



Detecting Credit card Fraud using R

Group 19:

Saswati Mohanty, Vishal Vindyala, Urshilah Senthilnathan, Sujeeth Ganta, Sara Kamal

Introduction to Dataset



The dataset contains transactions made by credit cards in September 2013 by European cardholders.

284,807 Transactions

492 Frauds

Variables

V1- V28
& Amount

Class ("0", "1")

Non-Fraudulent

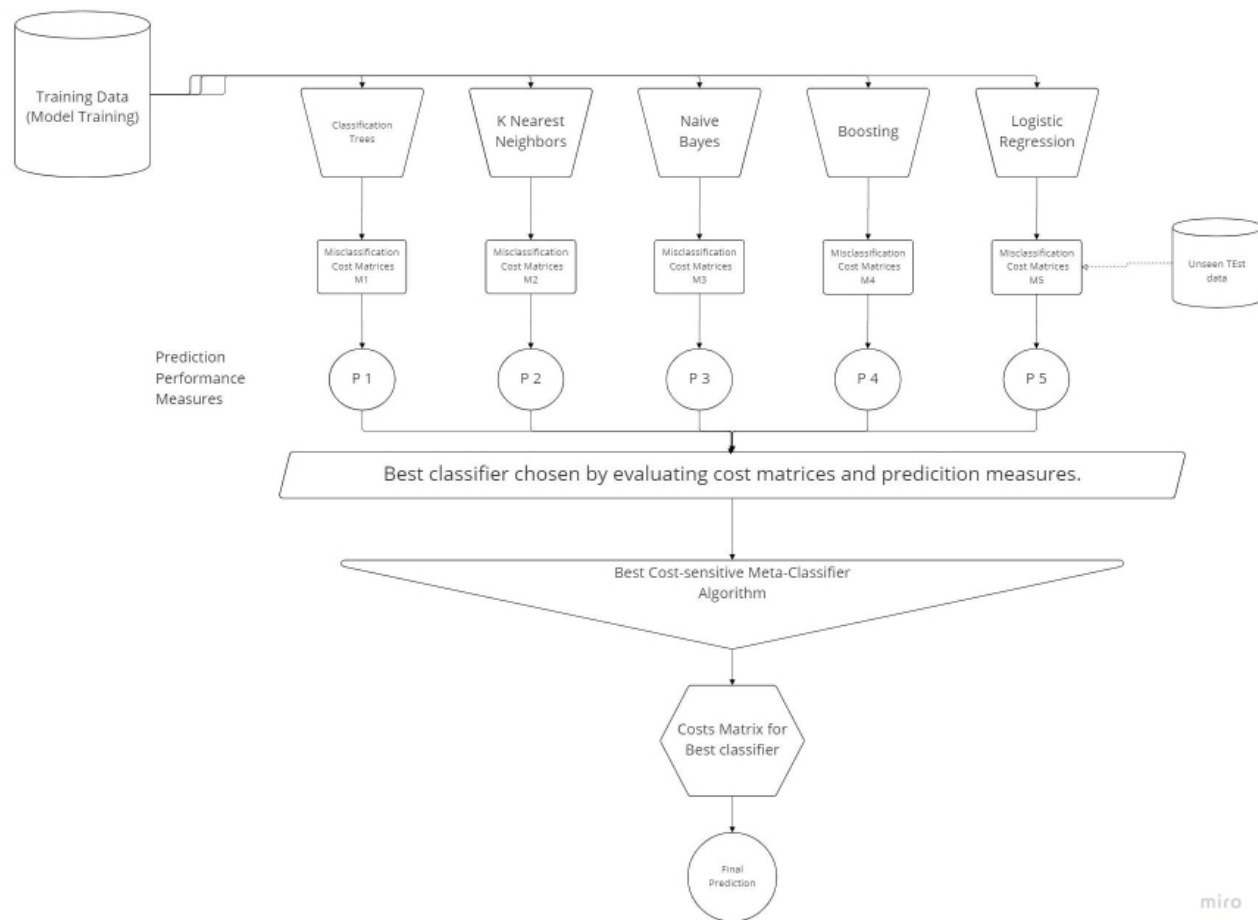
Fraudulent

Goals and Hypotheses



- To identify the best possible Machine Learning model which provides us with the best recall as the recall will be the measure for the rightful identification of the fraudulent class.
- Our approach of the proposed framework is to allow base-classifiers to fit traditionally with the individual cost-sensitive learning and then the undersampling methods is incorporated in the best model selected to fit the cost-sensitive meta-classifier.
- The predictive recall of the trained cost-sensitive meta-classifier and base classifiers were visualized using Tableau.
- We have chosen Supervised machine learning models with Boosting as an ensemble method.

Proposed Model Architecture



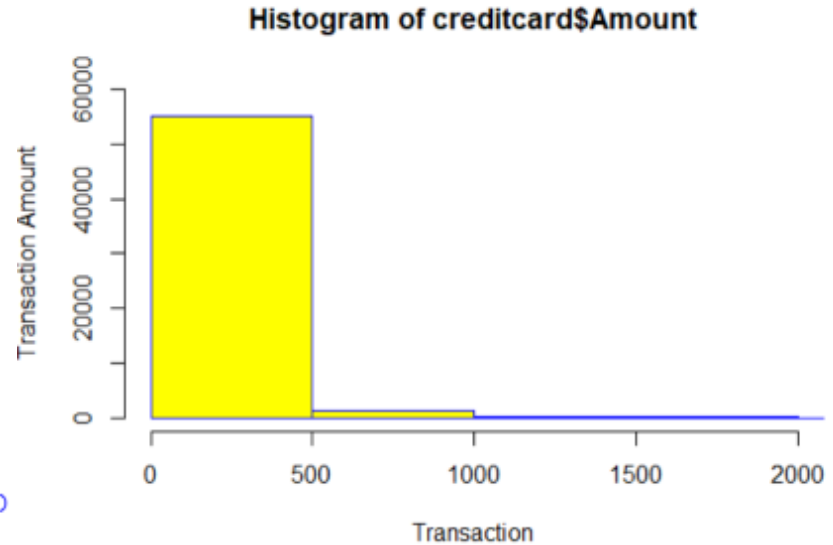
Initial Observations: Exploratory Plots using R

Graph of class distribution:



- 1: fraudulent transactions.
- 0: non fraudulent transactions
- Dataset is highly unbalanced
- Fraudulent transactions only account for 0.172% of all transactions.

Histogram of Amount:



- Transaction amounts are mostly less than \$500 for the largest chunk of transactions.

Calculating Misclassification costs



- Card fraud transactions stored by financial issuers are very small compared to legitimate transactions, which results in a high imbalance credit card dataset.
- The process of determining misclassification costs for fraudulent transactions depends on the nature of the transaction as well as the financial institution.
 - Factors to consider include losses of lenders' earnings due to misclassifying a potential fraudulent transaction as a regular one. The amount of the transaction may also play a role since the loss to the financial institution is equal to the amount of money lost.

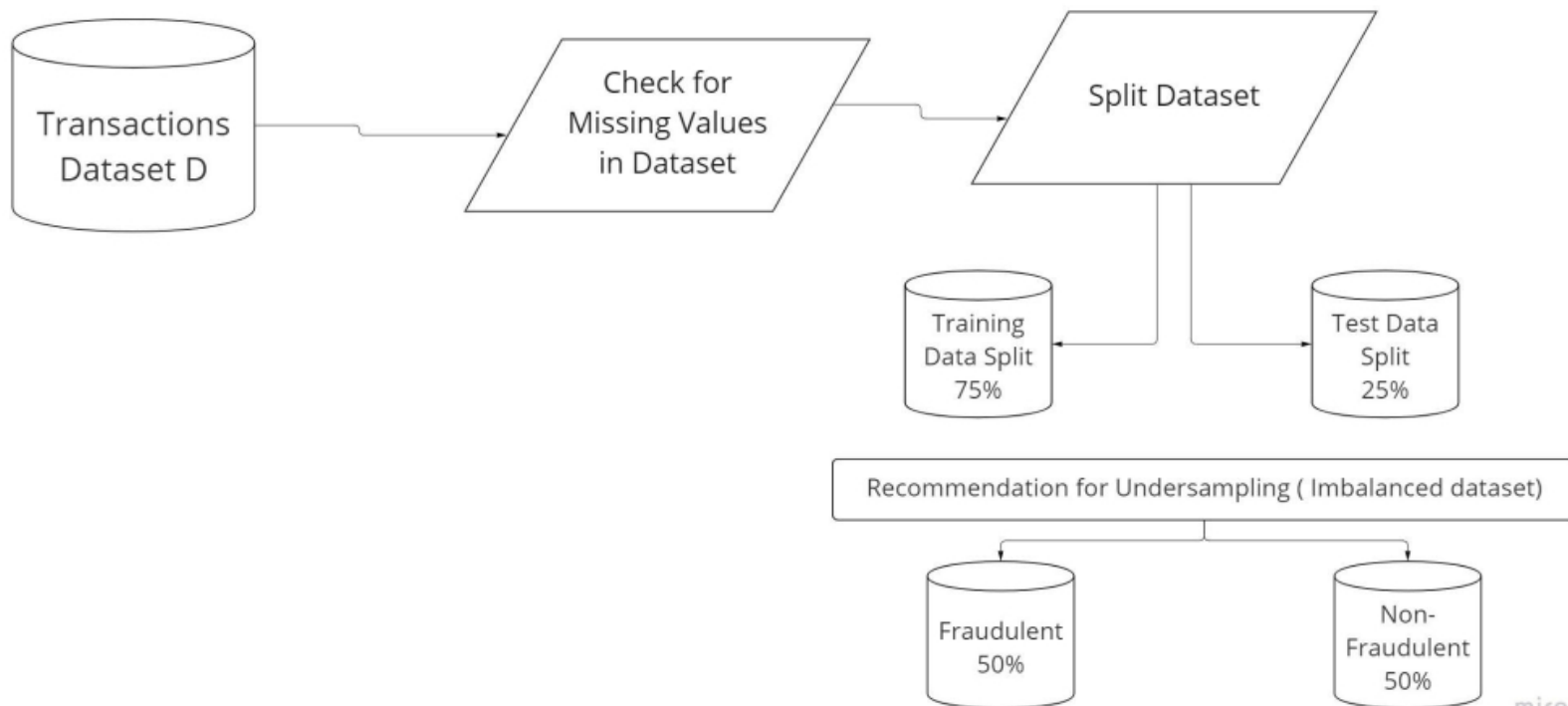
Cost Matrix

Mean of the Amount - Type 1 & Type 2 error

Mean- Type 1 Error and Max-Type 2 error

Max of the Amount- Type 2 error &
Zero for type 1

Data Pre-processing



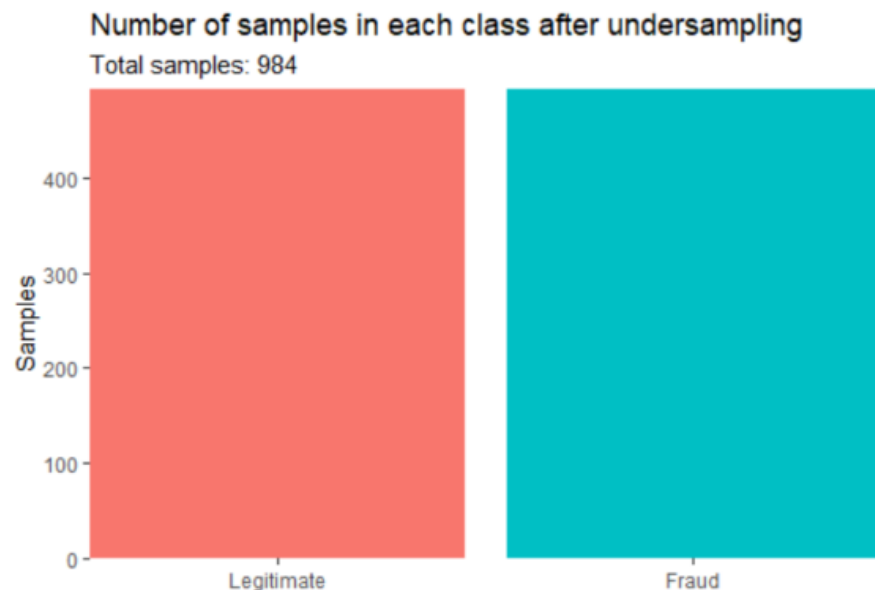
Under- Sampling



For banks or credit card companies to gain real insight from imbalanced datasets, they use resampling. Undersampling is one resampling method, which can be used alone or in conjunction with oversampling. Through an undersampling technique, businesses remove certain events from the majority class, which is made up of the non-fraudulent transactions. The goal is to create a balanced dataset that reflects the real world and can most accurately detect fraudulent transactions.

Under-sampling the dataset:

492 Fraudulent and 492 Non fraudulent (50:50 distribution)

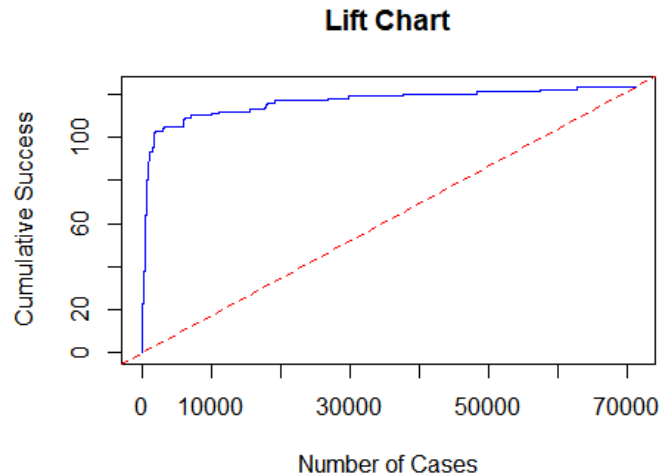




Base Classifier

Model 1: Naive Bayes

- Naive bayes was chosen because of its ability to take factored categorical variables as the dependent variable.
- The model assumes independence among the independent variables.
- We use a cutoff value of 0.4 to classify fraudulent activities with a train test split of 75%-25%.
- Naive Bayes performs well with large datasets as demonstrated by high accuracy.
- But due to lower threshold level there are also high number of false positives which might lead to increased misclassification costs if costs are associated to false positives



	predicted	
actual	0	1
0	69341	1737
1	27	96

Model 2: K Nearest Neighbors



- As a default, the number of neighbors was set to 5. Despite achieving 99.8 percent accuracy with the stated index of "k," the output from the Confusion Matrix has a few flaws.
 - One of these limitations was that the model did not predict any occurrences of "fraud", despite the fact that some actual cases of fraud were incorrectly classified as non-fraud.
 - Another disadvantage was that the obtained accuracy did not exceed the No-Information rate (if all the observations were predicted to be non-fraud). Because all of the cases were classified as non-fraud, the two metrics were equivalent.
- Overall, it is difficult to say if using KNN is a worthwhile attempt for achieving the maximum potential accuracy at this point.

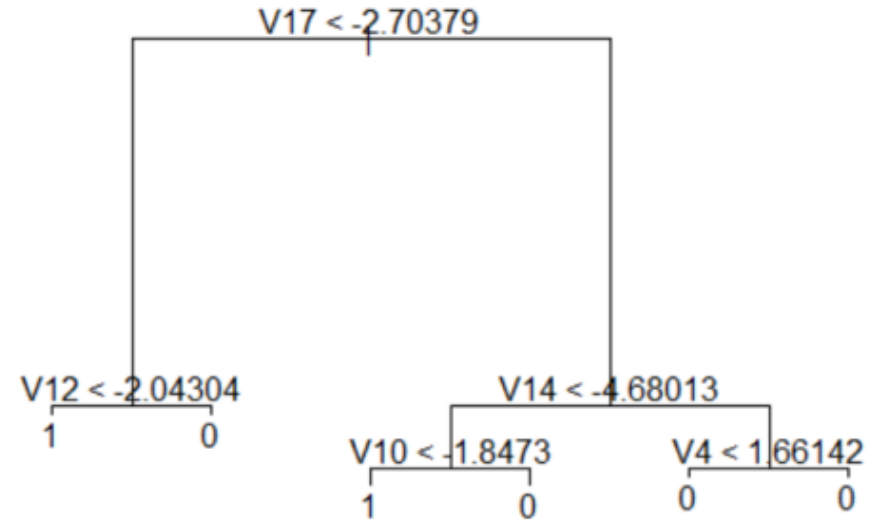
Model 3: Classification Trees

- Classification trees are used to identify the most important predictors.
- The trees are pruned at 5 and 15 to see if accuracy varies with how much we prune the tree - it does not.
- Misclassification costs are the lowest out of all

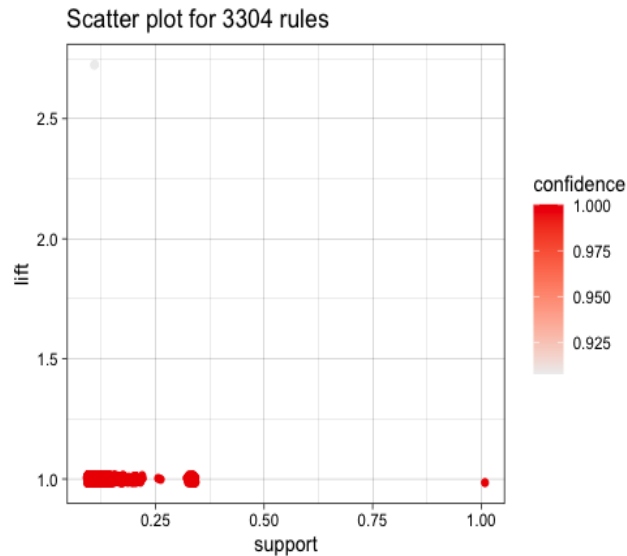
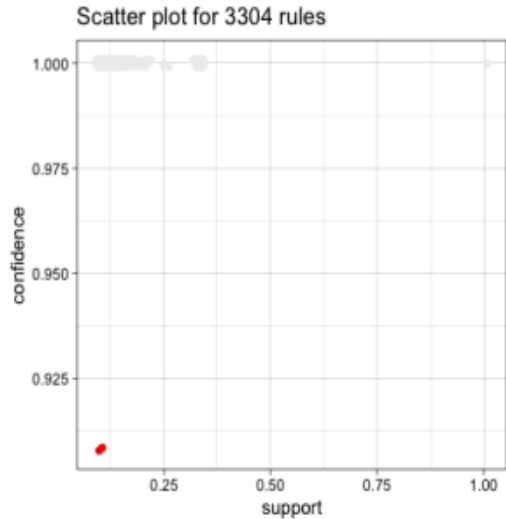
the models

#summary shows the variables actually used in the model are:

"V17" "V12" "V14" "V10" "V4"



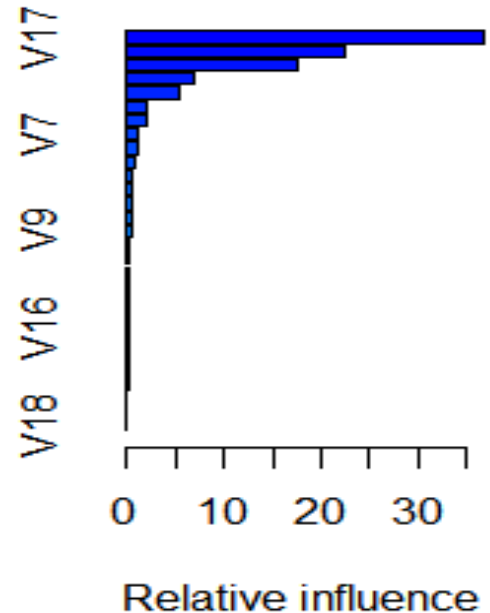
Model 4: Association Rules



- R revealed 3304 rules with a few variables having exactly the same support, confidence and lift.
- No clear relationship between variables. Also hard to interpret because variables are unlabeled.

Model 5: Boosting

- Boosting method of 100 trees with a tree depth of 4 was used and had an accuracy of 0.9985955 with a recall of 0.766
- This model gave the second best recall and with further increase in number of trees it will outperform naive bayes model and will give lesser false positives which means lower misclassification costs



Model 6: Logistic Regression

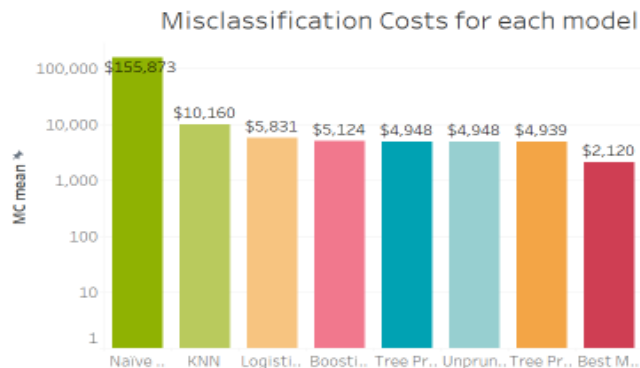
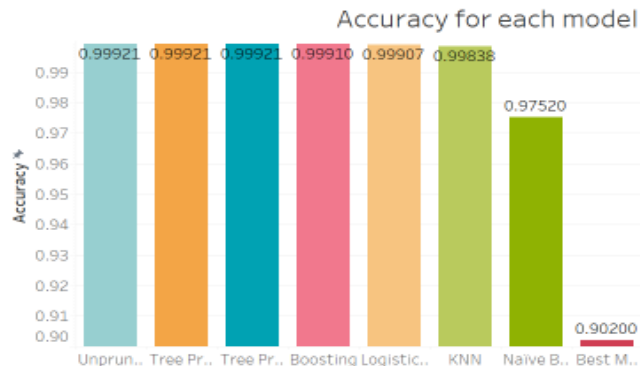
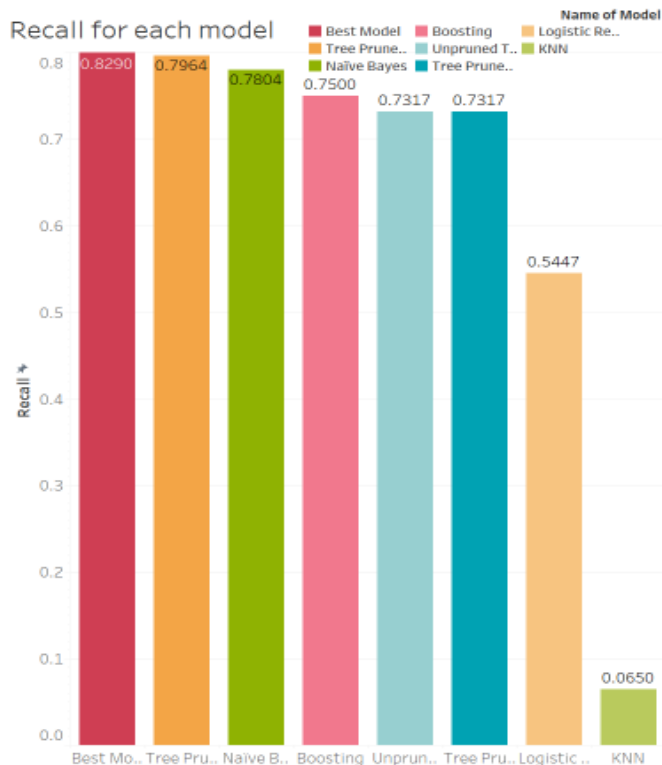
- Logistic regression is done to see if can predict class accurately and what variables are highly related to it.
- Some of the most significant variables were those also indicated by classification trees.
- Recall for this model is the lowest but accuracy remains high.
- V4, V8, V10, V13, V14, V20, V21, V22, V27, V28

```
(Intercept) -8.583e+00 3.092e-01 -27.753 < 2e-16 ***
Time        -3.495e-06 2.787e-06 -1.254 0.209755
V1          4.860e-02 4.931e-02 0.986 0.324307
V2          1.276e-02 6.674e-02 0.191 0.848416
V3         -1.807e-02 6.756e-02 -0.267 0.789101
V4          6.683e-01 8.639e-02 7.735 1.03e-14 ***
V5          5.855e-02 8.541e-02 0.686 0.493016
V6         -1.540e-01 1.015e-01 -1.517 0.129299
V7         -2.967e-02 8.069e-02 -0.368 0.713098
V8         -1.953e-01 3.982e-02 -4.906 9.29e-07 ***
V9         -3.014e-01 1.315e-01 -2.292 0.021930 *
V10        -6.169e-01 1.139e-01 -5.414 6.15e-08 ***
V11        -4.116e-02 9.896e-02 -0.416 0.677426
V12         2.549e-01 1.089e-01 2.340 0.019268 *
V13        -3.778e-01 1.038e-01 -3.640 0.000273 ***
V14        -7.270e-01 7.904e-02 -9.197 < 2e-16 ***
V15        -9.553e-02 1.042e-01 -0.917 0.359389
V16        -1.886e-01 1.452e-01 -1.299 0.194075
V17        -1.423e-01 8.376e-02 -1.699 0.089405 .
V18         1.208e-02 1.519e-01 0.080 0.936602
V19         6.426e-02 1.135e-01 0.566 0.571160
V20        -3.482e-01 1.003e-01 -3.471 0.000519 ***
V21         3.945e-01 7.745e-02 5.093 3.52e-07 ***
V22         8.366e-01 1.652e-01 5.064 4.10e-07 ***
V23        -1.256e-01 7.171e-02 -1.752 0.079793 .
V24         9.788e-02 1.848e-01 0.530 0.596276
V25         8.247e-02 1.596e-01 0.517 0.605277
V26         9.575e-02 2.286e-01 0.419 0.675262
V27        -8.018e-01 1.452e-01 -5.524 3.31e-08 ***
V28        -4.818e-01 1.367e-01 -3.524 0.000425 ***
Amount      7.588e-04 4.622e-04 1.642 0.100666
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


(Dispersion parameter for binomial family taken to be 1)

Null deviance: 5431.8 on 213605 degrees of freedom
Residual deviance: 1548.6 on 213575 degrees of freedom
AIC: 1610.6

Comparison of Performance Measures



Best Model: Rationale



Name of Model	Accuracy	Recall	MC mean	MC mean & max	MC zero and mean
Classification Trees	0.9992135	0.7317	4947.57	849840.321	847808.3
Classification Trees Pruned at 5	0.9992135	0.7317	4947.578	849840.321	847808.3
Classification Trees Pruned at 15	0.9992135	0.7964	4938.578	593812.217	590896.7
Logistic regression	0.999073	0.54471	5830.887	1439588.456	1438705
Boosting	0.9991	0.766	5124.279	721352.48	719100.23
KNN	0.9983849	0.065	10160.21	2954483	.2954483
Naïve Bayes	0.9752	0.7804	155,872.93	823662.87	693661.3



Best Classifier

Findings from Best Base Classifier Model



- Out of all the models Naive Bayes performs the best and identifies most number of fraudulent activities correctly.
- The accuracy is 0.9752251 and the recall is 0.7804878 which is the highest among all the models
- It also has a very high number of false positives. If there are costs associated with false positives then this model will have a very high misclassification costs
- Boosting is the second best model with recall 0.7666.
- As number of trees are increased boosting performs exponentially better
- Here the number of trees are 500 and the tree depth is 4, if increased further at some point it will outperform the naive bayes model for this dataset and will have lower false positives

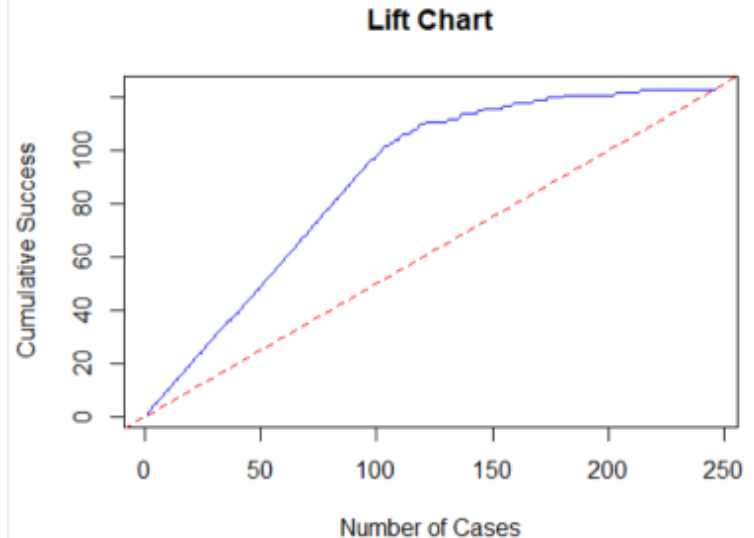
Best Classifier with Under-sampling

Naive Bayes algorithm modelled for Under-sampled data with 984 observations (50% Fraudulent and 50% Non-fraudulent data.

Performance measures:

4 % increase in Recall

```
> (Acc = (cm[1,1]+cm[2,2])/sum(cm))  
[1] 0.902439  
> (recall = (cm[2,2]/(cm[2,1]+cm[2,2])))  
[1] 0.8292683
```



Conclusion



With the increasing reliance on electronic payments in the recent times, business owners and e-commerce companies along with other major stakeholders such as card manufacturers, payment operators, and customers (the cardholders), credit cards face increasing security threats. That calls for a regular need to increase the efficiency and effectiveness of the previous fraud detection schemes with the aim of building reliable and secure payment systems.

Recommendations:

A Data pre-processing tool which uses more than our best model (984 observations data) to model the classifier as this is very minute fraction of the data. Under-sampling can be done more logically rather than random selection. Then, account for the misclassification costs by varying the costs for each transaction using the appropriate classifier with a varying cost matrix. Finally, ensemble the results to achieve the best predictive sensitivity and accuracy.

Sources



Haomin Wang, Gang Kou & Yi Peng (2021) Multi-class misclassification cost matrix for credit ratings in peer-to-peer lending, Journal of the Operational Research Society, 72:4, 923-934, DOI: 10.1080/01605682.2019.1705193 To link to this article:

<https://doi.org/10.1080/01605682.2019.1705193>

<https://rpubs.com/DeclanStockdale/799284>

<https://www.sciencedirect.com/science/article/pii/S2468227620302027>



Thankyou