Brainstormers

CE 784A Final Submission: Semester -2021-22 (II)

Classify Driver Behaviour Using Shallow and Deep-Learning Methods

Govind Gupta(190346), Mohd. Irfan Mahir(18807425)

1. Introduction

Due to the increasing number of vehicles and a vast transportation network globally, it is important to consider drivers' and pedestrians' safety. Most vehicle crashes in any country are due to distracted drivers, and these are usually the fatal ones. Past writing recommends that a forceful driving style is one of the main sources of car crashes that can prompt passings, material misfortunes, and wounds. It will be very difficult and cumbersome to handle these things manually, and hence, some alternate and more effective approaches will be very beneficial for all. Driving styles assume a fundamental part in guaranteeing street traffic security. Many solutions regarding this issue have been proposed so far based on some machine learning approaches, mainly known as "Driver Behaviour Profiling". Driver behaviour profiling is a procedure to categorise drivers as safe/risky at any point in time based on driving patterns, records, and other conditions. Hence, driver behaviour profiling tries to understand better and improve driver behaviour. All the current approaches mainly do driver profiling as safe/aggressive based on a safety score generated by a trained model, but in reality, this should not be the case because it solely depends on the circumstances. It can change from country to country, at different times and the mental health of drivers. Thus, the objective of our project is to identify the best algorithm to predict potentially risky driving using driver behaviour.

2. Driver Behaviour Overview

Drivers' behaviour tends to change anytime based on their physical and mental conditions, leading them to other thoughts or engage in other activities. Some researchers broadly classified driver behaviour change into two categories: internal and external distractions. Internal distractions are the ones based on the environment or activities inside the vehicle, whereas the external distractions are mostly due to the external environment like billboards, road rages, e.t.c.

Driver behaviour sensing can be done using the cumulative response of three broad methods: vehicle control data, physiological data and visual data. Vehicle control data is derived from the positioning, frequency, speed, and other vehicle properties. For instance, we can know the driver's behaviour from the positioning, acceleration, frequency and standard deviation of the angle made by the steering wheel. Similarly, physiological data can be extracted from drivers' health conditions, heart rate, e.t.c. Whereas visual data can be examined using the videos and images of drivers by looking at the body posture, head positioning, facial expressions, e.t.c. We will be using vehicle control data to identify driver behaviour for our project. Vehicle control data is found from smartphone sensors which are further converted and used to identify driver behaviour. We designed a model for identifying potentially risky drivers within large data sets and training them instead of training all the drivers. The logic that we are using for our approach is that the data profile of potentially risky driving behaviours will look quite similar to the data profile of non-risky driving of the same behaviour. Then a potentially risky left turn would have a data profile that outlies the average data profile, say all left turns in the dataset. In this way, we will use vehicle control data and classify driver behaviour as risky or not risky.

3. Literature Review

In [1], three different algorithms: logistic regression, random forest and Artificial Neural Networks (ANN), were used to predict driving behaviour. The results show that the accuracy of random forests in classifying the potentially risky behaviour was better than the other two.

In [2], they trained three deep-learning algorithms, CNN (Convolutional Neural Networks), RNN (Recurrent Neural Networks) and DNN (Deep Neural Networks), to classify the segments according to the score assigned. Apart from this, they used various techniques to enhance the accuracy of the algorithms. Functions such as Dropout and early stopping were used. Hyperparameters like learning rate of 0.001, epochs, and batch size are also used. They segmented the data into a group of 1-10 seconds and compared the results for all three algorithms. CNN has the best accuracy among the other two, RNN and DNN. Thus, they suggested CNN for detecting driving behaviour that is risky in Malaysia.

In [3], RF gave a good accuracy with an average F1 score of around 0.62 among all three classes (i.e., normal, drowsy and aggressive). The RF provides a consistent precision, recall, and F1 score, which implies very few chances of under-fitting and overfitting. LR was run on a range of parameters choices for “solver” and regularisation parameter, “C”. After obtaining the preferred parameter values, all the accuracy metrics were determined. ROC curves of all the classes are highly inconsistent and varied. Class 0 (normal) gives poorer results than the aggressive and drowsy classes. The ANN classifier obtained almost similar results on both datasets (i.e., traffic status and lane detection), with an overall accuracy not exceeding 0.30.

In [4], the driving information is gathered by a 3-pivot accelerometer, which records the sidelong, longitudinal and vertical speed increases. An information change way has been created to separate useful insights highlights from crude 3-hub sensor information and use AI calculations to recognise drivers. To kill the predisposition brought about by the sensor establishment and to guarantee the relevance of their methodology, they present an information alignment strategy. Four essential directed characterisation calculations are utilised to collect correlation information. For further development of the grouping execution, they present various classifier frameworks, which consolidate the results of a few classifiers. The arbitrary timberlands (RFs) calculation has the best presentation on exactness, accuracy, and review among the four fundamental calculations. RFs calculation wins in running speed.

In [5], information is gathered from 50 drivers from two unique urban communities in a naturalistic setting. Five highlights are disengaged from the crude information.The dataset was isolated into five distinct models(Support Vector Machines (SVM), Artificial Neural Networks (ANN), fuzzy logic, k-Nearest Neighbour (kNN), and Random Forests (RF)). Further, these models have been assessed as far as a bunch of execution measurements and real tests. The test results from execution measurements showed that SVM beat the other four models, achieving an average accuracy of 0.96, F1-Score of 0.9595, Area Under the Curve (AUC) of 0.9730, and Kappa of 0.9375. These outcomes show that the proposed technique might uphold specialists in concluding which ML model performs better for driving-styles classification.

In [6], RF gave satisfying accuracy. They used ROC as an ML matric. The ROC mean value of LR was more than 0.90 .

Although the proposed procedure has effectively shown a systematic assessment of driving-styles models, it has specific impediments that ought to be tended to and noted. First, it isn't prescribed to foster order models utilising little datasets because of the potential deficiencies of model speculation restrictions. Another limit is that we didn't consider outside variables, such as atmospheric conditions and day season, that could influence driving styles. These papers have implemented different ML models on varying datasets. We intend to apply most of these models to a given dataset with similar features and identify the best model to predict potentially risky driving behaviour.

4. Methodology

This project was modelled as a multi-label supervised learning classification issue with driving event kinds as labels. The purpose of this project is to identify the best machine learning algorithms to detect driver behaviour. Our evaluation pipeline started with raw data extracted from smartphone sensors and converted to Earth's coordinate system. These translated sensor data files are used to generate attribute vectors. These generated attribute vector datasets are further used to train, test and evaluate the performances of different MLAs. Then using the error functions, we identify the best algorithm.

4.1. Data

Dataset we have used has been collected using smartphone sensors. The experiment was performed on four different car trips of around 13 minutes each. Data consists of accelerometer reading, linear acceleration, magnetometer reading and gyroscope reading. Data consists of 7 different driving event types and a number of sequences for each event. A total of 69 sequences were obtained.

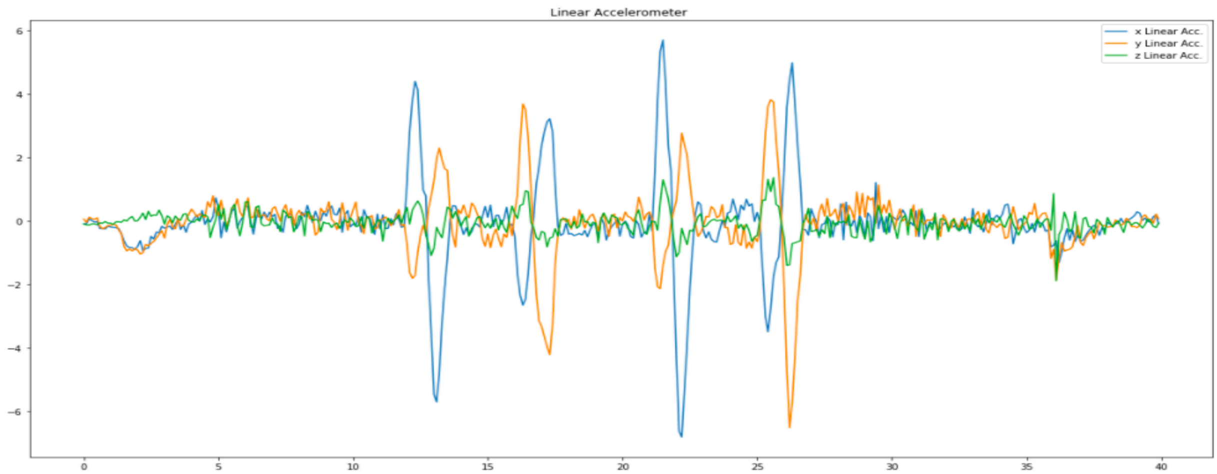
Event Type	**No of Sequences**
Non-Aggressive	14
Aggressive braking	12
Aggressive Acceleration	12
Aggressive left turn	11
Aggressive right turn	11
Aggressive right lane change	5
Aggressive left lane change	4
Total	**69**

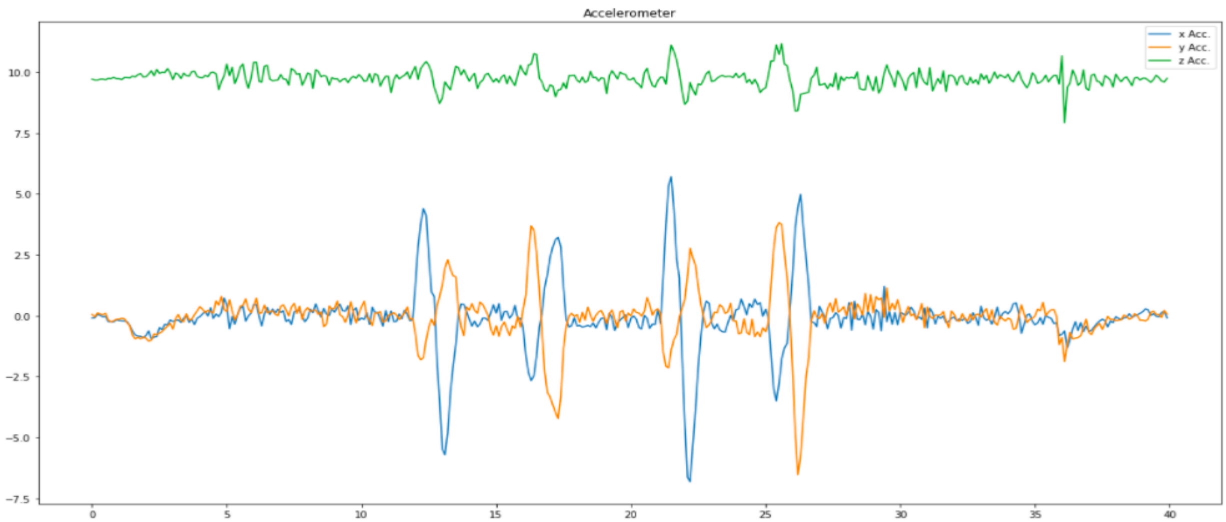
4.2. Data Pre-processing

We clumped together all the data files for each vehicle based on the time sequences available in the label files for each car trip. We converted the time feature (from the start time of the device) from nanoseconds to centiseconds (because data containing labels have a precision of centiseconds). We have used MinMaxScaler to scale the features to a given range to make the data centred for better convergence. We oversampled the data using RandomOverSampler from the imblearn library to get rid of the imbalanced dataset problem.

4.3. Data Visualisation

We also visualised the data for vehicle 2 for the first 40 seconds to observe the trends it is following. The sudden change in x and y acceleration values shows aggressive left and right turns or aggressive left lane change and right lane change.





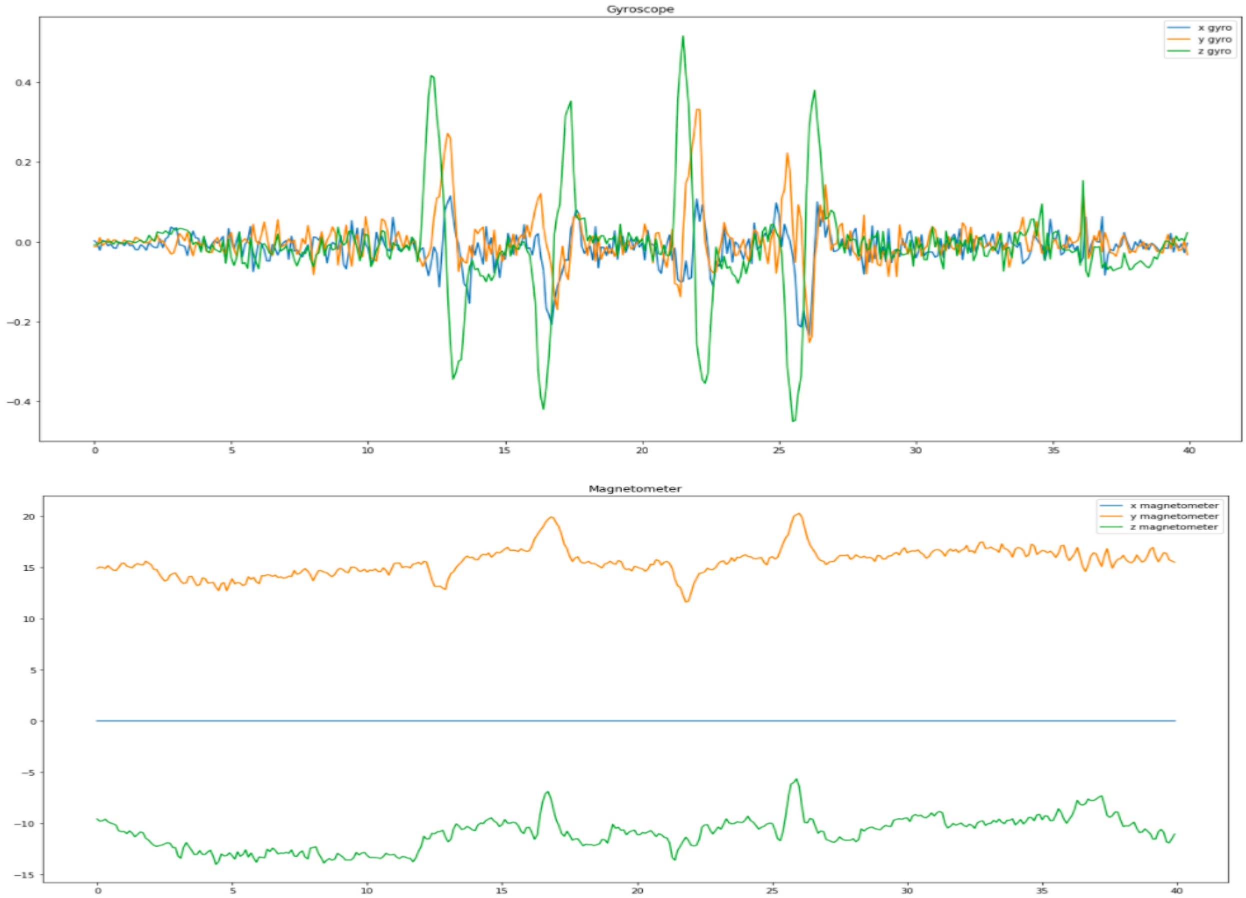


Fig 1: Data Visualisation

4.4. Generating Attribute Vectors

The data we used was not the whole raw dataset, rather we discarded some attributes which we didn't require like timestamp, time in nanoseconds, e.t.c. We used the time window partition technique to generate more attributes from the data. We chose a time window and then merged them to get the minimum, maximum, mean, and standard deviation for each attribute of each window. We chose a time window of 0.5 seconds to generate the attribute vectors. We only chose the data points where there is an event happening (from label files for each vehicle) and discarded all the other data points. Upon successfully generating the attribute vectors we got a total of 69 sequences cumulative of all the different event types. The data generated contains a total of 48 attributes and 1 label class.

4.5. Evaluation Assembly

The generated attribute vector datasets are further used to train, test and evaluate the performances of different MLAs (Machine Learning Algorithms). Thus, the best algorithm is identified. We used 70% data for training and the remaining for validation and testing of the different MLAs used. We used ANN, SVM and RF on the generated dataset for classification accuracy and chose the best one. We used the Keras framework for the implementation of all the algorithms. We used different algorithm configurations and tried different parameters, no of layers, no of epochs, e.t.c. to avoid the problem of overfitting and get satisfactory results.

Algorithm	**Parameters**	**Values**
ANN	Activation function	ReLU, Softmax
	Loss criterion	Sparse categorical cross-entropy
	Optimizer	Adam, RMSProp
RF	Number of trees	100, 200, 300
	Maximum depth	10, 15, 20
SVM	Kernel	Linear, Sigmoid, RBF
	C	0.001 - 2
	γ	2^{-9} - 2

In ANN we used the softmax (because of multi-class classification) activation function in the final layer along with the sparse categorical cross-entropy as a loss function because we had integer label classes. in all the five hidden layers we have used the ReLU activation function. We have trained ANN models with an epoch size of 300 and a validation split of 0.1.

In SVM we used the RBF kernel because our data is linearly inseparable and also because RBF resembles KNN and overcomes the space complexity problems. The values of C and γ have been chosen based on different trials and errors to get better accuracy. The C value of 2.5 gave a better decision boundary with a not much higher γ as it will increase the complexity (curvature weight) of the decision boundary.

We have used a total of 300 trees for the classification and to make the computations less expensive we have used “gini” criterion to find the optimum split of the features.

5. Results

We used three machine learning namely, artificial neural network, support vector machine and random forest to train from 70% of the provided dataset and then test and validate the remaining dataset. We used the F1 score to get the accuracy score of the model. We also visualised the confusion matrix obtained for the test data for different models.

5.1. Artificial Neural Network

Accuracy score obtained for ANN is 0.88. Thus, the accuracy of the model is 88%.

	precision	recall	f1-score	support
0	0.89	0.84	0.86	19
1	1.00	1.00	1.00	4
2	0.33	0.40	0.36	5
3	1.00	1.00	1.00	13
4	0.93	0.82	0.87	17
5	1.00	1.00	1.00	17
6	0.79	0.86	0.83	22
accuracy			0.88	97
macro avg	0.85	0.85	0.85	97
weighted avg	0.88	0.88	0.88	97

Fig 2 . Classification report for ANN model

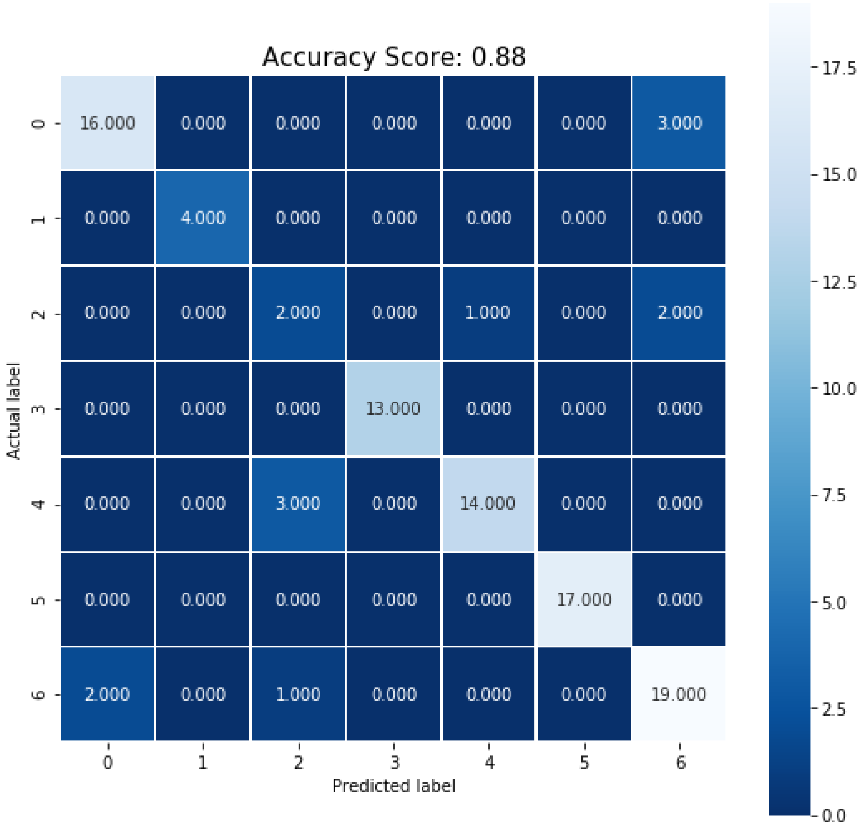


Fig 3. Confusion Matrix using ANN model

5.2. Support Vector Machine

Accuracy score obtained for SVM is 0.86. Thus, the accuracy of the model is 86%.

	precision	recall	f1-score	support
0	0.78	0.95	0.86	19
1	0.40	0.50	0.44	4
2	0.75	0.60	0.67	5
3	1.00	0.92	0.96	13
4	0.94	0.88	0.91	17
5	1.00	1.00	1.00	17
6	0.80	0.73	0.76	22
accuracy			0.86	97
macro avg	0.81	0.80	0.80	97
weighted avg	0.86	0.86	0.86	97

Fig 4. Classification report for SVM model

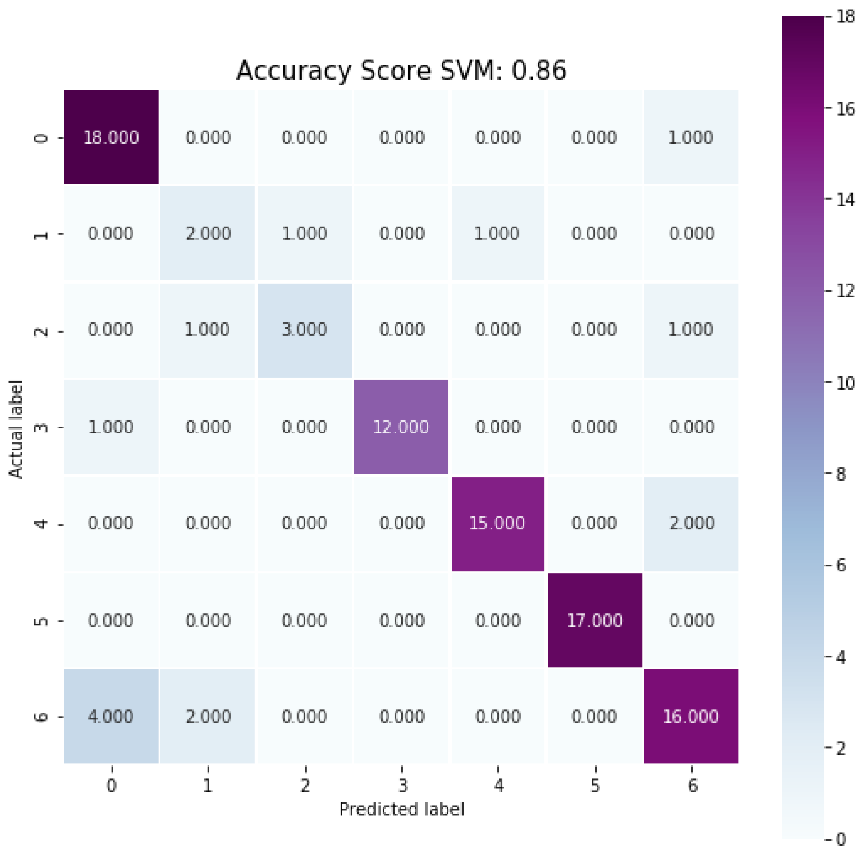


Fig 3. Confusion Matrix using SVM model

5.3.Random Forest

Accuracy score obtained for RF is 0.92. Thus, the accuracy of the model is 92%.

	precision	recall	f1-score	support
0	0.86	1.00	0.93	19
1	0.80	1.00	0.89	4
2	1.00	0.80	0.89	5
3	0.81	1.00	0.90	13
4	0.94	0.88	0.91	17
5	1.00	0.94	0.97	17
6	1.00	0.82	0.90	22
accuracy			0.92	97
macro avg	0.92	0.92	0.91	97
weighted avg	0.93	0.92	0.92	97

Fig 2 . Classification report for Random Forest model

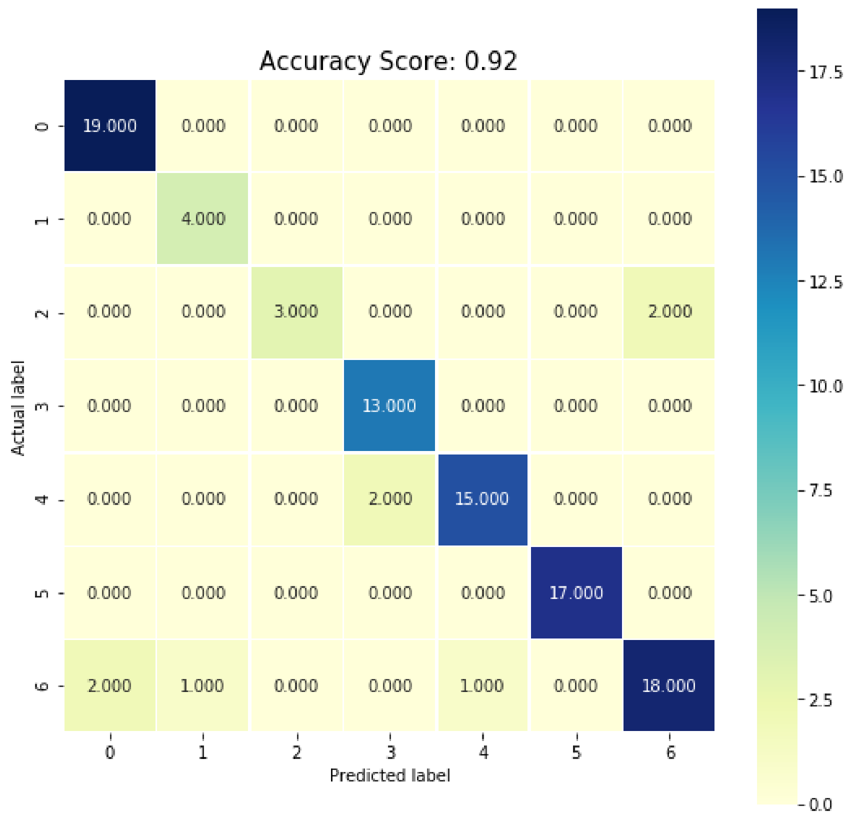


Fig 3. Confusion Matrix using Random Forest model

6. Conclusion and Future Work

In this project, we used ANN, SVM and RF to predict aggressive/ non-aggressive driver behaviour.Out of three, best accuracy was shown by random forest with an accuracy score of .92.Minimum accuracy was shown by SVM with an accuracy score of 0.86.ANN gave accuracy score 0.88. Thus, after analysing three algorithms, we found that the random forest algorithm is the best algorithm out of three with an accuracy of 92%.

As mentioned earlier, we have used only vehicle control data to examine driver behaviour but it depends on many other factors namely, physiological data of the driver and visual data of head and eye position of the driver, hands position of the driver (to detect any other activity the driver is involved in). Thus, we can add physiological data and visual data along with vehicle control data and examine driver behaviour more precisely.

We used ANN, SVM and RF to predict the driver behaviour in this project. We can increase our domain and further add more MLAs such as CNN, RNN, DNN. This will help us increase our accuracy and find more optimal algorithms to analyse and predict driver behaviour.

In domains like the freight management domain, automated, real-time driver behaviour profiling allows managers to systematise campaigns aiming to improve driver's score, decrease accidents, increase resource economy and vehicle lifetime. And likewise, In the insurance telematics domain, driver behaviour profile identification can make car insurance cheaper by rewarding drivers with good driving scores same as credit scores based on past driver behaviour, instead of only considering group-based statistics, e.g., age, gender, marital status, etc. The models that severs the above purposes can easily be developed once driver behaviour profiling is done(purpose of this project).

7. Work Distribution

- **Govind:** Literature review of paper [1],[2],[3], proposal writing, Report compilation and editing, Data Visualisation, Presentation Writing and Compilation, ANN application, Final report writing.
- **Irfan:** Literature review of paper [4],[5],[6], paper selection, proposal writing, Data Preprocessing, Evaluation Assembly, RF and SVM application, Presentation Writing and Compilation, Final report writing.

Plagiarism Accuracy:- 14% Similarity in Turnitin

References

[1] Charles Marks, Arash Jahangiri, and Sahar Ghanipoor Machina, “Identifying and Labeling Potentially Risky Driving: A Multistage Process Using Real-World Driving Data,” Journal of Advanced Transportation Volume 2021, Article ID 8819094, 11 pages.

[2] Al-Hussein, W.A.; Por, L.Y.; Kiah, M.L.M.; Zaidan, B.B., "Driver Behavior Profiling and Recognition Using Deep-Learning Methods: In Accordance with Traffic Regulations and Experts Guidelines, " Int. J. Environ. Res. Public Health 2022, 19, 1470.

[3] Ghandour, R.; Potams, A.J.; Boulkaibet, I.; Neji, B.; Al Barakeh, Z., "Driver Behavior Classification System Analysis Using Machine Learning Methods, ". Appl. Sci. 2021, 11, 10562.

[4] Li, Z.; Zhang, K.; Chen, B.; Dong, Y.; Zhang, L., "Driver identification in intelligent vehicle systems using machine learning algorithms, " IET Intell. Transp. Syst. 2018, 13, 40–47

[5] Silva, I.; Eugenio Naranjo, J., "A systematic methodology to evaluate prediction models for driving style classification, " Sensors 2020, 20, 1692.

[6] Rohon Das; Pabitra Mohan Khilar, "Driver Behaviour Profiling in VANETs: Comparison of Ensemble Machine Learning Techniques, "978-1-7281-0419-5/19/, 2019.

[7] Ferreira, Jair Júnior et al. “Driver behavior profiling: An investigation with different smartphone sensors and machine learning.” *PloS one* vol. 12,4 e0174959. 10 Apr. 2017