dataset including; count plots, histograms, scatter plots, and bar charts.

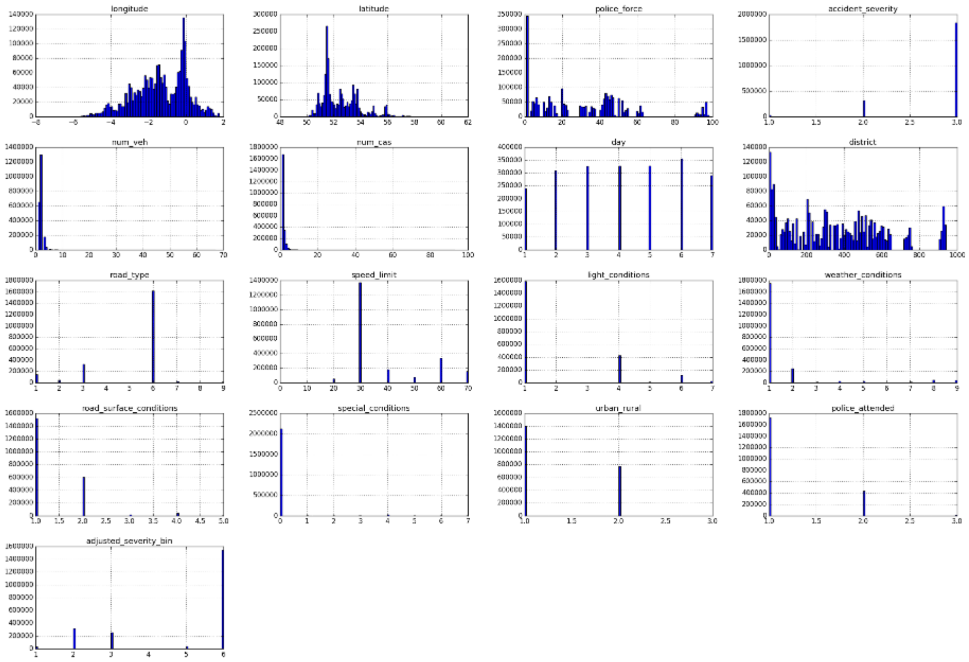


*Figure 8- initial histograms*

An initial view of the plotted histograms showed that a number of attributes were quite heavily skewed, providing a good indication of the attributes which may require additional attention.

The ‘special_conditions_at_site’ attribute was dropped as there were very few instances where special conditions existed, meaning the attribute would not add any value to the analysis.

## Accident Severity

The candidate Y (predictor value) “accident_severity” value, falls under the category of skewed attributes which need more attention. The ‘accident_severity’ variable showed an extreme skew, with the vast majority of accidents labelled as “3” (or ‘slight severity’).

For this reason, the ‘proxy_severity’ became the candidate Y value, as this value would provide a more balanced variable for which predictions could be made. The ‘proxy_severity’ value was generated by taking the ‘adjusted_severity’ (from the linked “Adjustments” dataset) value to the power of 3:

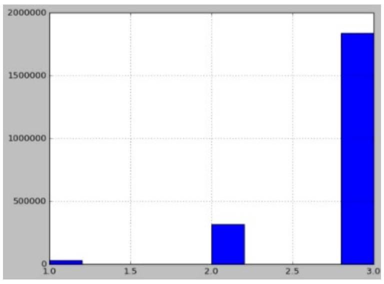


*Figure 9 - accident-severity*

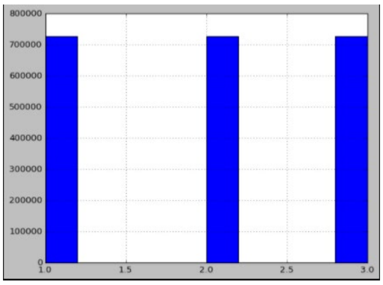


*Figure 10 - proxy-severity*

This proxy-severity should better demonstrate effectiveness of the machine learning classification models.

## Geographical Coordinates

To gain insight into the location the accidents were occurring, each accident was plotted out based on its geographical coordinates. The colour of each point was set to the severity of the accident; however, it was noted that due to the high volume of accidents, the majority of which had a severity of ‘slight’, it was hard to gain much insight from this, other than that the majority of accidents appear to occur in the more populous cities in the UK.

To account for this, the longitude and latitude coordinates of each record were grouped to create coordinate bins, and the average severity of these bins was calculated. A new plot, figure 5 below, was created with the size of the point set to the vehicle count and the colour of the point set to average severity, which provided some more detail.

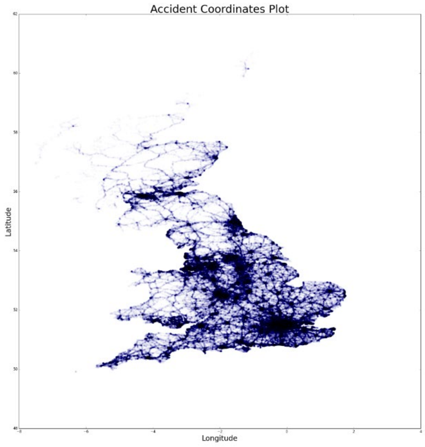


*Figure 11 - accidents plotted by coordinates.*

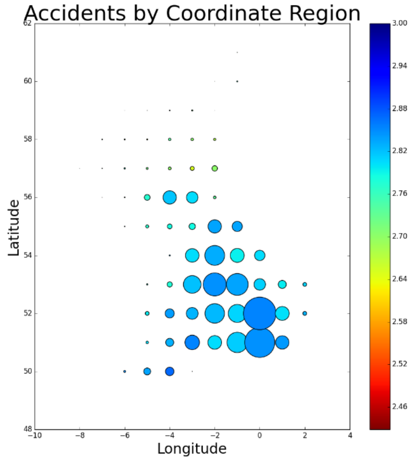


*Figure 12 - accidents plotted by coordinate group and severity.*

These grouped geocoordinate variables should be better suited to machine learning models, as utilising the original coordinates would likely not have added much value or may have overfit the model.

## Attribute Grouping / Binning

As identified by the multiple attribute histogram and various other analysis techniques performed in Jupyter Lab, such as using Pandas built in descriptive and summary methods, a number of attributes were identified as having one value accounting for the majority of the instances and some values with very few entries.

For these attributes, a grouping of values was performed to reduce the number of categories available with the aim of improving performance. Some examples of this included grouping ‘weather’, ‘light’, and ‘road surface’ conditions into a binary option of good or not good, to create more balanced categories. The speed limit attribute was also grouped to under or over 30mph.

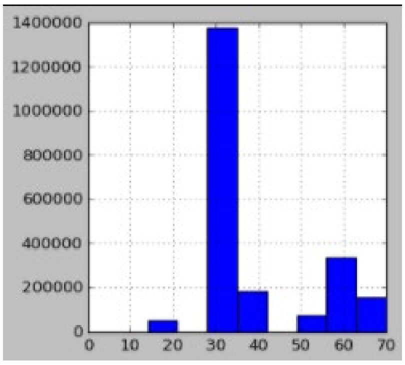


*Figure 13- initial speed limit histogram*

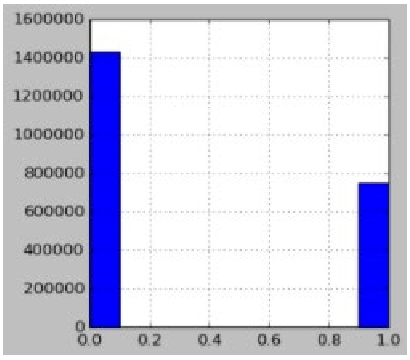


*Figure 14- grouped speed limit histogram (over 30mph?)*

## Additional Exploratory Analysis

Additional analysis was carried out to investigate the count of accidents over time, and ‘year’ and ‘month’ categories were created from the accident date variables, for use in machine learning.

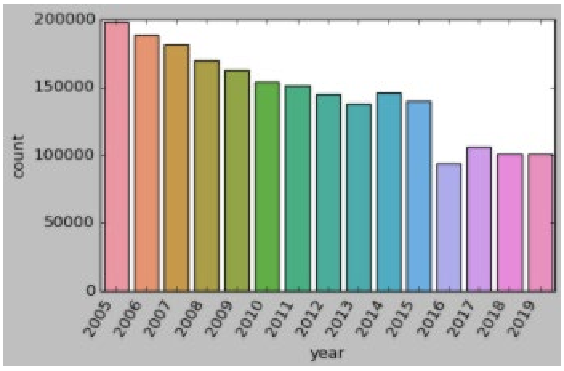


*Figure 15 - accident count by year*

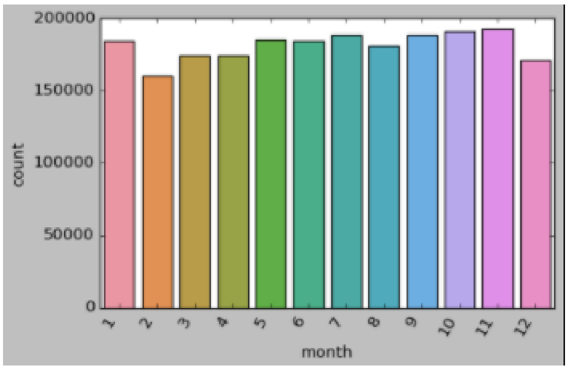


*Figure 16- accident count by month*

As well as some brief analysis of which police forces experienced the highest number of accidents of each proxy severity value.

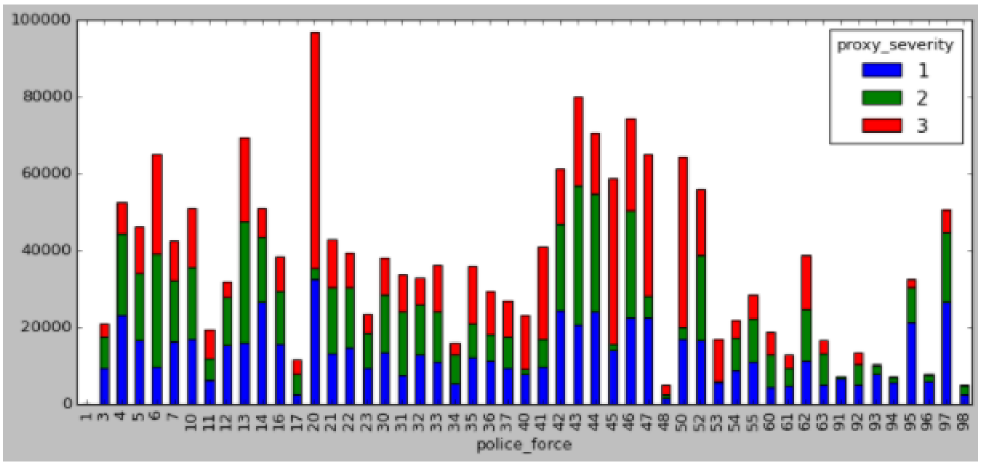


*Figure 17 - proxy severity by count and police force (excluding metro police)*

## Training Machine Learning Model

The final activity of the first iteration of data preparation and analysis was to train the first machine learning model with our prepared dataset.

Some final preparatory adjustments were made prior to training the first model, the first of which was to drop the remaining categorical attributes which contained too many categories, but which no grouping method had been devised. The attributes dropped at this point were "police force", "highway", and "district".

The categorical values which had non-numeric labels were then encoded using the "OneHotEncoder", from the SKLearn library, which transforms categorical attributes into a series of 'dummy' columns with a value of 1 or 0, for each possible category.

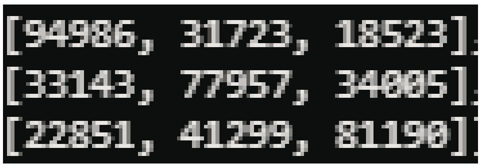
A stratified sample was created from the full dataset, ensuring that a proportionate selection of records was maintained from each year, creating a random training and testing split (80:20). The predictor variables were separated from the label to be predicted ("proxy severity"), and two machine learning models were trained for an initial view of performance.

The results of the models' accuracy scores for this first iteration were not as high as might have been expected, but this will continue to be fine-tuned throughout future iterations analysis. The preliminary results of this iteration's Decision Tree and Random Forest machine learning models can be seen below.

### Decision Tree:

Accuracy: 58.33%

Confusion Matrix:

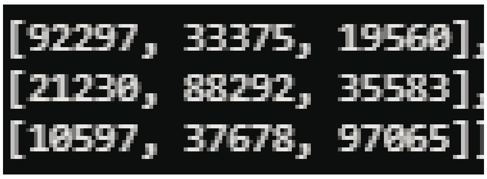


```
[94986, 31723, 18523]
[33143, 77957, 34005]
[22851, 41299, 81190]
```

### Random Forest:

Accuracy: 63.73%

Confusion Matrix:



```
[92297, 33375, 19560],
[21230, 88292, 35583],
[10597, 37678, 97065]]
```