Final project - Credit card fraud detection

Sreenivasulu Somu

2024-08-10

Introduction

In this report, we summarize the final phase of our research project, focusing on the analysis of credit card transactions to detect fraudulent activities. We aim to provide a coherent narrative that tells a story with the data, transitioning from analysis to the proposed implementation of a solution.

Problem Statement

The primary goal of this analysis is to develop a machine learning model capable of detecting fraudulent credit card transactions with high accuracy. The dataset provided from Kaggle is heavily imbalanced, with fraudulent transactions being a small fraction of the total transactions. The challenge is to address this imbalance while ensuring that the model does not produce too many false positives or negatives, which could either inconvenience customers or allow fraud to slip through undetected.

How the Problem Statement Was Addressed

Data Description and Preparation

The dataset consists of anonymized features labeled as `V1` through `V28`, which are likely the result of a PCA transformation, along with two additional features: `Time` and `Amount`. The `Class` feature indicates whether a transaction is fraudulent (`1`) or legitimate (`0`).

The data pre-processing steps included:

1. **Time Conversion**: The `Time` feature was converted from seconds to a datetime format for easier analysis.
2. **Class Renaming**: The `Class` feature was renamed to `is_fraud` for clarity, and converted to a factor with levels `Legitimate` and `Fraudulent`.
3. **Missing Values and Duplicates**: The dataset was checked for missing values and duplicates, which were removed if found.

Methodology

To tackle the extreme class imbalance, the following steps were implemented:

1. **Resampling Techniques**: Methods such as oversampling the minority class or undersampling the majority class were considered to balance the dataset.
2. **Model Selection**: Various Machine Learning models were evaluated, including Logistic Regression, Decision Trees, Random Forest, and Gradient Boosting Machines.
3. **Evaluation Metrics**: Given the imbalance, traditional accuracy metrics were insufficient. Precision, Recall, F1-Score, and AUC-ROC were used to evaluate model performance.

Recommendation for Model Implementation

After evaluating different models, the Random Forest classifier was recommended for implementation due to its balance between precision and recall, along with a relatively high AUC-ROC score. This model was also robust to overfitting and provided feature importance scores, which could help in understanding which features contribute most to detecting fraud.

Analysis

The analysis provided several key insights:

1. **Class Imbalance**: The dataset was confirmed to be highly imbalanced, with fraudulent transactions making up less than 1% of the data. This highlighted the need for careful handling to avoid a biased model.
2. **Feature Importance**: The Random Forest model identified specific features, particularly `V4`, `V10`, and `Amount`, as being more significant in predicting fraud. This insight could help in refining fraud detection systems.
3. **Temporal Patterns**: The time-based analysis did not reveal significant patterns in the occurrence of fraud, suggesting that fraud detection models should focus more on transactional features rather than temporal ones.

Implications

For financial institutions, the implementation of a robust fraud detection model could significantly reduce financial losses due to fraud. A model like the one developed in this project can help in early detection of fraudulent transactions, allowing for quick intervention before substantial losses occur. Customers would benefit from increased security, leading to greater trust in the institution's ability to protect their assets.

The analysis has significant implications for consumers and financial institutions:

- **Enhanced Fraud Detection:** Implementing the recommended model could improve the accuracy of fraud detection systems, reducing false positives and negatives.
- **Consumer Protection:** By accurately identifying fraudulent activities, consumers can be protected from unauthorized transactions, enhancing trust in financial services.

Limitations

This analysis has several limitations:

1. **Anonymized Data**: The lack of clear definitions for the features (V1-V28) makes it difficult to interpret the model's decisions and could limit the ability to improve the model based on domain knowledge.
2. **Imbalanced Dataset**: Despite resampling techniques, the model might still struggle with the minority class due to the extreme imbalance.
3. **Model Generalization**: The model's performance could vary on different datasets, especially if the data distribution changes over time.
4. **Real-time Implementation:** The analysis did not consider the computational requirements for real-time fraud detection, which is crucial for practical applications.

Addressing Key Questions

Handling High Dimensionality To effectively handle the high dimensionality of the dataset, we can use techniques such as Principal Component Analysis (PCA) to reduce the number of features while retaining most of the variance.

Interpreting Anonymized Features Feature importance can be interpreted using model-specific methods such as the importance() function in random forests or SHAP values for tree-based models.

Incorporating Time-Based Patterns Time-based patterns can be incorporated by engineering features that capture temporal trends, such as transaction frequency over time or time since the last transaction.

Detecting and Handling Concept Drift To detect and handle concept drift, we can employ techniques such as monitoring model performance over time, using adaptive learning algorithms, or retraining models periodically with the latest data.

Concluding Remarks

This research provides a foundation for developing an effective credit card fraud detection system. While the Random Forest model showed promise, further refinement and continuous monitoring are necessary to ensure its effectiveness in a real-world setting. Future work could involve exploring more advanced techniques like ensemble methods, deep learning models, or incorporating domain-specific knowledge to further enhance the model's performance.