CIS 5200 Project: Predicting Road Traffic Collision Severity

Rishi Ghia, Raj Anadkat, Victoria Fethke
University of Pennsylvania

December 14, 2022

Abstract

Road Traffic Collisions in the United Kingdom have been steadily falling since the 1960s. Nonetheless, there continues to be immense potential in lives and costs saved by safety policies tailored to address the factors contributing to crash severity. In this project, we analyze a 1.6M dataset by the Department of Transport in the United Kingdom via models with F1 scores of 72.99% with K Nearest Neighbors and 85.85% using a Random Forest Classifier. We have also run an SVM model, but the final executation did not finish before the submission deadline, running at over 3 hours already.

1 Motivation

Road Traffic Collisions (RTCs) in the United Kingdom have seen a steady downward trend over the last decades; from 6,352 in 1979 to 1,516 in 2022 [2]. Furthermore, among 32 countries in IRTAD (International Road Traffic and Accident Database), a database of validated traffic data, the UK has the fourth lowest road fatalities per 100 000 inhabitants, only behind Norway, Sweden, and Iceland [4]. Nonetheless, road crashes continue to cost the UK 33.4 billion GBP in 2019, representing 1.5% of GDP [4]. Fatalities caused 11% of these costs, despite them only making up 5 % of total crashes[4]. Better understanding the factors affecting the severity of road crashes has immense potential to save lives and costs to society.

In November, the UK government published detailed information on crashes and corresponding vehicle information which will be crucial to unpicking the variables that contribute to the fatality of car crashes [3]. Such forecasting methods for traffic accident severity have been shown as effective for road safety assessments and resulting policies to make roads safer [8]. This problem is suitable for Machine Learning methods because of the large datasize and numerous contributing factors, enabling models with limited risk of underfitting and providing insights that would not be possible with traditional methods. The problem requires multi-class classification into categories of slight, serious, and fatal, that can be addressed with machine learning models inclusing k-Nearest-Neighbors, Random Forest, and multi-class Support Vector Machines. With this approach, we aim to predict Road Traffic Collision severity based on crash and vehicle data to help reduce road injury and death.

2 Related Work

Road crash severity predictions is frequently covered in literature with the aim of contributing to an Intelligent Transport System (ITS) [6]. The field started in the early 80s with linear regression models [5], which were improved to Poisson regression models as crashes can be suitably modeled via a Poisson distribution. However, Poisson regression models have the over-dispersion problem, where variance is larger of smaller than the mean value, reducing the confidence in the interpretation of the model. This is why researchers like Shaik and Hossain have recently applied Negative binomial and Gamma regression models [9]. The field further developed to integrate methods like k-Nearest Neighbors for predicting the duration of traffic accidents, which have been found to correlate with accident severity [10].

Machine learning algorithms have become increasingly popular for predicting crash severity. Komol et al. [6] used KNN, SVM and Random forests on 17 distinct road user attributes as well as weather, vehicle, driver, period, road, traffic and speed data for each category of vulnerable road users from Queensland, Australia between 2013 and 2019. Accuracies found were relatively low, at 72.30% for motorcyclists, 64.45% for bicycles and 67.23% for pedestrians for the Random Forest model, which was evaluated as the best overall classifier. The authors noted that these Machine Learning methods are particularly useful for this problem as they are better at handling noisy data. Given that Komol et al. [6] identified crash severity for various road participants, we wanted to extend this approach by analyzing severity of crashes of road participants in the United Kingdom.

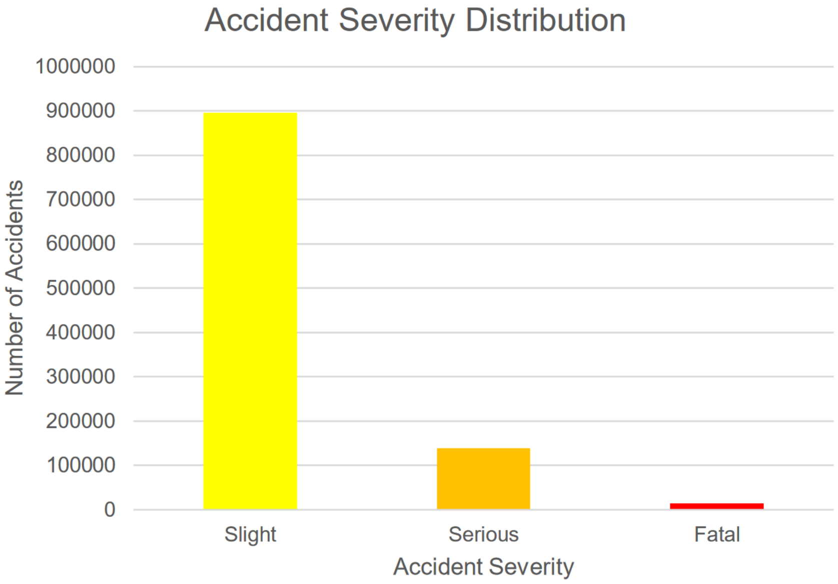
3 Dataset

3.1 Dataset description

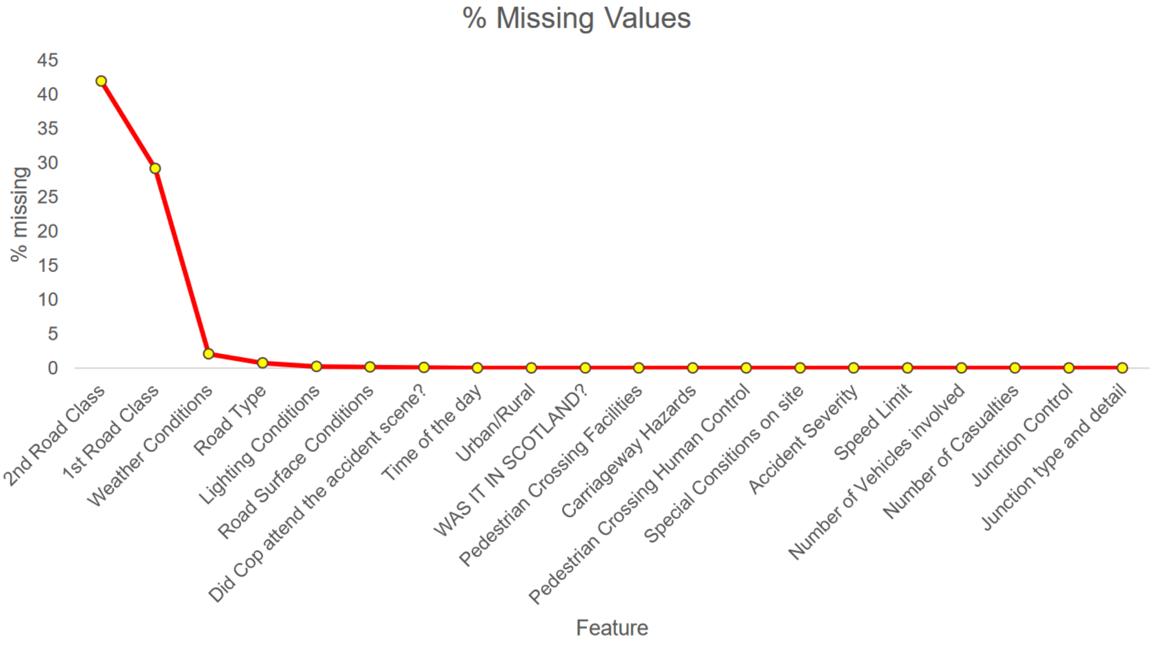
Our dataset, originally from the Department of Transport, but published on Kaggle by Salman Khaliq and Rich Gregson (https://www.kaggle.com/datasets/salmankhaliq22/road-traffic-collision-dataset), has 1.6 Million entires and 55 features. Yet, we observe that 8 of these features are completely irrelevant to performing predicting of car crash severity, and thus we remove them from the dataset.

3.2 EDA Visualizations

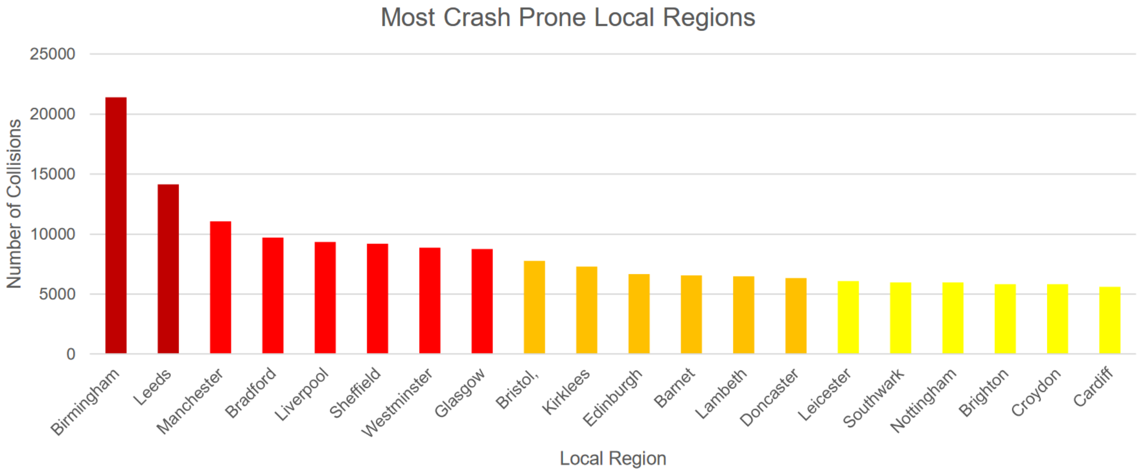
The figure below shows the class imbalance among the slight, serious and fatal categories initially.



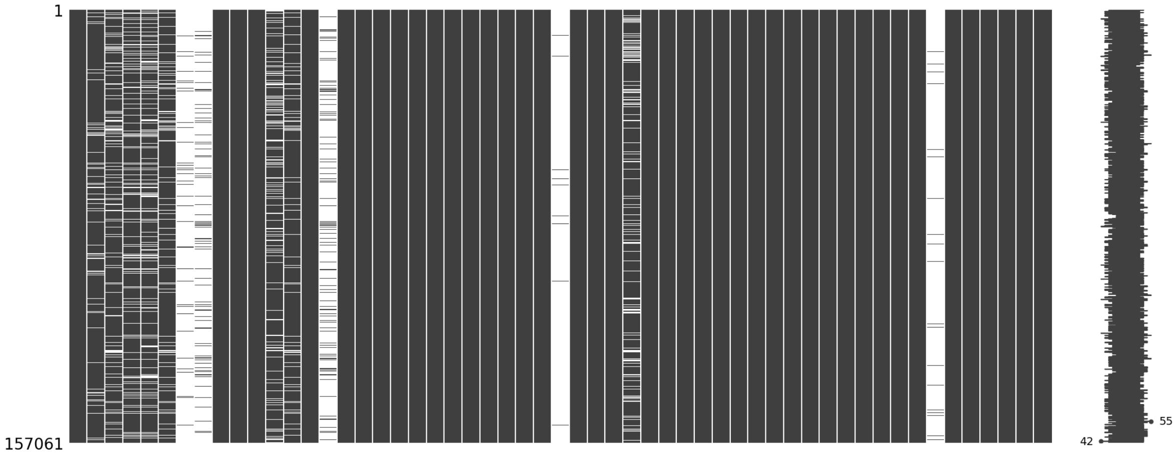
Next, we find here the % of missing values that exist for each datapoint.



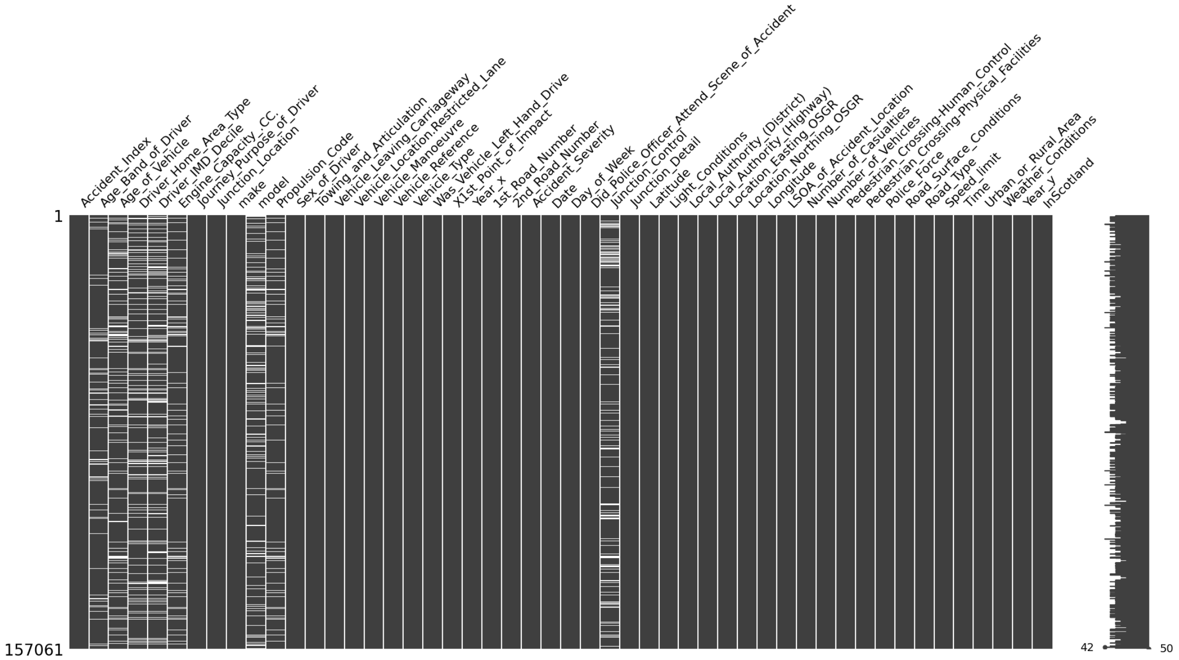
We also explore the physical locations of the most crashes below.



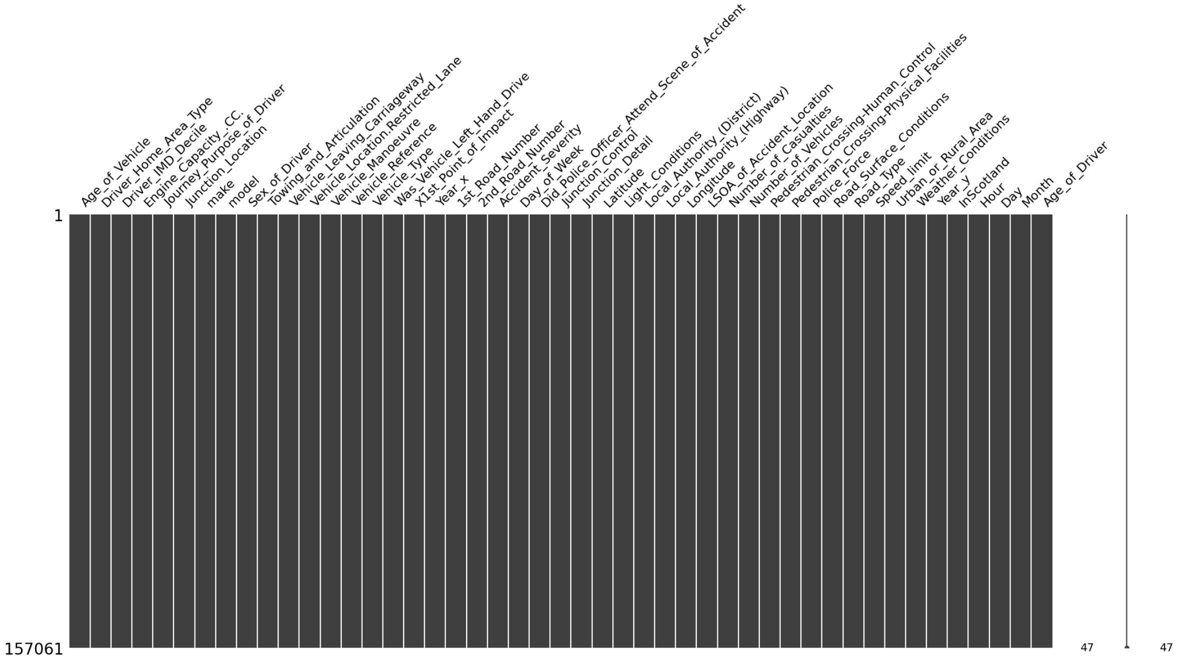
Below, within the image, we see a great visualization of the missing values in our dataset



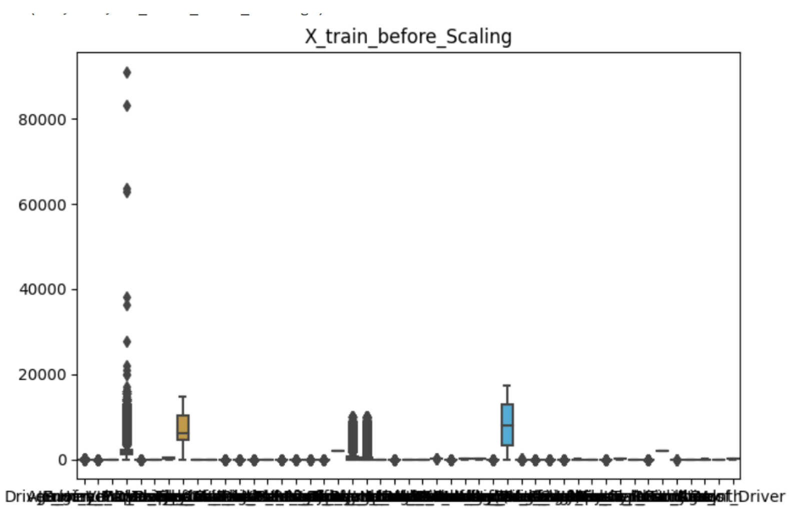
We now identify each of the features that contain over 75% of missing values, and shall drop them from our dataset. In the visualization seen below, we see another visualization of the dataset with an improvement, as we have now dropped all the 75% missing data columns. It is visible more full.

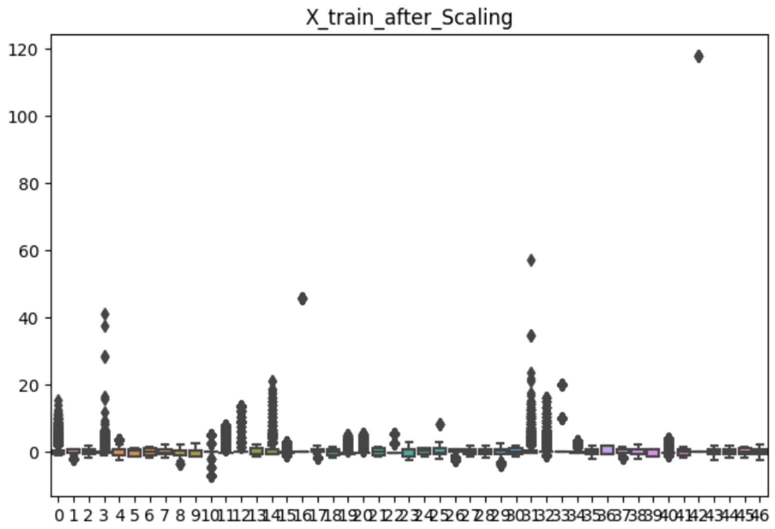


Seen below is our final dataset, fully cleaned up. We apologise efor the presence of propulsion code, but we did not have enough time before submission to remove it. But we had a chance to run the code partially, in order to achieve the accurate visualization for the report.



We also performed feature scaling using sklearn's fit transform.





4 Problem Formulation

We formulate our problem as a supervised multi-class classification problem with the goal to predict Road Traffic Collision severity based on the three categories of slight, serious and fatal as defined by the Department of Transport in the United Kingdom. We first split the dataset into training data (80%) and test data (20%) and performed feature scaling to improve training time and prevent the optimization from getting stuck in a local optima. Moreover, we adressed class imbalance, which will be explained in more detail in a later section.

We avoid using accuracy as an evaluation metric as the class imbalance would mean an unreasonably high accuracy for the slight accident category. Furthermore, it is more important to predict fatal crashes accurately instead of slight accidents to minimize deaths. Instead, we will use the F1 score an the evaluation metric as it takes into account precision and recall.

5 Methods

5.1 Baseline Model - kNN

We chose kNN as our baseline model due to its suitability for multiclass problems and not having to assume linear seperability as for logistic regression. We implement this via from sklearn.neighbors import KNeighborsClassifier.

5.2 Evaluation Metrics

Multi-class F1 is selected as the evaluation metric due to it capturing both precision and recall (it is the harmonic mean of the two). It is preferable for problems with class imbalance as in our dataset and when the goal is to find a balanced measure between precision and recall (Type I and II errors)[1]. For multi-class problems, the F1 score is calculated for each class separately. F1 score is calculated as follows:

$$F1 = \frac{2TP}{2TP + FP + FN} = \frac{2 * Precision * Recall}{Precision + Recall}$$

Furthermore, for SVM accuracy evaluation, we will be using one-vs rest strategy for multi-class evaluation from sklearn.multiclass's OneVsRestClassifier.

5.3 Class Imbalance Handling

We initially observed that there was a pretty severe class imbalance problem with our dataset. Of the 3 classes of accident severities, namely slight, serious and fatal, each represented 85.43%, 13.18%, 1.38%. As a result, we had to perform class balancing, which we did through an analysis of missing values. We saw that the set of slight accidents had 82.67% missing values. Which added up to 740641 datapoints. We decided to drop these, which left our dataset with 307934 datapoints. When we further merged this dataset with the vehicle data, we were left with a total of 162172 datapoints, with the final percentages being 56.11%, 39.41% and 4.47%. We did this in accordance with the feedback received after our presentation to the TA. Finally, we performed feature selection by dropping the useless/irrelevant features. The following were dropped:

- sr_no
- driver home area type (can get that from driver imd)
- journey purpose of driver (either missing or other, no inference)
- vehicle reference
- was the vehicle left hand drive (less than 0.001% yes)
- In Scotland
- year_x
- year_y
- Accident_index

5.4 Random Forest

We chose Random Forest as a comparison model as it is suitable for multi-class classification as an ensemble learning based on multiple decision trees. We chose this over a decision tree due to higher accuracy possibility. The ensemble of decision trees also minimizes the risk of overfitting.

Package: sklearn.ensemble.RandomForestClassifier

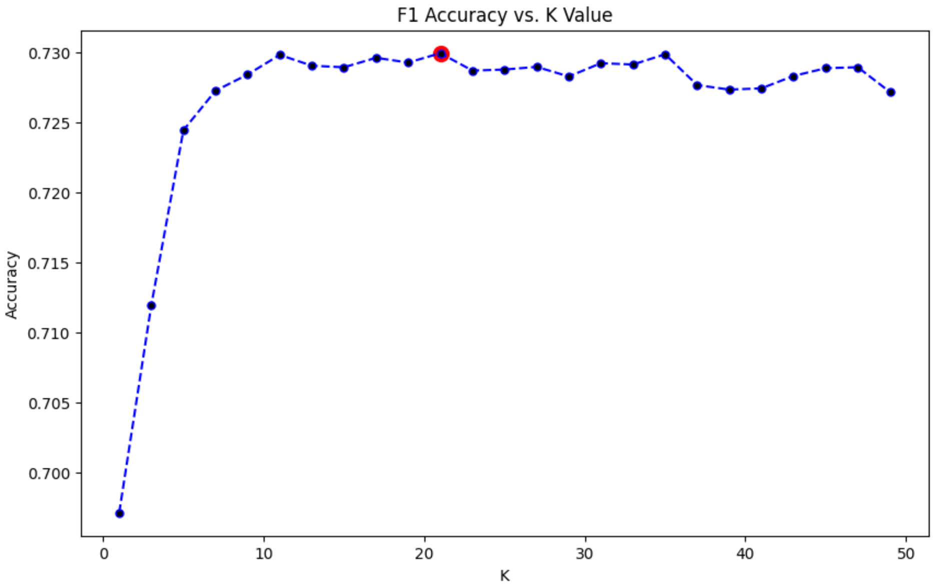
5.5 Support Vector Machines

Support Vector Machines are also suitable for multi-class classification: it aims to find the best decision boundary splitting the data into two or more classes via maximum margin. This method takes a long time to run which lead to some issues due to its O(n2) time complexity. Package: sklearn.svm.SV

6 Experiments and Results

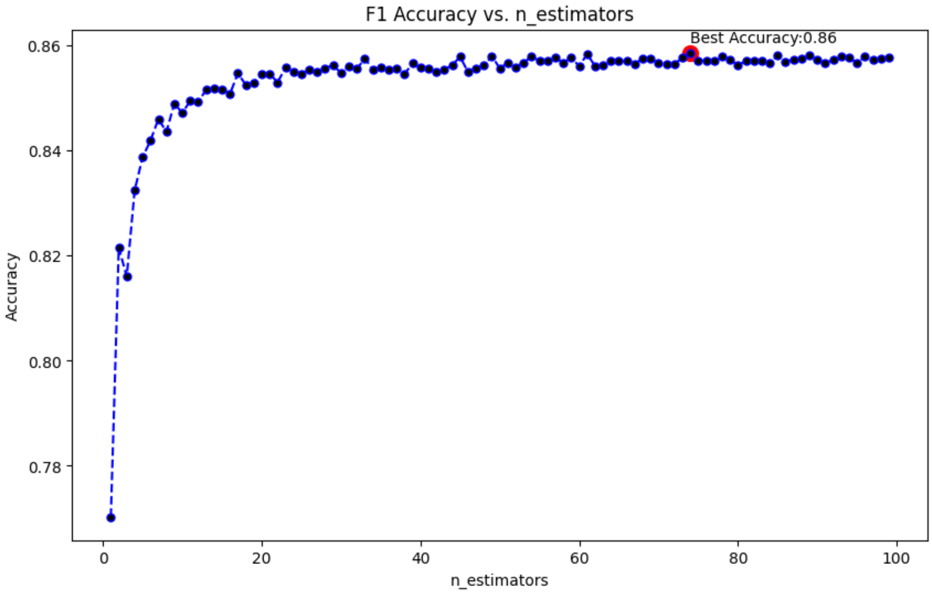
6.1 KNN

We found an optimal k value at k=21 corresponding to an F1 score of 0.7299360.



K	F1 Score
1	0.6971275718722334
3	0.7119507635731034
5	0.7244410414840466
7	0.7272559255869656
9	0.7284140408426998
11	0.7298143029259891
13	0.7290412988116537
15	0.7289296901868199
17	0.7296080268660153
19	0.7292648705286702
21	0.7299359836361622
23	0.7286941651158888
25	0.7287680939918038
27	0.7289599745440065
29	0.7282735161039466
31	0.7292217220817333
33	0.7291273304803103
35	0.7298592773124264
37	0.727659487031837
39	0.7273356355369587
41	0.7274282592212951
43	0.7283117368451038
45	0.7288644914044663
47	0.7289342721707228
49	0.7271688317035677

6.2 Random Forest



6.3 SVM

We tried implementing 2 different strategies for Multi-Label classification within Support Vector Machines. One was Support-Vector Machines with One-vs-Rest heuristic to perform multi-class classification and the other one was a Polynomial kernel and RBF kernel based strategy.

Since there are three classes in the classification problem, the One-vs-Rest method will break down this problem into three binary classification problems:

Problem 1 : Severe vs [Fatal, Slight]

Problem 2 : Slight vs [Fatal, Severe]

Problem 3 : Fatal vs [Severe, Slight]

A major downside of this method was it took a lot of computation time to fit the model. In order to do this, many models have to be created. For a multi-class problem with 'n' number of classes, 'n' number of models have to be created, which may slow down the entire process, especially for a dataset sized like ours, and very limited computational power. However, it is very useful with datasets having a small number of classes, where we want to use a model like SVM or Logistic Regression. We also implemented a RBF and Polynomial Kernel based SVM Multi Classifier but it made no improvements in the runtime, with our final models running for several hours apiece. We have at.

7 Conclusion and Discussion

7.1 Model Comparison

The two primary models we hae run are the Random Forest Classifier and the K Nearest Neighbors. The accuracies we have got at the maximum end are : 85.85% and 72.99% respectively. The primary issues with the KNN models were:

1. Best with low number of features

2. More features, more data; more data, kNN overfits

3. For large n, biases towards the Mode of the data

Due to this, we found that our model overfit heavily on the training data, obtaining an accuracy of over 95%, while achieving an accuracy of just 72.99% on the validation set. Further, as the dimensionality of our dataset, even after performing feature selection was rather high, at 48 features, that compounded the overfitting problem further.

We also observed that as the n increased, the model just outputted an "Average" of the dataset, making all k's over 30 not useful for our purposes.

To fix these issues, we ran a Random Forest Classifier, as that tends not to overfit easily for large n's and high dimensionality. We observed that due to the ensembling of weak learners, overfittting was kept in check, and at n estimators of value 74, we found our peak accuracy of 85.85%. This higher F1 accuracy was great as we have a pretty imbalanced dataset in terms of class imbalance.

We also wished to run an SVM model, as we believed the hinge loss would have been incredible to test on this dataset, since it would even penalize correct predictions within the margin, allowing for the accuracy to be pushed further.

7.2 Improvement Suggestions

A key improvement for our model woukld be to add more datapoints in the fatal accident category in order to ensure that we can avoid class imbalance. We have done our best with the size of the dataset present, but in order to fix this issue further, more, and better datapoints are required.

Additiaonlly, we can use SMOTE oversampling to fix the problem of multiclass imbalance. But SMOTE sometimes comes with its own limitations:

SMOTE is not very good for high dimensionality data Overlapping of classes may happen and can introduce more noise to the data.

Thus, an alternative is to adjust the class weights to balance the distribution.

Class weights modify the loss function directly by giving a penalty to the classes with different weights. It means purposely increasing the power of the minority class and reducing the power of the majority class. Therefore, it gives better results than SMOTE.

Another key improvement we could make is to infer the value of a certain feature in a datapoint through other features present, instead of performing basic imputation. Often we find that certain features can be very well inferred from a few other datapoints, and this would help improve overall model accuracy significantly.

7.3 Future Work

To make this model useful for accident management systems, it is important to include the factor of accident duration, as argued by Zong et al [11]. This is necessary in order to predict and thereby enact site and traffic management better after an accident. This is based on results from Nam and Mannering [7] that showed a correlation between the fatality of a crash and accident duration. Hence, an extension of this work can be to combine it with accident duration data to create traffic and safety management recommendations.

References

[1] Baeldung. F-1 score for multi-class classification. https://www.baeldung.com/cs/multi-class-f1-score, 2022.

[2] Department for Transport. Reported road casualties in great britain: 2019 annual report. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/922717/reported-road-casualties-annual-report-2019.pdf, 2020.

[3] Department for Transport. Road safety data. https://www.itf-oecd.org/sites/default/files/united-kingdom-road-safety.pdf, 2022.

[4] International Transport Forum. Road safety report 2021: United kingdom. 2021.

[5] Sarath C Joshua and Nicholas J Garber. Estimating truck accident rate and involvements using linear and poisson regression models. *Transportation planning and Technology*, 15(1):41–58, 1990.

[6] Md Mostafizur Rahman Komol, Md Mahmudul Hasan, Mohammed Elhenawy, Shamsunnahar Yasmin, Mahmoud Masoud, and Andry Rakotonirainy. Crash severity analysis of vulnerable road users using machine learning. *PLoS one*, 16(8):e0255828, 2021.

[7] Doohee Nam and Fred Mannering. An exploratory hazard-based analysis of highway incident duration. *Transportation Research Part A: Policy and Practice*, 34(2):85–102, 2000.

[8] Biswajeet Pradhan and Maher Ibrahim Sameen. Predicting injury severity of road traffic accidents using a hybrid extreme gradient boosting and deep neural network approach. pages 119–127, 2020.

[9] Md Shaik, Quazi Sazzad Hossain, et al. Application of statistical models: Parameters estimation of road accident in bangladesh. *SN Computer Science*, 1(5):1–10, 2020.

[10] Lun Zhang, Qiuchen Liu, Wenchen Yang, Nai Wei, and Decun Dong. An improved k-nearest

neighbor model for short-term traffic flow prediction. *Procedia-Social and Behavioral Sciences*, 96:653–662, 2013.

[11] Fang Zong, Huiyong Zhang, Hongguo Xu, Xiumei Zhu, and Lu Wang. Predicting severity and duration of road traffic accident. *Mathematical Problems in Engineering*, 2013, 2013.