Cyber Data Analytics

Lab 1 Assignment

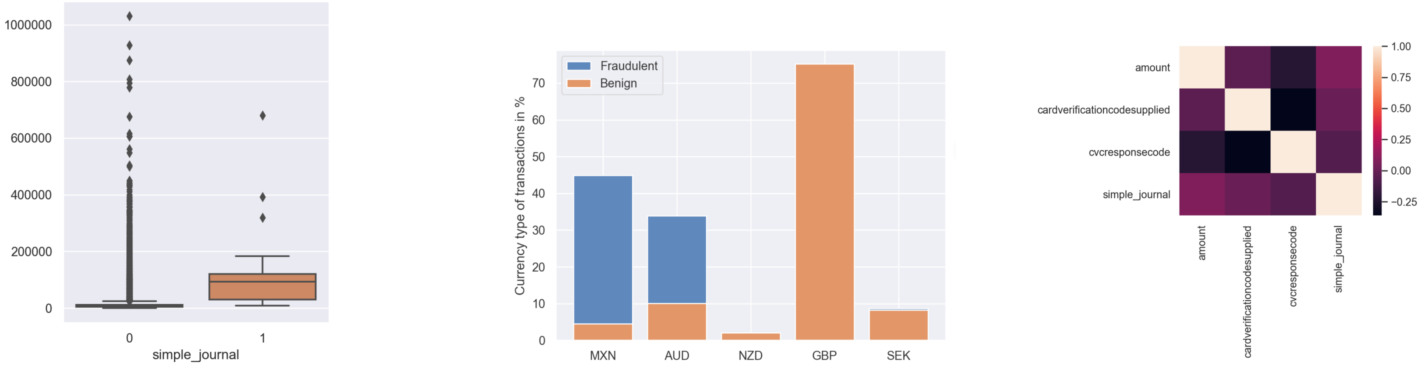
Kees Fani (4437179) and Jody Liu (4920392)

1 Visualization task

For the visualization task, we made visualizations for all the available features. The ones with the most telling relationship are shown in this section.

Prior to the visualizations we explored the data to gain insights in what features are available and how they could relate with one another. Since the task of the assignment is to identify fraudulent transactions, we made Figure 1a as initial visualization. Figure 1a shows a boxplot of the amount of a transaction for fraudulent- and benign cases. In order to make this comparison, we converted all amounts to standard currency; in this case this was the Euro.

While stronger outliers appear for benign cases (`simple_journal '0'`), it shows that most fraudulent transactions (`simple_journal '1'`) have less outliers and are on the median higher compared to benign cases.



(a) Boxplot of the average transaction amount

(b) Currency type of the transactions in percentage

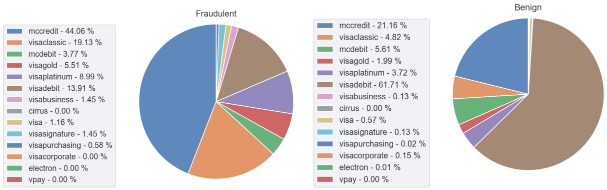
(c) Correlation matrix

Figure 1: Visualizations of the transactions

Figure 1b shows the currency type of the transactions. Interestingly, most fraudulent payments were done with Mexican Peso and Australian dollar. While benign cases mostly in Pond Sterling whereas fraudulent had zero transactions in Pond Sterling.

Figure 1c shows a correlation matrix between different numeric-valued features. This shows that there is no significant relationship between the features and `simple_journal` feature. We do see however a strong negative correlation between the CVC code supplied and the CVC response code. This is in line with what the feature values entail. High number of CVC response code means there is no match or that the CVC code is not checked. Thus when there is a CVC response code (a binary feature) given with value 1, then there is also no match in the supplied CVC code (or it is not checked).

Lastly, in order to see what payment type were used for these transactions, Figure 2 shows pie charts for the type of card used for a transaction. This shows that most card types are rarely used for both parties, but also shows that there is a difference between the most-used types for both parties. These are for fraudulent cases MC Credit (44%) and Visa Classic (19%); for benign transactions these are Visa Debit (62%) MC Credit (21%)



(a) Pie chart for Fraudulent cases

(b) Pie chart for Benign cases

Figure 2: The card type used for the transactions in percentage

2 Imbalanced data task

2.1 Classification choice

For the imbalanced data task we used the three classifiers: Random Forest, k Neighbours and Decision Tree. Since the provided code used kN classifier, we decided to use this algorithm as a means of a baseline. The choice for the other two classifiers are further explained in Section 3.
Since the goal is to understand the effect of SMOTE on imbalanced data, we did not perform any parameter tuning for the classifiers.

2.2 The transaction data and SMOTE

For each classifier we used a training-test split of 65-35. This ratio was chosen because of the requirement to find at least 100 positive cases in the test data.
The data distribution of the data can be seen below:

```
Training data                                  Test data
Before SMOTE, counts of label '1': 221         Counts of label '1': 124
Before SMOTE, counts of label '0': 145060      Counts of label '0': 78105

After SMOTE, counts of label '1': 145060
After SMOTE, counts of label '0': 145060
```

In this case we removed the rows with 'Refused' and with empty column values. All the categorical features were factorized to numeric values and the data was shuffled to make the data time-independent

Figure 3 illustrates the ROC curves for the three classifiers with and without the SMOTE technique. Except for the decision tree the AUC score is higher for data with SMOTE than without. This indicates that an oversampling of the minority class for a balanced data set decreases the number of false positives in the data. This rebalancing on the training data results in a better classification on test data. The lower score for a decision tree with SMOTE might have to do with the model itself, where the hyperparameters could be changed to improve the score. Anoyhrt notable difference is the score achieved by the different classifiers. k Neighbours performs worse. This might be due to the chosen k-value, which highly depends on the number of dimensions, data points and type of features. Decision Trees can be helpful for classifiers with categorical features. Since the data set contain many categorical features, it can perform better than a kN classifier. Finally, the Random Forest performs best with the lowest False Positive rate.

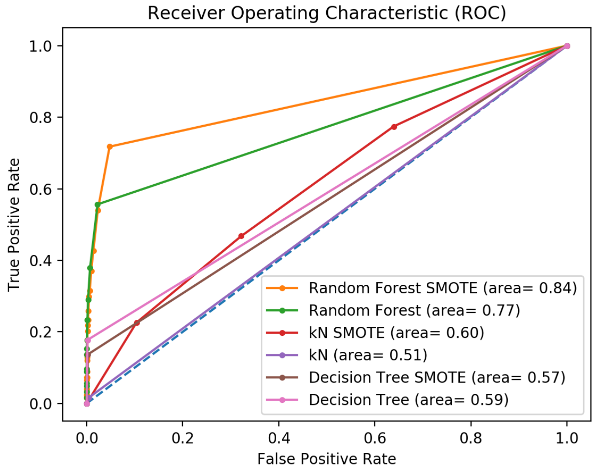


Figure 3: ROC Curves for three classifiers with and without SMOTE

3 Classification task

3.1 Preprocessing and after-processing of the data

The preprocessing was similar to Section 2. In order to improve the performance of Section 3, we made a few added changes in the data.
Aside from the visualizations in Section 1, the following information about the data can be computed:

```
    1) Labels
    Chargeback: 345
    Refused: 53,346
    Authorised: 0
    Settled: 236,691
    2) Empty Values:
    3) Unique email id’s: 220,952
    5) Unique card id’s: 234,814
```

As already known, the data is highly imbalanced with only 345 cases fraudulent.
From the visualizations in Section 1, it is shown that features as amount, currency type and shopping country are notable features. Interestingly, based on the above points (3) and (5) there are more unique card id’s than unique email addresses. This can indicate that people have more than one card and could be a potential indication of fraud. We therefore decided to not remove the above mentioned features.
Lastly, the correlation matrix in Section 1 did not show a strong relationship between the label and the CVC Supplied or CVC code. These two features were therefore removed to reduce the dimensionality of the classified models.

3.2 Classification choice

Because the data hold many categorical features, a classifier that can deal with such feature types are decision trees. Moreover, due to the ensembling-properties of the Random Forest as stated in [1] we decided to use this as second classifier in Section 2 and as blackbox-algorithm for Section 3. It is assumed that an ensembling algorithm such as Random Forest would perform better because of the generation of multiple trees. This would create a more robust model than a single classifier. As third classifier for Section 2 and as whitebox-algorithm for Section 3, we therefore chose the single decision tree to evaluate its performance and compare this to the Random Forest.

3.2.1 Configuration

For both the whitebox- and blackbox algorithms we used 10-fold cross validation. The parameters were chosen on the basis of [2] and empirical testing. We performed 10-fold cross validation on a train- and validation set and then computed for each fold the precision, recall and a confusion matrix on a test set. Furthermore, we applied SMOTE in each fold on the training data based on the results in Section 2. For each fold there were more than 100 True Positive cases and at most 1000 False Positive cases. This is highlighted by the precision and recall scores explained in Section 3.3.

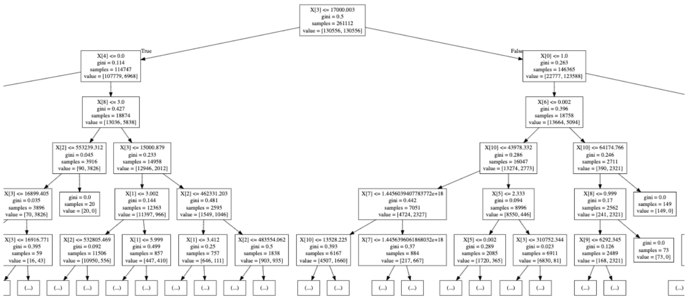
3.3 Results Classifiers

```
Random Forest
Average precision:  0.85
Average recall:  0.58

Decision Tree
Average precision:  0.48
Average recall:  0.66
```

The random forest performs better than the decision tree with a higher precision, but with lower recall. Since the imbalanced data contain more benign cases than fraudulent ones, the likelihood of misclassifying more benign cases as fraudulent is higher than fraudulent cases being misclassified as benign. Since the random forest has an ensembling-characteristic (taking many single decision trees) it is able to create a more robust model against False Positives than a single decision tree. However, the decision tree has less False Negatives than the random forest based on the recall. This could be because the misclassification of many trees lowers the overall average recall than when a decision tree computes a single value.

Aside from the performance of a random forest, we can also look at the classification path by creating a visualization of the decision tree. Figure 4 is a subtree of the complete tree. Features that appear most important in the tree are Amount, Issuercountrycode, Currencycode and IP address. With the visualizations in Section 1 highlighting some features with a notable relationship between fraudulent case and a feature, we could deduce that most fraudulent cases have transactions with a certain amount in the regions of Mexico or were payed in Mexican Peso at certain places.



X[3] <= 17000.003 gini = 0.5 samples = 261112 value = [130556, 130556]
True
False
X[4] <= 0.0 gini = 0.114 samples = 114747 value = [107779, 6968]
X[0] <= 1.0 gini = 0.263 samples = 146365 value = [22777, 123588]
X[8] <= 3.0 gini = 0.427 samples = 18874 value = [13036, 5838]
X[6] <= 0.002 gini = 0.396 samples = 18758 value = [13664, 5094]
X[2] <= 553239.312 gini = 0.045 samples = 3916 value = [90, 3826]
X[3] <= 15000.879 gini = 0.233 samples = 14958 value = [12946, 2012]
X[10] <= 43978.332 gini = 0.286 samples = 16047 value = [13274, 2773]
X[10] <= 64174.766 gini = 0.246 samples = 2711 value = [390, 2321]
X[3] <= 16899.405 gini = 0.035 samples = 3896 value = [70, 3826]
gini = 0.0 samples = 20 value = [20, 0]
X[1] <= 3.002 gini = 0.144 samples = 12363 value = [11397, 966]
X[2] <= 462331.203 gini = 0.481 samples = 2595 value = [1549, 1046]
X[7] <= 1.4456039407783772e+18 gini = 0.442 samples = 7051 value = [4724, 2327]
X[5] <= 2.333 gini = 0.094 samples = 8996 value = [8550, 446]
X[8] <= 0.999 gini = 0.17 samples = 2562 value = [241, 2321]
gini = 0.0 samples = 149 value = [149, 0]
X[3] <= 16916.771 gini = 0.395 samples = 59 value = [16, 43]
X[2] <= 532805.469 gini = 0.092 samples = 11506 value = [10950, 556]
X[1] <= 5.999 gini = 0.499 samples = 857 value = [447, 410]
X[1] <= 3.412 gini = 0.25 samples = 757 value = [646, 111]
X[2] <= 483554.062 gini = 0.5 samples = 1838 value = [903, 935]
X[10] <= 13528.225 gini = 0.393 samples = 6167 value = [4507, 1660]
X[7] <= 1.4456396061868032e+18 gini = 0.37 samples = 884 value = [217, 667]
X[5] <= 0.002 gini = 0.289 samples = 2085 value = [1720, 365]
X[3] <= 310752.344 gini = 0.023 samples = 6911 value = [6830, 81]
X[9] <= 6292.345 gini = 0.126 samples = 2489 value = [168, 2321]
gini = 0.0 samples = 73 value = [73, 0]
(...)

Figure 4: Decision Tree

(0) issuercountrycode
(1) txvariantcode
(2) bin
(3) amount
(4) currencycode
(5) shoppercountrycode
(6) shopperinteraction
(7) creationdate
(8) accountcode
(9) mail_id
(10) ip_id
(11) card_id

References

[1] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data mining for credit card fraud: A comparative study," *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, 2011.

[2] M. R. Segal, "Machine learning benchmarks and random forest regression," 2004.