Module 6 - Final Exam

ALY6020: Predictive Analytics

Lectured by

Prof. Amin Karimpour

College of Professional Studies

Northeastern University

Submitted by

Student: Sudhamshu Vidyananda

NU ID: 001527328

Degree: Master of Professional Studies - Analytics

Date of submission: 10/28/2023

INTRODUCTION

In predictive Analytics, we have seen all the different kinds of models used to predict the dependent variable and see how well the model is behaving by comparing the accuracy of different models. Below is the summary of each of the model seen so far:

Nearest Neighbor Model: The Nearest Neighbor model, specifically the k-Nearest Neighbors (k-NN) method, is a sort of instance-based learning in which predictions are made based on the majority class of the dataset's 'k' nearest data points. The distance between two points, which is frequently Euclidean, establishes their proximity. It is non-parametric, which means it makes no underlying assumptions regarding data distribution. Despite its simplicity and effectiveness, k-NN can be computationally demanding for big datasets due to the necessity to compute distances to all training samples for each prediction.

Regression Model: Regression models attempt to forecast a continuous result variable using one or more predictor variables. The most frequent is linear regression, which assumes a straight line between the predictors and the result. Other forms, such as logistic regression (used for binary classification), polynomial regression, and ridge/lasso regression, deal with various sorts of correlations and complexity.

Classification Model: Based on input characteristics, classification algorithms anticipate discrete labels or categories. Logistic Regression (despite its name), Support Vector Machines (SVM), and k-NN are a few examples. These models function by creating feature space decision boundaries that best distinguish various classes. The nature of these boundaries (linear or non-linear) and how they are calculated differ depending on the method.

Decision Tree Model: Based on the value of input attributes, Decision Trees divide the data into subgroups. This procedure is performed recursively, culminating in a decision-tree-like model. They can tolerate non-linearities and can be used for both regression and classification. Random Forests and Gradient Boosted Trees are ensemble approaches that use decision trees to build many trees and aggregate their findings for more robust and accurate predictions.

Neural Network Model: A subset of machine learning models inspired by the structure of the human brain is neural networks. They are made up of linked nodes or "neurons" that are grouped into three layers: input, hidden, and output. Data is sent through the network, where it is transformed using weights and activation functions. Neural networks are extremely effective at capturing complicated, non-linear interactions and serve as the foundation for most current deep learning applications, such as image and speech recognition.

To summarize, the choice of model is frequently determined by the nature of the data and the task at hand. While simpler models such as linear regression or k-NN may be enough for some datasets, complicated issues or huge, high-dimensional datasets may necessitate more sophisticated models such as neural networks or ensemble approaches.

ANALYSIS

Part 1

Please provide only textual answers and 2-3 sentences around the rationale of why that is (no coding is required)

1. Give a definition of impurity. Please follow that up with a business example of low and high impurity.

The degree of disorder or unpredictability in a set is referred to as impurity. Impurity evaluates the homogeneity of labels inside a subset in machine learning, notably in decision trees and their ensemble counterparts. Impurity is 0 if a subset is totally homogenous (it contains only data points from one class). In contrast, if a subset is evenly distributed across several classes, its impurity is at its highest. Gini impurity and entropy are two common impurity measurements.

Business Case Study:
Consider a fruit vendor with baskets of fruits.

Low Impurity:
One basket is labeled "Apples" and exclusively includes apples. Because it contains only one variety of fruit, this basket is an excellent illustration of minimal impurity or high purity.
Business Implication: Because the vendor knows this basket includes solely apples, it is simple for him to sell or price it. It is not necessary to inspect or sort the contents.

High Impurity:
Another basket is labeled "Mixed Fruits" and contains nearly equal amounts of apples, oranges, bananas, and grapes. This basket has a high impurity level since it contains a variety of fruit species, making it heterogeneous.

Business Implication:
The vendor's ability to sell or price this basket becomes increasingly difficult. He may need to guarantee a healthy mix in every sale, or he may need to sort and segregate the fruits if a buyer requests only one variety. The procedure takes longer and may necessitate additional resources.

During training, the purpose of decision trees is frequently to minimize impurity: while determining how to divide the data at each node, the algorithm picks the split that results in the largest drop in impurity.

2. Compare and discuss similarities or differences between AIC and R2?

Nature and Function:

AIC: This is a model selection metric. AIC assesses a model's goodness-of-fit while punishing models for their complexity. A model with a low AIC score fits the data very well while utilizing fewer parameters.

R-squared: It is a measure of how much of the variation in the dependent variable can be predicted by the independent variables. Essentially, it assesses the model's explanatory capacity. A R-squared of 1 indicates that the model explains all the variability in the response data around its mean.

Range of Values:

AIC: There is no such thing as a limited range for AIC. Lower AIC values imply models that fit better, although there is no intrinsic "good" or "bad" number. It is more beneficial when comparing many models; normally, the model with the lowest AIC is selected.

R-squared: The value ranges from 0 to 1. A score of 0 implies that the model explains no variance, whereas a value of 1 suggests that it explains all variances.

Penalization:

AIC: AIC penalizes models for having too many parameters. This helps to avoid overfitting, which occurs when a model fits the training data very well but performs badly on fresh, previously unknown data.

R-squared: Model complexity is not penalized in R-squared. Adding more variables to a model will always raise (or maintain) the R-squared value, which might be deceptive at times.

Case Studies:

AIC: AIC is frequently used in the context of model selection among several models. It's very popular for choosing time series models.

R-squared: Commonly used in linear regression to determine the model's goodness-of-fit. It indicates how well the model predicts the dependent variable, but it does not always imply whether the model is acceptable or if forecasts will be true.

Limitations:

AIC: While AIC is useful for model comparison, it does not give an absolute measure of the quality of fit.

R-squared: This value can be unnaturally large if the model is overfit. It does not offer a comprehensive picture; additional measures, such as corrected R-squared or RMSE, may be evaluated in addition to it.

In conclusion, while both AIC and R-squared give insight into model performance, they serve distinct objectives. AIC is mainly concerned with model selection, balancing fit, and complexity, whereas R-squared quantifies the fraction of variability explained by the model. Both measurements are useful in their own settings and should be used with caution.

3. Name the three activation function techniques mentioned in class for Neural Networks. Provide an example of a business problem leveraging at least one such technique.

The 3 activation function techniques we learnt in the class for Neural Networks are:

Sigmoid Activation Function
Because its output may be read as a probability, the sigmoid function is widely utilized in the output layer of binary classification problems. The output range for Sigmoid Function ranges from [0,1]

ReLU (Rectified Linear Unit) Activation Function
Because it allows the model to account for non-linearities without being too computationally costly, ReLU has become the default activation function for many types of neural networks. The output range for ReLU ranges from [0, ∞].

Tanh (Hyperbolic Tangent) Activation Function
The zero-centered sigmoid is similar to sigmoid but with output values centered around 0. The output range for Tanh function ranges from [-1,1].

Business Problem Example Leveraging ReLU Activation Function

Consider a business situation in which a real estate company needs to forecast home values based on factors such as location, number of bedrooms, accessibility to facilities, and so on. They decide to capture intricate connections between these elements using a neural network.

The ReLU activation function can be used by the neural network's hidden layers in this circumstance. The reason for selecting ReLU is its efficiency and ability to cope with the vanishing gradient problem, which might impede deep network training. Because the prediction goal is regression (predicting a continuous value), the output layer may utilize a linear activation function. The network may learn non-linear patterns in the data by utilizing ReLU in the hidden layers, allowing it to make more accurate predictions regarding house prices.

4. Why is it important to find balance in the learning rate of a Gradient Boosted Model? Is there a learning rate for Random Forest?

The learning rate, often known as the "shrinkage" parameter in the context of Gradient Boosting, is critical to the model's success. Let's look at its significance and how it relates to Gradient Boosted Models and Random Forests.

Importance of Balance in the Learning Rate for Gradient Boosted Models

Preventing Overfitting: A lower learning rate can strengthen the boosting technique by making the optimization more thoughtful. This can help to avoid overfitting to training data.

Model Convergence: A learning rate that is too high may lead the model to converge too fast to a poor solution, while a learning rate that is too low may cause the model to converge extremely slowly, requiring many trees to attain excellent performance. This can be computationally costly and may not always result in ideal performance.

Trade-off with Tree Count: There is frequently a trade-off between learning rate and tree count. A slower learning rate will often need more trees in the ensemble to sustain. Thus, tuning the learning rate often involves adjusting the number of trees as well.

Learning Rate in Random Forest

Random Forests do not have the same learning rate as Gradient Boosted Models. This is because Random Forests and Gradient Boosted Models work on fundamentally distinct ideas. Random Forests construct each tree individually by bootstrapping samples and using a subset of split characteristics. The final prediction is an ensemble outcome (for example, a majority vote for classification or an average for regression). Enhanced Gradient Models, on the other hand, construct trees in a sequential fashion. Each tree attempts to remedy the mistakes committed by the one before it. In this case, the learning rate governs how much each tree contributes to the final forecast. This is a type of regularization.

In conclusion, the learning rate in Gradient Boosted Models is critical for balancing high performance without overfitting and ensuring efficient training. Random Forests, on the other hand, do not use a learning rate in the same way because of their independent tree-building mechanism.

5. Provide two examples of Machine Learning models that are suitable to support decision-making by executive management. Name two optimization algorithms used in Machine Learning.

Machine Learning Models Suitable for Executive Decision-making

1. ***Time Series Forecasting Models***

These models forecast future values based on already observed values. Anything from stock prices to sales numbers may be predicted using time series models.

Executive Decision-Making Use Case: A retail executive might utilize time series forecasting to estimate future sales, assisting them in making decisions regarding inventory, personnel, and promotions. For example, they can forecast product demand for the approaching Christmas season using models like as ARIMA (Autoregressive Integrated Moving Average) or Prophet (created by Facebook).

2. ***Clustering Models (K-Means Clustering)***

Clustering models group data points that are similar to one another. They represent a type of unsupervised learning.

Executive Decision-Making Use Case: A telecoms executive could use clustering to divide their client base into separate groups based on consumption trends, demographics, or service preferences. This may be used to inform focused marketing efforts, tailored offers, or service package strategies to improve client retention and sales.

Optimization Algorithms in Machine Learning:

1. ***Gradient Descent***

Gradient descent is used in neural networks to optimize the network's weights when training them. The model iteratively refines its predictions by computing the gradient of the loss function with respect to each weight and modifying the weights in the direction that decreases the loss. Gradient descent variants such as Stochastic Gradient Descent (SGD), Mini-batch Gradient Descent, and optimization approaches such as Adam or RMSprop improve on the fundamental gradient descent algorithm.

2. ***Friedman's Gradient Boosting Algorithm***

The gradient boosting approach iteratively adds trees to the model to maximize a differentiable loss function. At each step, a decision tree is fitted to the loss function's negative gradient (or "pseudo-residuals"). This method is like gradient descent, but in function space, with each tree correcting the flaws of its predecessor. Both optimization methods are critical in structuring machine learning models to create accurate predictions and insights, which executives may then use to make educated decisions.

Part 2

You have a dataset about whether or not someone got in an accident that year. This insurance company needs to understand which features of a person or vehicle make them more likely to get in an accident.

Clean missing values first, but no exploratory info is needed. You need to predict the Outcome column.

1. Clean the dataset and discuss how you cleaned each variable with missing values and why you chose that method.

Data Cleaning is a very important step in model building and it should be the priority and the first step taken up in data analysis. . In the car_insurance campaign dataset, we have a dimension of **10000 rows × 17** columns. In this dataset, we have a total of **6** categorical variables and **11** numerical variables. **CREDIT_SCORE** is having a total of **982** null values and **ANNUAL_MILEAGE** is having a total of **957** null values. I have also taken a percentage of these missing data and CREDIT_SCORE has **9.82%** of missing data and ANNUAL_MILEAGE is having **9.57%** of missing data. I have decided to use **median value** of these columns to replace the null values and not remove it since there is high risk of losing valuable data and median is also less sensitive to outliers than the mean. I also checked if the dataset contains any of these special characters **['!@#$%^&*(),.?":{}|<>']** and found none. I checked if the dataset contains any **duplicates** and found no duplicates in the dataset. I have used **IQR (Interquartile Range)** method to see if the numeric columns data point is an outlier. CREDIT_SCORE is having a total of **57** outliers, ANNUAL_MILEAGE is having a total of **273** outliers, SPEEDING_VIOLATION is having a total of **588** outliers, PAST_ACCIDENTS is having a total of **285** outliers, DUIS is having a total of **1882** outliers, POSTAL_CODE is having a total of **484** outliers. Based on the nature of the dataset (car_insurance) and the columns with outliers, I have decided to not remove the outliers for **SPEEDING_VIOLATIONS, DUIS, PAST_ACCIDENTS** as these columns might have higher values for risky drivers. It's essential for an insurance dataset to capture such risk. Removing these might lead to loss of valuable data. Thus, it's advisable to leave these outliers as-is. Outliers in **CREDIT_SCORE** might not be errors but indicate people with very high or very low credit scores. It might be best to cap these values at the lower and upper bounds to avoid extreme values without eliminating the data points. With respect to **ANNUAL_MILEAGE** its plausible for some individuals to drive very little or a lot within a year. Again, capping might be a suitable approach to ensure that the dataset doesn't have extreme values. **POSTAL_CODE** again might not be an error because it can represent any place in the country and capping the data will be a better option in this case as well. The outliers for the CREDIT_SCORE , POSTAL_CODE and ANNUAL_MILEAGE columns have been capped at their respective lower and upper bounds. The other columns with outliers (SPEEDING_VIOLATIONS, DUIS, PAST_ACCIDENTS) have been left unchanged as they capture essential risk characteristics in the car insurance context. Post capping, the CREDIT_SCORE values now lie between **0.167 and 0.871**, POSTAL_CODE values now lie

between **10,238 and 66,555.5** and ANNUAL_MILEAGE value now lies between **5500 and 17,500**.

Exploratory Data Analysis

Distribution plot for numeric columns

In this dataset, we have a total of 11 numerical columns but considering only 10 as ID is irrelevant in this analysis. namely: *['CREDIT_SCORE', 'ANNUAL_MILEAGE', 'SPEEDING_VIOLATIONS', 'DUIS', 'PAST_ACCIDENTS','VEHICLE_OWNERSHIP', 'MARRIED','CHILDREN','POSTAL_CODE','OUTCOME']*.

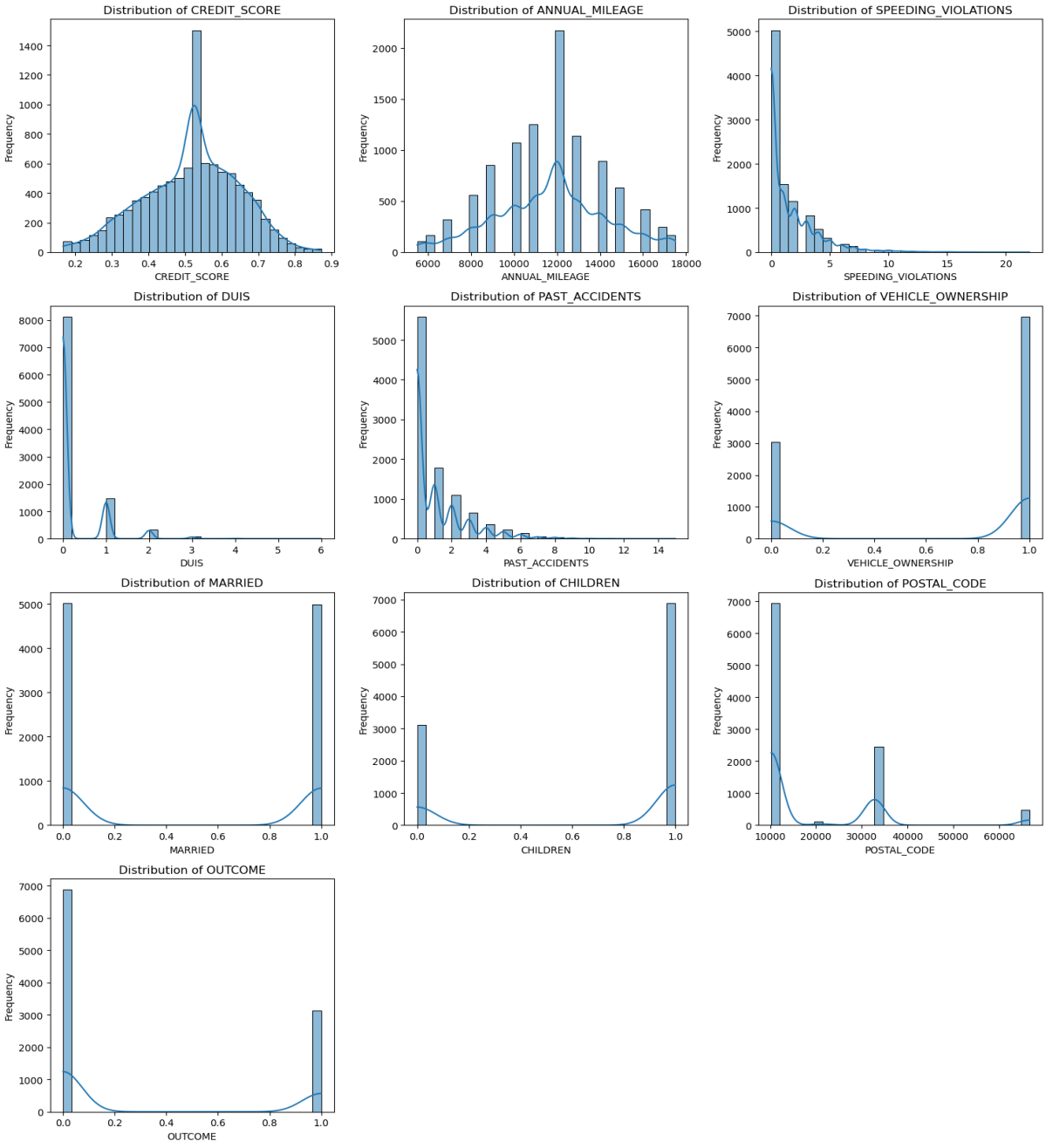

Fig 1.1

In the above plot **(Fig 1.1),** we see that, for CREDIT_SCORE, the distribution appears to be roughly bell-shaped, with a slight skew towards higher scores. For ANNUAL_MILEAGE, there is a concentration of data around the 10,000 to 15,000 mileage range, which might be indicative of average driving behavior. For SPEEDING_VIOLATIONS, most people have 0 or very few speeding violations, with a rapid decrease as the number of violations increases.
A vast majority have 0 DUIs, with a few having 1, 2 or more. For PAST_ACCIDENTS, many people have not had past accidents, but there is a significant number who have had 1 or 2. We see a lot of people owning a vehicle compared to people not owning one. We also see that, the number of people married/unmarried are almost the same number in count. We see that the people having children is more than people not having. POSTAL_CODE shows a significant peak at 10000 and between 30000 and 40000. The OUTCOME variable shows more people/vehicle have not met with an accident compared to the ones who met with an accident.

Bar plot for categorical variables

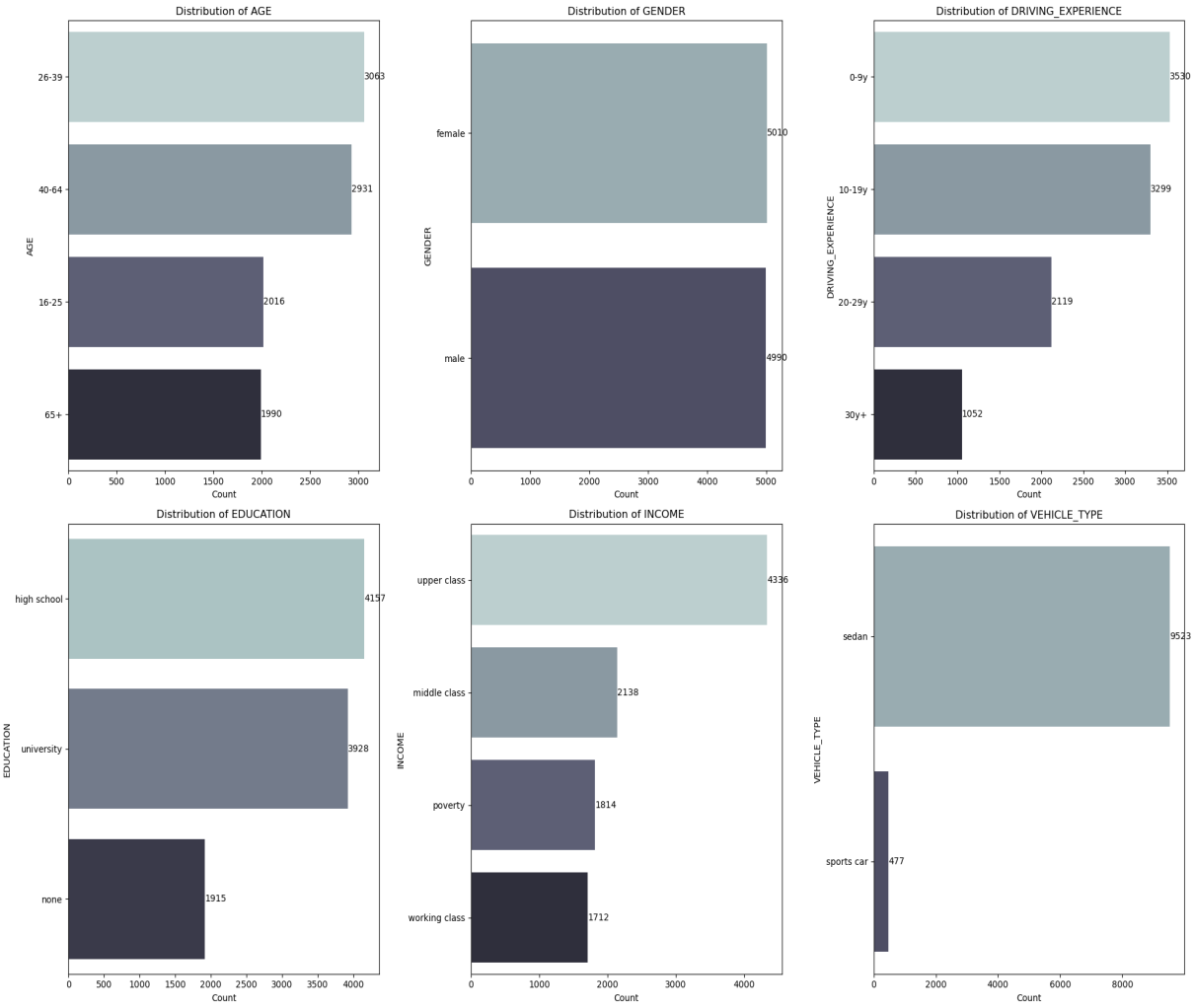

Fig 1.2

In the above plot **(Fig 1.2)**, we see that, 3063 of the people belong to the age group of 26-39 and 1990 of them belong to 65+. As per the Age plot, adults are comparatively more than old

age people. As per the Gender plot, female ratio is more than male ratio. As per Driving_Experience plot, most individuals have driving experience in the range of 0-9 years, followed by those with 10-19 years of experience. As per the Education plot, high school students are comparatively more than university students and other professionals. As per the Income plot, upper class people are more than middle class and working class. Higher income people are comparatively more than middle class, poverty and working class. As per the Vehicle_Type plot, majority of individuals drive sedans, with a smaller portion driving sports cars.

Bar plot for OUTCOME (target) variable

As per the below bar plot **(Fig 1.3)**, we see that the number of accidents that occurred is less. We have only 3133 reported. 6867 individuals/vehicles have not met with an accident.

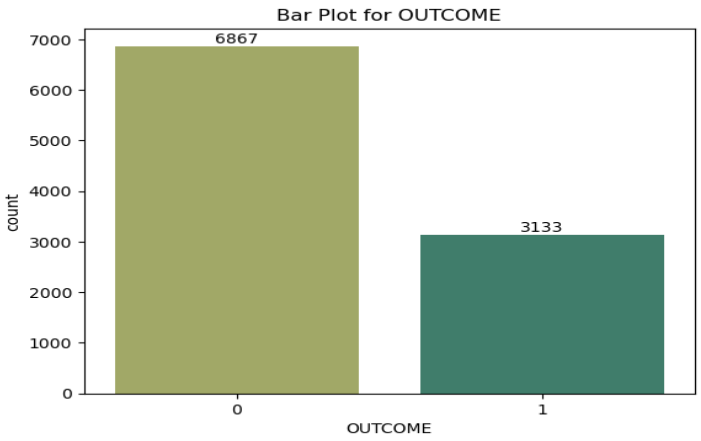

Fig 1.3

Bar plot for VEHICLE_TYPE, AGE, DRIVING_EXPERIENCE and MARRIED w.r.t OUCOME target variable

As per the below plot **(Fig 1.4)**, we see that, a greater number of accidents have occurred with Sedan Vehicle_Type. 65+ age group have a smaller number of accidents reported compared to 16-25 years. 0-9 years driving experience individuals have reported more accidents than 10-19 y, 20-29y and 30+y. We also see that unmarried individuals have met with more accidents than married ones.

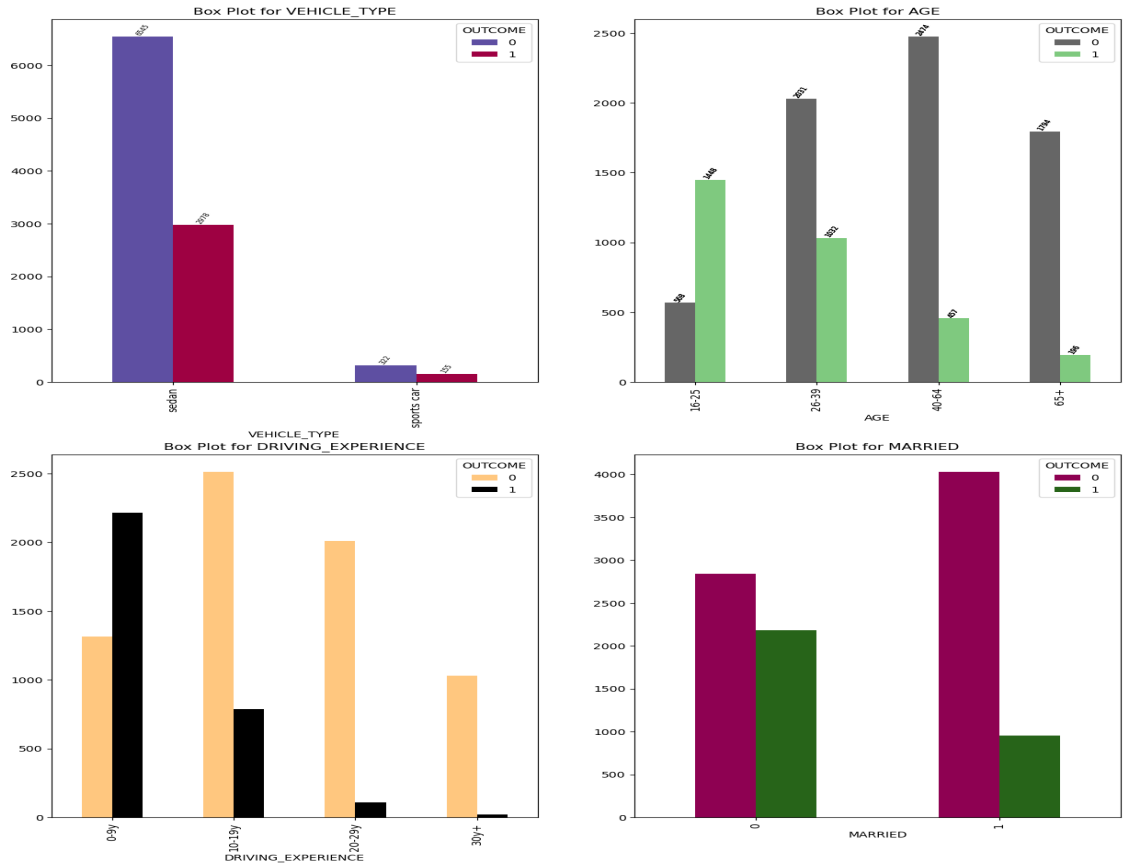

Fig 1.4

Correlation Matrix

```
OUTCOME                1.00
ANNUAL_MILEAGE         0.18
GENDER                 0.11
POSTAL_CODE            0.11
VEHICLE_TYPE           0.01
ID                    -0.01
INCOME                -0.05
EDUCATION             -0.09
DUIS                  -0.19
CHILDREN              -0.23
MARRIED               -0.26
SPEEDING_VIOLATIONS   -0.29
CREDIT_SCORE          -0.31
PAST_ACCIDENTS        -0.31
VEHICLE_OWNERSHIP     -0.38
AGE                   -0.45
DRIVING_EXPERIENCE    -0.50
Name: OUTCOME, dtype: float64
```

Fig 1.5

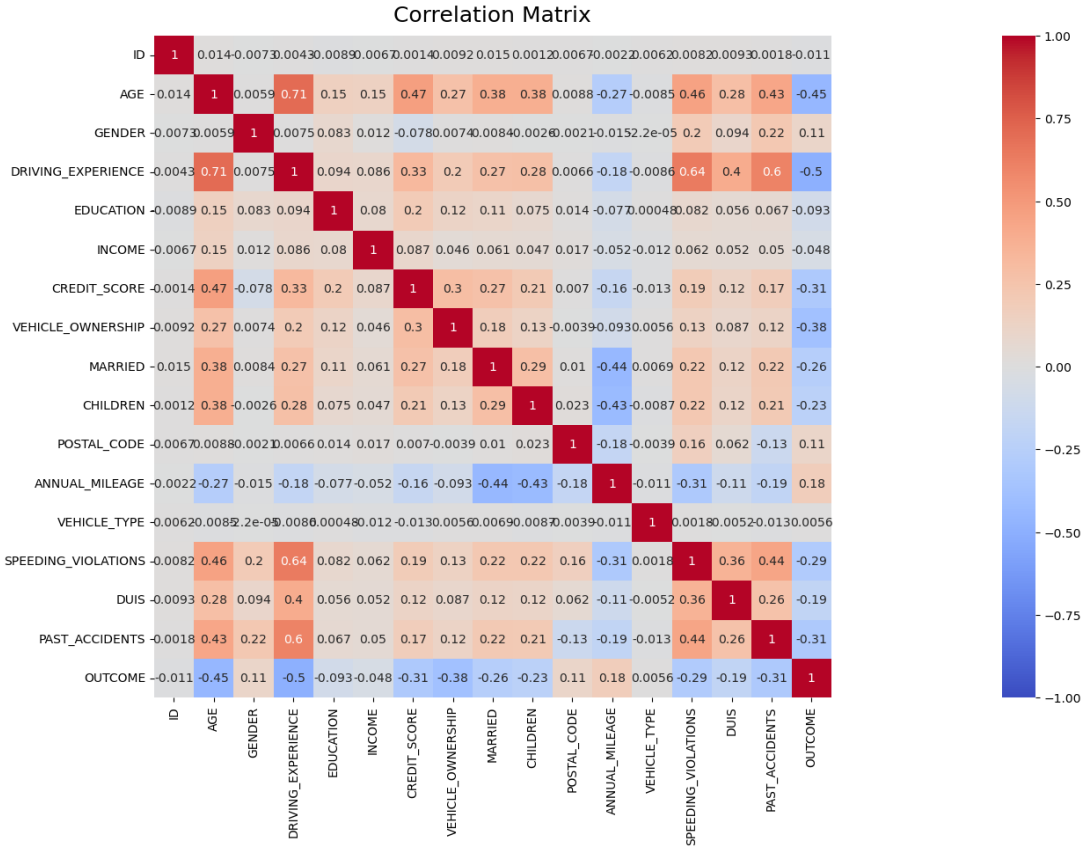

Fig 1.6

As per the above correlation value and matrix, we see that OUTCOME variable is strongly correlated to ANNUAL_MILEAGE having correlation value of 18%. GENDER and POSTAL_CODE is correlated with each other having 11% correlation value which in turn shows multicollinearity. OUTCOME is least correlated to DRIVING_EXPERIENCE with correlation value of -50%. DRIVING_EXPERIENCE and AGE are highly correlated having a value of 71% which is correct. Higher the age, more driving experience and vice versa.

2. Build a logistic regression model and discuss the significant variables. Provide a table of all significant variables and their coefficients (a snippet of the data is not acceptable and if there are no variables at .05 or under, feel free to expand to .1). From your initial thoughts, which variable sticks out to you as intriguing that it is significant and why. How could this information be useful to the insurer?

I have taken dependent variable as *OUTCOME* and independent variables as *['CREDIT_SCORE', 'ANNUAL_MILEAGE', 'SPEEDING_VIOLATIONS', 'DUIS', 'PAST_ACCIDENTS', 'VEHICLE_OWNERSHIP', 'INCOME', 'EDUCATION', 'AGE', 'DRIVING_EXPERIENCE, 'MARRIED', 'CHILDREN', 'VEHICLE_TYPE, 'POSTAL_CODE' ,'GENDER',].*

I have split the dataset into training and test dataset and called the logistic regression method.

```
Optimization terminated successfully.
         Current function value: 0.383190
         Iterations 7
                           Logit Regression Results
==============================================================================
Dep. Variable:                OUTCOME   No. Observations:                 7000
Model:                          Logit   Df Residuals:                     6984
Method:                           MLE   Df Model:                           15
Date:                Sat, 28 Oct 2023   Pseudo R-squ.:                  0.3840
Time:                        15:41:48   Log-Likelihood:                -2682.3
converged:                       True   LL-Null:                       -4354.3
Covariance Type:            nonrobust   LLR p-value:                     0.000
==============================================================================
                         coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const                  0.6026      0.278      2.169      0.030       0.058       1.147
AGE                   -0.1120      0.048     -2.334      0.020      -0.206      -0.018
GENDER                 0.8718      0.073     11.943      0.000       0.729       1.015
DRIVING_EXPERIENCE    -1.6033      0.085    -18.954      0.000      -1.769      -1.438
EDUCATION             -0.0482      0.040     -1.201      0.230      -0.127       0.030
INCOME                 0.0182      0.032      0.574      0.566      -0.044       0.080
CREDIT_SCORE          -1.1074      0.301     -3.681      0.000      -1.697      -0.518
VEHICLE_OWNERSHIP     -1.6798      0.075    -22.466      0.000      -1.826      -1.533
MARRIED               -0.4021      0.080     -5.050      0.000      -0.558      -0.246
CHILDREN              -0.1379      0.080     -1.730      0.084      -0.294       0.018
POSTAL_CODE         3.121e-05   2.49e-06     12.514      0.000    2.63e-05    3.61e-05
ANNUAL_MILEAGE      7.256e-05   1.64e-05      4.436      0.000    4.05e-05       0.000
VEHICLE_TYPE           0.1366      0.157      0.869      0.385      -0.172       0.445
SPEEDING_VIOLATIONS    0.0418      0.030      1.409      0.159      -0.016       0.100
DUIS                   0.0779      0.088      0.889      0.374      -0.094       0.250
PAST_ACCIDENTS        -0.1296      0.042     -3.056      0.002      -0.213      -0.046
==============================================================================
```

Fig 1.7

As per the above snapshot **(Fig 1.7)**, we received a R-squared value of 38% which is not a great value. The below table **(Fig 1.8)** shows the table of all significant variables and their co-efficient.

	Coef.	Std.Err.	z	P>|z|	[0.025	0.975]
const	0.602554	0.277855	2.168593	3.011359e-02	0.057969	1.147139
AGE	-0.112011	0.047986	-2.334230	1.958367e-02	-0.206061	-0.017960
GENDER	0.871828	0.072999	11.942975	7.065003e-33	0.728752	1.014904
DRIVING_EXPERIENCE	-1.603318	0.084590	-18.954036	4.089824e-80	-1.769111	-1.437525
CREDIT_SCORE	-1.107379	0.300864	-3.680660	2.326310e-04	-1.697062	-0.517696
VEHICLE_OWNERSHIP	-1.679753	0.074768	-22.466117	8.904483e-112	-1.826296	-1.533210
MARRIED	-0.402117	0.079630	-5.049821	4.422249e-07	-0.558189	-0.246045
POSTAL_CODE	0.000031	0.000002	12.514404	6.227350e-36	0.000026	0.000036
ANNUAL_MILEAGE	0.000073	0.000016	4.435567	9.183047e-06	0.000040	0.000105
PAST_ACCIDENTS	-0.129645	0.042427	-3.055717	2.245228e-03	-0.212801	-0.046490

Fig 1.7

Here, all the p-values are less than 0.05 and GENDER is having the highest co-efficient value of 87%. POSTAL_CODE and ANNUAL_MILEAGE is having co-efficient close to 0 but rest of the variables have a negative coefficient value.

VEHICLE_OWNERSHIP is intriguingly significant. The negative coefficient indicates that car owners are less likely to register an accident. This may seem contradictory because one may expect car owners to drive more and hence be more likely to meet with accidents. However, vehicle owners may take better care of their vehicles or be more experienced drivers.

This information can help the insurer in a variety of ways:

Pricing Policies: Knowing which variables have a substantial impact on claims might help insurance companies price their policies more properly.
Marketing Tailoring: The insurer can develop tailored marketing campaigns. Offering discounts to car owners, for example, may attract a clientele that is less inclined to submit claims.
Risk assessment: This process helps to determine which variables enhance or lessen the risk of an event occurring.

Classification Report

Below **(Fig 1.7)**, shows the classification report for logistic regression model. We got an accuracy of 80% which is a good number and a precision score of 70%.

```
Accuracy Score = 0.8
Precision Score = 0.7
              precision    recall  f1-score   support

           0       0.84      0.88      0.86      2063
           1       0.70      0.62      0.66       937

    accuracy                           0.80      3000
   macro avg       0.77      0.75      0.76      3000
weighted avg       0.79      0.80      0.80      3000
```

Fig 1.7

Confusion Matrix

Below **(Fig 1.8)**, shows the confusion matrix for logistic regression model. We see that 1813 of the data were predicted as accident occurred which is correct and 583 of them were reported as accident not occurred which is correct. 250 of them were predicted as accident occurred but were not occurred and 354 were predicted as not occurred but were occurred.

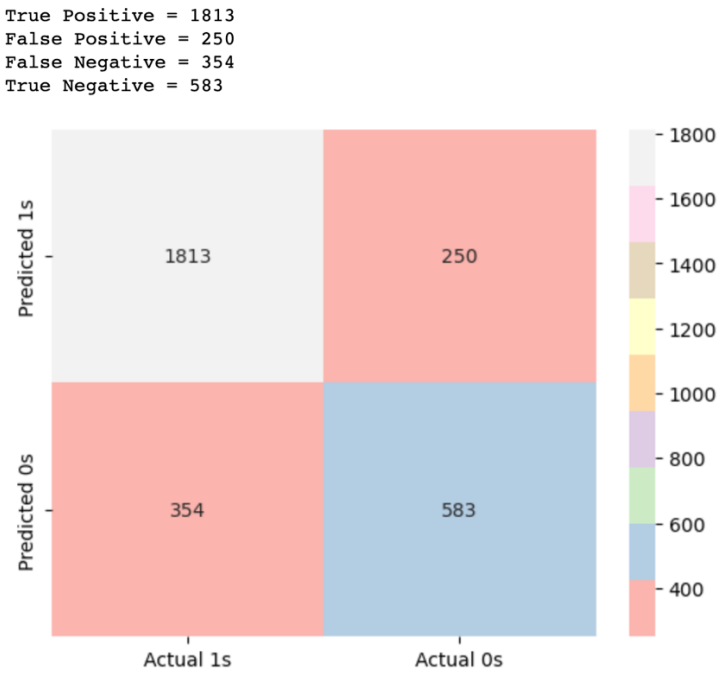

Fig 1.8

ROC curve and ROC score

We received an AUC score of 0.75 (75%) which is a good number and can say that the model did a pretty good job in predicting the outcome variable.

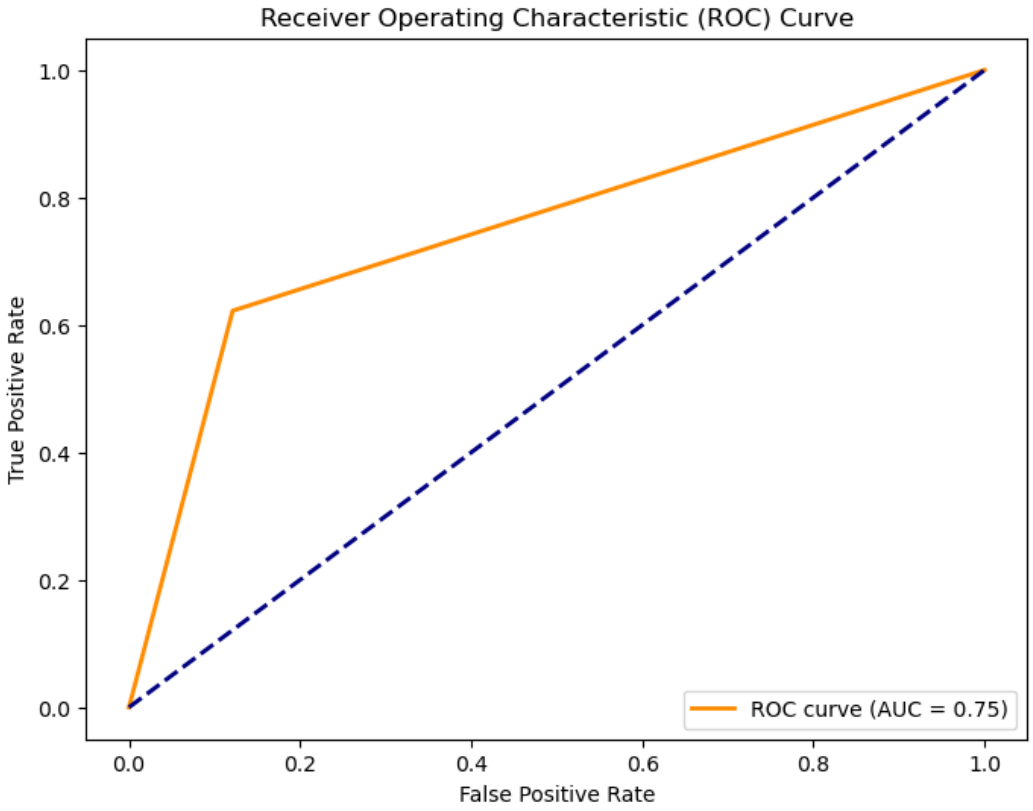

Fig 1.9

3. Run a few non-ensemble models (only ones used in class) using (1000 iterations). Address the accuracy of each model and why you choose that model. Which model is the most accurate?

4. Build a confusion matrix for each model; discuss which part of the confusion matrix a company would want to reduce, and which model does the best at doing so.

The non-ensemble models I will be including is KNN Model, Random Forest Model, Decision Tree Model, Gradient Boosting Model, Linear Regression Model along with Forward/Backward Selection, Neural Networks Model and SVM Model.

KNN Model

Because k-NN doesn't make any assumptions about the relationships between features, it can naturally capture interactions between features without any additional work. Given its simplicity, k-NN can serve as a quick baseline model to benchmark more complex models against.

```
Accuracy KNN Model : 0.8
Precision Score = 0.73
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      2063
           1       0.73      0.58      0.65       937

    accuracy                           0.80      3000
   macro avg       0.78      0.74      0.75      3000
weighted avg       0.80      0.80      0.80      3000
```

Fig 2.0

As per the above snapshot **(Fig 2.0)**, we see that, we received an accuracy of 80% which is similar to logistic regression model. We received a precision score of 73% which is slightly higher than logistic regression model.

Confusion Matrix:

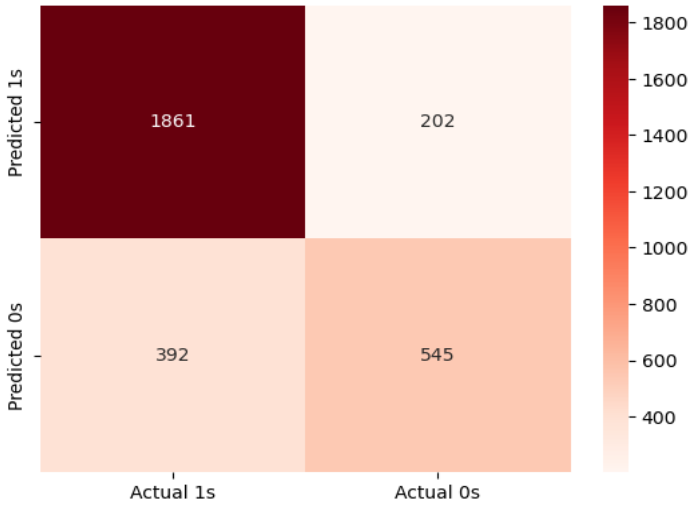

Fig 2.1

From the above confusion matrix **(Fig 2.1)**, we see that, 1861 of them were predicted as accident occurred correctly and 545 of them were predicted as accident not occurred which were predicted correctly. 202 of them were predicted as accident occurred but were not occurred and 392 of them were predicted as accident not occurred but were actually occurred.

ROC curve

From the below graph **(Fig 2.2)**, we get an AUC score of 86% which is a very good score compared to logistic model.

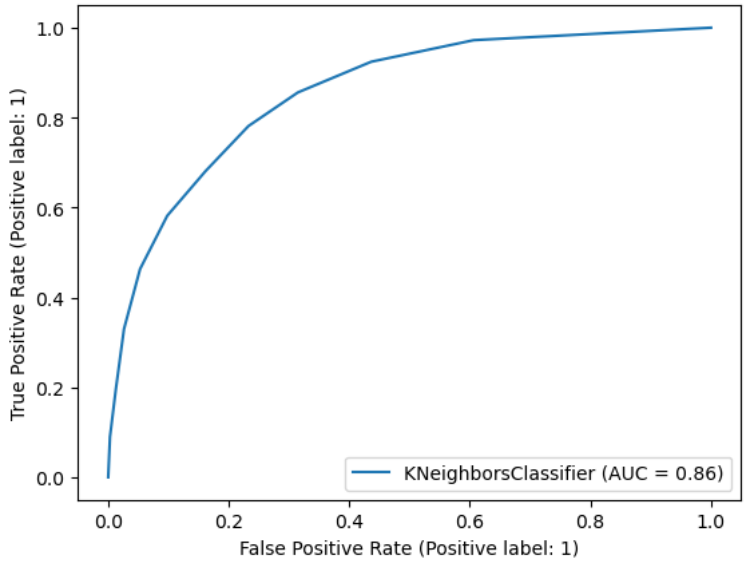

Fig 2.2

Random Forest Model

One of the standout features of Random Forest is its ability to provide a ranking of the importance of input features in predicting the target. This can be invaluable for understanding the data and the prediction task. It can handle binary features, categorical features, and numerical features without any need for scaling.

The below snapshot **(Fig 2.3)** shows an accuracy of 82% for random forest model which is slightly higher than logistic model and KNN model. We have a precision score of 73% which is equal to KNN model.

```
Accuracy Random Forest =  0.82
Precision Score = 0.73
Classification Report:
              precision    recall  f1-score   support

           0       0.85      0.89      0.87      2063
           1       0.73      0.65      0.69       937

    accuracy                           0.82      3000
   macro avg       0.79      0.77      0.78      3000
weighted avg       0.81      0.82      0.81      3000
```

Fig 2.3

Confusion Matrix

The below matrix **(Fig 2.4),** 1838 of them were predicted as accident occurred correctly and 613 of them were predicted as accident not occurred which were predicted correctly. 225 of them were predicted as accident occurred but were not occurred and 324 of them were predicted as accident not occurred but were actually occurred.

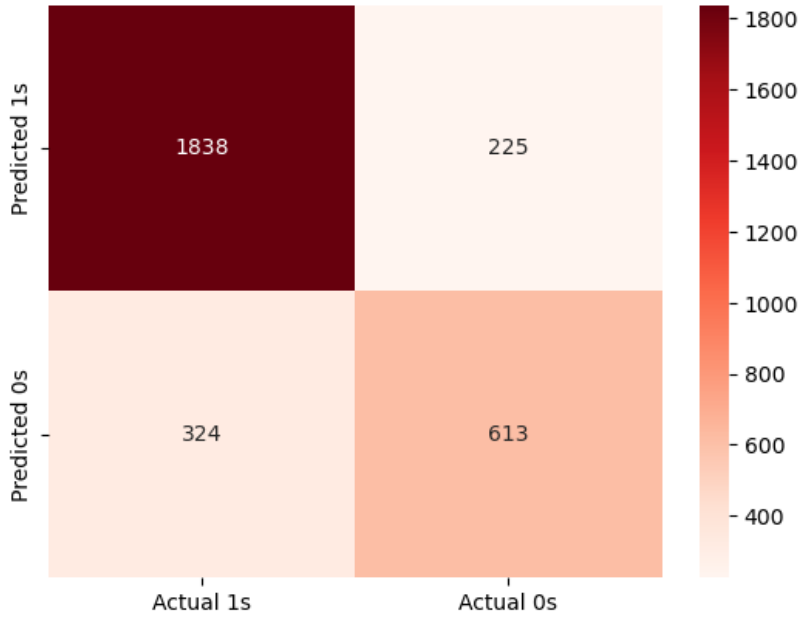

Fig 2.4

ROC curve

The below graph **(Fig 2.5)** shows an AUC score of 0.89 which is a very good score when compared to logistic regression model and KNN model. This shows to be a good fit.

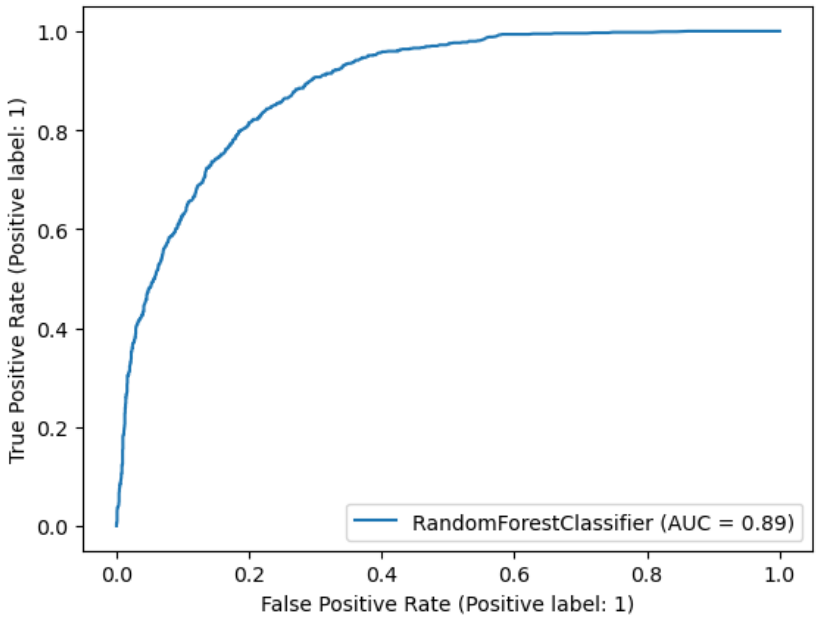

Fig 2.5

Gradient Boosting Model

Gradient Boosting often provides superior predictive accuracy compared to other algorithms. Gradient Boosting is robust to outliers in the output space (via robust loss functions) and provides good performance even if the data has not been carefully pre-processed. Like Random Forest, Gradient Boosting provides a framework for feature importance which can be invaluable in understanding which variables are driving predictions.

As per the below snapshot **(Fig 2.6)**, we have an accuracy score of 82% which is similar to random forest model. We have a precision score of 73% which is similar to KNN model and random forest model. This model has behaved more or equally to random forest.

```
Accuracy GradientBoosting =  0.82
Precision Score = 0.73
Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.88      0.87      2063
           1       0.73      0.70      0.71       937

    accuracy                           0.82      3000
   macro avg       0.80      0.79      0.79      3000
weighted avg       0.82      0.82      0.82      3000
```

Fig 2.6

Confusion Matrix

The below matrix **(Fig 2.7),** 1815 of them were predicted as accident occurred correctly and 657 of them were predicted as accident not occurred which were predicted correctly. 248 of them were predicted as accident occurred but were not occurred and 280 of them were predicted as accident not occurred but were actually occurred.

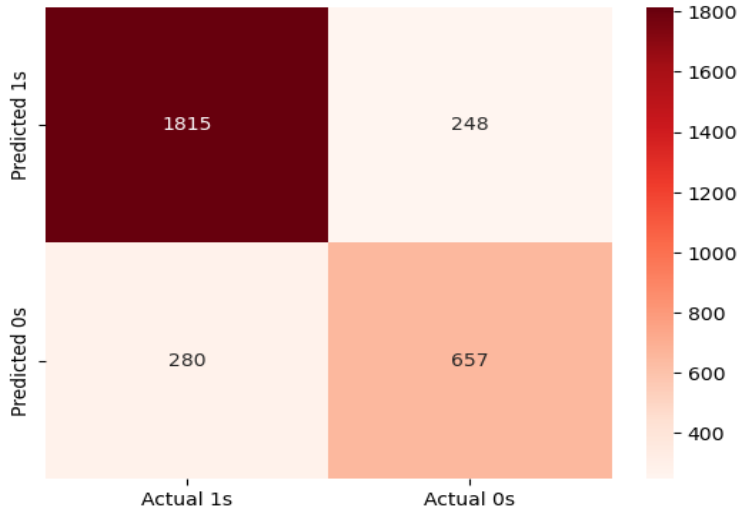

Fig 2.7

ROC curve

In the below graph **(Fig 2.8)**, we received an AUC score of 89% which is equal to the AUC score generated by random forest model. This also shows a good fit model.

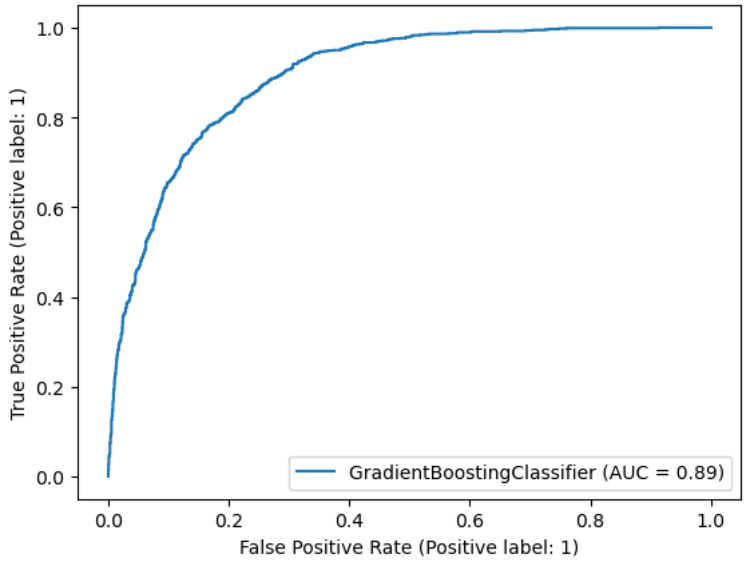

Fig 2.8

Decision Tree Model

Decision trees inherently perform feature selection by choosing the most informative features at the top nodes, which can be particularly useful when dealing with datasets with many features. Decision trees provide an intuitive framework for feature importance, helping to identify which variables are most influential in making predictions.

As per the below snapshot **(Fig 2.9)**, we see an accuracy of 82% which is similar to gradient boost and random forest models but received a precision score of 71% which is lesser than KNN model, random forest and gradient boost model.

```
Decision Tree Model Accuracy: 0.82
Precision Score = 0.71
Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.87      0.87      2063
           1       0.71      0.72      0.72       937

    accuracy                           0.82      3000
   macro avg       0.79      0.79      0.79      3000
weighted avg       0.82      0.82      0.82      3000
```

Fig 2.9

Confusion Matrix

Below matrix shows that, 1793 of them were predicted as accident occurred correctly and 674 of them were predicted as accident not occurred which were predicted correctly. 270 of them were predicted as accident occurred but were not occurred and 263 of them were predicted as accident not occurred but were actually occurred.

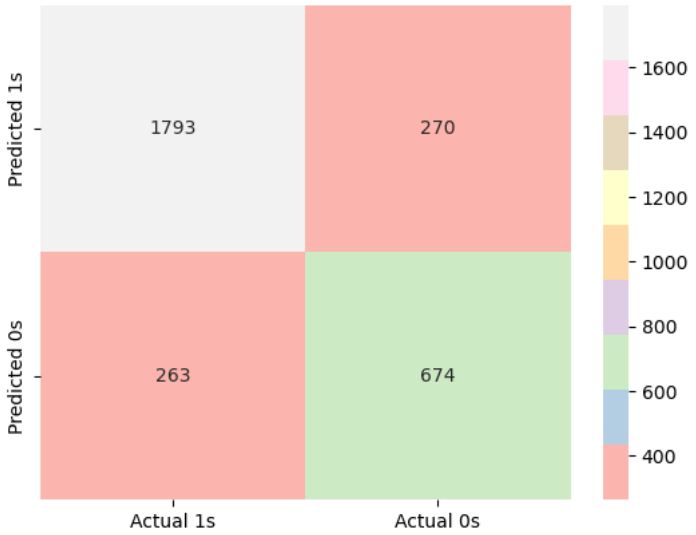

Fig 3.0

ROC curve

The below graph **(Fig 3.1)** shows an AUC score of 88% which is a good score and shows a good fit. But this value is slightly lesser than Random Forest and gradient boosting model.

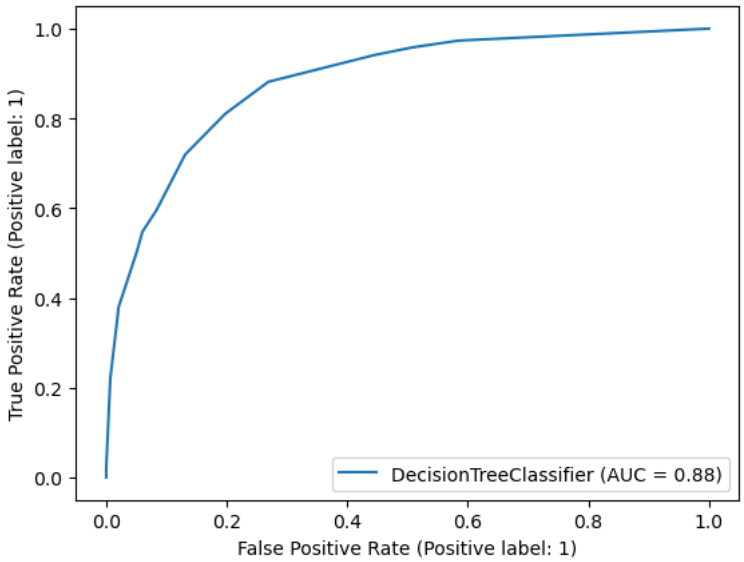

Fig 3.1

Neural Network Model

Neural networks, especially deep ones, have the capacity to model extremely complex, non-linear relationships. They excel in problems where the relationships between variables are intricate and not easily discernible. Neural networks, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), can work with raw data like pixels in an image or audio waveforms, eliminating the need for manual feature engineering in many cases.

Attempt 1 using 'sgd' solver

Here, we received an accuracy score of 69% which is the least score compared to all the models and precision score is 0 which is a indication of not a good model.

```
Neural Network Accuracy score 1 =  0.69
Precision Score = 0.0
Classification Report:
              precision    recall  f1-score   support

           0       0.69      1.00      0.81      2063
           1       0.00      0.00      0.00       937

    accuracy                           0.69      3000
   macro avg       0.34      0.50      0.41      3000
weighted avg       0.47      0.69      0.56      3000
```

Fig 3.2

Confusion Matrix

The below matrix shows that 2063 of them were predicted as accident occurred correctly and 0 of them were predicted as accident not occurred which were predicted correctly. 0 of them were predicted as accident occurred but were not occurred and 937 of them were predicted as accident not occurred but were actually occurred. This matrix reveals that the neural network predicted no accidents (class 1) at all, resulting in all predictions for class 1 being false negatives.

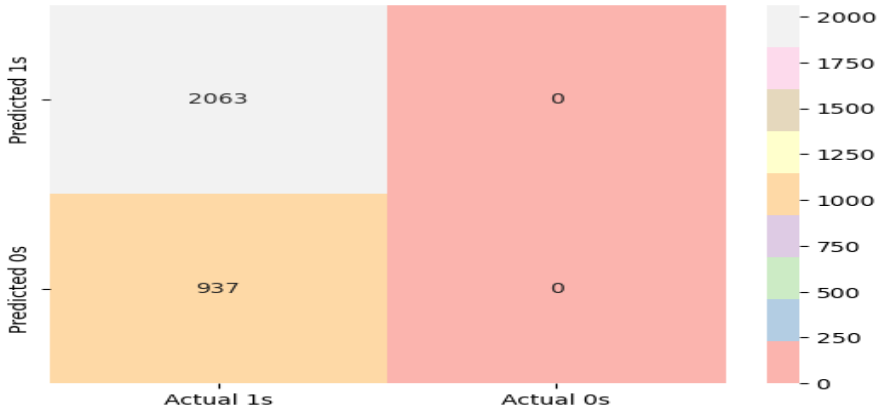

Fig 3.3

ROC curve

The graph shows an AUC score of 50% which is not a good score and shows a bad fit model.

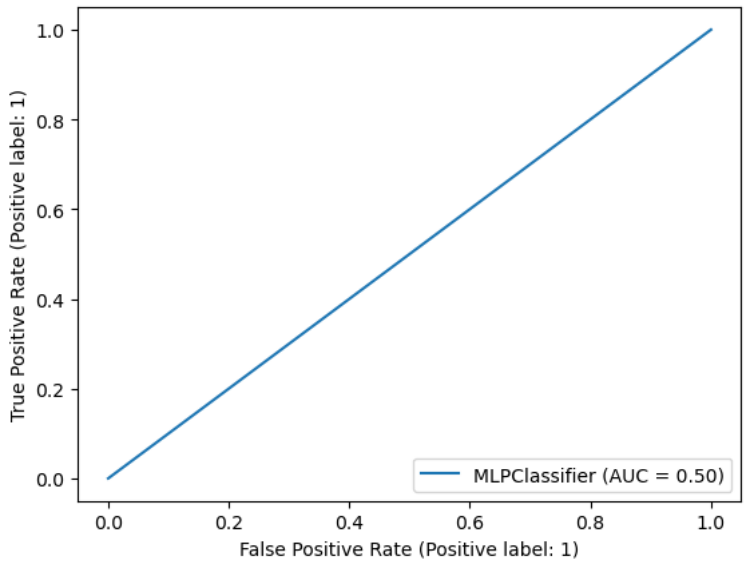

Fig 3.4

Attempt 2 using different activation function

We will be scaling the X_train and X_test data.

The training set accuracy is 85% whereas the test set accuracy is 81%. This is a better model than the SGD solver which had an accuracy of 69%.

```
Training set Accuracy: 0.8488571643829346
Test set Accuracy: 0.8103333115577698
94/94 [==============================] - 0s 504us/step
Classification Report
              precision    recall  f1-score   support

           0       0.82      0.93      0.87      2063
           1       0.78      0.55      0.64       937

    accuracy                           0.81      3000
   macro avg       0.80      0.74      0.76      3000
weighted avg       0.81      0.81      0.80      3000
```

Fig 3.5

Confusion Matrix

Below matrix shows that 1916 of them were predicted as accident occurred correctly and 515 of them were predicted as accident not occurred which were predicted correctly. 147 of them were predicted as accident occurred but were not occurred and 422 of them were predicted as accident not occurred but were actually occurred.

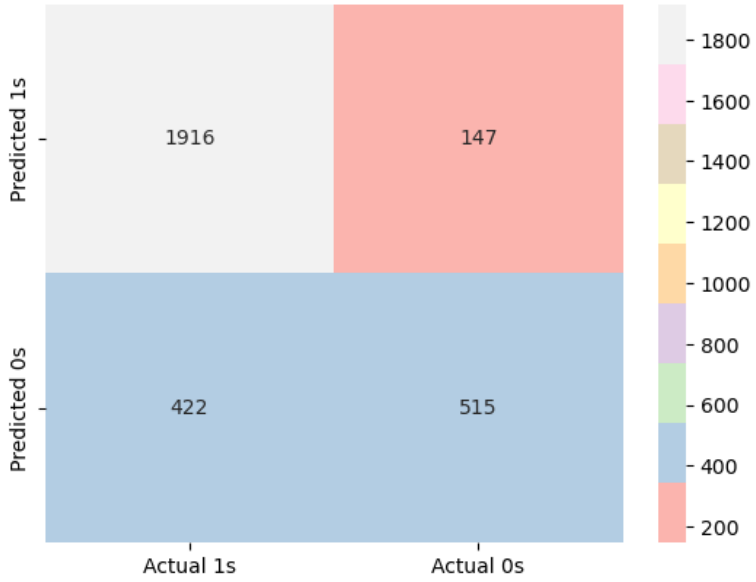

Fig 3.6

ROC curve

The graph shows that the AUC score is 50% which again says that the model is not a good fit.

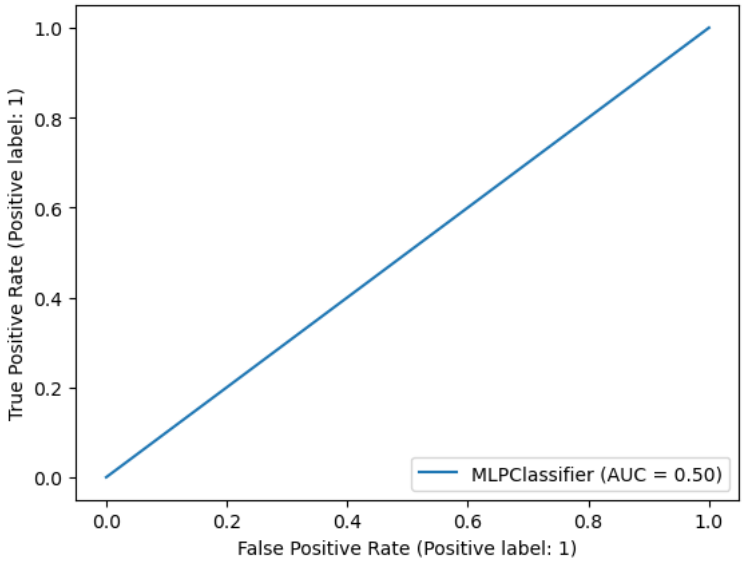

Fig 3.7

Linear Regression Model

Linear regression is an excellent tool for understanding the relationships between variables, making it particularly useful for exploratory data analysis. Linear regression can be extended to model multiple independent variables (multiple linear regression), as well as interactions between variables.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                OUTCOME   R-squared:                       0.374
Model:                            OLS   Adj. R-squared:                  0.372
Method:                 Least Squares   F-statistic:                     277.7
Date:                Fri, 27 Oct 2023   Prob (F-statistic):               0.00
Time:                        20:11:00   Log-Likelihood:                -2920.5
No. Observations:                7000   AIC:                             5873.
Df Residuals:                    6984   BIC:                             5983.
Df Model:                          15
Covariance Type:            nonrobust
==============================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const              0.6185      0.037     16.803      0.000       0.546       0.691
AGE               -0.0237      0.007     -3.404      0.001      -0.037      -0.010
GENDER             0.1089      0.010     11.433      0.000       0.090       0.128
DRIVING_EXPERIENCE -0.1633     0.008    -19.533      0.000      -0.180      -0.147
EDUCATION         -0.0067      0.005     -1.320      0.187      -0.017       0.003
INCOME             0.0095      0.004      2.166      0.030       0.001       0.018
CREDIT_SCORE      -0.1842      0.040     -4.595      0.000      -0.263      -0.106
VEHICLE_OWNERSHIP -0.2629      0.010    -25.845      0.000      -0.283      -0.243
MARRIED           -0.0527      0.010     -5.060      0.000      -0.073      -0.032
CHILDREN          -0.0311      0.011     -2.815      0.005      -0.053      -0.009
POSTAL_CODE     3.978e-06   3.24e-07     12.260      0.000    3.34e-06    4.61e-06
ANNUAL_MILEAGE  8.549e-06   2.12e-06      4.036      0.000     4.4e-06    1.27e-05
VEHICLE_TYPE       0.0127      0.021      0.611      0.541      -0.028       0.053
SPEEDING_VIOLATIONS 0.0009     0.003      0.319      0.750      -0.005       0.006
DUIS              -0.0039      0.009     -0.452      0.651      -0.021       0.013
PAST_ACCIDENTS    -0.0054      0.004     -1.525      0.127      -0.012       0.002
==============================================================================
Omnibus:                      336.915   Durbin-Watson:                   1.997
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              197.952
Skew:                           0.268   Prob(JB):                     1.04e-43
Kurtosis:                       2.375   Cond. No.                     2.67e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.67e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Fig 3.8

Here, we see that the R-squared value is 37% which is lesser than logistic regression model and the least when compared to other model. The accuracy score is 82% which is slightly higher than logistic regression and precision score is 73%.

```
Accuracy Score = 0.82
Precision Score = 0.73
              precision    recall  f1-score   support

           0       0.85      0.89      0.87      2063
           1       0.73      0.64      0.68       937

    accuracy                           0.81      3000
   macro avg       0.79      0.77      0.78      3000
weighted avg       0.81      0.81      0.81      3000
```

Fig 3.9

Confusion Matrix

Below matrix shows that 1844 of them were predicted as accident occurred correctly and 601 of them were predicted as accident not occurred which were predicted correctly. 219 of them were predicted as accident occurred but were not occurred and 336 of them were predicted as accident not occurred but were actually occurred.

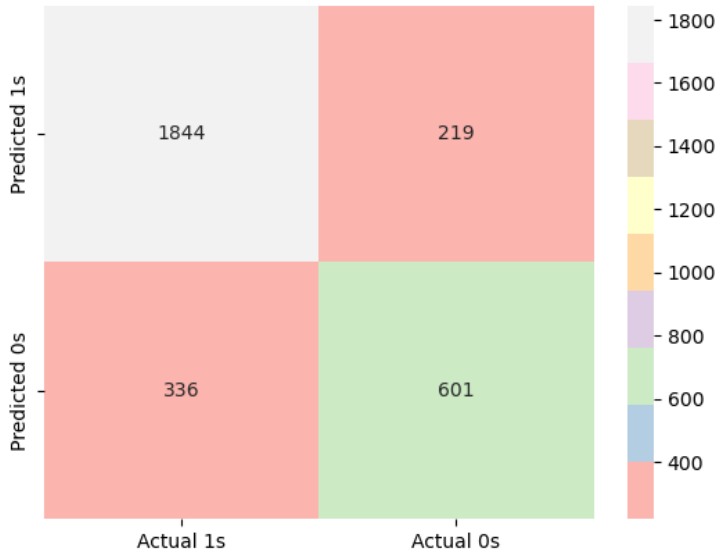

Fig 4.0

<u>Forward Selection/Backward Selection Model</u>

Forward Selection Model

In scenarios where there are many potential predictor variables, evaluating all possible combinations can be computationally prohibitive. Forward selection provides a more tractable approach than exhaustive search methods. The forward selection method is an iterative method in which we start with having no features in the model. In each iteration, we keep adding the feature which best improves our model until the addition of a new variable does not improve the performance of the model.

After running forward selection, I am receiving the below result:

```
Selected Features (Forward Selection): ['AGE', 'GENDER', 'DRIVING_EXPERIENCE', 'EDUCATION', 'INCOME', 'CREDIT_SCORE',
'VEHICLE_OWNERSHIP', 'MARRIED', 'CHILDREN', 'POSTAL_CODE', 'ANNUAL_MILEAGE', 'PAST_ACCIDENTS']
Minimum AIC: 8379.04
```

The AIC score is 8379.04. Since this value is little higher, we can say it is not a good model and we require lower AIC value.

Backward Selection Model

Backward selection starts with a full model that includes all potential predictor variables. This allows for the consideration of the full complexity of the model from the outset. In scenarios where there are many predictors, starting with a full model (as in backward elimination) can sometimes be more feasible than starting with the simplest model (as in forward selection) and trying to add in every possible predictor one by one.

After running backward selection, I am receiving the below result:

```
Selected Features (Backward Selection): ['AGE', 'GENDER', 'DRIVING_EXPERIENCE', 'EDUCATION', 'INCOME', 'CREDIT_SCOR
E', 'VEHICLE_OWNERSHIP', 'MARRIED', 'CHILDREN', 'POSTAL_CODE', 'ANNUAL_MILEAGE', 'PAST_ACCIDENTS']
Minimum AIC: 8379.04
```

Here, we receive an AIC score of 8379.04 which is same as forward selection and can qualify to not be a good fit model.

SVM Model

The core principle of SVM is to find the hyperplane that best divides a dataset into classes. It does so by maximizing the margin between the hyperplane and the nearest data point from either class, which can lead to better generalization on unseen data. SVMs have been found to be effective in domains where the data might be unstructured or semi-structured, such as text and images.

The below snapshot **(Fig 4.1)** shows that the accuracy of the model is 75% which is a moderate value and lesser than logistic and linear regression models. The precision score is the least which is around 57%.

```
SVM Model Accuracy: 0.75
Precision Score = 0.5726937269372694
Classification report:
              precision    recall  f1-score   support

           0       0.90      0.72      0.80      2063
           1       0.57      0.83      0.68       937

    accuracy                           0.75      3000
   macro avg       0.74      0.77      0.74      3000
weighted avg       0.80      0.75      0.76      3000
```

Fig 4.1

Confusion Matrix

The confusion matrix below **(Fig 4.2)**, shows that 1484 of them were predicted as accident occurred correctly and 776 of them were predicted as accident not occurred which were predicted correctly. 579 of them were predicted as accident occurred but were not occurred and 161 of them were predicted as accident not occurred but were actually occurred.

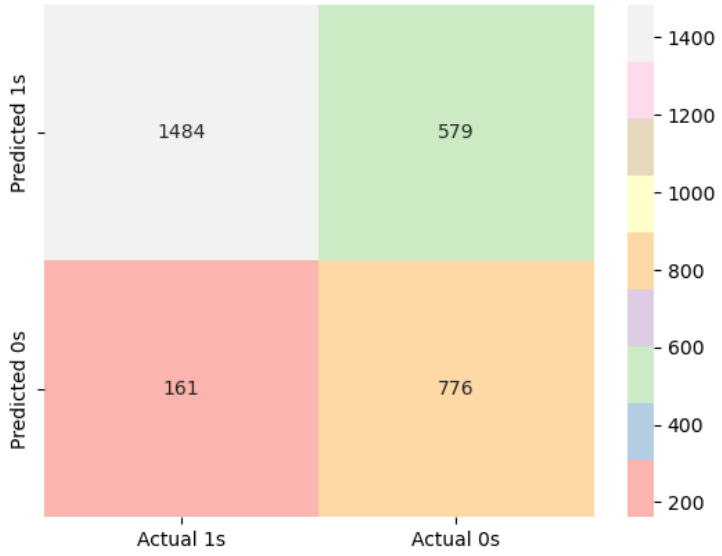

Fig 4.2

Inference:

The confusion matrix has below quadrants:

True Positives (TP): Actual accidents correctly predicted as accidents.
True Negatives (TN): Actual non-accidents correctly predicted as non-accidents.
False Positives (FP): Actual non-accidents incorrectly predicted as accidents.
False Negatives (FN): Actual accidents incorrectly predicted as non-accidents.

The best model, from a car insurance perspective, would be the one that minimizes the “riskiest” mistakes. The definition of "riskiest" would vary based on the business context. For instance, in a car insurance scenario, a False Negative (failing to catch an accident) might be much riskier than a False Positive (flagging a legitimate non-accident as accident). Hence, we need to make sure that both False Negative and False Positive needs to be reduced.
By comparing all the confusion matrix, I can infer the below:

- If we want to minimize False Positives, the Neural Network model with FP=0 stands out, but it comes with a very high FN of 937. This might be too high a trade-off.

- If we want to minimize False Negatives, the Decision Tree model with FN=263 is the lowest, but it also has a relatively high FP of 270.
- If we're looking for a balance between the two, the Gradient Boost model seems to strike a relatively good balance with FP=248 and FN=280.

As per the model accuracy, Random Forest Model, Gradient Boost Model, Decision Tree Model and Linear Regression Model shows the highest accuracy of 82%.

5. In two paragraphs minimum, discuss the features that were important and significant from the models. Use that to provide a specific recommendation to the insurance company of what they need to look out for when someone applies for car insurance with them and how they can reduce that risk when someone uses their insurance.

Significant Features Discussion:

The **Random Forest** and **Decision Tree** models, along with the **Linear Regression** model and **Gradient Boost** Model, all pointed towards a set of features that significantly influenced the likelihood of a person getting into an accident. Key features include **AGE**, indicating that age plays a pivotal role in risk assessment, with younger or older drivers potentially posing different levels of risk. The **GENDER** feature also stands out, suggesting that one gender might be more accident-prone than the other. **DRIVING_EXPERIENCE** is another crucial feature, emphasizing the intuitive understanding that the number of years someone has been driving can impact their risk of accidents. Other features such as **VEHICLE_OWNERSHIP**, **MARRIED**, **CHILDREN**, and **ANNUAL_MILEAGE** were also highlighted, suggesting that factors related to personal life and driving habits can significantly affect the likelihood of accidents.

Recommendation to the Insurance Company:

When assessing potential clients, the insurance company should prioritize examining the **AGE, GENDER, and DRIVING_EXPERIENCE** of the applicant. Younger drivers or those with limited driving experience might pose a higher risk and differentiated premium rates or additional training might be considered for such groups. Furthermore, given the significance of the **VEHICLE_OWNERSHIP** feature, the company could offer different plans or rates for those who own their vehicles versus those who don't. Marital status and the presence of children might indicate more responsible driving behavior, so offering family packages or discounts for married individuals could be beneficial. Lastly, monitoring **ANNUAL_MILEAGE** can help in predicting risk; those who drive more are naturally exposed to higher risk. Offering incentives for lower annual mileage or implementing

tracking systems to monitor driving habits can both act as deterrents for reckless driving and provide data-driven insights for future risk assessments.

CONCLUSION

In this project, we have used Logistic regression, SVM model, Decision tree model, Random Forest Classifier model, Gradient Boosting Classifier model, KNN Model, Linear Regression, Forward/Backward Selection and Neural Network Model to check to understand which features of a person or vehicle make them more likely to get in an accident using OUTCOME target variable. After building all the models, we found that Random Forest Model, Gradient Boost Model, Decision Tree Model and Linear Regression Model is giving the highest accuracy of 82% compared to all other models. Neural network using 'SGD' solver is giving the least accuracy of 69%. I have used classification report, confusion matrix to check on the report in detail. Age, gender, driving experience, and aspects of personal life emerged as crucial determinants, painting a holistic picture of an individual's risk profile. The models not only emphasized the quantitative importance of these features but also allowed for a qualitative understanding of human behavior and its interplay with risk. For an insurance company, these findings are invaluable. They pave the way for data-driven decision-making, enabling the company to design more accurate and tailored insurance packages, and ultimately, ensuring the safety and satisfaction of their clientele. In an ever-evolving landscape, staying abreast with such insights allows insurance companies to remain competitive, relevant, and most importantly, effective in their primary role of risk mitigation.

REFERENCES

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer.

MacKay, D. J. (1997). *Ensemble learning for hidden Markov models* (pp. 1-10). Technical report, Cavendish Laboratory, University of Cambridge.

Menard, S. (2002). Applied logistic regression analysis (No. 106). Sage.

Novaković, J. D., Veljović, A., Ilić, S. S., Papić, Ž., & Tomović, M. (2017). Evaluation of classification models in machine learning. *Theory and Applications of Mathematics & Computer Science*, *7*(1), 39.

Wan, E. A. (1990). Neural network classification: A Bayesian interpretation. *IEEE Transactions on Neural Networks, 1*(4), 303-305.

APPENDIX

The python jupyter notebook used are given below:

ALY6020_SudhamshuVidyananda_Week4_Fall_2023