ACTL3143 Project Report Part 2: Predicting Crash Types in Australian Roads Fatalities

By Danny Li (z5416979) – 25T2 ACTL3143

Abstract

This report investigates two deep learning approaches to classify road crash types as single or multiple vehicle incidents. The problem will use the Australian Road Deaths Database to predict against the metrics of F1-score, AUC-ROC, precision, recall and accuracy. A random forest classifier is used as the baseline model for the feedforward neural network and residual neural network. Key processes in this report are exploratory data analysis, data preprocessing, training the baseline model, tuning and training the deep learning models and analysis.

Problem Specification

The motivation for analysing this issue is that vehicle incident predictions is vital to many areas of concern. This includes insurance pricing and risk assessments for motor policies, public safety programs and interventions for higher risk communities, and identification of dangerous road conditions to improve road design and safety.

The problem is a binary classification problem on the target variable ‘Crash Type’ being ‘Single’ or ‘Multiple’. The models’ performance will be measured against the following metrics: precision, recall, F1-score to balance precision and recall, AUC-ROC to evaluate the overall classification performance, and accuracy.

Data Preprocessing

The data used is from the Australian Road Deaths Databased collected by the Bureau of Infrastructure and Transport Research Economics. The dataset contains demographic details about the incidents and circumstances of the crash comprised of temporal, road and vehicle features. The data dictionary and input features for all models is attached in the appendix.

The database consists of 57430 observations from January 1989 – May 2025. The data was cleaned by removed 153 observations (0.3%) with missing critical features[1] and I assumed that unknown values for other features were had a ‘missing’ response. The database was then split using a 60/20/20 split of training, validation and testing stratified on ‘Crash Type’ to ensure class balance. The split is also optimal as it accounts for changes in the time variant as a larger train may overfit on testing and not account for changes in behaviour from 1989 to 2025.

The features used in all the models were preprocessed based on the type of variable they were. The numerical features (‘Time’, ‘Speed Limit’, ‘Age’) were processed using standard scaler. The nominal categorical features ('State', 'Road User', 'Gender', 'Bus Involvement',

[1] ‘Time’, ‘Speed Limit’, and ‘Crash Type’

'Heavy Rigid Truck Involvement', 'Articulated Truck Involvement', 'Christmas Period', 'Easter Period') were processed using one-hot encoding.

Exploratory Data Analysis

This section will explore the patterns in road fatalities through the visual analysis of six key figures. Figures 1-3 shows patterns in the circumstances of the crash and figures 4-6 contain details of the person killed.

Figure 1 shows the distribution of crash types revealing that single-vehicles fatalities comprise a majority of the dataset. This suggests a higher prevalence for incidents involving non-vehicles such as stationary objects. It also signifies a class imbalance in 'Crash Type' with 55% of crash types being 'Single'.

Figure 2 examines how crash types are distributed by speed limit. 40-60 km/h roads (urban roads) and 80-100 km/h roads have the greatest frequency for both crash types. Multiple vehicle fatalities are more likely to occur for 80-100 km/h roads indicating severe outcomes at higher speeds.

Figure 3 reveals a temporal link between the time and the type of crash. Single vehicle crashes occur disproportionately more than multiple vehicle crashes during night between 12 AM – 6 AM, linked to impaired driving or fatigue. Multiple vehicle crashes occur more frequently during afternoon between 12 PM – 6 PM coinciding with high traffic density and evening rush hour.

Figure 4, 5 and 6 highlight the victim profiles. Figure 4 show that drivers constitute the largest proportion of fatalities followed by passengers and pedestrians. Figure 5 show the age distribution of fatalities with peaks among young adults (20-30 years). This reflects risks factors such as inexperience. Figure 6 reveals that males account for 72% of fatalities linked to higher risk driving behaviour.

Random Forest (Benchmark Model)

This section will summarise the random forest benchmark model implemented in Part 1 of the report. Random forest is preferred as it handles mixed data types without the need for extensive preprocessing and feature engineering, and it is robust to outliers.

The model used the validation AUC-ROC score to tune the hyperparameters of the random forest using Bayesian Optimisation. This approach was chosen over grid search cross validation as it required less searches and is more efficient. The hyperparameters are:

Hyperparameter	Search space	Optimal hyperparameter
Number of estimators	100 – 500	223
Max depth	5 – 50	36
Min sample split	2 – 20	5
Min sample leaf	1 – 10	10
Max features	Sqrt / log2 / None	Sqrt
Class weight	Balanced / None	Balanced
Boostrap	True / False	False

Feedforward Neural Network

This section outlines the feedforward neural network implementation used to predict crash types. Feedforward neural networks are a class of neural networks that process inputs through fully connected layers from the input layer through multiple hidden layers to the output layer. The network is unidirectional and does not contain any loops.

The implementation used to predict crash types consists of (1) an input with batch normalisation, (2) multiple dense layers with Rectified Linear Unit (ReLU) activation, (3) dropout layers to prevent overfitting, and (4) a final output layer with one neuron with a sigmoid activation function for binary classification.

The feedforward neural network is hyperparameter tuned using Bayesian optimisation on the number of layers, the number of neurons per dense layer, the dropout rate per layer, and the learning rate. Bayesian optimisation is chosen for its probabilistic approach that significantly reduces the number of trials needed. The optimisation process utilised 15 trials with early stopping of patience at 3, epochs of 20, and optimisation of validation AUC-ROC. The search space is summarised in the following table with the optimal hyperparameters listed.

Hyperparameter	Search space	Optimal hyperparameter
Number of layers	1 – 4 (step of 1)	2
Layer 1 units	32 – 256 (step of 32)	64
Layer 1 dropout	0.1 – 0.6 (step of 0.1)	0.1
Layer 2 units	32 – 256 (step of 32)	192
Layer 2 dropout	0.1 – 0.6 (step of 0.1)	0.1
Learning rate	1e-4 – 1e-3 (sampling ‘log’)	0.0007957180395613706

Residual Neural Network

This section outlines the residual neural network implementation used to predict crash types. Residual neural networks introduces skip connection to address the vanish/exploding gradients problems. Skip connections provide an alternative path for the gradient allowing them to converge faster. The implementation used to predict crash types consists of (1) initial dense layer to project inputs to higher dimensions, (2) residual blocks each with two dense layers with batch normalisation and skip connections with ReLU activation function, and (3) a final output layer with one neuron with sigmoid activation.

The residual neural network is also tuned using Bayesian optimisation on the number of layers, dropout per layer, and learning rate. The method used is identical to the one used to tune the feedforward neural network. The search space is summarised in the following table with the optimal hyperparameters listed.

Hyperparameter	Search space	Optimal hyperparameter
Number of layers	1 – 4 (step of 1)	4
Initial units	64 – 256 (step of 32)	64
Layer 1 units	64 – 512 (step of 32)	256
Layer 1 dropout	0.1 – 0.6 (step of 0.1)	0.5

Layer 2 units	64 – 512 (step of 32)	192
Layer 2 dropout	0.1 – 0.6 (step of 0.1)	0.3
Layer 3 units	64 – 512 (step of 32)	384
Layer 3 dropout	0.1 – 0.6 (step of 0.1)	0.4
Layer 4 units	64 – 512 (step of 32)	64
Layer 4 dropout	0.1 – 0.6 (step of 0.1)	0.5
Learning rate	1e-4 – 1e-3 (sampling 'log')	0.0003558617008728565

Analysis

This section evaluates the performance of the predictive models in predicting crash types. The metrics are summarised:

	F1-Score	AUC-ROC	Precision (Single)	Recall (Single)	Precision (Multiple)	Recall (Multiple)	Accuracy
Random Forest	0.7058	0.8062	0.7754	0.6971	0.6666	0.7500	0.7207
Feedforward NN	0.6748	0.8005	0.7344	0.7652	0.6933	0.6573	0.7170
Residual NN	0.6789	0.7987	0.7380	0.7636	0.6941	0.6643	0.7192

Some key observations are that Random Forest outperforms both neural networks in terms of F1-score and AUC-ROC suggesting a better balance between precision and recall. Its higher precision for single vehicle (0.7754) crashes show fewer false positive while its recall for multiple vehicle crashes (0.75) show that it can identify most instances of multiple vehicle incidents.

Another key observation is that the feedforward neural network achieves higher recall for single vehicles incidents (0.7652) but has a lower precision for multiple vehicle incidents (0.6933). This shows that it tends to misclassify some multiple vehicle incidents as single vehicle incidents indicating overfitting into the majority class. The residual neural network achieves a marginally better F1-score (0.6789) than the feedforward neural network likely due to the skip connection reducing gradient issues.

From the confusion matrices (figure 9), it shows that random forest balances misclassifications with slightly higher false negatives. This is contrasted with both deep learning models which have more false positives.

Practical Implications

As discussed in problem specification, the information provided by the models is very useful in many areas. The random forest model has the highest precision for single vehicle crashes so may be more useful than the deep learning models in risk assessments for motor insurance or life insurance policies. This is because they are interested in individual behaviour where single vehicles crashes is most likely to derive from driver behaviour and precision is necessary to prevent mispricing risks. In terms of road safety, the feedforward neural network's high single vehicle incidents recall may assist in scenarios targeting night time driving or road design. This can be done by identifying roads using the feedforward model and installing preventative solutions such as increased lighting. Although the random forest is better than the residual neural network, the residual neural network has greater potential due to its architecture and may justify its use when further tuned or combined with additional data.

Ethical Considerations

The use of neural networks to predict crash types poses ethical concerns as it fundamentally uninterpretable and risks of discrimination if used in insurance or policy decisions. Neural networks would have to rely on techniques such as 'LIME' or 'SHAP' to be explainable but may risk reducing accuracy. Even then, users of explained models may rely on oversimplistic answers and be lured by a false sense of transparency. If using a dataset that is not debiased, model predictions would be biased and explanation techniques could masks these biases further.

Conclusion

This report compared a random forest benchmark against feedforward and residual neural networks for binary classification on crash data. The random forest achieved the highest F1-score (0.7058) and AUC-ROC (0.8062) on the test. The random forest's greater interpretability and greater predictive power makes it preferred to actuarial applications while neural networks could be used in conjunction with more feature rich data.

References

Bureau of Infrastructure and Transport Research Economics. (2025). Australian Road Deaths Database-ARDD. Retrieved from https://www.bitre.gov.au/statistics/safety/fatal_road_crash_database

Angwin, J., Larson, J., Kirchner, L., & Mattu, S. (2016). Machine Bias. Retrieved from https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Adaloglou, N. (2020). Intuitive Explanation of Skip Connections in Deep Learning. Retrieved from https://theaisummer.com/skip-connections/

Appendix A – Data, Figures and Tables

Data dictionary

Variable name	Description	Datatype
Crash ID	National crash identifying number (9-19 characters)	Numeric
State	Australian state/territory where crash occurred (NSW, VIC, QLD, etc.)	Nominal categorical
Month	Month of crash (1-12)	Numeric
Year	Year of crash (1989–current)	Numeric
Dayweek	Day of week (Monday–Sunday)	Ordinal categorical
Time	Time of crash (hh:mm:ss)	Numeric
Crash Type [Target]	Number of vehicles involved (Single, Multiple, Unknown)	Nominal categorical
Bus Involvement	Whether a bus was involved (Yes, No, -9)	Nominal categorical
Heavy Rigid Truck Involvement	Whether a heavy rigid truck was involved (Yes, No, -9)	Nominal categorical
Articulated Truck Involvement	Whether an articulated truck was involved (Yes, No, -9)	Nominal categorical
Speed Limit	Posted speed limit at crash location (km/h)	Numeric
Road User	Type of road user killed (Driver, Pedestrian, Cyclist, etc.)	Nominal categorical
Gender	Sex of killed person (Male, Female, Unknown)	Nominal categorical
Age	Age of killed person (years)	Numeric
National Remoteness Areas 2021	Remoteness classification (Major Cities, Inner Regional, etc.)	Ordinal categorical
SA4 Name 2021	Statistical Area Level 4 name	Free-form text
National LGA Name 2021	Local Government Area name	Free-form text
National Road Type	Type of road (National or State Highway, Arterial Road, Sub-arterial Road, Collector Road, Local Road, Access Road, Busway, Pedestrian Thoroughfare)	Ordinal categorical

Christmas Period	Whether crash occurred during Christmas Period (Yes, No)	Nominal categorical
Easter Period	Whether crash occurred during Easter (Yes, No)	Nominal categorical

Table 1

Inputs

Road features	State, Speed Limit, National Road Type
Victim features	Road User, Age, Gender
Vehicle involvement features	Bus involvement, Articulated Truck Involvement, Heavy Rigid Truck Involvement
Temporal features	Dayweek, Time, Christmas Period, Easter Period
Target feature	Crash Type

Table 2

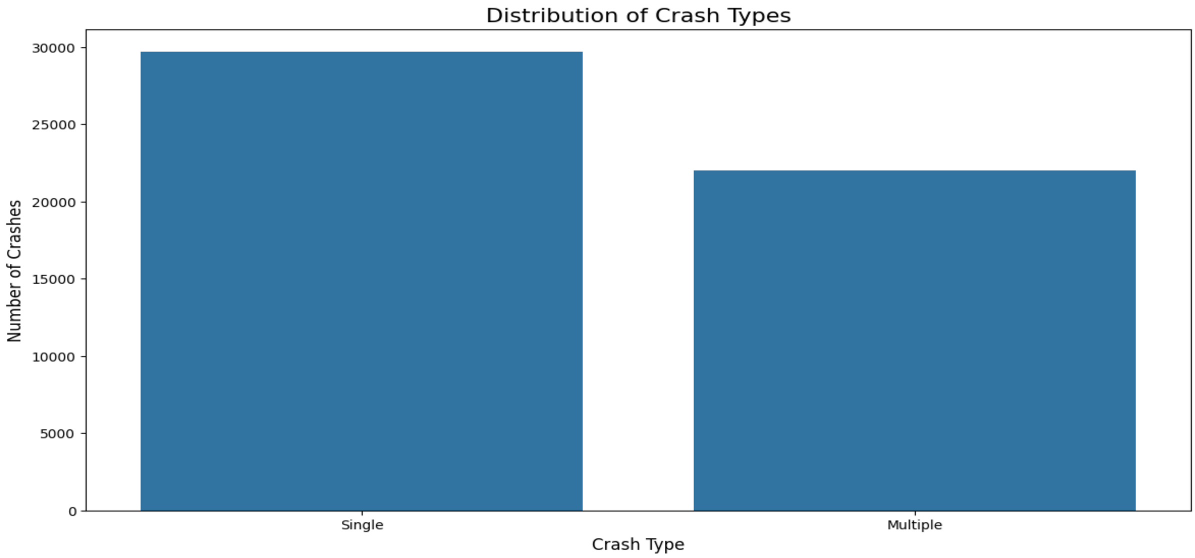

Figure 1

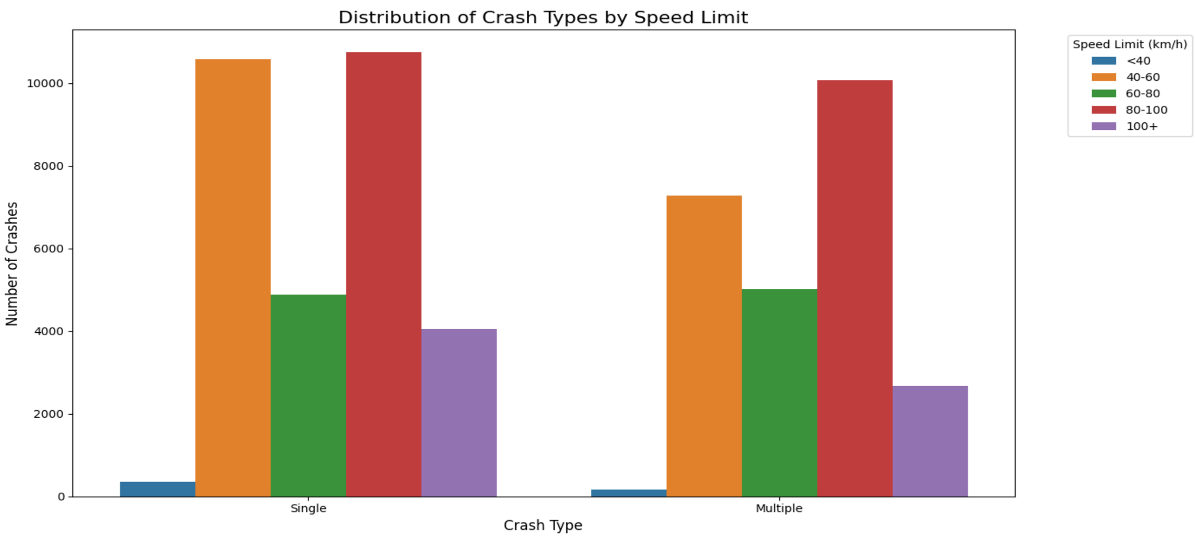

Figure 2

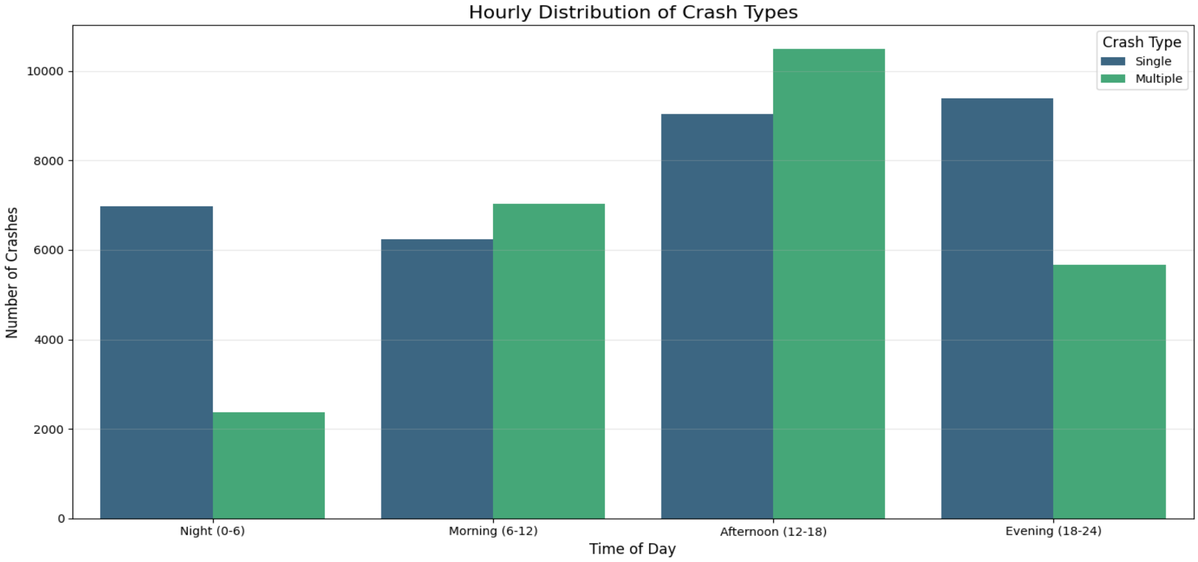

Figure 3

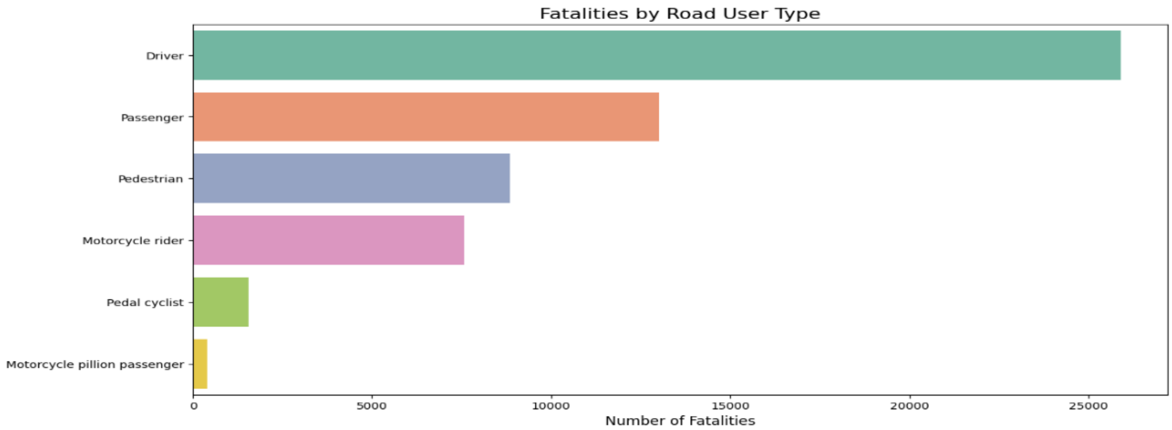

Figure 4

Figure 5

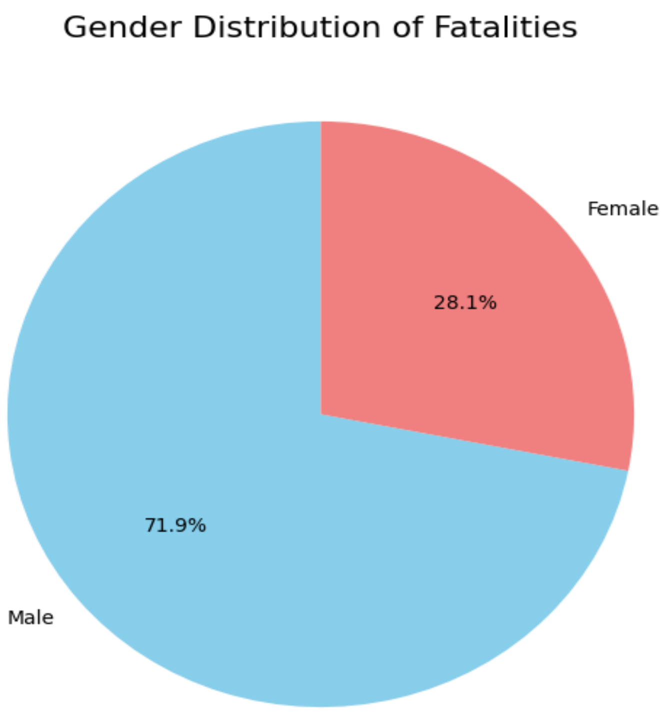

Figure 6

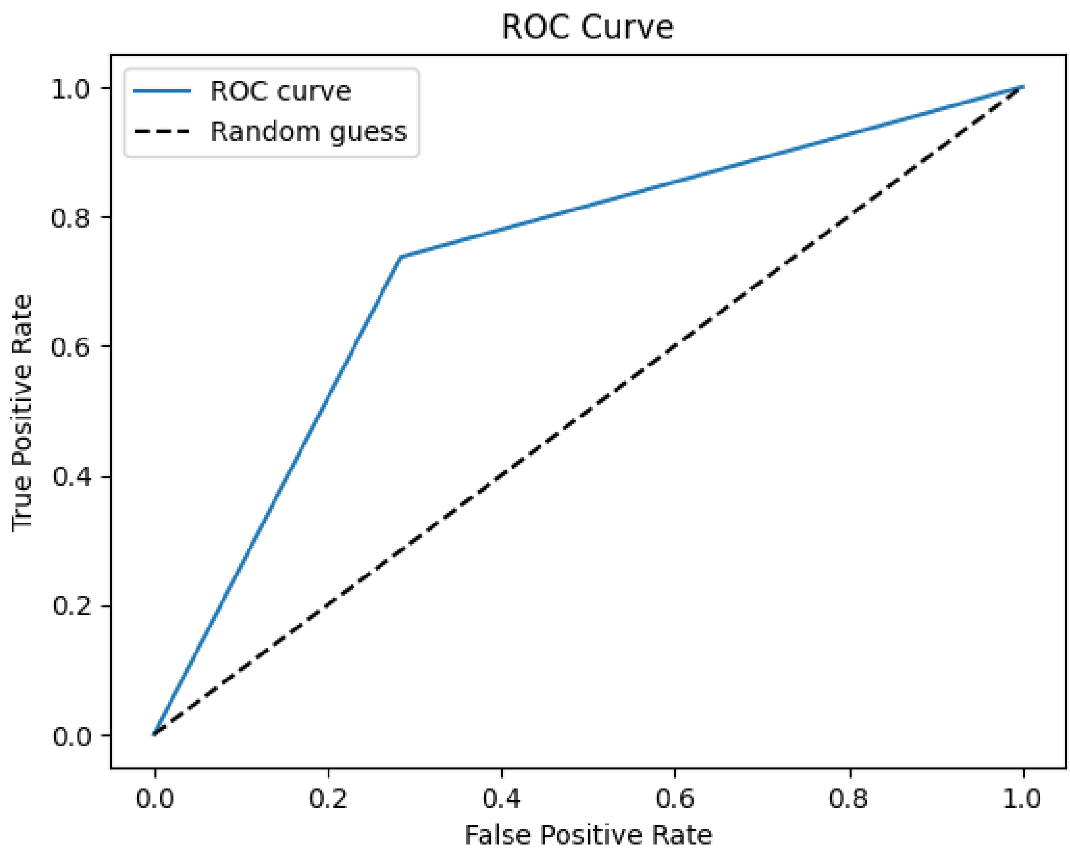

Figure 7

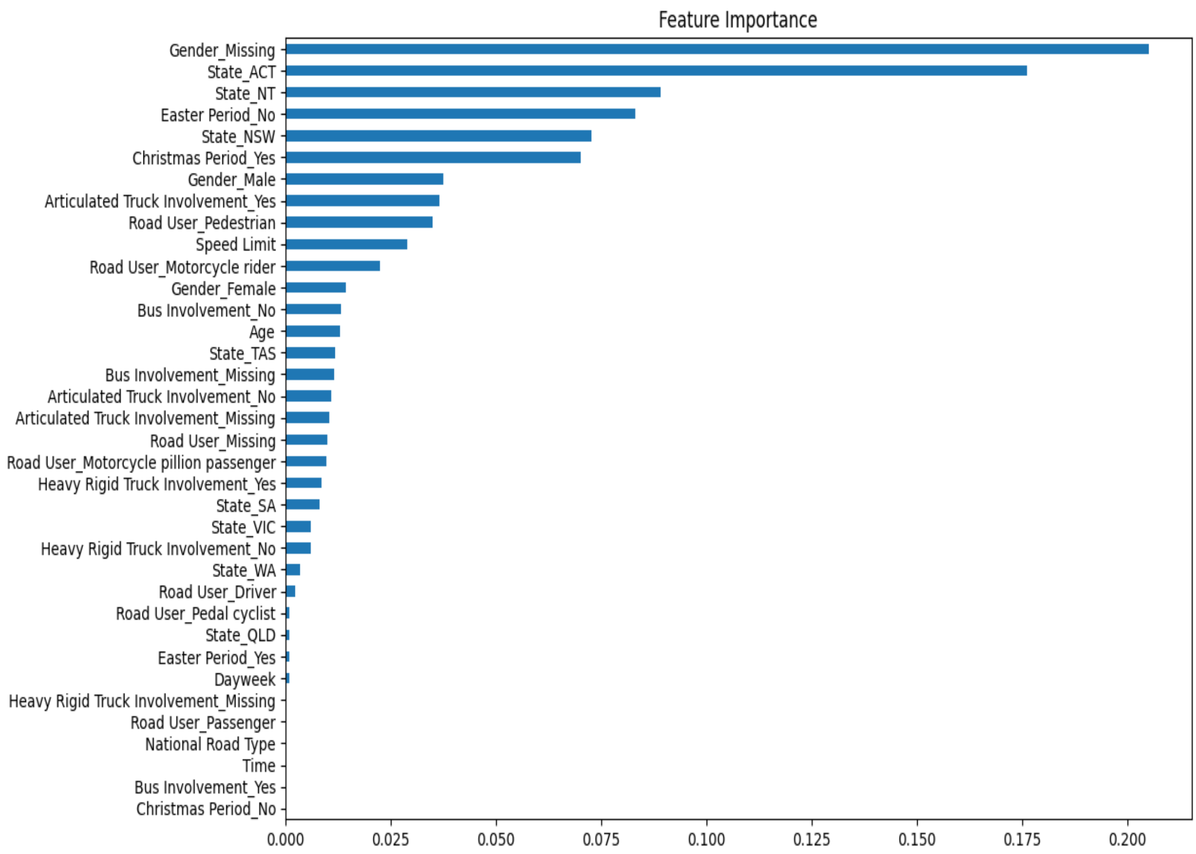

Figure 8

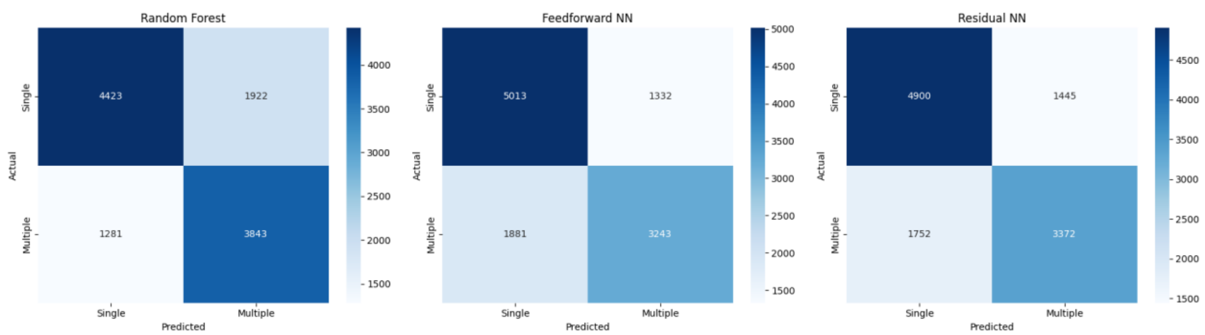

Figure 9

Appendix B – Generative AI Usage

- "Provide some suggestions on what my second deep learning model should be. I am using a feedforward nn as my first model and my task is to binary classify some data"
- "How do I mount my files explicitly in Google colab"
- "Please help me generate a table to show all my results in python. I have a function that returns 'classification report' which has the keys 'Precision', 'Recall' and 'Accuracy' on the data 'Single' and 'Multiple' "
- "Am I able to save my pickle file as a keras file for my random forest implementation. It uses sklearn but I want to save it as keras so I can load it and be consistent with my deep learning models."

- “Please turn my GridSearch hyperparameter tuning into Bayesian Optimisation hyperparameter tuning”
- “Please check over this work to make sure that it makes sense or if I need to add anything more”
- Code errors debugged using in-built Google Colab Gemini