Detecting Credit Card Fraud Under a Statistics and Machine Learning Lens

By Vivek Gopalan

1. Introduction

Credit card fraud is a real-world problem that affects thousands of individuals throughout the globe. The impact of fraudulent charges on the lives of everyday people can be catastrophic and detrimental, especially to the finances of the cardholder/customer. In addition to causing financial distress to the cardholder/customer, fraudulent charges can adversely impact a customer's credit score which diminishes their chances of buying a house, a car, or potentially finding a job. Furthermore, the leak of personal information that arises when credit card fraud occurs can increase the likelihood of identity theft. This a very unfortunate scenario that can destroy the livelihood of cardholders and can be extremely difficult to recover from. Besides the cardholder being adversely impacted, fraudulent charges also financially impact the bottom-line of credit card companies as they usually absorb the loss incurred by the customer and reimburse the customers who were fraudulently charged to promote brand loyalty. Credit card companies therefore proactively use data analytics to build predictive models that can help them identify fraudulent charges in real-time and prevent financial harm from occurring to the respective cardholders and to safeguard their profitability. The development of accurate predictive models can help credit card companies identify key factors in detecting and preventing combat theft and fraud.

2. Data Set Information

I downloaded real time credit card data of European cardholder from September of 2013 from Kaggle to gain a better understanding of how credit card companies are using machine learning and data analytics to detect fraudulent charges. Based on this sample credit card data, we can gain insights into how credit card companies collect information on charges and then determine what type of predictive modeling techniques can be implemented with the data at hand. After analyzing the data set, we observe that information on the amount charged, time, and location is given for all the transactions. To protect the confidentiality of the customer credit data, the location data (variables V1-V28) has been transformed using a PCA transformation. As a result of the PCA transformation on the location data, all the variables in the data set contain numerical values. In addition to these parameters, there is an additional feature variable called "Class," which identifies which credit card charges are fraudulent and which charges are not fraudulent. The Class variable is binary with a value of 1 for every fraudulent credit card charge and a value of 0 for every non-fraudulent credit charge. When analyzing the distribution of fraudulent and non-fraudulent charges in our data set, we notice that there are 492 credit card charges classified as fraudulent and 284,315 credit card charges classified as non-fraudulent. The percentage of the credit card data set that consists of fraudulent charges would be equal to $\frac{492}{284{,}315+492} * 100 = 0.172\%$ Hence, we observe that the data set is highly imbalanced as there is an extremely unequal number of fraudulent and non-fraudulent charges.

A representation of all the variables in the credit card data set:

Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23
0	-1.3598071	-0.07278117	2.5363467	1.3781552	-0.33832077	0.46238778	0.23959855	0.09869790	0.3637870	...	-0.018306778	0.277837576	-0.11047391
0	1.1918571	0.26615071	0.1664801	0.4481541	0.06001765	-0.08236081	-0.07880298	0.08510165	-0.2554251	...	-0.225775248	-0.638671953	0.10128802
1	-1.3583541	-1.34016307	1.7732093	0.3797796	-0.50319813	1.80049938	0.79146096	0.24767579	-1.5146543	...	0.247998153	0.771679402	0.90941226
1	-0.9662717	-0.18522601	1.7929933	-0.8632913	-0.01030888	1.24720317	0.23760894	0.37743587	-1.3870241	...	-0.108300452	0.005273597	-0.19032052
2	-1.1582331	0.87773675	1.5487178	0.4030339	-0.40719338	0.09592146	0.59294075	-0.27053268	0.8177393	...	-0.009430697	0.798278495	-0.13745808
2	-0.4259659	0.96052304	1.1411093	-0.1682521	0.42098688	-0.02972755	0.47620095	0.26031433	-0.5686714	...	-0.208253515	-0.559824796	-0.02639767

V24	V25	V26	V27	V28	Amount	Class
0.06692807	0.1285394	-0.1891148	0.133558377	-0.02105305	149.62	0
-0.33984648	0.1671704	0.1258945	-0.008983099	0.01472417	2.69	0
-0.68928096	-0.3276418	-0.1390966	-0.055352794	-0.05975184	378.66	0
-1.17557533	0.6473760	-0.2219288	0.062722849	0.06145763	123.50	0
0.14126698	-0.2060096	0.5022922	0.219422230	0.21515315	69.99	0
-0.37142658	-0.2327938	0.1059148	0.253844225	0.08108026	3.67	0

3. Data set analysis and Pre-Processing Methods

i. To get a better sense of the data, it is imperative to conduct a statistical analysis as well as split the raw data into training and test sets. Statistical analysis was conducted to identify variables which can affect the accuracy of any predictive model that is constructed; this is especially important if there too many unnecessary variables. With the location (variables V1-V28), amount, time, and class, there a total of 31 variables in this data set. The response variable(Y) is the Class variable, and the objective is to build accurate models that can predict and classify fraudulent credit card charges. A graph of the data was generated (Figure 1 shown below) to determine if the amount of money charged looked different between fraudulent and non-fraudulent charges. According to this graph, more fraudulent charges occur than non-fraudulent charges when the transaction amount is around 100, and between 300 and 400.

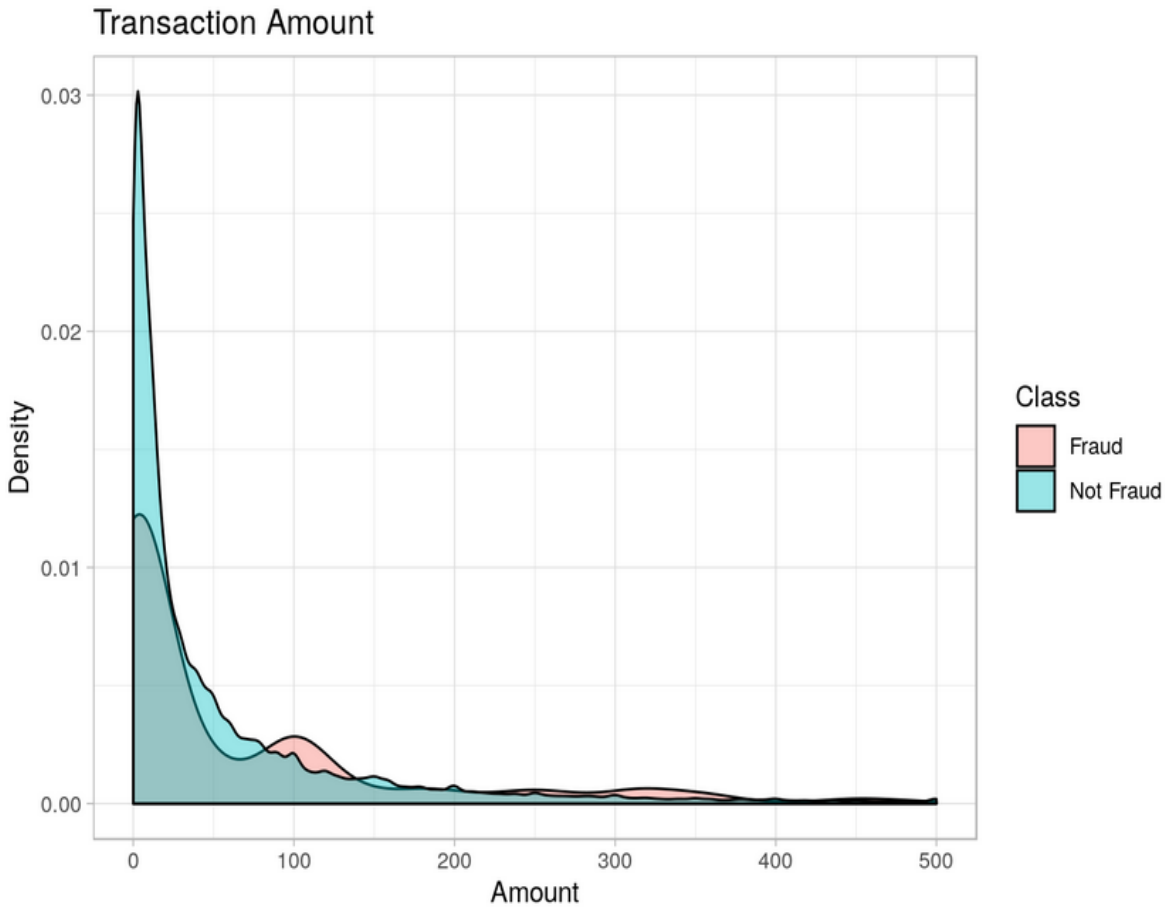


Figure 1: Graph to determine any difference in "Amount charged" between Fraudulent and Non-Fraudulent charges

ii. Data Set Challenges:

A. Extreme class imbalance: The extreme class imbalance in this data set is a major issue that prevents the construction of an accurate and predictive classification models, especially when there are only two classes. When the number of fraudulent charges makes up an extremely small fraction of the overall data set, we can generate new training sets by resampling the data in our training sets in order to ensure that the number of fraudulent and non-fraudulent charges are roughly equal when building predictive models. Some potential ways to resample the data set include over-sampling and under-sampling. However, these methods of resampling the data come with potential drawbacks.

With oversampling, you replicate observations excessively from the minority class to the point where the generated data is not random. Hence, when constructing predictive models, the lack of randomness in the (over-sampled) training data generated would affect the accuracy of any machine learning model especially when the model performance is evaluated on the test set. However, with under-sampling,

you lose out on valuable pieces of data when cutting out entries from the majority class to match the number of entries in the minority class. Because of the extreme class imbalance between fraudulent and non-fraudulent charges, we would lose almost all the entries in the majority class when under-sampling. Hence, when resampling the data set, it is imperative to choose a sampling method that generates artificial data that is random and does not lose any valuable pieces of data during the sampling process. When implementing ROSE sampling, it uses a statistical method known as bootstrapping that generates artificial data based on the existing data by using random sampling techniques. With ROSE sampling, the number of fraudulent and non-fraudulent credit card charges are roughly equal, which means that the data set is not imbalanced, and we did not lose or unnecessarily repeat any observations in the original data through the process of resampling. Please see figure below for more details.

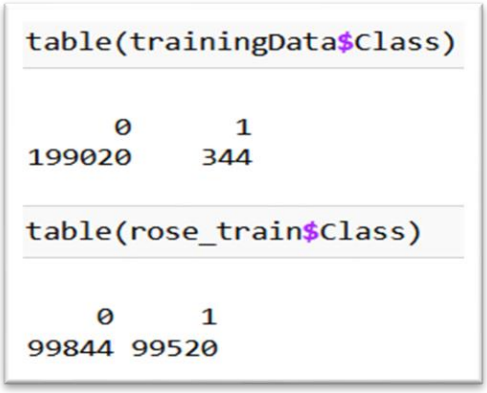


Figure 2: Use of ROSE Sampling to balance training set data ahead of classification model development

B. Multicollinearity: In addition, when there are multiple explanatory variables (in this case there are 30) and one response variable (the Class variable), the issue of multicollinearity can arise, and we should check whether the explanatory variables are highly correlated to one another. If this is the case, then any regression model constructed to classify fraudulent charges would not produce accurate or significant results and would not be useful. However, the Principal Components Analysis (PCA) transformation on the location data, has addressed any sort of multicollinearity among the explanatory variables V1-V28 in the data set. A correlation matrix is shown below (Figure 3) to assess the Pearson correlation (R-value) between all the explanatory variables in the credit card data set and we observe that the correlation between all the explanatory variables is approximately zero. This means that if a regression is performed on the credit card data set, fairly accurate results can be produced since there is no multicollinearity present.

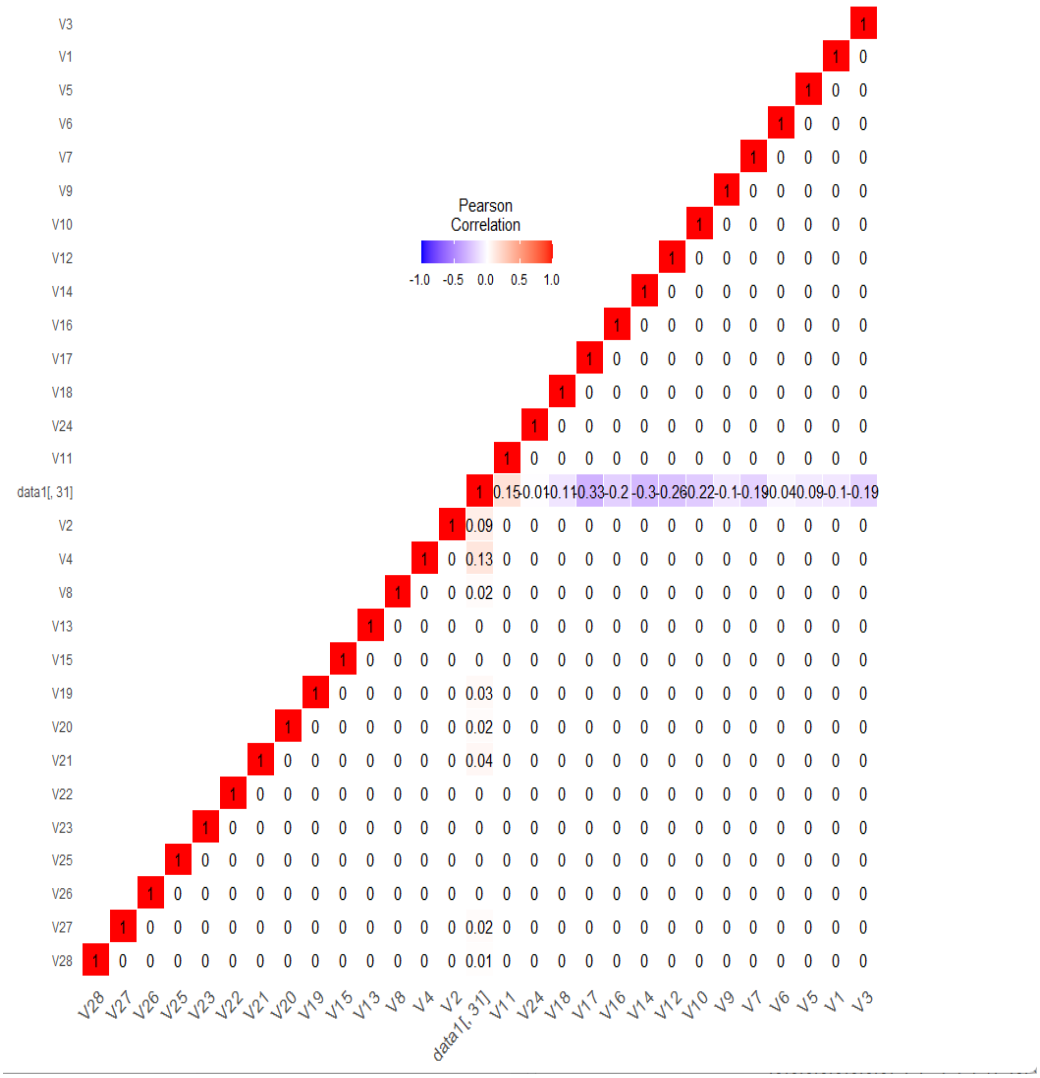


Figure 3: Correlation matrix with Pearson correlation (R-value) between all explanatory variables

4. Predictive Modeling Techniques to Identify Fraudulent Charges

The following five machine learning approaches were used to analyze the data and build predictive classification models.

i. Logistic Regression
ii. Decision Trees(CART)
iii. Gradient Boost(GBM)
iv. Support Vector Machine (SVM)
v. Artificial Neural Networks(ANN)

Typically, Confusion Matrix are not useful when dealing with imbalanced data, and *Area Under the Precision Recall Curves*[3] are used to compare the error rates for the various machine learning models. However, as ROSE sampling was used to balance the training set data, the typical ROC AUC values and Confusion Matrix statistics can now be computed and used to compare their performance relative to each other.

i) Logistic Regression on Training Set

Our response variable Class is binary and takes on values of 0 and 1 and takes on a value of 1 for every fraudulent charge and a value of 0 for every non-fraudulent charge. Since there is no multicollinearity present among our predictor variables, a logistic regression model on our generated training set can help us determine which variables are significant in predicting fraudulent charges. Based on the results of the logistic regression on the training set, we can do a model selection and discard predictor variables that are not statistically significant and see how the revised logistic regression model fares on the test set. Based on the results of the Logistic Regression and the diagnostic plots generated to verify the accuracy of this predictive model, we can come to the necessary conclusions about how this model stacks up compared to other predictive models that are built to classify fraudulent charges.

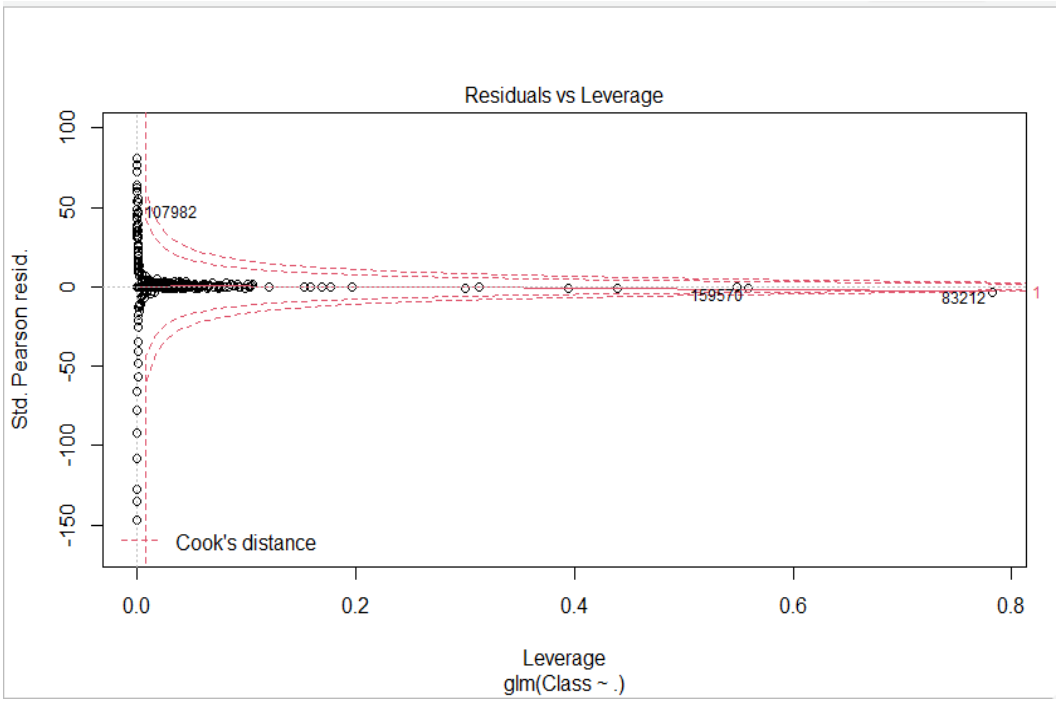
Results of Logistic Regression Model on Training Set Data:

```
Call:
glm(formula = Class ~ ., family = "binomial", data = training_set)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.4686  -0.0296  -0.0194  -0.0124   4.1907

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -8.645945   0.167777 -51.532  < 2e-16 ***
V1           0.075466   0.045423   1.661 0.096633 .
V2           0.005186   0.060217   0.086 0.931370
V3          -0.004433   0.051196  -0.087 0.931005
V4           0.664921   0.078337   8.488  < 2e-16 ***
V5           0.094736   0.071984   1.316 0.188146
V6          -0.141237   0.086739  -1.628 0.103461
V7          -0.134051   0.069222  -1.937 0.052801 .
V8          -0.143667   0.035313  -4.068 4.73e-05 ***
V9          -0.378153   0.119421  -3.167 0.001543 **
V10         -0.745186   0.098590  -7.558 4.08e-14 ***
V11         -0.047503   0.089384  -0.531 0.595105
V12          0.121163   0.099717   1.215 0.224341
V13         -0.351685   0.097590  -3.604 0.000314 ***
V14         -0.557308   0.070375  -7.919 2.39e-15 ***
V15         -0.115221   0.098129  -1.174 0.240325
V16         -0.139091   0.140434  -0.990 0.321960
V17          0.030152   0.077014   0.392 0.695418
V18         -0.078980   0.143912  -0.549 0.583136
V19          0.076363   0.107842   0.708 0.478885
V20         -0.472347   0.082108  -5.753 8.78e-09 ***
```

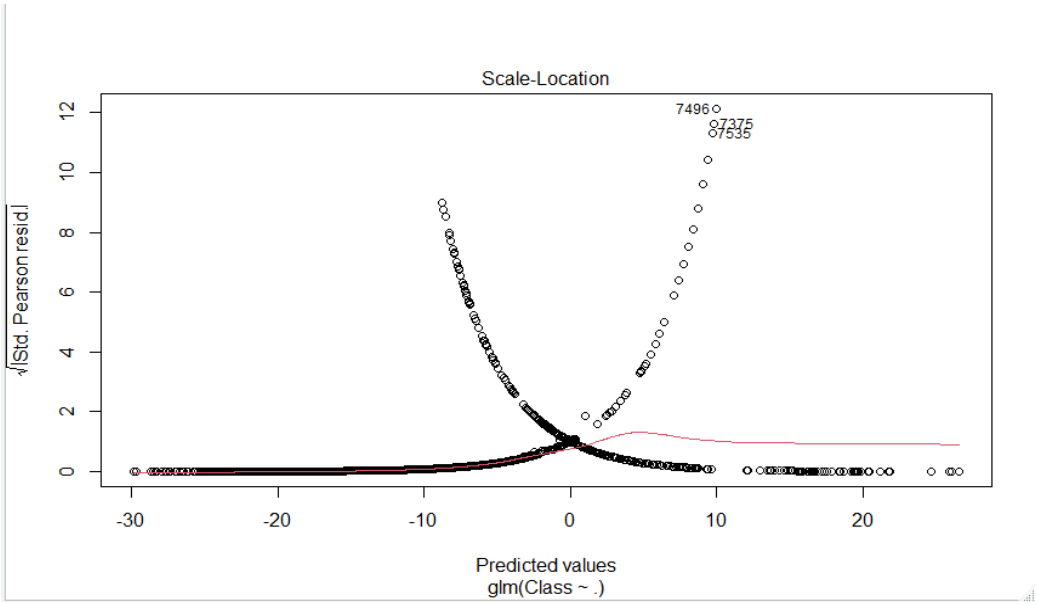
```
V21          0.414977   0.067124   6.182 6.32e-10 ***
V22          0.726419   0.149840   4.848 1.25e-06 ***
V23         -0.081491   0.061044  -1.335 0.181889
V24          0.249532   0.176166   1.416 0.156641
V25         -0.104263   0.146905  -0.710 0.477873
V26          0.096037   0.213304   0.450 0.652540
V27         -0.849087   0.115931  -7.324 2.41e-13 ***
V28         -0.314820   0.086237  -3.651 0.000262 ***
Amount       0.303059   0.092516   3.276 0.001054 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5064.6  on 199363  degrees of freedom
Residual deviance: 1625.9  on 199334  degrees of freedom
AIC: 1685.9

Number of Fisher Scoring iterations: 11
```

Diagnostic Plots of Logistic Regression Model on Training Set Data:

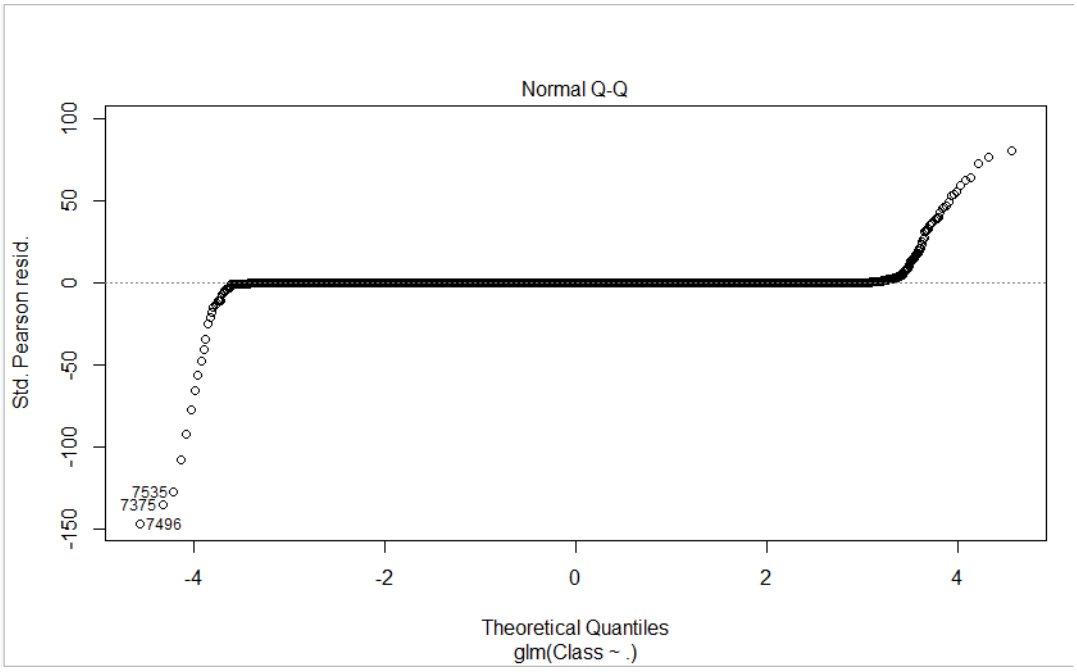
Residual vs. Leverage Plot:



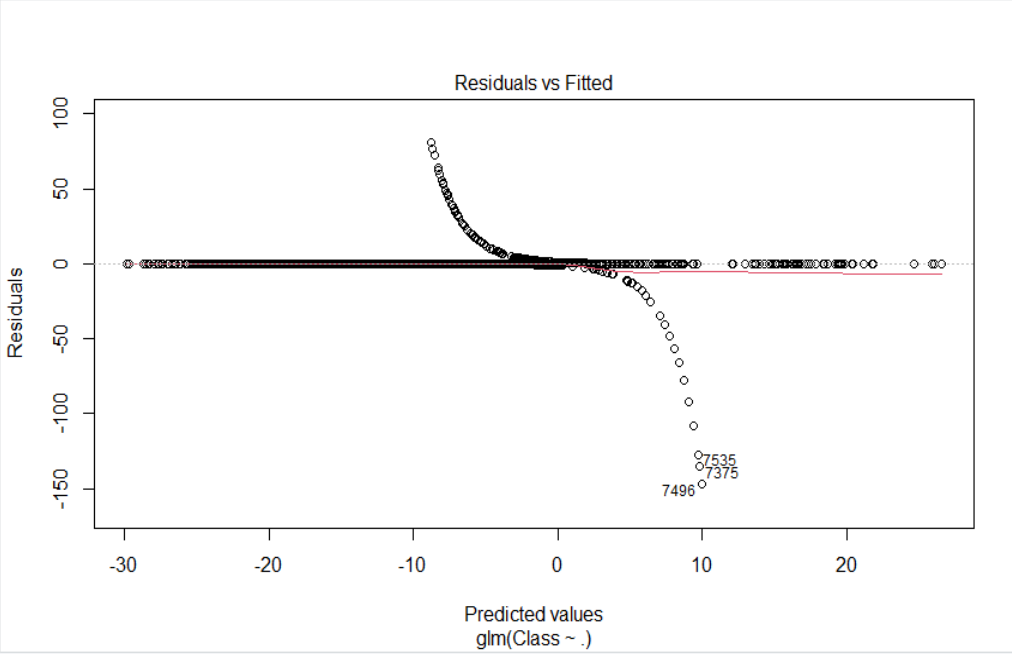
Scale-Location Plot:



QQ Plot:



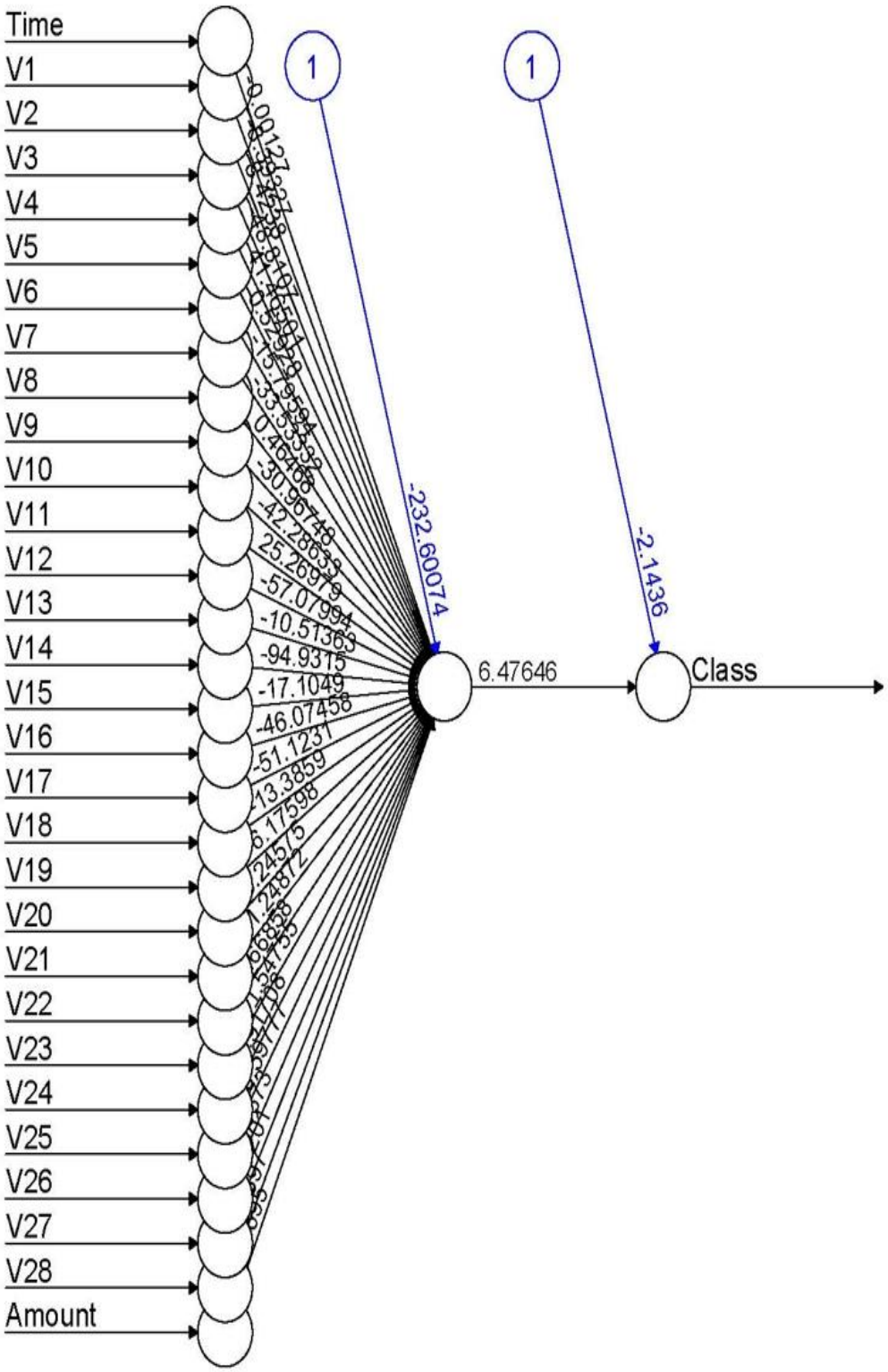
Residuals vs. Fitted Plot:



Summary of Results and Plots for Logistic Regression Model:

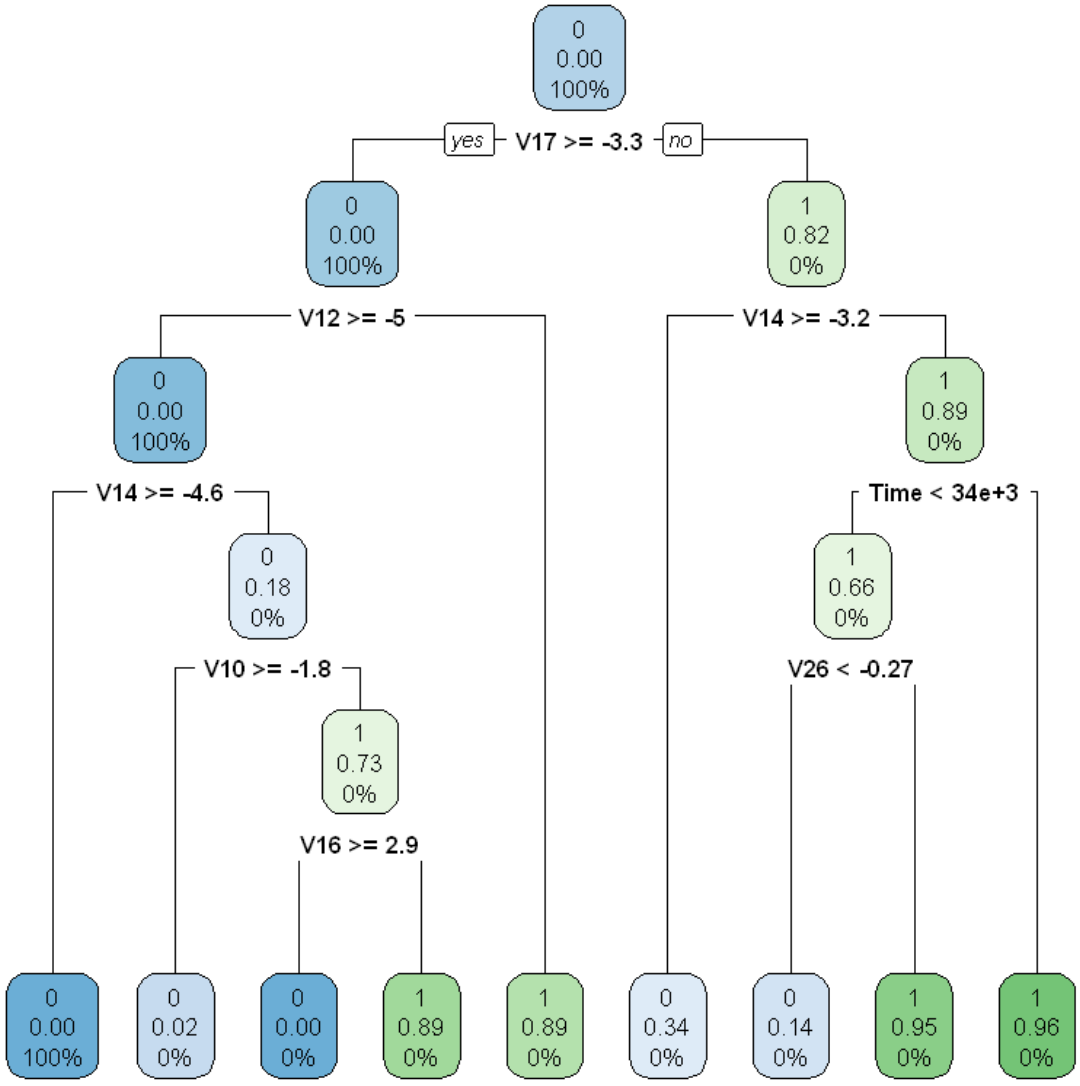
The Residuals vs Leverage shows us the Cook distances for all the residuals generated by the logistic regression model. The Cook distances are calculated to show if there any influential outliers that affect our logistic regression model. The plot only shows that that there are three influential outliers. The Scale – Location , QQ, and Residual vs. Fitted Plot also verify that there are three influential outliers in our logistic regression model. The Residual vs. Fitted plot shows a discernable trend which shows that the distribution of our data is skewed. This is accurate as the response variable Class is highly imbalanced and fraudulent credit card charges only consists of 0.172% of all credit card charges in the data set. The QQ plot generated also verifies the skewness in our data as there is a discernable trend with our residuals that also shows that the distribution of our data is not normal. Furthermore, we observe that only the variables location and amount are statistically significant with predicitng the output of our logisitc regression model. The variable time does not show up as a statistically significant variable. When developing other predictive models to classify fraudulent charges, we will look to see what these various models tell us about our credit card data and whether there are any similarities or differences in the patterns they uncover about the overall data and what this tells us about classifying fraudulent credit card charges.

Artificial Neural Network Model (ANN) model on Training Data:

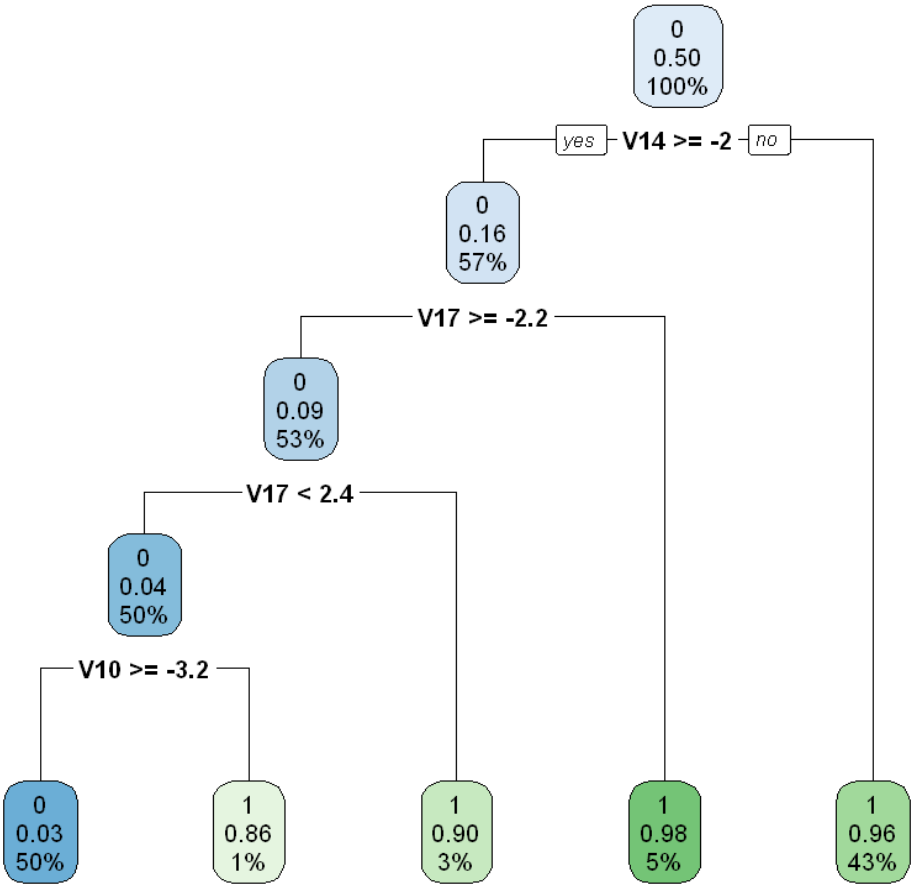


The Artificial Neural Network model provides us weights on our 30 predictor variables, which tells us the significance of each predictor variable in predicting the response variable Class. One interesting thing to note about this model is that the input variable Time is given a weight of approximately 0. This shows that the variable Time is not significant in classifying fraudulent credit card charges. However, the other predictor variables in the input layer receive significant weightages, which further proves that the location of the charges is the most accurate predictor variable in determining fraudulent credit card charges.

Decision Tree Cart (CART) on imbalanced training set data:



Decision Tree Cart (CART) on balanced training set data: ROSE Sampling



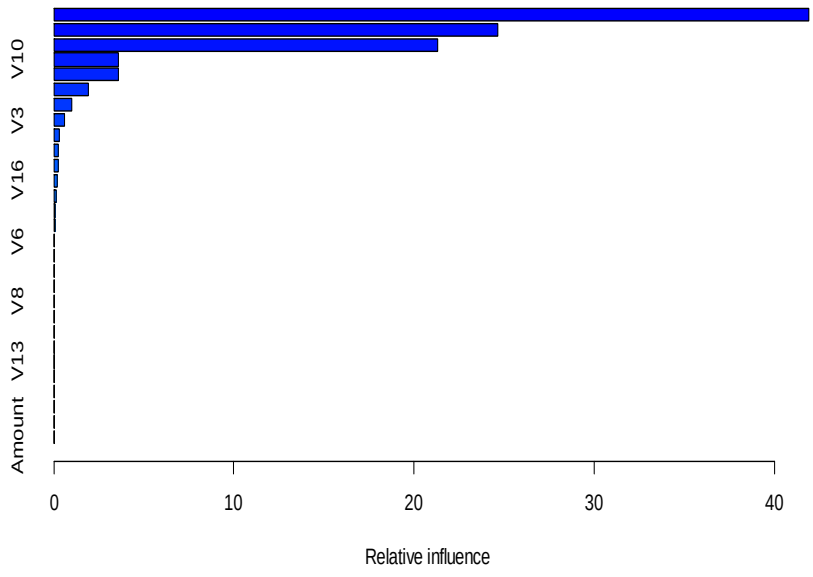
When comparing the decision trees from the ROSE sampled training data and the unbalanced training data, we observe that conducting the decision tree analysis helps us narrow down on the specific variables that help us predict the response variable Class. Another important thing to note about the decision tree analysis on the re-sampled training data versus the decision tree analysis on the un-balanced training data is the variable time is no longer a factor in the CART tree when the classes are balanced with the ROSE sampling. Implementing this predictive model with a balanced dataset further proves that the time of the credit card charges does not play any sort of role in the big picture of classifying fraudulent credit charges.

SVM Model:

```
Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  1
      gamma:  1
    epsilon:  0.1


Number of Support Vectors:  111010
```

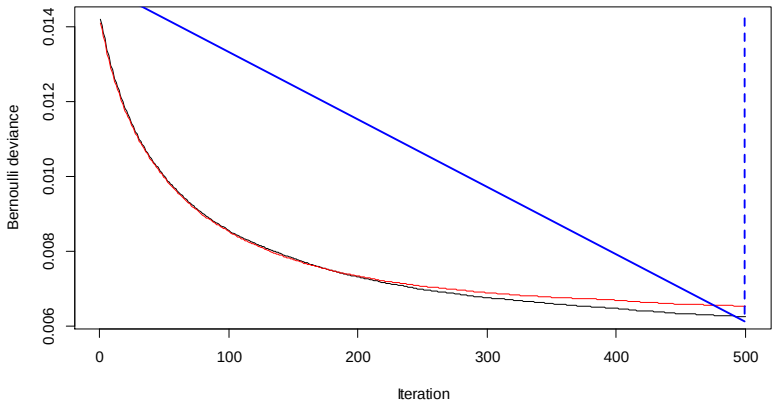
The SVM generated an unusually high number of Support Vectors, which helps in producing more accurate results in classifying fraudulent credit card charges. The SVM model was trained on the re-sampled training data generated by ROSE sampling.

Gradient Boosting Method (GBM):



Relative Influence of Predictor Variables in determining the response variable Class:

	var	rel.inf
V12	V12	49.089272500
V17	V17	20.386298195
V14	V14	13.064792911
V10	V10	5.438455106
V20	V20	4.054524815
V11	V11	2.751366654
V7	V7	1.339229562
V9	V9	1.179859636
V26	V26	0.751362702
V4	V4	0.533214917
V3	V3	0.529538026
V16	V16	0.301351937
Amount	Amount	0.132456419
V28	V28	0.119347406
V1	V1	0.109077389
V8	V8	0.056856690
V6	V6	0.044395507
V18	V18	0.033333272
V13	V13	0.021392419
V15	V15	0.020848797
V25	V25	0.015535467
V24	V24	0.011544070
V21	V21	0.007380694
V22	V22	0.005932401
V27	V27	0.002632510
V2	V2	0.000000000
V5	V5	0.000000000
V19	V19	0.000000000
V23	V23	0.000000000



As shown by the Gradient Boosting Algorithm, we observe that the variable location has the highest relative influence compared to the variables amount charged or the time. However, we see that the amount charged still plays some significance in determining fraudulent credit card charges, especially if the amount charged is unusually high. The gradient boosting algorithm also proves that the variable time does not play a huge role in predicting fraudulent charges and is not listed as a variable that has any sort of relative influence at all in classifying fraudulent credit card charges.

5. **Results (Performance evaluation of different models):**
   Support Vector Machine (SVM) Confusion Matrix and Statistics: ROSE Sampling

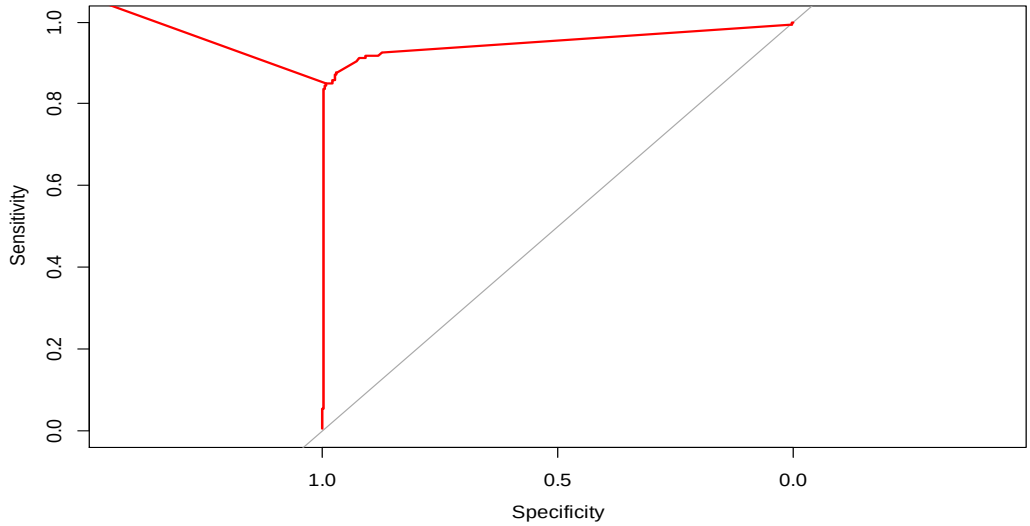
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 85110   185
         1    30   118

               Accuracy : 0.9975
                 95% CI : (0.9971, 0.9978)
    No Information Rate : 0.9965
    P-Value [Acc > NIR] : 5.729e-08

                  Kappa : 0.5222

 Mcnemar's Test P-Value : < 2.2e-16
```

```
            Sensitivity : 0.9996
            Specificity : 0.3894
         Pos Pred Value : 0.9978
         Neg Pred Value : 0.7973
             Prevalence : 0.9965
         Detection Rate : 0.9961
   Detection Prevalence : 0.9983
      Balanced Accuracy : 0.6945

       'Positive' Class : 0
```

Decision Tree Cart (CART) Confusion Matrix and Statistics: ROSE Sampling

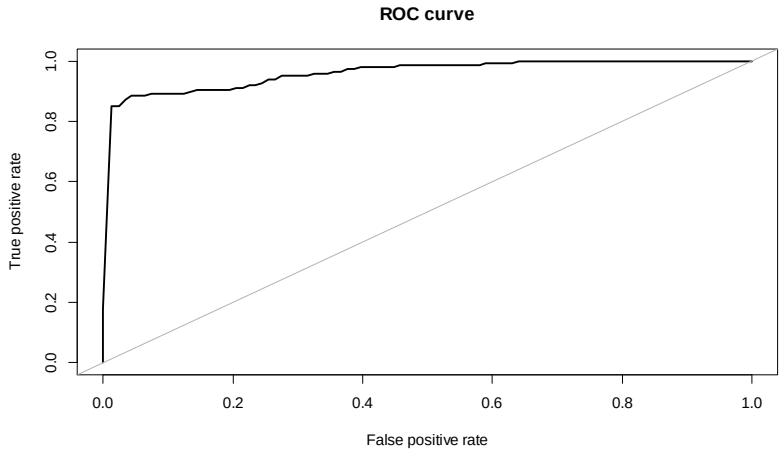
```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 83125  2170
         1    14   134

               Accuracy : 0.9744
                 95% CI : (0.9734, 0.9755)
    No Information Rate : 0.973
    P-Value [Acc > NIR] : 0.005508

                  Kappa : 0.1064

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.99983
            Specificity : 0.05816
         Pos Pred Value : 0.97456
         Neg Pred Value : 0.90541
             Prevalence : 0.97303
         Detection Rate : 0.97287
   Detection Prevalence : 0.99827
      Balanced Accuracy : 0.52900

       'Positive' Class : 0
```

AUC Curve of Gradient Boost Model:




```
Call:
roc.default(response = test_data$Class, predictor = gbm_test,     plot = TRUE, col = "red")

Data: gbm_test in 85295 controls (test_data$Class 0) < 148 cases (test_data$Class 1).
Area under the curve: 0.9513
```

AUC Curve of Logistic Regression Model:




```
> roc.curve(test_data$Class,lr_predict_test, plotit = TRUE)
Area under the curve (AUC): 0.960
>
```

The different performance metrics typically used to evaluate classification models are Recall, Precision, Accuracy, F1 values and AUC curves.

The following terms are defined to compute the model performance metrics.

- **True positive (TP):** A prediction that correctly predicts the class membership
- **True negative (TN):** A prediction that correctly indicates the absence of a class membership
- **False positive (FP):** A prediction which wrongly indicates a class membership
- **False negative (FN):** A prediction which wrongly indicates a class membership

The equations for calculating these metrics are shown in Figure 4 below

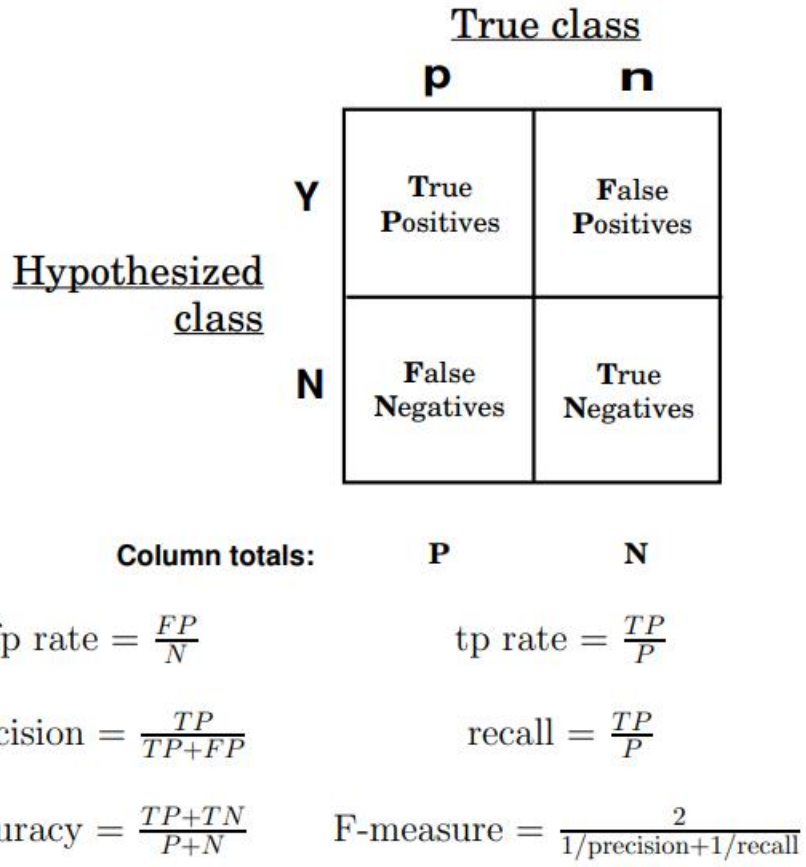


Fig 4: Confusion matrix & common metrics calculated from it (Fawcett T.(2004). *Machine learning, 31(1), 1-38.*)

Therefore:

A. **Accuracy:** It is the ratio of correct predictions to total predictions.
   Accuracy = (TP+TN)/(TP+FP+FN+TN)
B. **Recall (Sensitivity)** is the metric that evaluates a model's ability to predict true positives of each available class.
   - It is a measure of the effectiveness of predictions and indicates the fraction of an actual category that is correctly predicted and 'retrieved.'
     Recall = TP/(TP+FN), or, **TN/(TN+FP)**

C. **Precision (Specificity)** is the metric that evaluates a model's ability to predict correctly within each available class. It measures the fraction of predictions that are correct in each category/class.
   Precision = TP/(TP + FP) or, **TN/(TN+FN)**
D. **ROC curve plots** the True Positive Rate (Y-Axis) vs. the False Positive Rate (1-Specificity) (X-Axis).
   - The Area Under this ROC curve indicates how well the classifier has predicted. The closer the AUC value is to 1, the better the prediction.

The table below shows the performance of the different Machine Learning approaches on this data set.

No.	Machine Learning Model	Accuracy	Recall (Sensitivity)	Precision (Specificity)	AUC (ROC)
1.	Logistic regression	0.961	0.9427	0.3864	0.960
2.	Gradient boost model	0.9513	0.9214	0.4532	0.9513
3.	CART decision tree model	0.9744	0.9993	0.05816	0.915
4.	Support Vector Machine (SVM)	0.9975	0.9996	0.3894	0.982
5.	Artificial Neural Networks(ANN)	0.950	0.9377	0.2143	0.954

Table 1: Comparison of Performance of the Different Machine Learning Approaches

Note: Confusion Matrices can only be constructed for predictive models built on balanced datasets.

6. Findings & Conclusions

i. This project has attempted to show how supervised machine learning classification models can be developed for imbalanced data sets through the use of resampling approaches such as ROSE, which uses bootstrapping to generate random artificial data based on existing data samples.

ii. The use of Principal Components Analysis to generate non-collinear (orthogonal) explanatory variables is a viable approach to data pre-processing that eliminates the need for Z-transformation or autoscaling of data that is typically done for machine learning model development.

iii. The time variable was the least informative and has no bearing on fraudulent charges whatsoever

iv. The location of the credit card charges is the most influential factor in determining fraudulent charges. If the distance between the cardholder and the credit card charge man is beyond a certain distance threshold

v. The use of PCA to transform the location data removes any sort of multicollinearity between all the predictor variables, which makes implementing a logic regression model a viable option when trying to classify/predict fraudulent credit charges

vi. The Gradient Boosting Algorithm shows which variables have the highest relative influence and provides further evidence that the location is a very crucial variable in predicting fraudulent charges.
vii. The logistic regression model helps us in determining which predictor variables are not significant
viii. The Artificial Neural Network (ANN) model helps confirm that time is not a significant variable and receives a weight of approximately zero in the input layer of the model when predicting Class.
ix. The Support Vector Machine(SVM) model had the highest accuracy in classifying fraudulent credit card charges.

7. References:

i. Andrew Estabrooks et. al., (2004) A multiple resampling method for learning from imbalanced datasets, *Computational Intelligence*, Vol.20., No. 1., pp18-36
ii. Menardi Giovanna, Torelli Nicola (2014) Rose: random over-sampling examples. *Data Min Knowl Dis* 28(1):92–122
iii. Fawcett, T. (2004). ROC graphs: Notes and practical considerations for researchers. *Machine learning*, 31(1), 1-38.
iv. Jesse Davis, Mark Goadrich, (2006) The relationship between Precision-Recall and ROC curves, *Proceedings of the 23rd International Conference on Machine Learning*
v. Kaggle (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud)
vi. Elements of Statistical Learning by Trevor Hastie, Robert Tibshirani, and Jerome Friedman (http://statweb.stanford.edu/~tibs/ElemStatLearn/)
vii. R for Data Science by Hadley Wickham & Garrett Grolemund (https://r4ds.had.co.nz/)