BANK ACCOUNT FRAUD ANALYSIS AND PREDICTIVE MODELING

MAY 6, 2023

GROUP 2

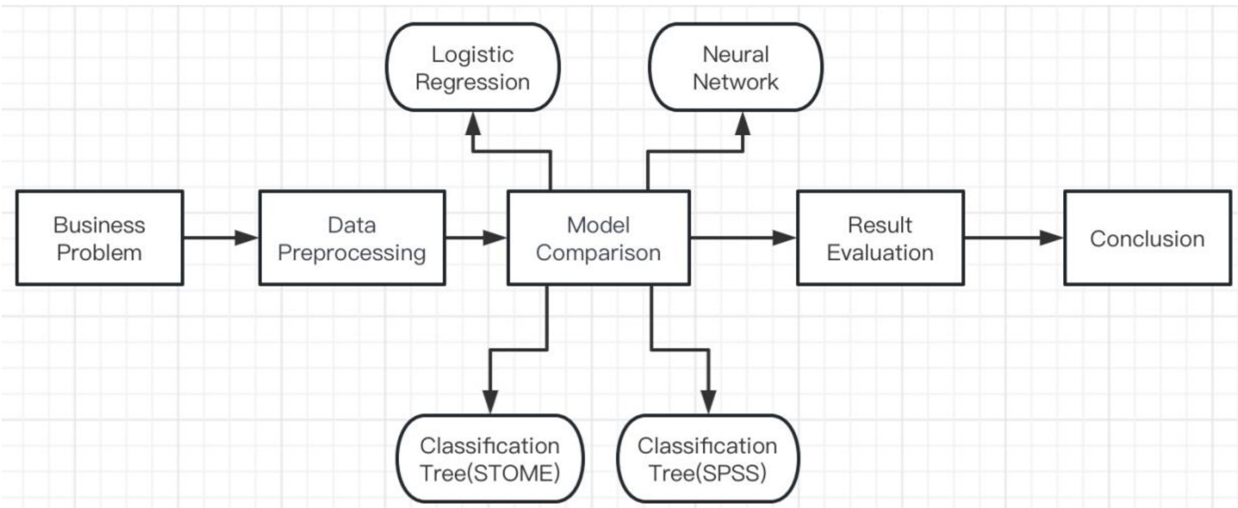
Member: Yuqi Chen, Yirong Wang, Haoyu Zhu, Xiaoyang Guo, Jianzhang Zhu

Abstract

In this study, we address the critical issue of bank account fraud by developing a predictive model that accurately identifies fraudulent applications. We utilize various machine learning models, including Logistic Regression, Decision Trees, and Neural Networks, and employ oversampling techniques to address the class imbalance. Our results indicate that the Neural Network model offers superior performance in detecting fraudulent transactions, with a high level of predictive accuracy. However, the models' limitations include potential overfitting and a lack of adaptability to evolving fraud patterns.

1. Introduction

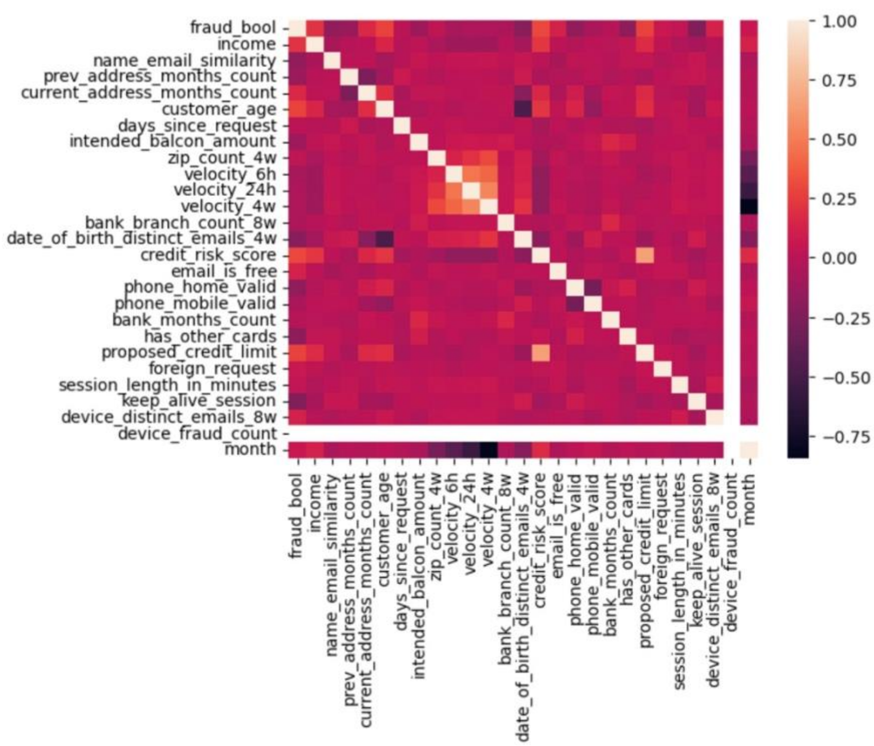
Bank account fraud has become an increasingly prevalent issue in recent years, with banks facing significant financial losses and customers experiencing severe consequences. The objective of this project is to identify potential fraud risks and develop a predictive model that accurately identifies fraudulent applications. By understanding the major factors influencing fraud and predicting the likelihood of customers being a fraud, banks can minimize losses and better protect their customers.



2. Dataset Description

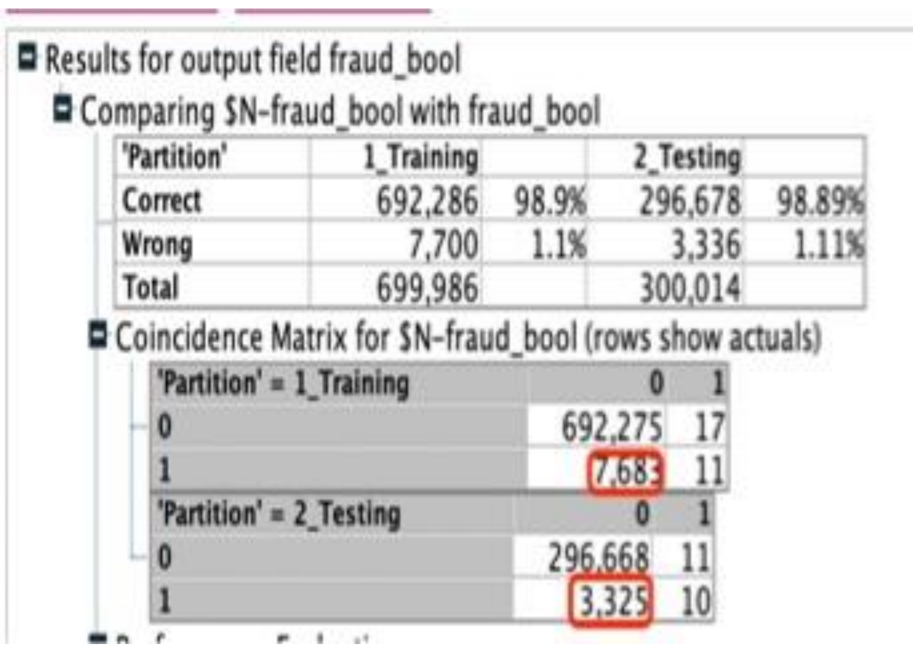
The Bank Account Fraud (BAF) dataset was obtained from Kaggle and consists of six different synthetic bank account fraud tabular datasets. The dataset is realistic, biased, imbalanced, dynamic, and privacy-preserving. The dataset is available in CSV format and has a size of 206.6 MB. The dataset contains variables such as Fraud_bool, income, name_email_similarity, customer_age, and others.

```
fraud_bool                                                [1, 0]
payment_type                                [AA, AB, AC, AD, AE]
employment_status                   [CA, CB, CC, CD, CE, CF, CG]
email_is_free                                             [0, 1]
housing_status                      [BA, BB, BC, BD, BE, BF, BG]
phone_home_valid                                          [1, 0]
phone_mobile_valid                                        [0, 1]
has_other_cards                                           [0, 1]
foreign_request                                           [0, 1]
source                                       [INTERNET, TELEAPP]
device_os             [windows, other, linux, macintosh, x11]
keep_alive_session                                        [0, 1]
dtype: object
```



3. Data Pre-Processing

During the pre-processing stage, correlations between different variables were examined to eliminate multicollinearity. An overview of categorical variables was also conducted. The original dataset was unbalanced, with 11008 instances tagged as 1 (fraudulent) and 988943 tagged as 0 (non-fraudulent). To address this issue, two different oversampling approaches were employed: SPSS and Synthetic Minority Over-sampling Technique (SMOTE).



Results for output field fraud_bool

Comparing $N-fraud_bool with fraud_bool

'Partition'	1_Training		2_Testing	
Correct	692,286	98.9%	296,678	98.89%
Wrong	7,700	1.1%	3,336	1.11%
Total	699,986		300,014	

Coincidence Matrix for $N-fraud_bool (rows show actuals)

'Partition' = 1_Training	0	1
0	692,275	17
1	7,683	11
'Partition' = 2_Testing	0	1
0	296,668	11
1	3,325	10

4. Methodology

To develop the predictive model, three machine learning techniques were utilized: Logistic Regression, Decision Trees, and Neural Networks. The models were trained on the oversampled datasets, and their performance was evaluated using the ROC AUC score on the test set.

4.1 Logistic Regression

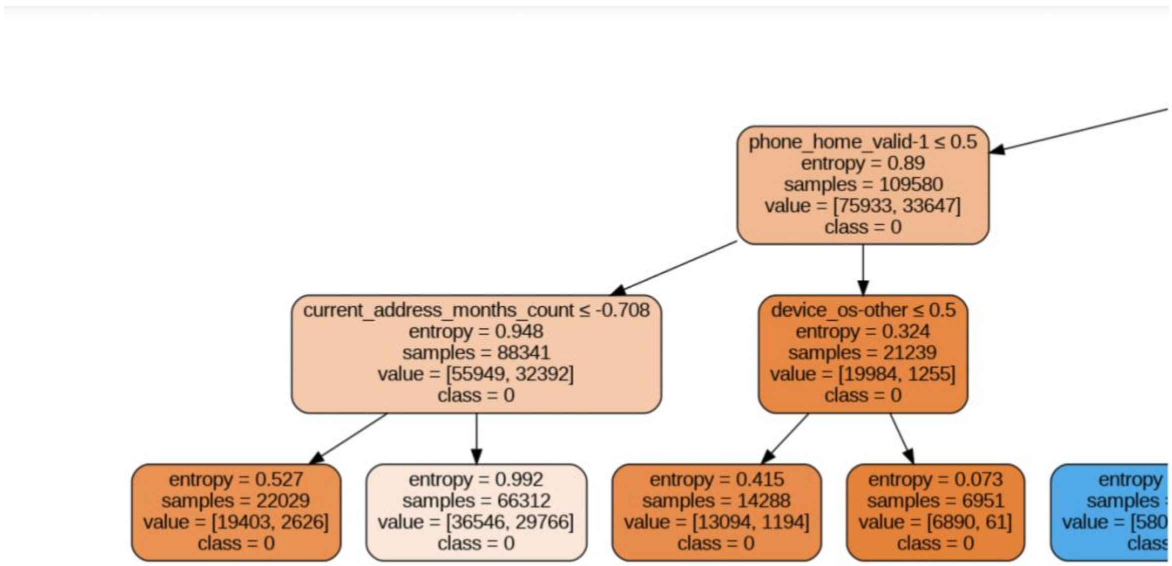
The optimal regularization parameter (alpha) was identified as 23.224. The top three positive predictors were device_os-windows, payment_type-AC, and device_os-macintosh, while the top three negative predictors were housing_status-BE, has_other_cards-1, and housing_status-BB.

```
[[169552  28473]
 [ 37970 100449]]
              precision    recall  f1-score   support

           0       0.82      0.86      0.84    198025
           1       0.78      0.73      0.75    138419

    accuracy                           0.80    336444
   macro avg       0.80      0.79      0.79    336444
weighted avg       0.80      0.80      0.80    336444

0.8025139399127343
```

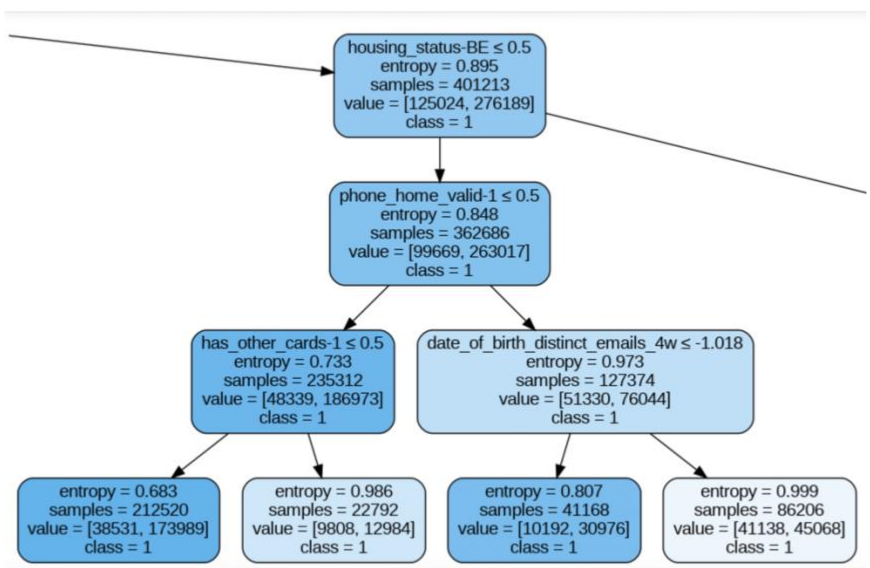
4.2 Decision Trees

Two different datasets were used for training the Decision Tree Classifier: the original dataset with SMOTE oversampling and the dataset with SPSS oversampling. In both cases, a tree depth of 5 was found to be the best hyperparameter.

SMOTE



After Oversampling



4.3 Neural Networks

The Neural Network model structure included input, hidden, and output layers with their corresponding weights and biases. The model's performance was evaluated using the ROC AUC score on the test set.

```
Section 3. Hidden (H) -> Output (O) - Weight (W):

H: 1 -> O - W: -1.9350125834770644
H: 2 -> O - W: 0.8531435831953775
H: 3 -> O - W: -0.7894580098546637
H: 4 -> O - W: -1.0022219644934918
H: 5 -> O - W: 0.5836083118660991
H: 6 -> O - W: 0.8517694546146137
H: 7 -> O - W: 1.5000824925765182
H: 8 -> O - W: 1.2729474687254079
H: 9 -> O - W: -1.2306863092733755

Section 4. Output (O) - Node Bias (B):

O - B: 4.669752633673926
```

5. Results and Evaluation

The performance of each model, based on the ROC AUC score, was as follows:

- Classification Tree (SPSS oversampling): 0.8084
- Logistic Regression: 0.7975
- Classification Tree (SMOTE): 0.7647
- Neural Network: 0.8984 (Best)

The Neural Network model demonstrated the best predictive performance compared to the other models. However, the oversampling techniques (SPSS oversampling and SMOTE) used to address class imbalance may result in overfitting the minority class.

6. Conclusion and Lessons Learned

The study of various credit card fraud detection models provided valuable insights into their performance, strengths, and weaknesses. The Neural Network model demonstrated the power and adaptability of deep learning techniques in solving complex, imbalanced classification problems. We recommend using the Neural Network model for credit card fraud detection.
By analyzing various models, we learned the importance of selecting the right model, addressing the class imbalance, considering interpretability vs. performance trade-offs, and ensuring continuous improvement by updating and re-evaluating models over time.

7. Limitations and Future Work

Despite the promising results obtained from the Neural Network model, there are some limitations that need to be addressed:

7.1 Overfitting: The models might be prone to overfitting, which can lead to poor performance on unseen data. This issue could be mitigated through more extensive hyperparameter tuning, cross-validation, and regularization techniques.

7.2 Evolving Fraud Patterns: The models do not account for evolving fraud patterns or external factors, such as changes in technology, regulations, or the economic environment. This limitation could affect the long-term effectiveness of the models as fraud detection mechanisms.

7.3 Evaluation Metrics: The evaluation of model performance primarily focused on the AUC-ROC metric. While this metric is useful for comparing different models, it does not provide a complete picture of model performance. Additional metrics and real-world performance testing should be considered to better understand the practical implications of deploying these models for fraud detection.

Future work could focus on addressing these limitations by exploring more advanced machine learning techniques, such as ensemble methods or unsupervised learning, and incorporating more comprehensive evaluation metrics. Additionally, the models could be updated and re-evaluated periodically to ensure their continued effectiveness in detecting fraud. Incorporating features that capture temporal patterns or external factors could also help improve model performance and adaptability to evolving fraud patterns.

References

[1] Kaggle. Bank Account Fraud Dataset - NeurIPS 2022. Retrieved from https://www.kaggle.com/datasets/sgpjesus/bank-account-fraud-dataset-neurips-2022?resource=download

[2] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321-357.

[3] He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence) (pp. 1322-1328). IEEE.

[4] SPSS Inc. (2017). IBM SPSS Statistics for Windows (Version 25.0). IBM Corp.