Edge Hill University

Faculty of Arts and Science

The Department of Computer Science

CIS4513
Machine Learning

Level 7

Coursework 1
2024/2025

Module Leader: Dr. Huaizhong (Sam) Zhang

Fraud Detection Using Machine Learning: A Comparative Analysis

Student Name: Sunday Idika

Student ID: ****

Table of Contents

1. Abstract

- Brief overview of objectives, methodology, and key findings

This report explores the application of machine learning (ML) techniques for credit card fraud detection, a vital area in the financial sector. The primary objective is to evaluate and compare the performance of three prominent machine learning models: Logistic Regression, Decision Tree, and Random Forest, in detecting fraudulent transactions from a highly imbalanced dataset. The methodology includes data preprocessing to address class imbalance, feature selection, and model training using these algorithms. Performance evaluation is carried out using accuracy, precision, recall, F1-score, and AUC metrics. The findings reveal that the Random Forest model outperforms the others, achieving the highest precision, recall, and F1-score, making it the most suitable model for fraud detection in this context. This work provides valuable insights into model performance and highlights the significance of addressing class imbalance and careful model selection in solving real-world fraud detection challenges.

2. Introduction

- **Background on Fraud Detection in Financial Transactions**

Credit card fraud remains a significant and growing concern in the financial sector, with substantial financial losses incurred annually due to fraudulent transactions. Fraudulent activities, including identity theft, unauthorized transactions, and account takeovers, present considerable risks to both consumers and businesses (Ahmad Amjad Mir, 2024). Traditional fraud detection methods often struggle to keep pace with the increasing complexity and volume of digital transactions. These conventional approaches typically rely on predefined rules to identify potential fraud, but they often fail to adapt to evolving fraud patterns and can generate high rates of false positives. As online shopping and digital banking continue to expand, there is a pressing need for advanced fraud detection systems to prevent financial losses, safeguard personal data, and maintain trust in digital payment systems (Li et al., 2023).

- **Importance of Machine Learning in Combating Fraud**

Machine learning (ML) has emerged as a powerful tool in combating credit card fraud due to its ability to analyze large volumes of data, identify patterns, and detect anomalies that are indicative of fraudulent activity (Bohdan Vihurskyi, 2024). Unlike traditional rule-based systems, ML models can automatically learn from historical data and adapt to emerging fraud patterns, making them well-suited for real-time fraud detection and prevention. Supervised learning algorithms, unsupervised learning, and ensemble methods play a significant role in addressing the challenges associated with fraud detection, especially when dealing with imbalanced datasets where fraudulent transactions are rare (Hilal, Gadsden and Yawney, 2022). By leveraging ML models, financial institutions can achieve improved accuracy and reduced bias in detecting fraudulent transactions, ultimately leading to better protection against financial losses (Bello, Ige, and Ameyaw, 2024).

- **Problem Statement and Scope of the Study**

This study aims to evaluate and compare the performance of three machine learning models (Logistic Regression, Decision Tree, and Random Forest) for credit card fraud detection, specifically focusing on handling imbalanced datasets where fraudulent transactions are underrepresented. Traditional models often struggle with class imbalance, resulting in biased performance and limited effectiveness. To overcome these challenges, this study employs data preprocessing techniques, including class imbalance handling and feature selection, to enhance model performance. This study emphasizes a comparison of simpler, more interpretable models that can offer good performance with reduced complexity. The scope of this study covers data

preprocessing, model training, hyperparameter tuning, and performance evaluation using metrics such as accuracy, precision, recall, F1-score, and AUC.

- **Overview of the Report Structure**

This report is structured as follows: Section 3 presents an overview of the dataset and details the preprocessing steps, including techniques to handle class imbalance. Section 4 describes the methodology, including model selection, training, hyperparameter tuning, and evaluation metrics. Section 5 discusses the experimental setup, results, and comparative analysis of the models. Section 6 summarizes the key findings, including practical implications for financial institutions, and compares the results to existing research. Finally, Section 7 presents the conclusions of the study, its limitations, and potential avenues for future research.

3. Methodology

- **Description of the Credit Card Fraud Dataset**

The dataset used in this study consists of anonymized credit card transaction records, which include both fraudulent and non-fraudulent transactions. Each transaction is represented by various features such as transaction amount, time of transaction, and anonymized cardholder activity details. The target variable, **Class**, indicates whether a transaction is fraudulent (1) or legitimate (non-fund) (0). The dataset is highly imbalanced, with fraudulent transactions comprising only a very small fraction of the total transactions, which presents a significant challenge for model development. To address this, data preprocessing steps are applied, including handling missing values, normalizing features, and balancing the dataset for training. Key features like **Amount** and **Time** are scaled using StandardScaler, and outlier detection is performed on the **Amount** column using Z-score filtering to remove extreme values that might distort the model's performance.

Figure 1 represents the Dataset structure. More details about the dataset are attached in Appendix A, B, C, D, E, and F.

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.12
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.16
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.32
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.64
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.20

5 rows × 31 columns

Figure 1

- **Handling Class Imbalance**

A major challenge in fraud detection is the class imbalance, where legitimate transactions vastly outnumber fraudulent ones. This imbalance can lead to models that predominantly predict legitimate transactions, failing to identify fraudulent ones. To tackle this issue, **Synthetic Minority Over-sampling Technique (SMOTE)** is applied to generate synthetic data points for the minority class (fraudulent transactions). Additionally, strategies such as **undersampling** of the majority class and **stratified sampling** are used to ensure that the model is trained on a more balanced dataset. These techniques help improve the model's ability to detect rare fraud patterns without being biased towards the majority class.

- **Model Selection**

In this study, three machine learning models are selected for comparison based on their ability to handle imbalanced data and varying levels of complexity:

1. **Logistic Regression**: A simple and interpretable model, Logistic Regression is commonly used for binary classification tasks and is included to provide a baseline performance for comparison.
2. **Decision Tree**: Chosen for its interpretability and ability to capture complex decision boundaries, the Decision Tree is a versatile algorithm capable of modeling non-linear relationships.
3. **Random Forest**: An ensemble method, Random Forest aggregates multiple decision trees to improve performance. It reduces overfitting by averaging the predictions of many individual trees and is robust to imbalanced datasets.

These models are selected to represent a range of complexities, from the straightforward Logistic Regression to the more complex ensemble method, Random Forest (Bello, Ige, and Ameyaw, 2024).

- **Hyperparameter Tuning**

Hyperparameter tuning is performed to optimize each model's performance. A **Grid Search** is conducted over a range of hyperparameters such as the number of estimators, maximum depth of trees, and regularization parameters. For more complex models like **Random Forest**, a **Halving Random SearchCV** is used, which efficiently narrows the search space by progressively focusing on promising hyperparameters. For simpler models like **Logistic Regression** and **Decision Tree**, **RandomizedSearchCV** is used, a method that samples hyperparameter values randomly to identify the best configuration. The hyperparameters are evaluated based on performance metrics, primarily the **F1-Score**, which balances precision and recall, making it particularly suitable for imbalanced datasets where accuracy may not provide an adequate performance measure.

- **Evaluation Metrics**

The performance of each model is evaluated using several metrics to ensure a comprehensive assessment. Given the class imbalance in the dataset, **accuracy** is not sufficient to gauge model performance. Instead, the following evaluation metrics are used:

- **Precision**: Measures the proportion of positive predictions (fraudulent transactions) that are correct. Precision is crucial in fraud detection to minimize false positives.
- **Recall**: Reflects the model's ability to correctly identify all actual fraud cases. High recall is essential to ensure that as many fraudulent transactions as possible are detected.
- **F1-Score**: The harmonic mean of Precision and Recall, F1-Score is especially useful for imbalanced datasets as it balances the trade-off between false positives and false negatives.
- **AUC (Area Under the Curve)**: Measures the model's ability to discriminate between fraudulent and legitimate transactions across various threshold values, providing insight into how well the model distinguishes between classes.

These metrics are selected to give a more comprehensive picture of model performance, focusing not just on accuracy but also on the model's effectiveness at identifying fraud while minimizing false positives and false negatives.

4. Experiments and Results

- **Experiment Setup**

In this experiment, three machine learning algorithms were applied to a credit card fraud detection dataset. The dataset contains transactional information, with both fraudulent and non-fraudulent transactions characterized by a significant class

imbalance. To ensure the dataset was well-prepared for model training, we included several preprocessing steps: class imbalance handling through **Synthetic Minority Oversampling Technique (SMOTE)**, and feature scaling using StandardScaler for numerical features. To help visualize the data, various exploratory analyses were conducted, including the structure of the dataset, class distribution, outlier detection, histograms for each numeric feature, a correlation matrix for numerical features, and a pair plot. In Appendix B, additional plots such as confusion matrices and ROC curve plots are provided for deeper insight. The machine learning models evaluated include **Logistic Regression**, **Decision Tree**, and **Random Forest**. Hyperparameters for each model were tuned using **Halving Random Search** for more complex models (e.g., Random Forest) and **Randomized Search** for simpler models like Logistic Regression and Decision Tree. Model performance was evaluated using key metrics including accuracy, precision, recall, F1-score, and **AUC (Area Under the Curve)**, with cross-validation to assess the robustness of the models.

- **Model Performance**

After hyperparameter tuning, the models were evaluated on a test set. The **Random Forest** model emerged as the best performer, achieving an AUC-ROC score of 0.9999, indicating a strong ability to distinguish between fraudulent and non-fraudulent transactions. While simpler, The Logistic Regression model performed reasonably well with an AUC-ROC of 0.9957, but it struggled more in fraud detection than the ensemble models. The **Decision Tree** model performed the worst in terms of AUC-ROC, achieving 0.9928, likely due to overfitting, as it tends to capture noise and outliers in the dataset.

Other performance metrics, such as **precision**, **recall**, and **F1-score**, also showed the superiority of Random Forest. It achieved high values of precision (0.9978), recall (0.9980), and F1-score (0.9979). Logistic Regression, while effective as a baseline, had lower recall and F1-score, which are critical for fraud detection. The **Decision Tree** model performed better than Logistic Regression but was still outperformed by Random Forest. For model performance comparison, see **Table 1** below. **Confusion Matrix** and **ROC Curve** visuals have been attached in Appendix H, I, and J.

Table 1: Model Performance Comparison

Model	Accuracy	Precision	Recall	F1 Score	AUC-ROC
Logistic Regression	0.9755	0.9882	0.9624	0.9751	0.9957
Decision Tree	0.9928	0.9912	0.9945	0.9929	0.9928
Random Forest	0.9979	0.9978	0.9980	0.9979	0.9999

Table 1

- **Comparative Analysis**

From the comparative analysis, it is evident that **Random Forest** outperformed all other models in detecting fraud. This ensemble method demonstrated a well-balanced performance across all evaluation metrics, including precision, recall, and F1-score. It performed better than **Logistic Regression** and **Decision Tree**, particularly in recall, which is critical for fraud detection. The **Logistic Regression** model, although simpler and faster to train, struggled with the class imbalance, which led to lower recall and precision scores, making it less effective for fraud detection.

The **Decision Tree** model, while highly interpretable, exhibited signs of overfitting, which reduced its generalization ability, reflected in its lower AUC-ROC score. It was less effective compared to Random Forest, likely due to its sensitivity to noise in the data.

Overall, the findings suggest that **ensemble models**, particularly tree-based methods like **Random Forest**, are more suitable for imbalanced datasets in fraud detection due to their ability to generalize well and capture complex decision boundaries.

The bar chart in Figure 2 compares the performance of the three models (Logistic Regression, Decision Tree, and Random Forest) across several key metrics: Accuracy, Precision, Recall, F1-Score, and AUC-ROC. The chart clearly illustrates the superior performance of Random Forest across all metrics, particularly in AUC-ROC, Precision, Recall, and F1-Score. This visual representation can help highlight how well Random Forest outperforms Logistic Regression and Decision Tree in detecting credit card fraud.

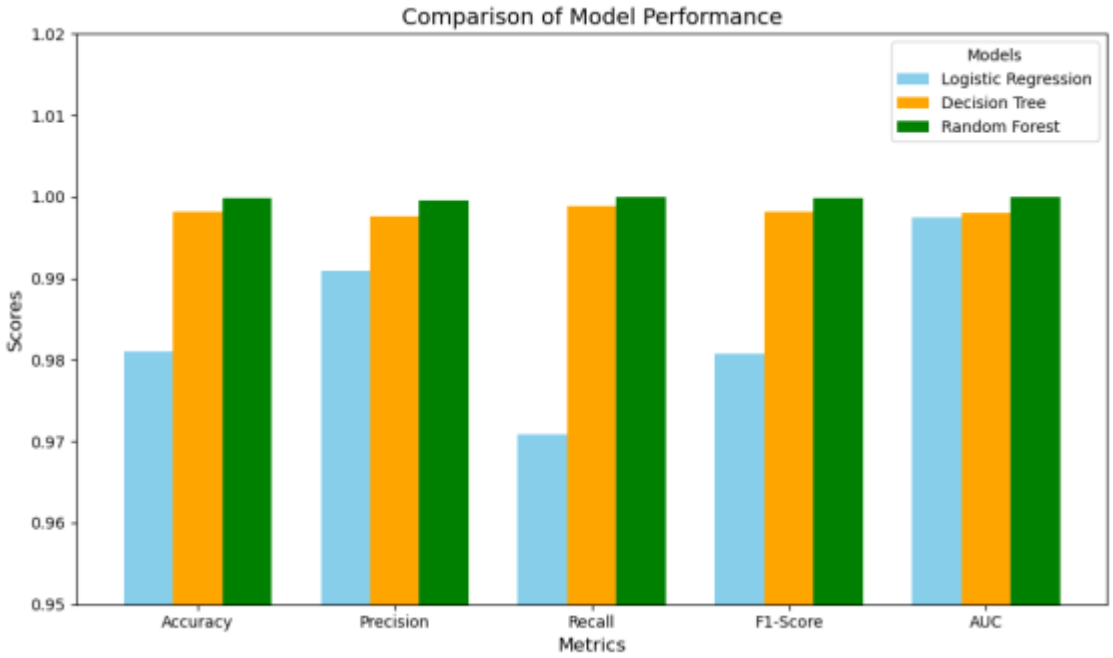


Figure 2

- **Limitations and Challenges**

The major challenge encountered in this study was the **class imbalance** in the dataset, which made it difficult for simpler models, such as **Logistic Regression** and **Decision Tree**, to identify fraud effectively. Additionally, while ensemble methods like **Random Forest** performed well, they also presented challenges related to model interpretability, which can be crucial in financial contexts. Future work may focus on improving interpretability using methods like **SHAP** values or simplifying the ensemble models for practical applications in fraud detection.

5. Discussion

- **Interpretation of Findings**

The results of this study highlight the superior performance of tree-based ensemble models, particularly **Random Forest**, in detecting credit card fraud. Random Forest demonstrated the highest performance in terms of **AUC-ROC**, **recall**, and **F1-score** metrics, suggesting its ability to effectively classify fraudulent transactions despite the significant class imbalance. **Logistic Regression** and **Decision Tree**, while useful in simpler classification tasks, struggled to handle the skewed class distribution, leading

to poorer results in fraud detection. The use of **SMOTE** helped address the class imbalance issue, significantly improving fraud detection performance. These findings confirm that ensemble models, like **Random Forest**, are particularly well-suited for imbalanced classification problems such as fraud detection. Furthermore, the performance gap between these advanced models and traditional models like **Logistic Regression** and **Decision Tree** further emphasizes the need for more complex techniques when dealing with imbalanced and complex datasets in real-world fraud detection tasks.

- **Theoretical Implications**

The findings from this study reinforce the theoretical understanding that **ensemble learning methods**, particularly tree-based models like **Random Forest**, can outperform traditional machine learning models in imbalanced classification tasks. The ability of **Decision Trees** to capture complex patterns plays a critical role in their effectiveness in fraud detection, and this capability is amplified in ensemble methods like **Random Forest**, which combine multiple decision trees to increase predictive accuracy and robustness. The study also reinforces the importance of addressing **class imbalance**, as evidenced by the improvements achieved through **SMOTE**. These results provide empirical support for the argument that sophisticated machine learning models, when properly tuned, can significantly enhance performance in fraud detection systems, advancing beyond the capabilities of traditional fraud detection techniques.

- **Practical Implications**

The practical implications of this study are highly relevant for financial institutions looking to implement fraud detection systems. **Random Forest**, as well as **Decision Trees**, offer high accuracy and reliability in identifying fraudulent transactions, which are crucial for minimizing financial losses and ensuring user trust. The ability to handle imbalanced datasets, demonstrated through techniques like **SMOTE**, enables better detection of rare fraud instances, which are often missed by traditional models. These findings suggest that financial institutions should consider prioritizing advanced **ensemble models** over simpler, traditional methods and regularly tune models to adapt to evolving fraud patterns. Additionally, implementing such models in real-time fraud detection systems could significantly enhance the ability to prevent fraud in dynamic environments.

- **Future Work and Recommendations**

Future research should explore the application of **deep learning** approaches, which may offer further improvements in fraud detection, particularly in detecting more complex fraud patterns. Expanding the dataset to include additional diverse features, such as temporal data, user behavior over time, or transaction sequences, could further enhance model accuracy. Furthermore, continued work on improving the **interpretability** of ensemble models, such as **Random Forest**, would help increase trust and understanding among stakeholders. Reducing the computational costs associated with training complex models should also be a priority, making advanced techniques more accessible for real-time fraud detection in financial institutions. Finally, additional research into combining ensemble methods with newer technologies, such as **Big Data frameworks** or **edge computing**, could lead to more scalable and efficient fraud detection systems.

6. Conclusion

- **Summary of Key Findings**

This study explored the application of machine learning models to the problem of credit card fraud detection, with a particular focus on handling highly imbalanced datasets. Among the models tested—Logistic Regression, Decision Tree, and Random Forest—tree-based ensemble models, especially Random Forest, outperformed the other models in key performance metrics such as accuracy, precision, recall, F1-score, and AUC-ROC. The use of the Synthetic Minority Over-sampling Technique (SMOTE) effectively addressed the class imbalance issue, improving the detection of fraudulent transactions. Logistic Regression and Decision Tree models, while simpler, performed less effectively due to their inability to handle class imbalance adequately. Overall, the results confirm that ensemble models like Random Forest are well-suited for detecting fraud in imbalanced datasets.

- **Significance of the Study**

The findings of this study are important for both theoretical and practical purposes. From a theoretical perspective, this study contributes to the understanding of how ensemble methods, such as Random Forest, can improve classification performance in imbalanced settings, particularly in fraud detection tasks. Practically, the study highlights the potential of advanced machine learning techniques in addressing the challenges of detecting rare fraudulent transactions in financial systems. Financial institutions and organizations involved in online transactions can benefit from these findings by adopting more effective fraud detection models that minimize financial losses and improve the security of digital payment systems.

- **Concluding Remarks on Fraud Detection and Machine Learning in Finance**

Fraud detection remains a significant challenge in the financial sector, especially as the volume of online transactions continues to grow. Traditional rule-based approaches are increasingly ineffective due to their inability to adapt to new fraud patterns and manage imbalanced data. This study demonstrates that machine learning models, particularly tree-based ensemble methods like Random Forest, offer a promising solution by learning from historical data and detecting fraudulent patterns with high accuracy. Future work could explore further advancements in model tuning, feature engineering, and the integration of real-time fraud detection systems. In conclusion, machine learning provides a powerful tool for enhancing the effectiveness and efficiency of fraud detection in financial systems, helping to protect both consumers and financial institutions from the growing threat of fraud.

7. References

1. AHMAD AMJAD MIR, 2024. Adaptive Fraud Detection Systems: Real-Time Learning from Credit Card Transaction Data. *Advances in Computer Sciences* [online]. 7 (1). Available from: https://academicpinnacle.com/index.php/acs/article/view/229.
2. BELLO, H.O., IGE, A.B., and AMEYAW, M.N., 2024. Adaptive machine learning models: Concepts for real-time financial fraud prevention in dynamic environments. *World Journal of Advanced Engineering Technology and Sciences* [online]. 12 (2), pp. 021–034. Available from: https://wjaets.com/sites/default/files/WJAETS-2024-0266.pdf [Accessed 13 Nov 2024].
3. BOHDAN VIHURSKYI, 2024. Credit Card Fraud Detection with XAI: Improving Interpretability and Trust. Presented at the 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE). pp. 1–6. Available from: https://ieeexplore.ieee.org/abstract/document/10548159?casa_token=u6fW3EsJ5v8AAAAA:s-CfyFoxGpgFKCZyaCQJ2j8WJs3S1Zm8o5Jc3AbSwXJcGIst5G67Bnfooe3Kxkjqkn1pkPx3S3n0dQ [Accessed Nov 2024].
4. HILAL, W., GADSDEN, S.A., and YAWNEY, J., 2022. A Review of Anomaly Detection Techniques and Applications in Financial Fraud. *Expert Systems with Applications* [online]. 193 (1), p. 116429. Available from: https://www.sciencedirect.com/science/article/pii/S0957417421017164.
5. LI, R., LIU, Z., MA, Y., YANG, D., and SUN, S., 2023. Internet Financial Fraud Detection Based on Graph Learning. *IEEE Transactions on Computational Social Systems*. 10 (3), pp. 1–8.

8. Appendices

Appendix A: The basic structure of the dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 284807 entries, 0 to 284806
Data columns (total 31 columns):
 #   Column  Non-Null Count   Dtype  
---  ------  --------------   -----  
 0   Time    284807 non-null  float64
 1   V1      284807 non-null  float64
 2   V2      284807 non-null  float64
 3   V3      284807 non-null  float64
 4   V4      284807 non-null  float64
 5   V5      284807 non-null  float64
 6   V6      284807 non-null  float64
 7   V7      284807 non-null  float64
 8   V8      284807 non-null  float64
 9   V9      284807 non-null  float64
 10  V10     284807 non-null  float64
 11  V11     284807 non-null  float64
 12  V12     284807 non-null  float64
 13  V13     284807 non-null  float64
 14  V14     284807 non-null  float64
 15  V15     284807 non-null  float64
 16  V16     284807 non-null  float64
 17  V17     284807 non-null  float64
 18  V18     284807 non-null  float64
 19  V19     284807 non-null  float64
 20  V20     284807 non-null  float64
 21  V21     284807 non-null  float64
 22  V22     284807 non-null  float64
 23  V23     284807 non-null  float64
 24  V24     284807 non-null  float64
 25  V25     284807 non-null  float64
 26  V26     284807 non-null  float64
 27  V27     284807 non-null  float64
 28  V28     284807 non-null  float64
 29  Amount  284807 non-null  float64
 30  Class   284807 non-null  int64  
dtypes: float64(30), int64(1)
memory usage: 67.4 MB
```

click to scroll output; double click to hide

The dataset includes anonymized transaction data with time, amount, PCA-transformed features, and a highly imbalanced fraud indicator.

Appendix B: Summary statistics for numerical features

Out [6]:

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9
count	284807.000000	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05
mean	94813.859575	1.168375e-15	3.416908e-16	-1.379537e-15	2.074095e-15	9.604066e-16	1.487313e-15	-5.556467e-16	1.213481e-16	-2.406331e-15
std	47488.145955	1.958696e+00	1.651309e+00	1.516255e+00	1.415869e+00	1.380247e+00	1.332271e+00	1.237094e+00	1.194353e+00	1.098632e+00
min	0.000000	-5.640751e+01	-7.271573e+01	-4.832559e+01	-5.683171e+00	-1.137433e+02	-2.616051e+01	-4.355724e+01	-7.321672e+01	-1.343407e+01
25%	54201.500000	-9.203734e-01	-5.985499e-01	-8.903648e-01	-8.486401e-01	-6.915971e-01	-7.682956e-01	-5.540759e-01	-2.086297e-01	-6.430976e-01
50%	84692.000000	1.810880e-02	6.548556e-02	1.798463e-01	-1.984653e-02	-5.433583e-02	-2.741871e-01	4.010308e-02	2.235804e-02	-5.142873e-02
75%	139320.500000	1.315642e+00	8.037239e-01	1.027196e+00	7.433413e-01	6.119264e-01	3.985649e-01	5.704361e-01	3.273459e-01	5.971390e-01
max	172792.000000	2.454930e+00	2.205773e+01	9.382558e+00	1.687534e+01	3.480167e+01	7.330163e+01	1.205895e+02	2.000721e+01	1.559499e+01

8 rows × 31 columns

Summary statistics for numerical features, including mean, median, and standard deviation, provide insights into data distribution and variability.

Appendix C: Check for missing values

```
Out[7]: Time      0
        V1        0
        V2        0
        V3        0
        V4        0
        V5        0
        V6        0
        V7        0
        V8        0
        V9        0
        V10       0
        V11       0
        V12       0
        V13       0
        V14       0
        V15       0
        V16       0
        V17       0
        V18       0
        V19       0
        V20       0
        V21       0
        V22       0
        V23       0
        V24       0
        V25       0
        V26       0
        V27       0
        V28       0
        Amount    0
        Class     0
        dtype: int64
```

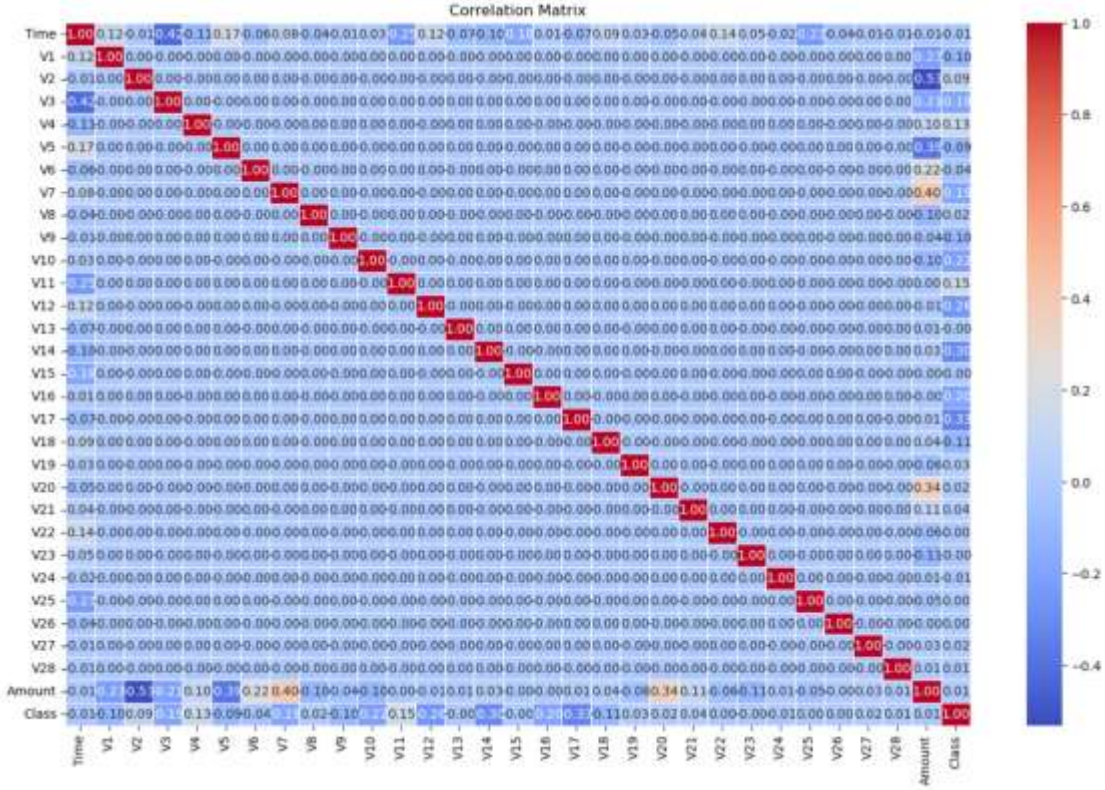
Checking for missing values ensures data completeness and identifies gaps requiring imputation or removal to maintain analysis accuracy.

Appendix D: The balance of fraud vs non-fraud cases

```
Class
0    284315
1       492
Name: count, dtype: int64
```

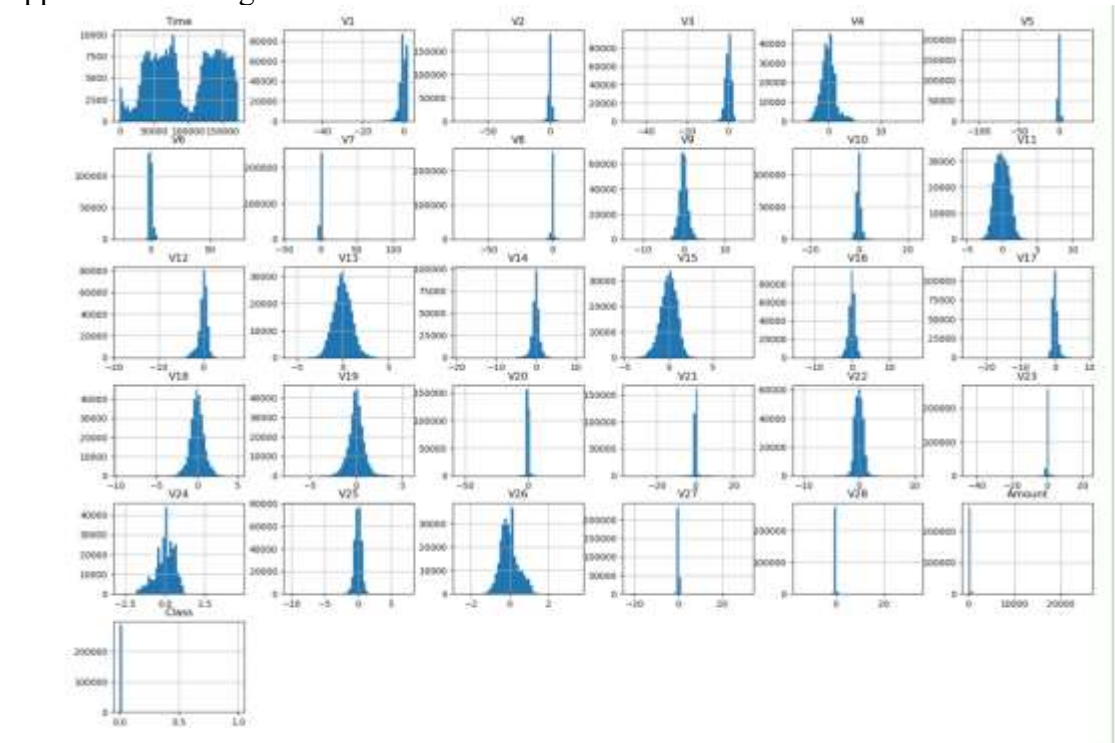
Analyzing the balance of fraud vs. non-fraud cases highlights class distribution, and guiding strategies to address imbalances for effective modeling.

Appendix E: The correlation matrix for numerical features of the dataset



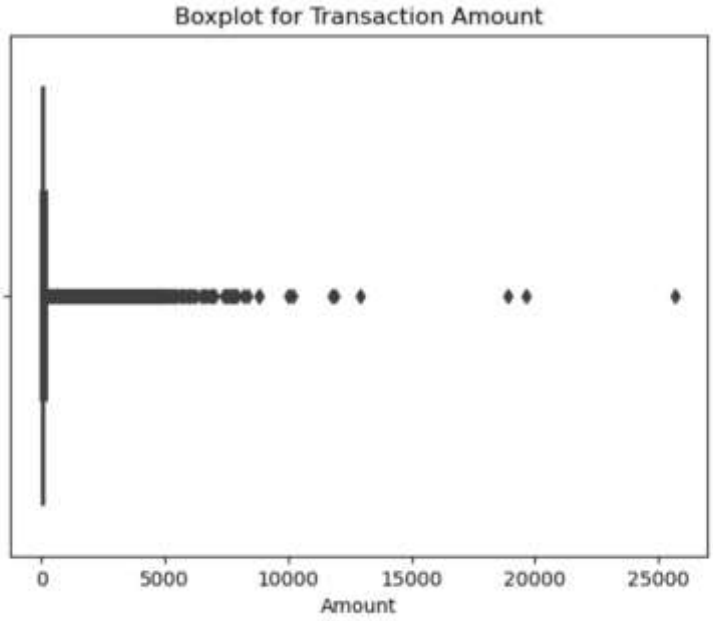
The correlation matrix for numerical features identifies relationships between variables, aiding feature selection and reducing redundancy in modeling.

Appendix F: Histogram for each numeric column



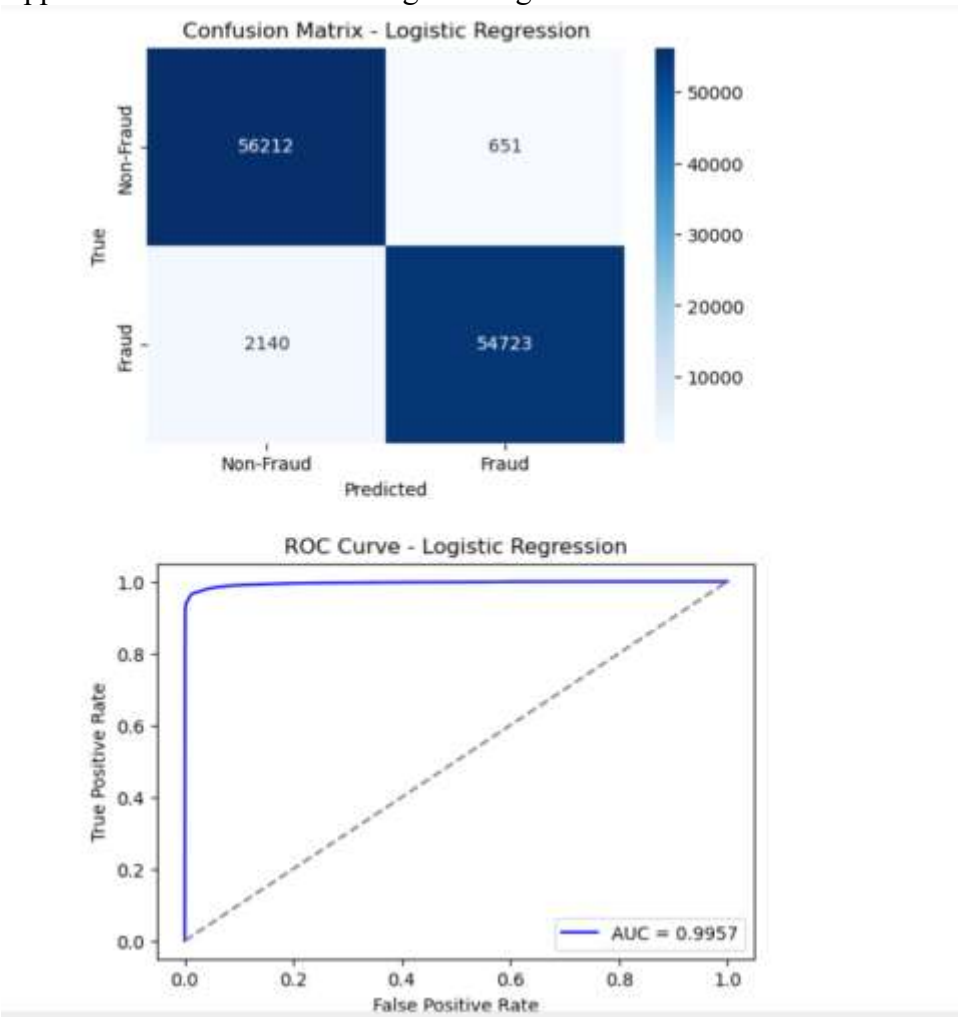
The histograms for each numeric column visualize the distribution of values, helping identify skewness, outliers, and feature scaling needs.

Appendix G: Boxplot for 'Amount' feature



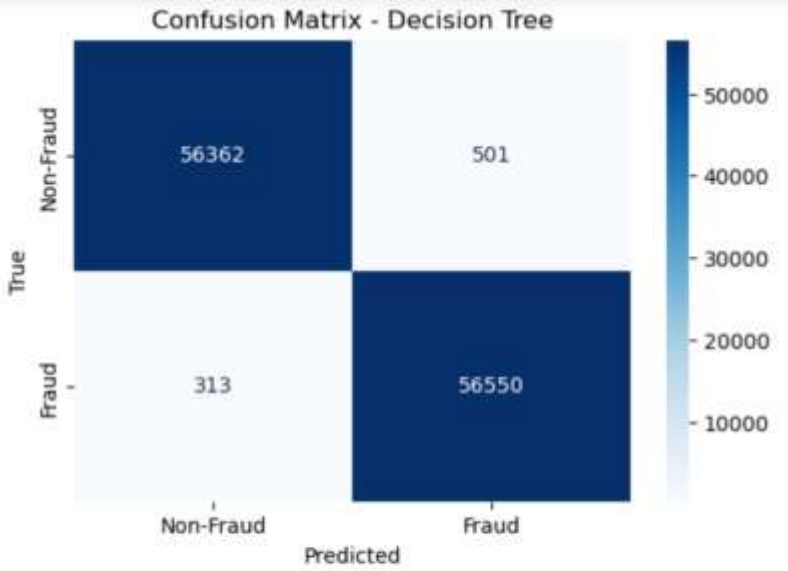
The boxplot for the 'Amount' feature helps identify outliers by displaying the spread and potential extreme values in the data.

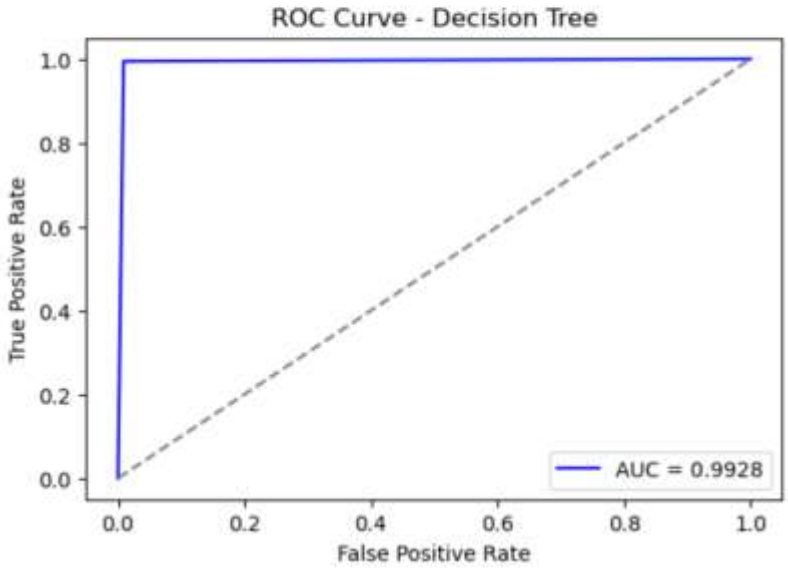
Appendix H: Evaluation for Logistic Regression



The evaluation of Logistic Regression with ROC and Confusion Matrix assesses the model's classification performance, visualizing its true vs. false positive/negative rates. The ROC curve shows the trade-off between sensitivity and specificity, while the confusion matrix provides insight into the true and false predictions.

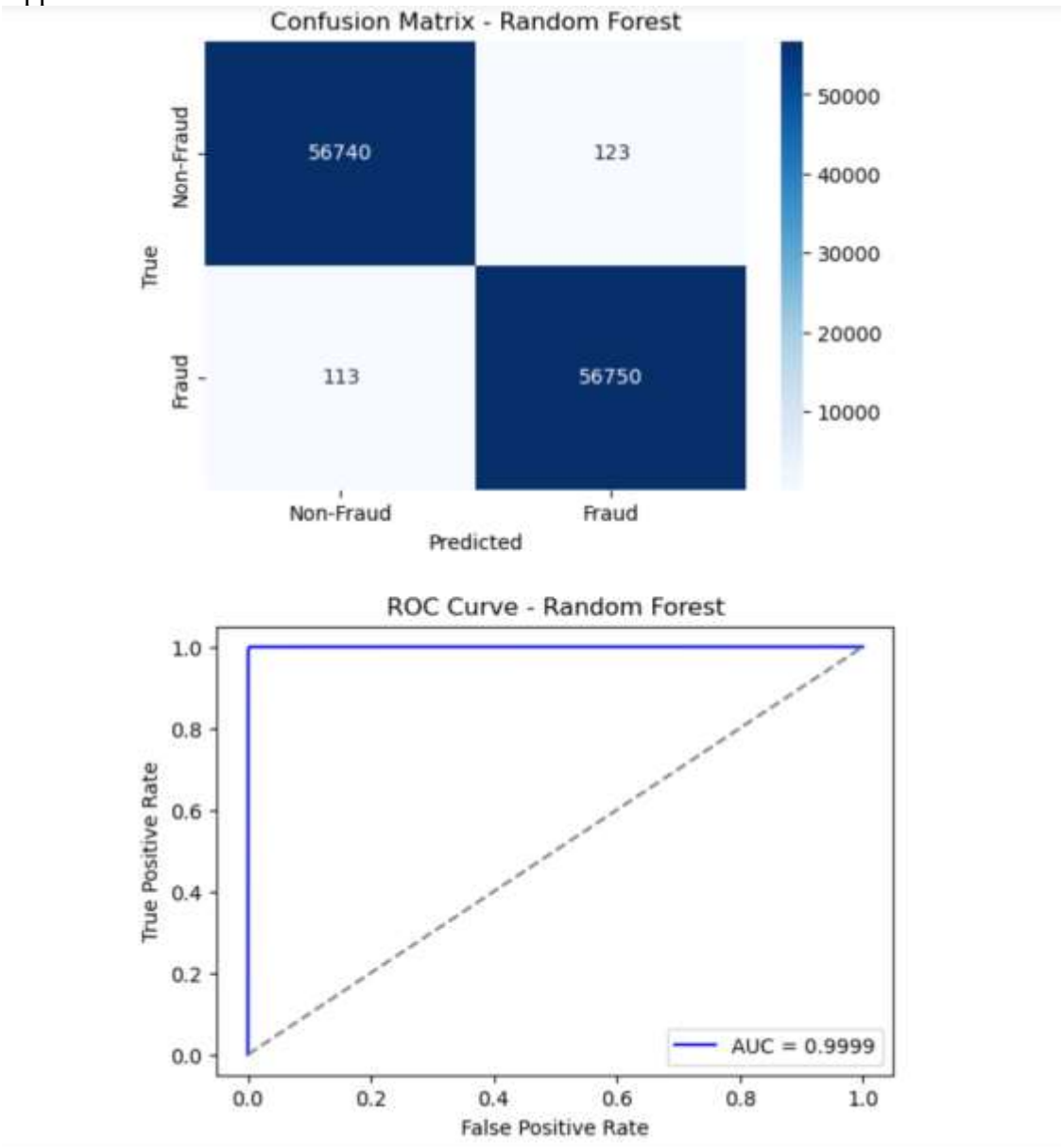
Appendix I: Evaluation of Decision Tree





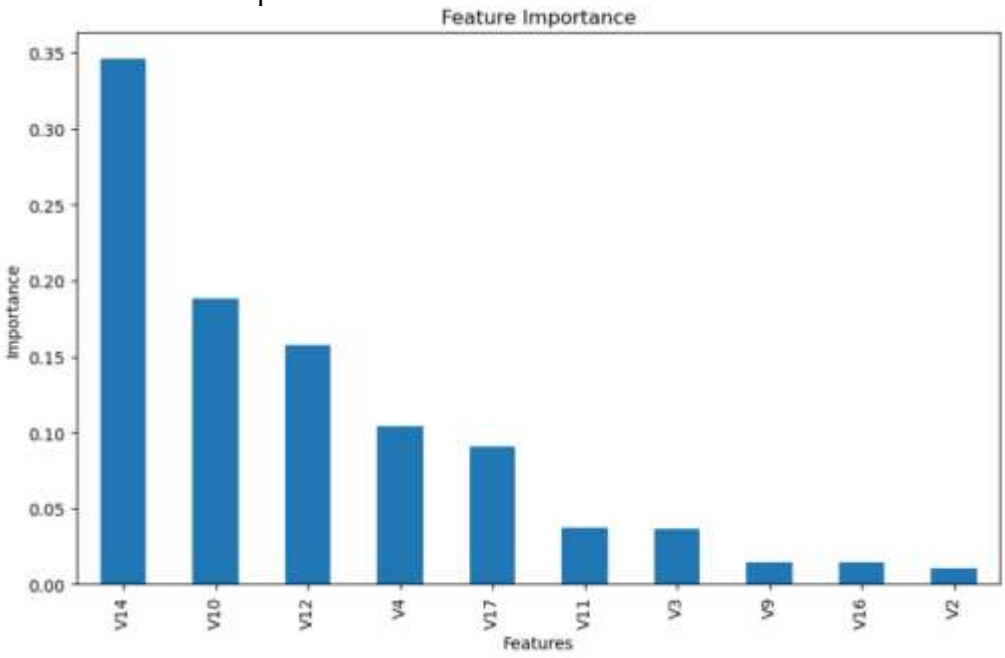
The evaluation of the Decision Tree model involves assessing its classification performance, including accuracy, precision, recall, and F1-score. The confusion matrix and ROC curve help visualize the model's ability to distinguish between fraudulent and non-fraudulent transactions. The decision tree model's interpretability is also a key feature, showing how decisions are made based on the data.

Appendix J: Evaluation for Random Forest



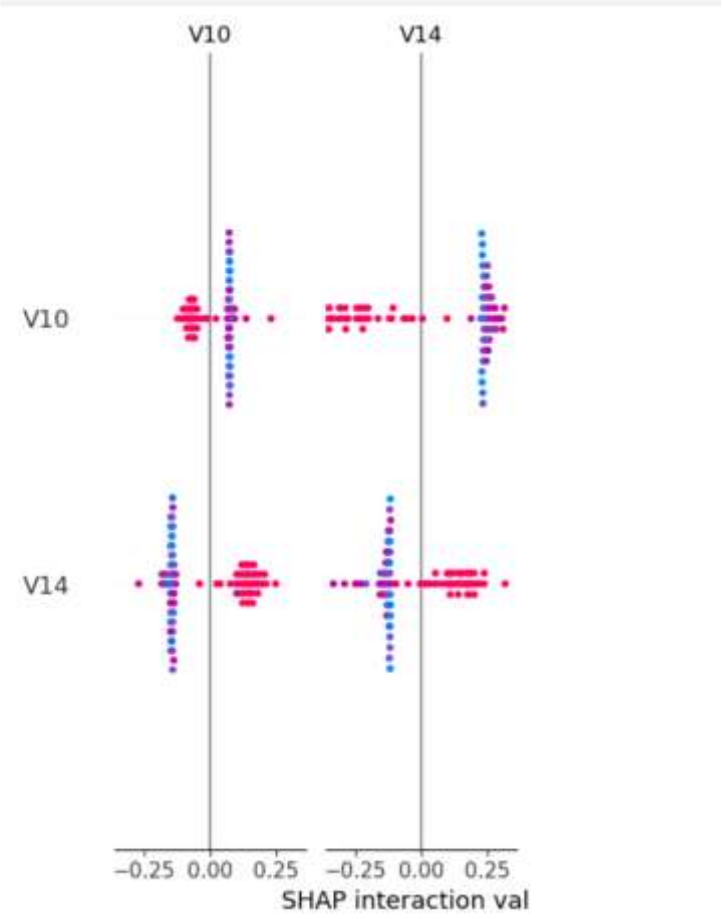
The evaluation of the Random Forest model involves analyzing its performance using accuracy, precision, recall, F1-score, and AUC metrics. The confusion matrix and ROC curve further illustrate its ability to distinguish between fraudulent and non-fraudulent transactions. Random Forest, being an ensemble method, reduces overfitting by averaging multiple decision trees, enhancing its robustness in fraud detection tasks.

Appendix K: Feature importance for Random Forest



Feature importance for Random Forest" highlights the most influential features in predicting fraudulent transactions. This analysis helps identify which variables, such as transaction amount or time, contribute the most to the model's decisions. By evaluating feature importance, we gain insight into the key drivers of fraud detection, which can guide further model refinement and feature selection.

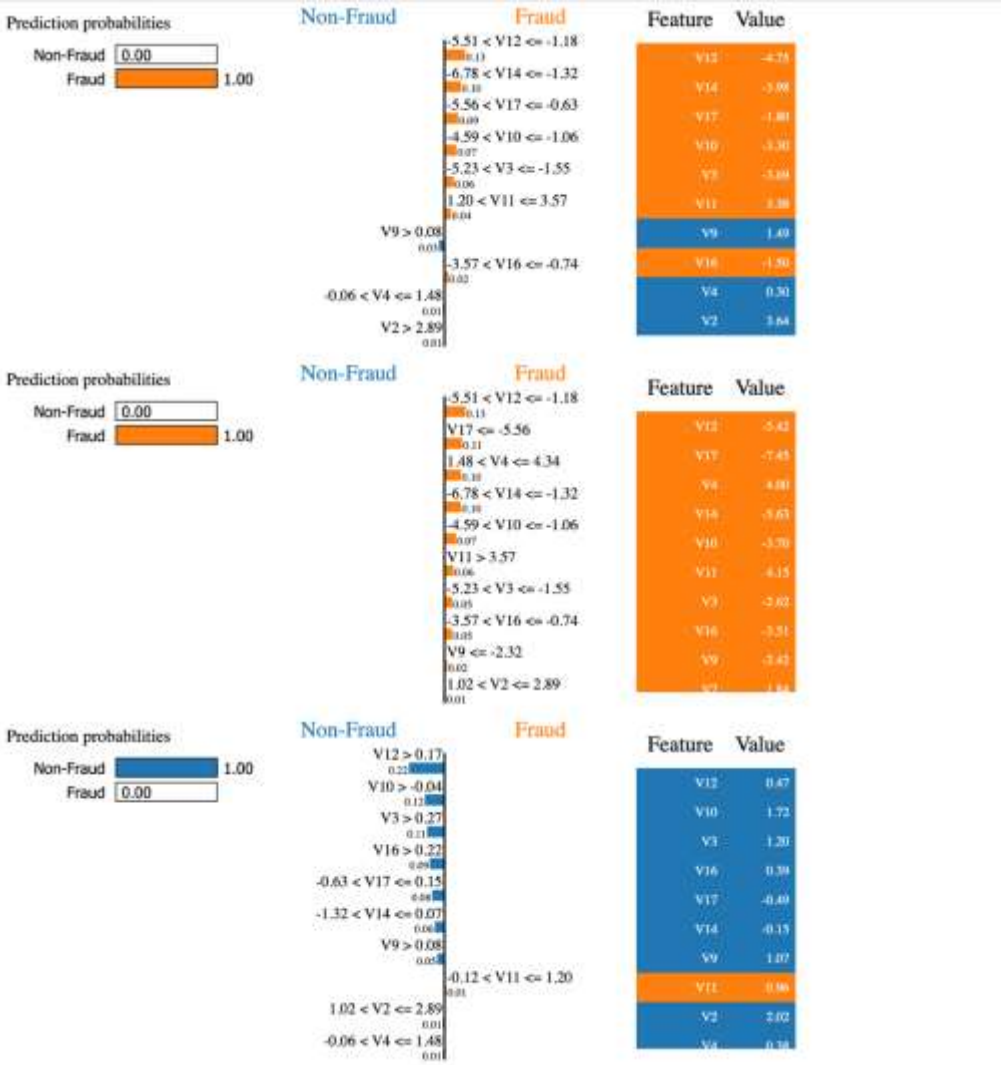
Appendix L: SHAP values using TreeExplainer



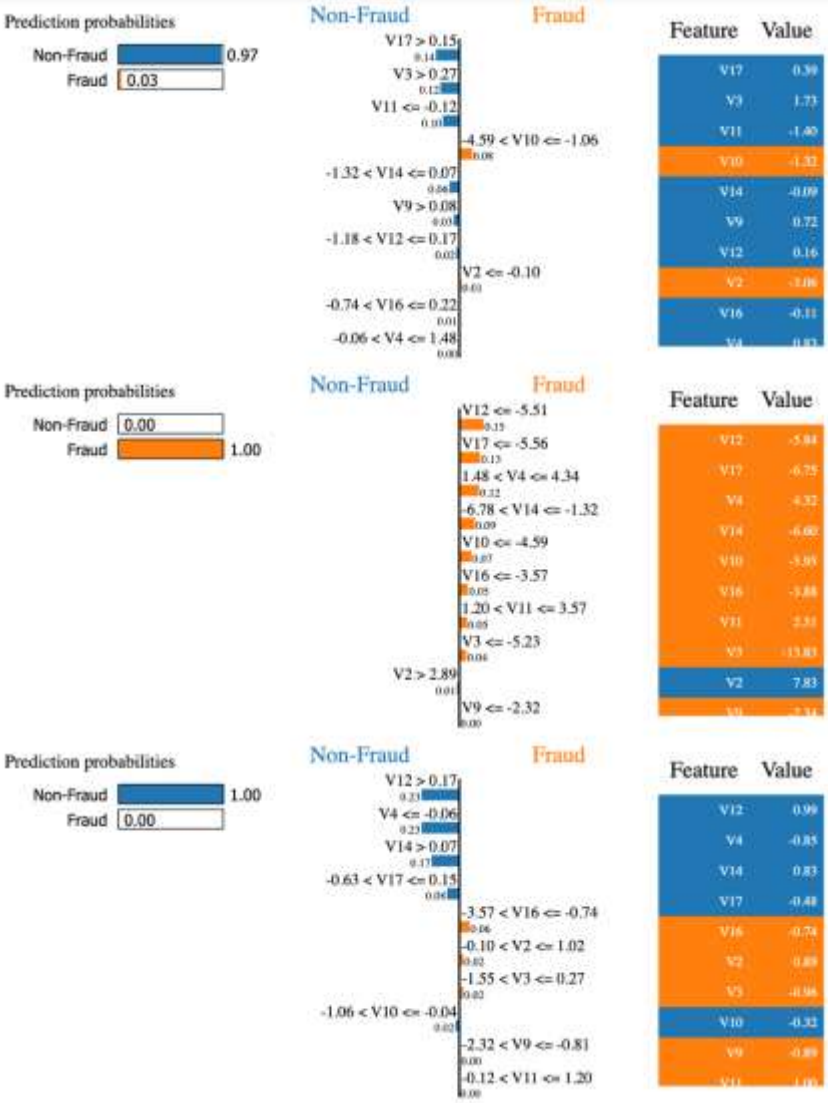
SHAP values using TreeExplainer" provide a detailed explanation of how each feature contributes to individual predictions in tree-based models. SHAP (SHapley Additive exPlanations) values quantify the impact of each feature on a specific prediction,

offering transparency and interpretability. Using TreeExplainer, which is optimized for tree models like Random Forest and Decision Trees, helps visualize and understand model decisions, making it easier to explain why a particular transaction was classified as fraudulent or legitimate. This improves model trust and transparency, which is crucial in sensitive financial applications.

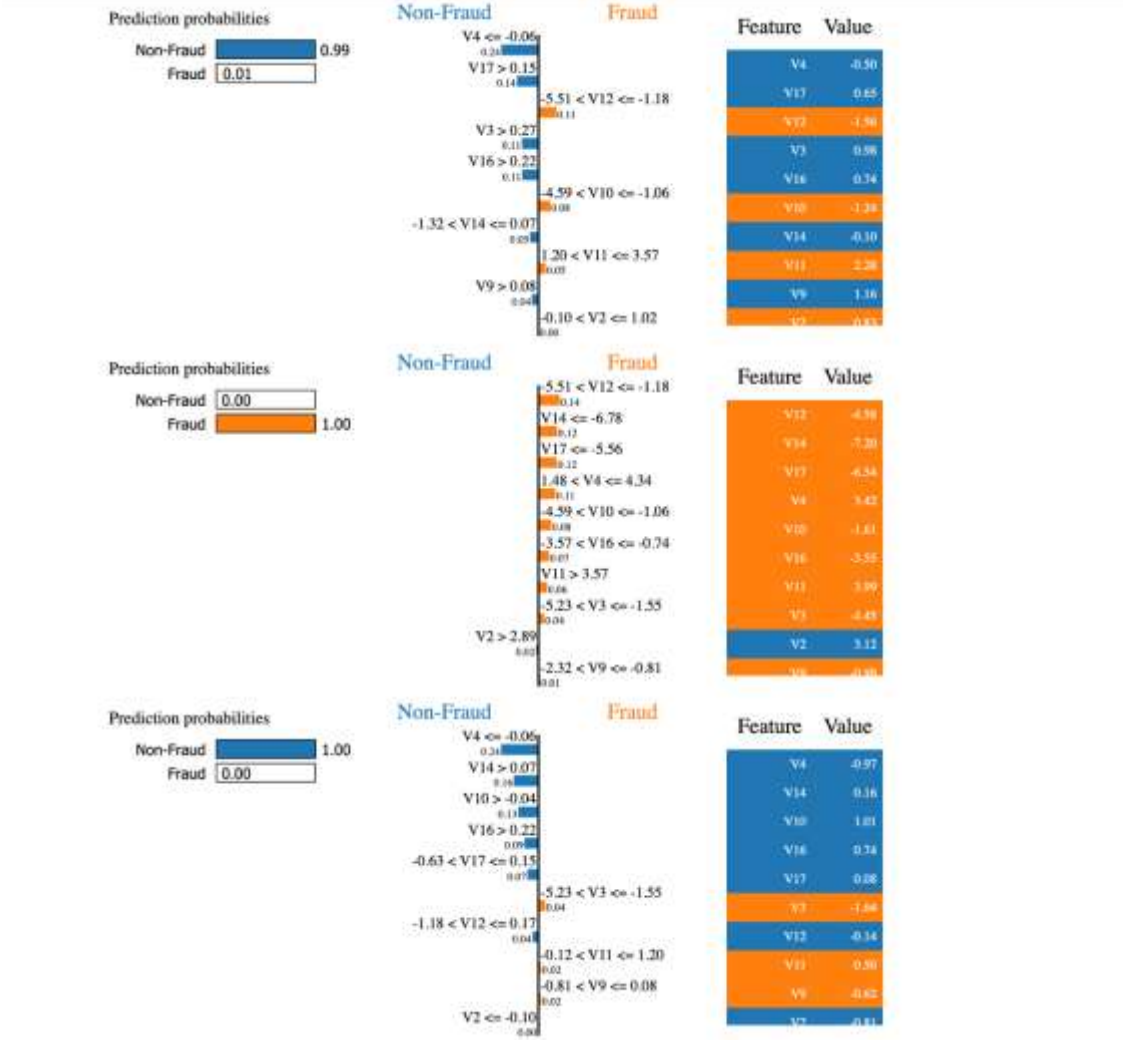
Appendix N: Prediction probabilities (for 10 instances)
First 3 instances:



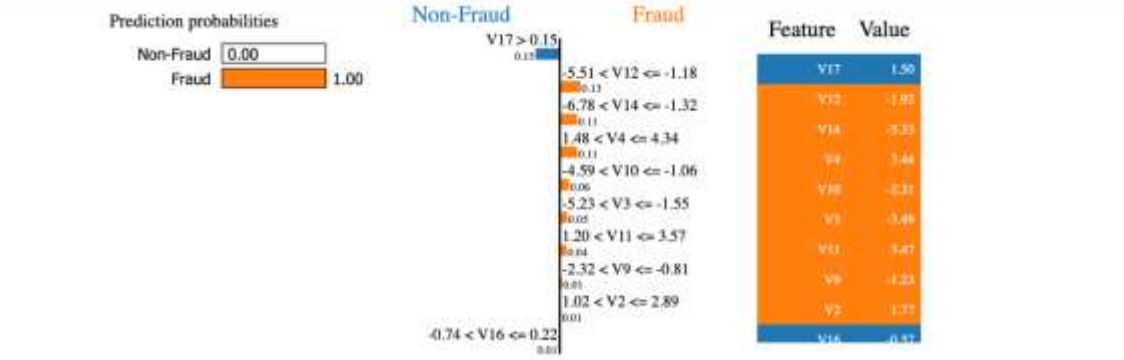
Second (3 instances)



Third (3 instances)

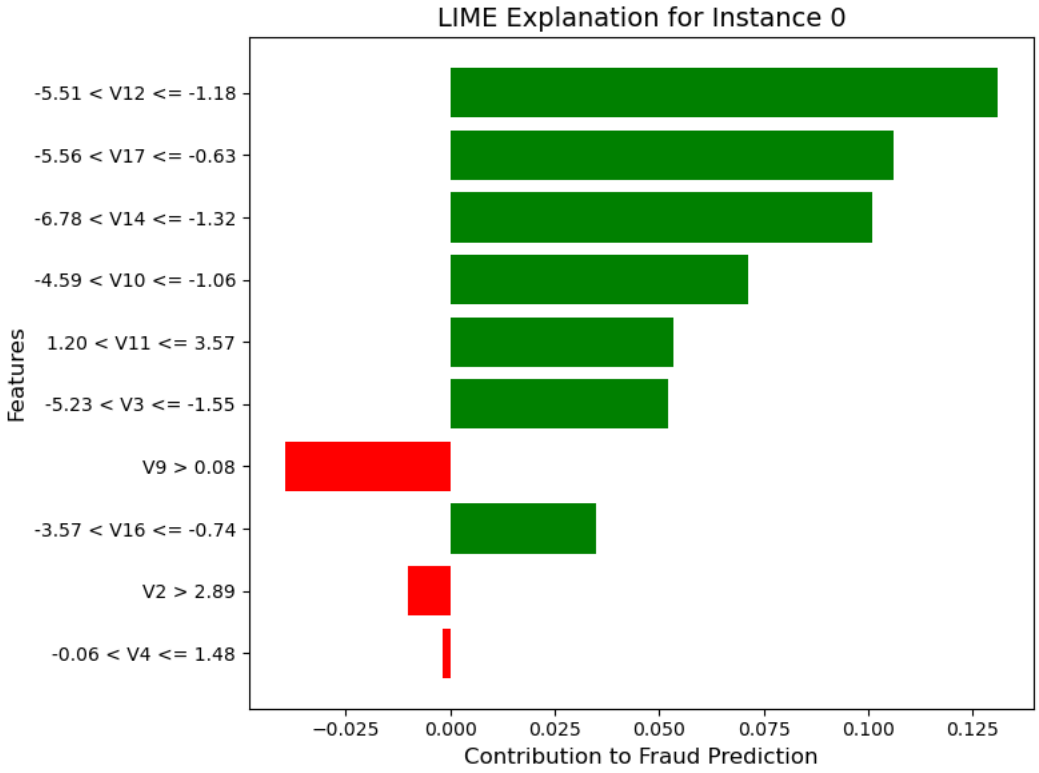


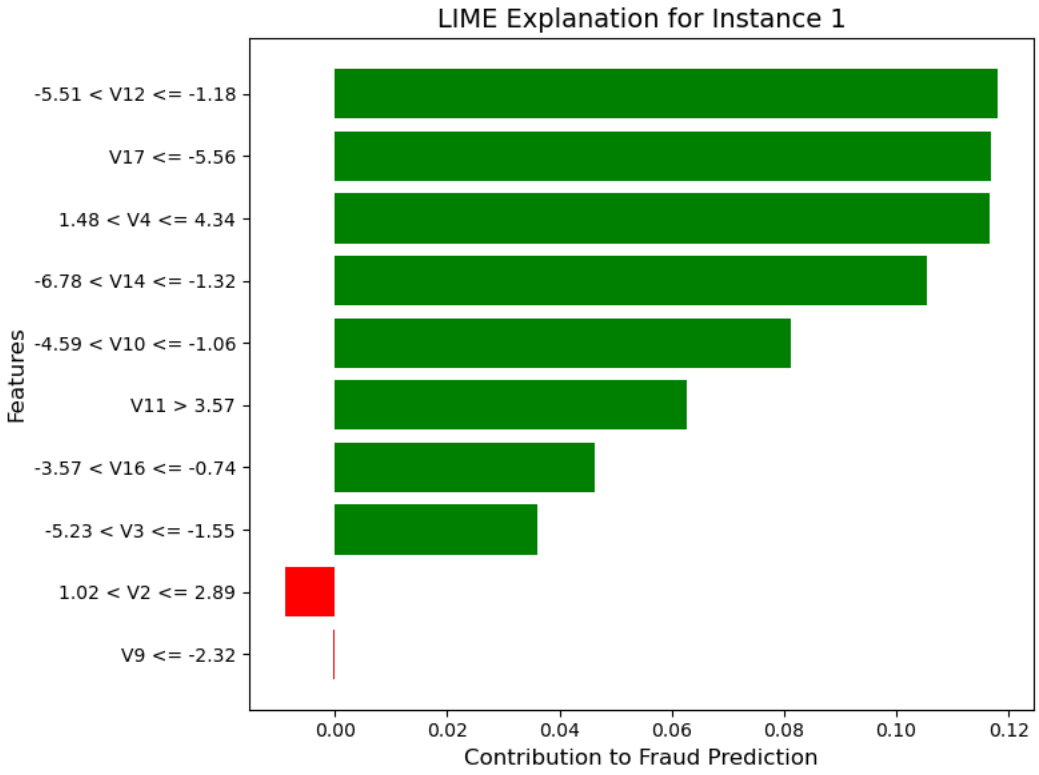
The 10th instances

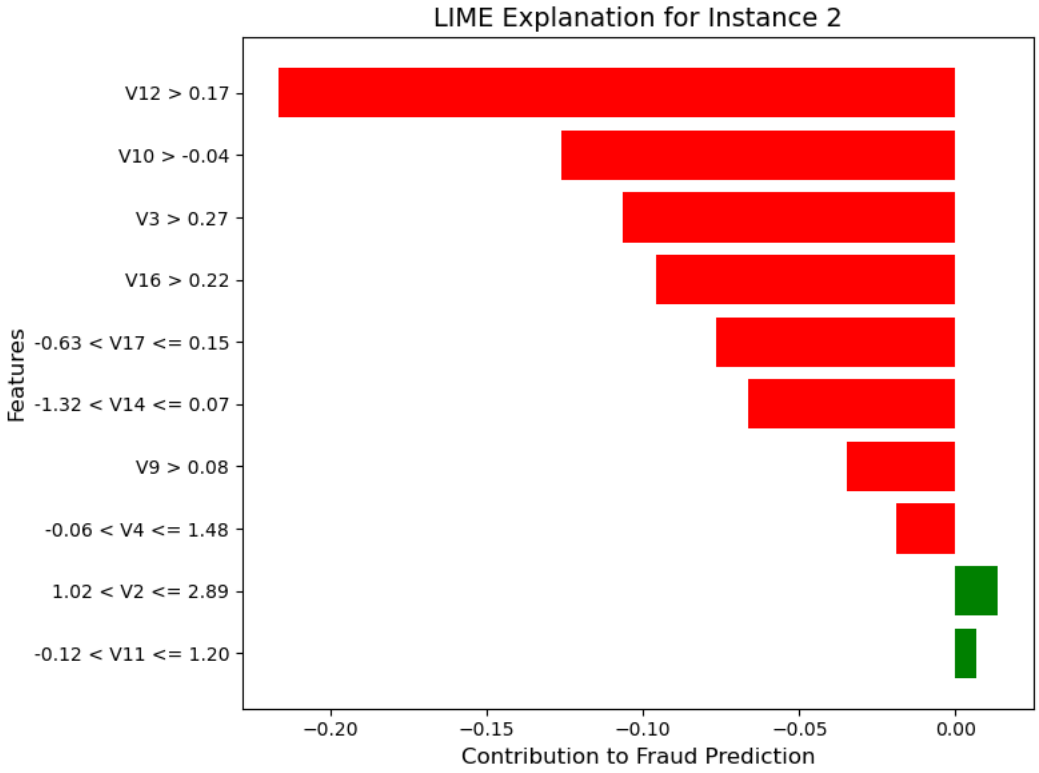


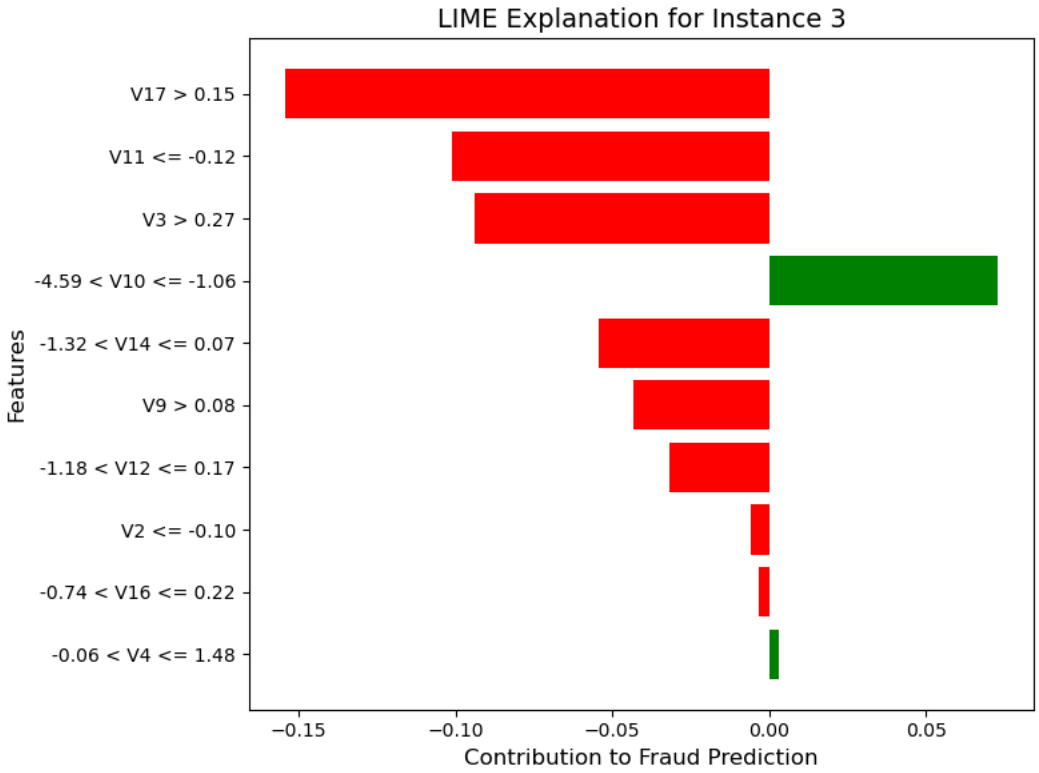
Prediction probabilities" refer to the likelihood that a given transaction is fraudulent or legitimate, as predicted by the model. These probabilities are generated by the machine learning model for each instance, typically ranging from 0 (non-fraudulent) to 1 (fraudulent). By analyzing the prediction probabilities, we can better understand the model's confidence in its decisions. This information can be used to set thresholds for fraud detection, helping organizations make more informed decisions, such as flagging transactions that exceed a certain probability of being

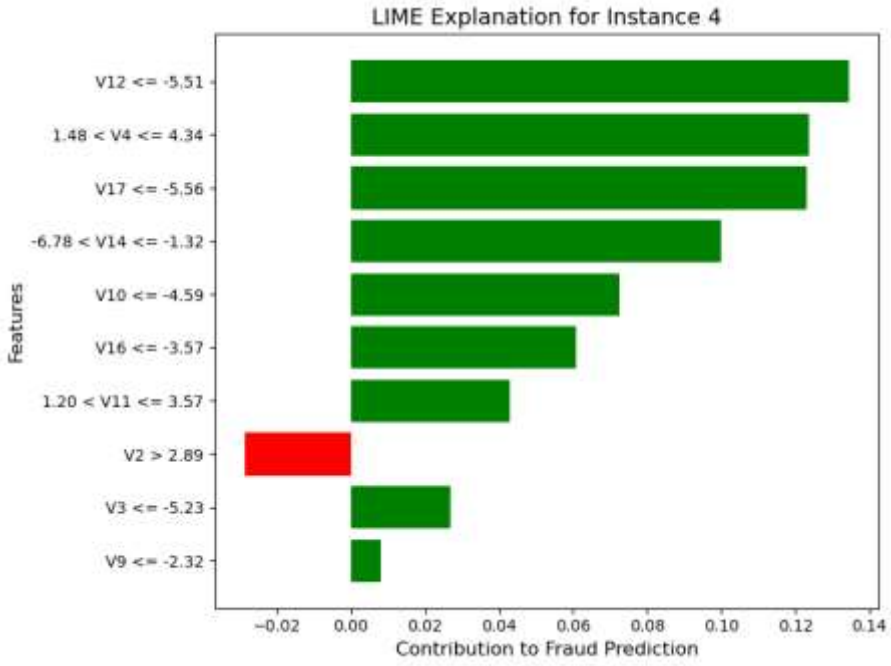
Appendix M: Lime Explanation For five instances


LIME Explanation for Instance 0
-5.51 < V12 <= -1.18
-5.56 < V17 <= -0.63
-6.78 < V14 <= -1.32
-4.59 < V10 <= -1.06
1.20 < V11 <= 3.57
-5.23 < V3 <= -1.55
V9 > 0.08
-3.57 < V16 <= -0.74
V2 > 2.89
-0.06 < V4 <= 1.48
Features
−0.025
0.000
0.025
0.050
0.075
0.100
0.125
Contribution to Fraud Prediction


LIME Explanation for Instance 1
-5.51 < V12 <= -1.18
V17 <= -5.56
1.48 < V4 <= 4.34
-6.78 < V14 <= -1.32
-4.59 < V10 <= -1.06
V11 > 3.57
-3.57 < V16 <= -0.74
-5.23 < V3 <= -1.55
1.02 < V2 <= 2.89
V9 <= -2.32
Features
0.00
0.02
0.04
0.06
0.08
0.10
0.12
Contribution to Fraud Prediction


LIME Explanation for Instance 2
V12 > 0.17
V10 > -0.04
V3 > 0.27
V16 > 0.22
-0.63 < V17 <= 0.15
-1.32 < V14 <= 0.07
V9 > 0.08
-0.06 < V4 <= 1.48
1.02 < V2 <= 2.89
-0.12 < V11 <= 1.20
Features
−0.20
−0.15
−0.10
−0.05
0.00
Contribution to Fraud Prediction

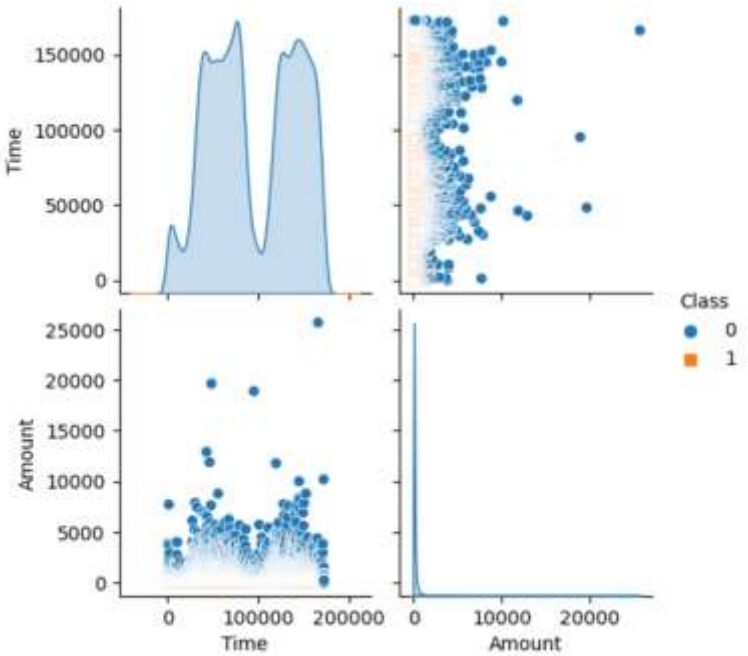

LIME Explanation for Instance 3
V17 > 0.15
V11 <= -0.12
V3 > 0.27
-4.59 < V10 <= -1.06
-1.32 < V14 <= 0.07
V9 > 0.08
-1.18 < V12 <= 0.17
V2 <= -0.10
-0.74 < V16 <= 0.22
-0.06 < V4 <= 1.48
Features
−0.15
−0.10
−0.05
0.00
0.05
Contribution to Fraud Prediction



For each instance:

- A **horizontal bar chart** is displayed showing the contributions of various features to the model's prediction.
- **Positive contributions (green)**: Features that increase the likelihood of the predicted class (Fraud).
- **Negative contributions (red)**: Features that decrease the likelihood of the predicted class.
- The y-axis lists the features, sorted by their impact on the prediction (largest impact at the top).

Appendix O: Visualizing Relationships with Pairplot



This visualizes pairwise relationships between features (Time, Amount, Class) using a pairplot to analyze fraud patterns.