**Detecting Credit Card Fraud**

NAME: PULLETIKURTHI SAI SREEKAR.

**Aim of Project**

The aim of this R project is to build a classifier that can detect credit card fraudulent transactions. I used a variety of machine learning algorithms like Decision Trees, Logistic Regression, Artificial Neural Networks and finally, Gradient Boosting Classifier that will be able to discern fraudulent from non-fraudulent ones.

**Dataset used**

For carrying out the credit card fraud detection, I used the [Card Transactions dataset](https://drive.google.com/file/d/1CTAlmlREFRaEN3NoHHitewpqAtWS5cVQ/view) that contains a mix of fraud as well as non-fraudulent transactions.

**Data Exploration**

First I imported the datasets that contain transactions made by credit cards. I then explored the data that is contained in the creditcard\_data dataframe. After displaying the creditcard\_data using the head() function as well as the tail() function, I proceeded to explore the other components of this dataframe.

**Data Manipulation**

In this section of the project, I scaled the data using the scale() function. I applied this to the amount component of our creditcard\_data amount. With the help of scaling, the data is structured according to a specified range. Therefore, there are no extreme values in the dataset that might interfere with the functioning of the model.

**Data Modelling**

After standardizing the entire dataset, I split the dataset into training set as well as test set with a split ratio of 0.80. This means that 80% of the data will be attributed to the train\_data whereas 20% will be attributed to the test\_data. I then found the dimensions using the dim() function.

**Fitting Logistic Regression Model**

In this section of the project, I fit the first model. I began with logistic regression. I used it for modelling the outcome probability of fraud/not fraud. I proceeded to implement this model on the test data. Once I summarised the model, I visualized it through plots. In order to assess the performance of the model, I portrayed the Receiver Optimistic Characteristics or ROC curve. For this, I first imported the ROC package and then plotted the ROC curve to analyse its performance.

**Fitting a Decision Tree Model**

Next, I implemented a decision tree algorithm to plot the outcomes of a decision through which I could conclude as to what class the object belongs to. I then implemented the decision tree model and plotted it using the rpart.plot() function. I specifically used the recursive parting to plot the decision tree.

**Artificial Neural Network**

Artificial Neural Networks are a type of machine learning algorithm that are modeled after the human nervous system. The ANN models are able to learn the patterns using the historical data and are able to perform classification on the input data. I imported the neuralnet package that allowed me to implement the ANNs. Then I proceeded to plot it using the plot() function. Now, in the case of Artificial Neural Networks, there is a range of values that is between 1 and 0. I set a threshold of 0.5, that is, values above 0.5 will correspond to 1 and the rest will be 0.

**Gradient Boosting (GBM)**

Gradient Boosting is a popular machine learning algorithm that is used to perform classification and regression tasks. This model comprises of several underlying ensemble models like weak decision trees. These decision trees combine together to form a strong model of gradient boosting. I implemented gradient descent algorithm in the model.

**AUC-ROC Curve**

In the last section of the project, I calculated and plotted an ROC curve measuring the sensitivity and specificity of the model. The print command plots the curve and calculates the area under the curve. The area of a ROC curve can be a test of the sensitivity and accuracy of a model.

**Conclusion**

Concluding plotted the respective performance curves for the models. I also learnt how data can be analysed and visualized to discern fraudulent transactions from other types of data.