# Credit Fraud Detection

Roman Micuda

August 18, 2024

**Abstract**

The objective of this assignment is to build a machine learning model to detect fraudulent transactions. You will work through the entire process of data handling, feature engineering, model training, evaluation, and deployment. This assignment will also involve exploring the ethical implications of fraud detection.

## 1 Introduction

Fraud detection in financial transactions is a critical challenge for financial institutions. The increasing volume of transactions and the sophistication of fraud techniques necessitate robust detection systems. In this project, we utilize a machine learning approach to detect fraudulent transactions using a publicly available dataset. The project follows a structured approach, including data collection, exploration, preprocessing, feature engineering, model building, evaluation, and deployment.

## 2 Data Collection and Exploration

### 2.1 Data Collection

For this project, we used the *Kaggle Credit Card Fraud Detection dataset.* This dataset contains transactions made by credit cards in September 2013 by European cardholders. The dataset is highly imbalanced, with a small percentage of transactions classified as fraudulent.

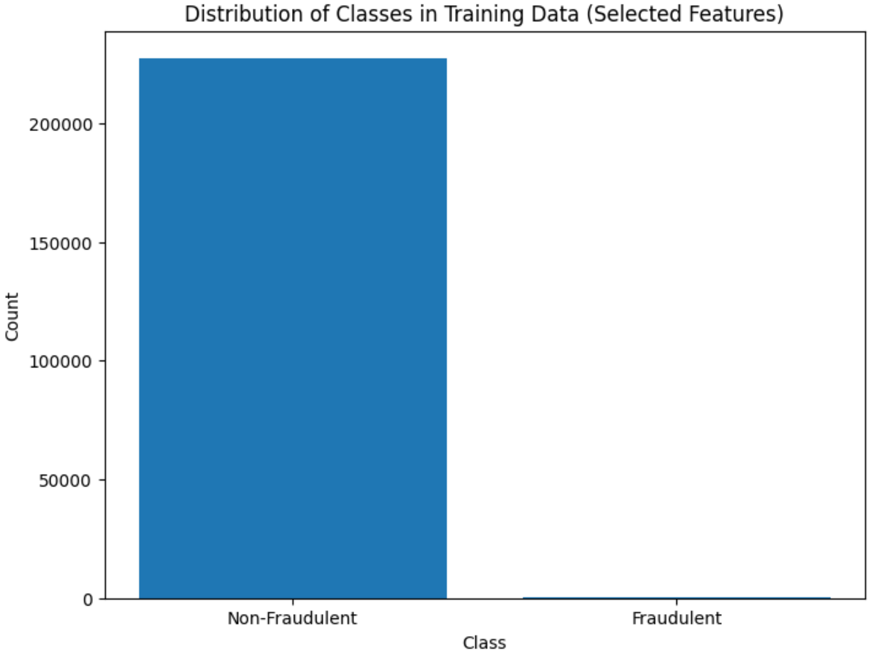

Figure 1: Distribution of Classes in Training Data (Selected Features)

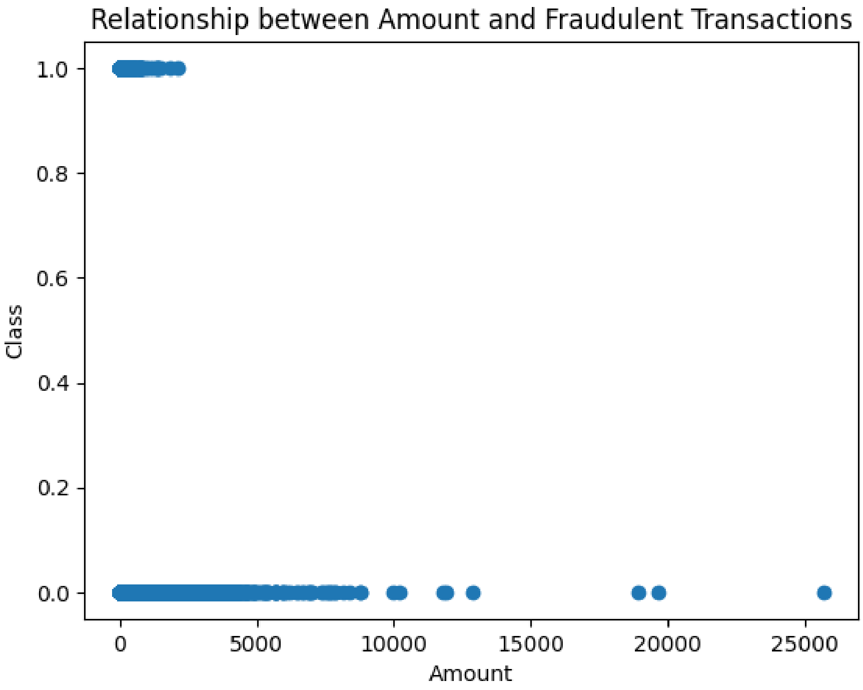

Figure 2: Relationship between Amount and Fraudulent Transactions

### 2.2 Exploratory Data Analysis (EDA)

Exploratory Data Analysis was conducted to understand the dataset better. We visualized the distribution of fraudulent and non-fraudulent transactions, examined feature distributions, and analyzed correlations. The class distribution revealed a significant imbalance between fraud and non-fraud cases.

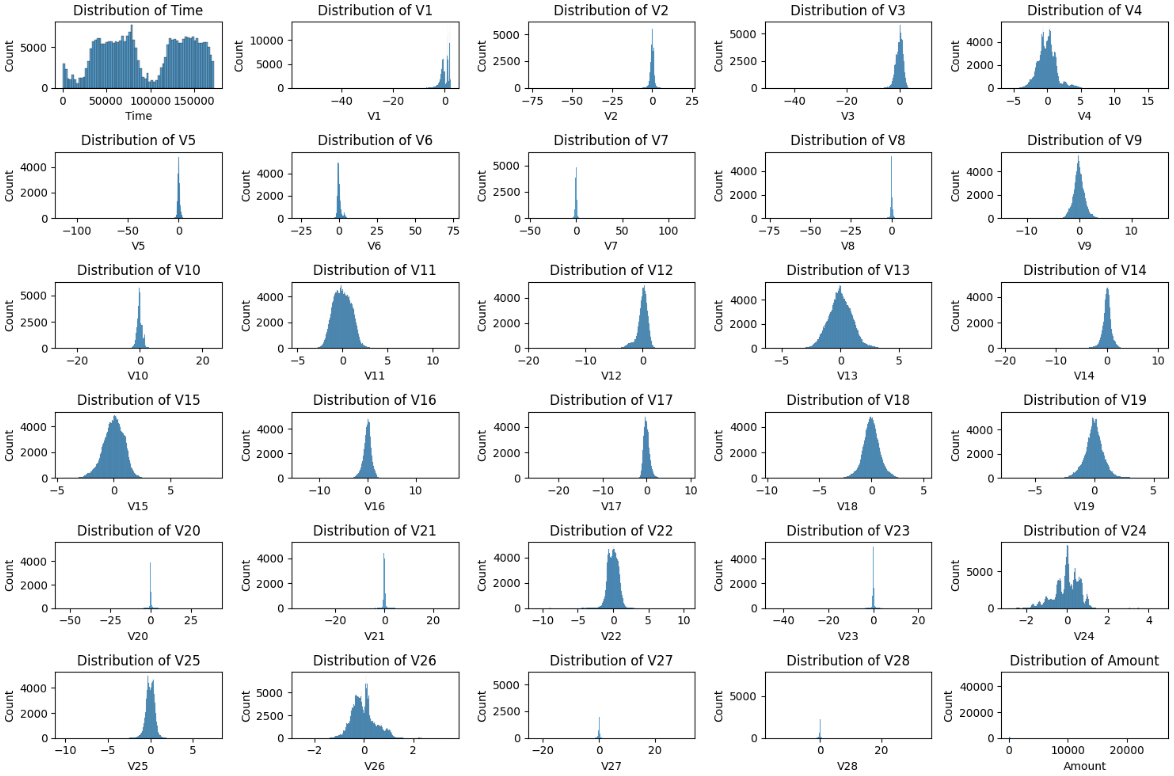

Figure 3: Feature distributions

## 2.3 Data Preprocessing

To handle missing values and outliers, we performed data cleaning. The features were then scaled using *StandardScaler* to normalize the input data. Finally, the dataset was split into training and testing sets to facilitate model evaluation.

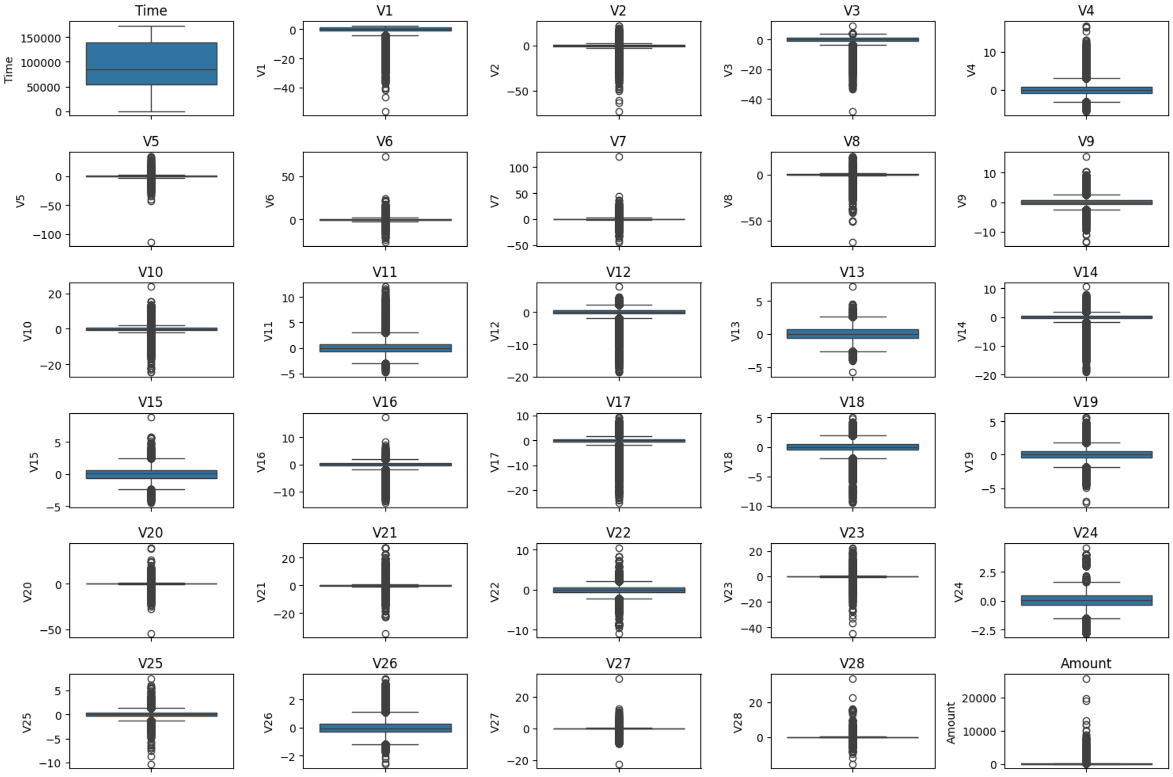

Figure 4: Outlier detection

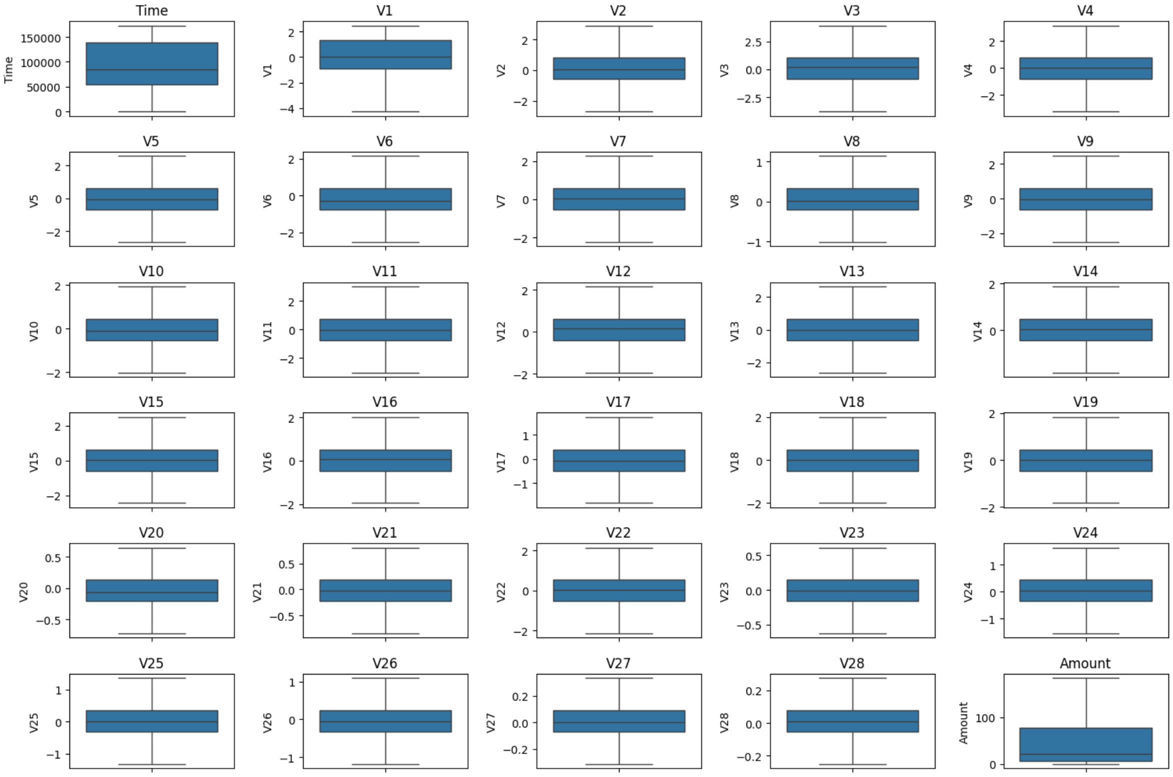

Figure 5: Verify outlier handling

# 3 Feature Engineering

## 3.1 Feature Selection

We analyzed the importance of the features using a *Random Forest Classifier*. The feature importances were visualized to identify the most influential features in predicting fraud.

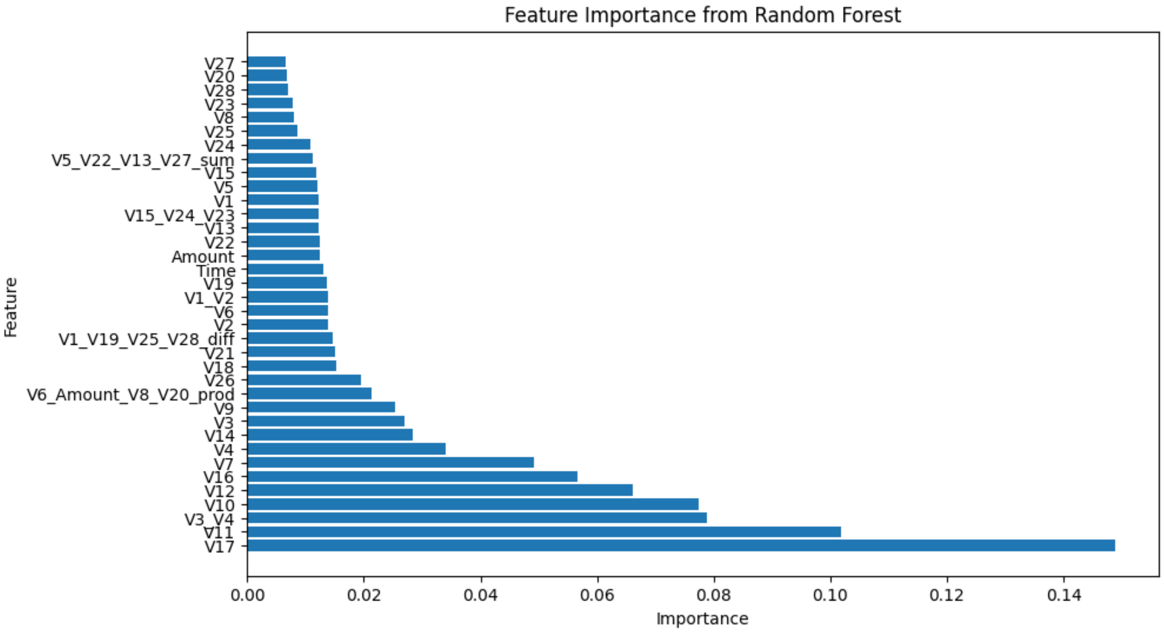

Figure 6: Feature Importance from Random Forest

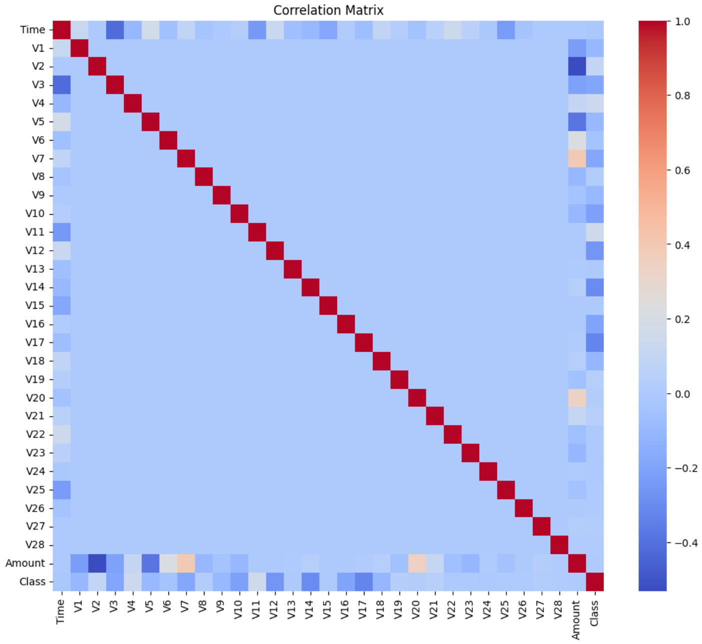

Figure 7: Correlation Matrix

## 3.2 Dimensionality Reduction

To reduce dimensionality and remove noise from the dataset, Principal Component Analysis (PCA) was applied. This step was crucial in improving model performance and computational efficiency.

# 4 Model Building

## 4.1 Algorithm Selection

We explored several machine learning algorithms, including Logistic Regression, Decision Trees, Random Forest, and Gradient Boosting. The choice of algorithms was justified based on their ability to handle imbalanced data and their interpretability in the context of fraud detection.

| Layer (type) | Output Shape | Param # |
|---|---|---|
| Linear-1 | [-1, 128] | 1,536 |
| ReLU-2 | [-1, 128] | 0 |
| Linear-3 | [-1, 64] | 8,256 |
| ReLU-4 | [-1, 64] | 0 |
| Linear-5 | [-1, 1] | 65 |
| Sigmoid-6 | [-1, 1] | 0 |
| **Total params:** | | **9,857** |
| **Trainable params:** | | **9,857** |
| **Non-trainable params:** | | **0** |

## 4.2 Model Training

Models were trained on the training dataset using cross-validation to ensure generalizability. Hyperparameter tuning was conducted to optimize model performance.

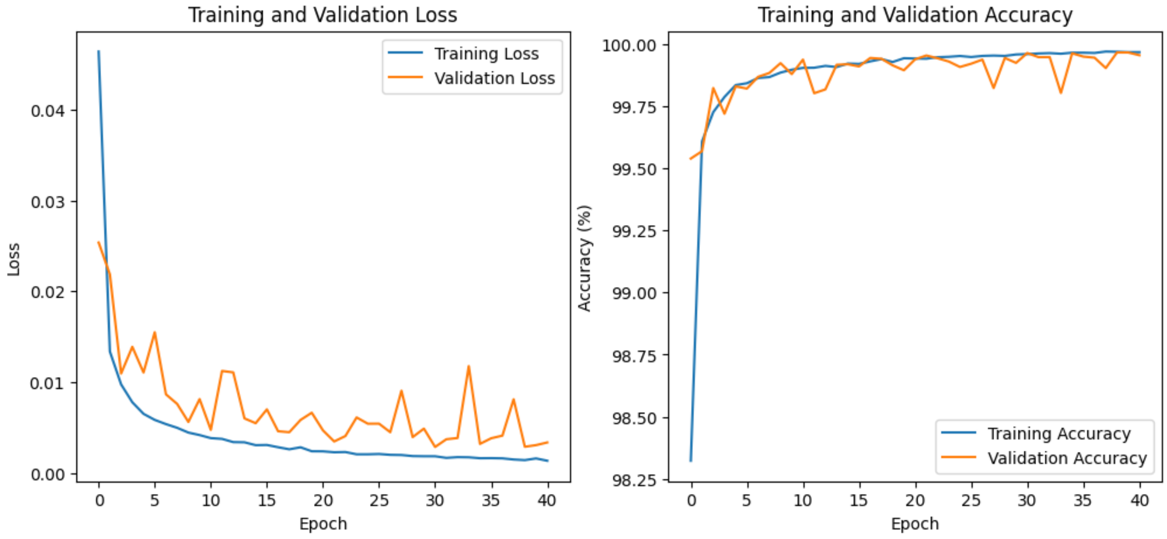

Figure 8: Loss and accuracy

## 4.3 Model Evaluation

Model performance was evaluated using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Special attention was given to the trade-off between precision and recall due to the importance of minimizing both false positives and false negatives in fraud detection.

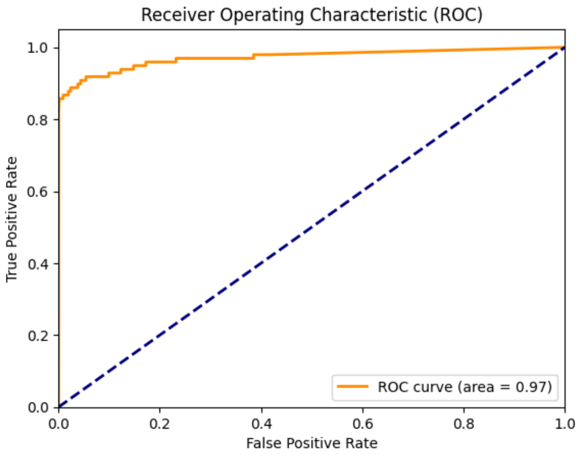

Figure 9: ROC

| | |
|---|---|
| **Test Accuracy:** | 0.9991 |
| **Test Precision:** | 0.6833 |
| **Test Recall:** | 0.8367 |
| **Test F1-score:** | 0.7523 |
| **Test ROC-AUC:** | 0.9180 |

## 5 Model Tuning

To address class imbalance, we implemented oversampling using SMOTE (Synthetic Minority Oversampling Technique). Additionally, cost-sensitive learning techniques were explored to further enhance the model's ability to detect fraud.

## 6 Ethical Considerations

Fraud detection systems raise several ethical concerns, particularly regarding false positives, false negatives, and privacy. False positives can lead to unnecessary scrutiny of legitimate transactions, while false negatives allow fraudulent transactions to go undetected. Ensuring the privacy of transaction data and mitigating biases in the model are also critical considerations in the deployment of fraud detection systems.

## 7 Conclusion

In conclusion, this project successfully developed a machine learning model for detecting fraudulent transactions. The model's performance was optimized through feature engineering, dimensionality reduction. Ethical considerations were addressed to ensure the responsible deployment of the fraud detection system.