Credit Card Fraud Detection Final Project Presentation

Xinyi Wang

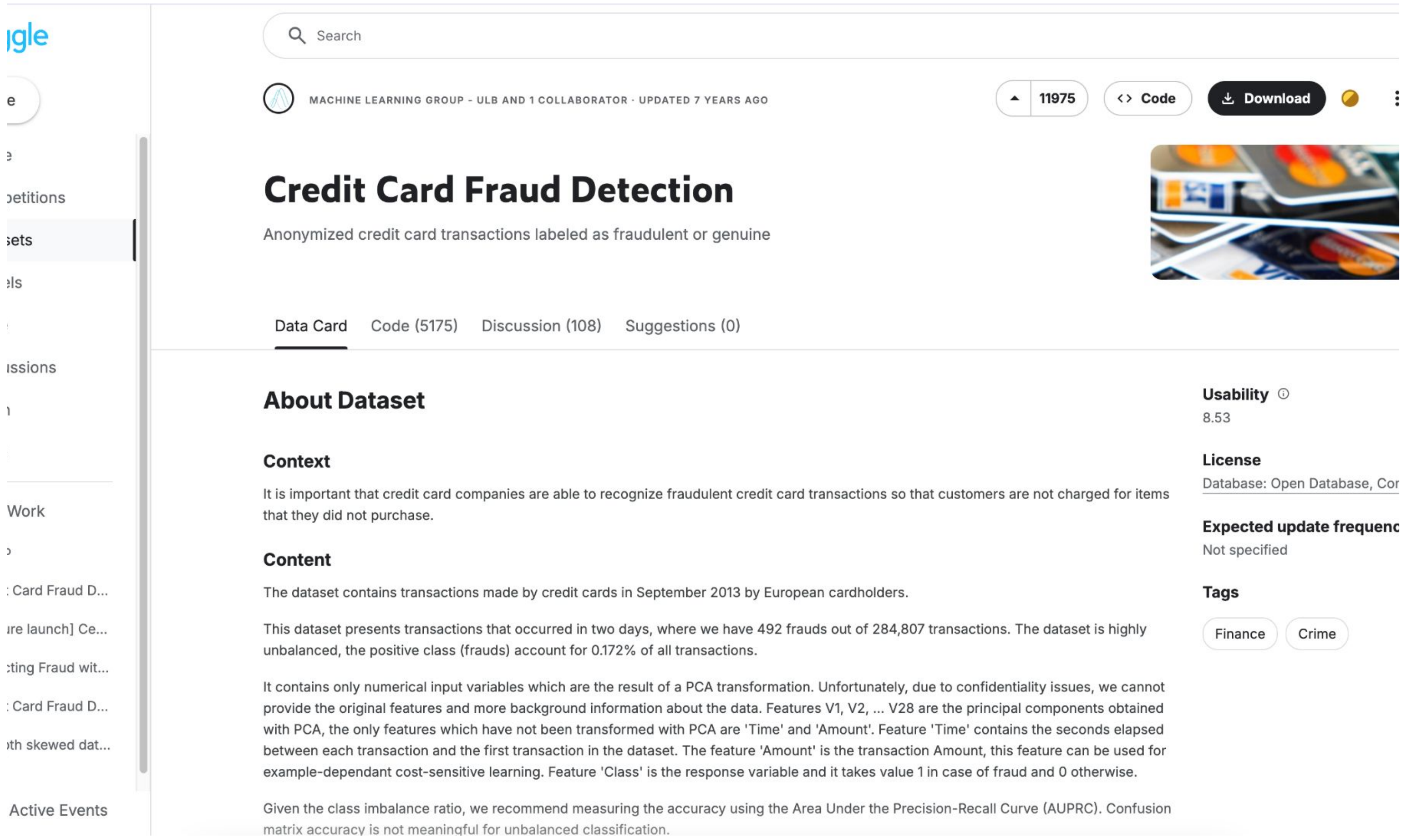
Overview

This project explores advanced data mining techniques to address severe class imbalance, high computational complexity, and evolving fraud patterns.

Credit card fraud detection remains a critical challenge in financial security, with global fraud losses exceeding billions of dollars annually.

We employ data preprocessing techniques such as feature scaling and Synthetic Minority Over-sampling Technique (SMOTE) to handle imbalanced data. Two machine learning models, **Random Forest** and **Neural Networks**, are implemented and evaluated based on precision, recall, and F1-score.

Dataset Characteristics



We use the Credit Card Fraud Detection Dataset from Kaggle, **containing 284,807 transactions with 492 fraud cases**. The dataset includes 28 PCA-transformed numerical features, a time variable, and the transaction amount.

https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data

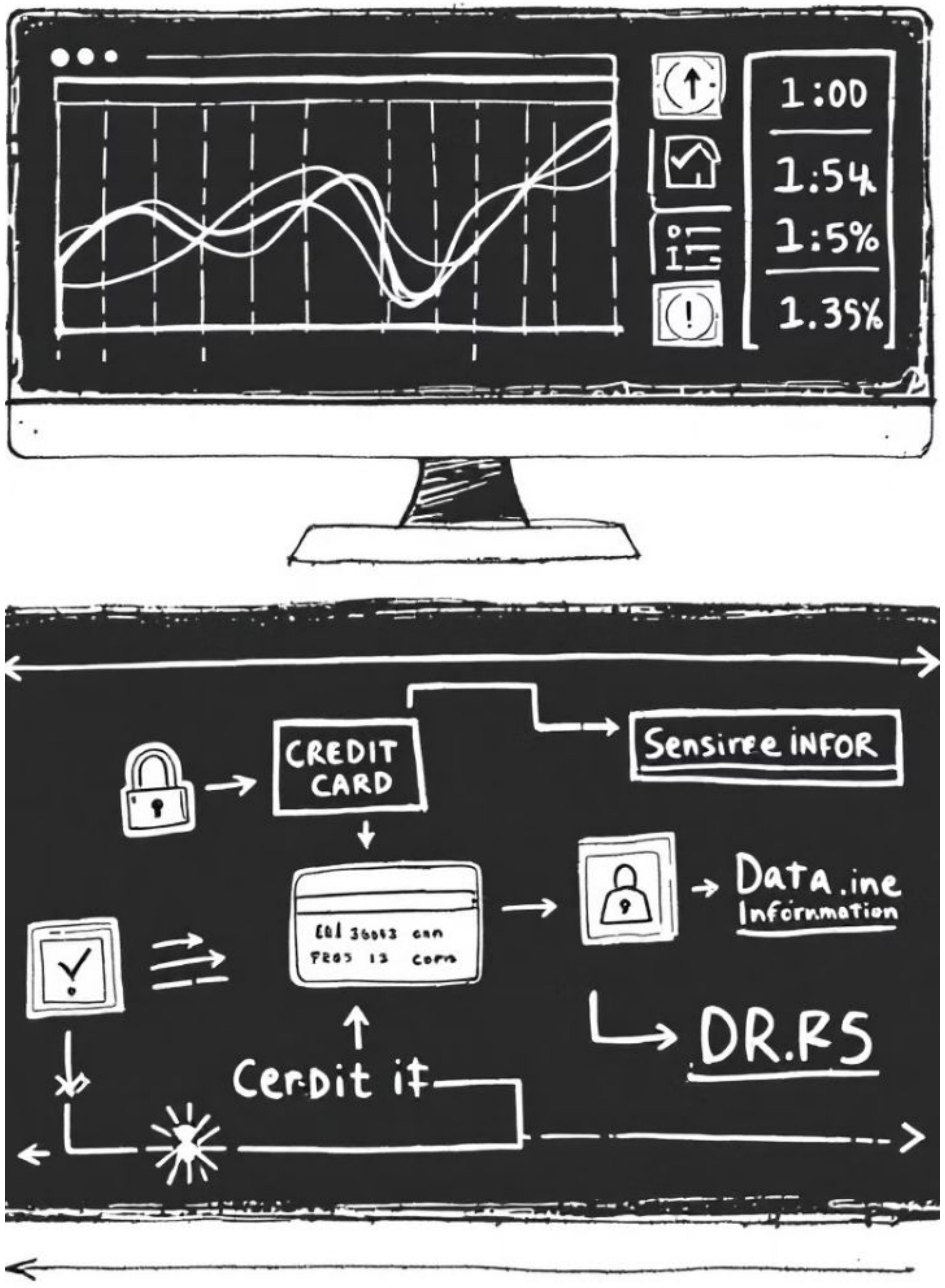
Problem Statement and Research Question

Problem Statement

Credit card fraud detection is challenging due to the imbalanced nature of fraud cases. In our dataset of 284,807 transactions, only 492 (0.17%) are fraudulent. Traditional classifiers struggle to accurately identify fraudulent transactions due to their tendency to favor the majority class.

Research Question

How does the application of SMOTE and deep learning techniques impact the accuracy and recall of credit card fraud detection models compared to traditional machine learning approaches? This question allows us to evaluate the effectiveness of our methodology.



Background and Related Work

1 Traditional Approaches

Traditional approaches include rule-based methods and supervised learning algorithms such as logistic regression, decision trees, and support vector machines. However, these methods struggle with highly imbalanced datasets.

2 SMOTE Technique

Synthetic data generation techniques such as the Synthetic Minority Oversampling Technique (SMOTE) have been introduced to balance class distributions. SMOTE has been widely applied in fraud detection, improving recall without significantly compromising precision.

3 Ensemble and Deep Learning

Ensemble learning methods such as Random Forest and XGBoost have demonstrated strong performance. Recently, deep learning techniques, including neural networks, have been explored for fraud detection due to their ability to capture complex feature relationships.

Project Implement Workflow

Problem Statement

Develop a machine learning model to accurately detect credit card fraud.

Data Processing

Clean and transform the dataset, including feature scaling, SMOTE implementation, and outlier detection.

Model Building

Train and evaluate several machine learning models, including Random Forest, Random Forest with SHAP analysis, and a Neural Network model.

Data Processing

1. **Feature Scaling improved model performance**
2. **SMOTE helped address class imbalance**
3. **Outlier detection identified problematic data points**

Next Steps: Model Building

1. **Deploy the Random Forest model in production**
2. **Explore the Random Forest with SHAP analysis model**
3. **Train a Neural Network model on more representative data**

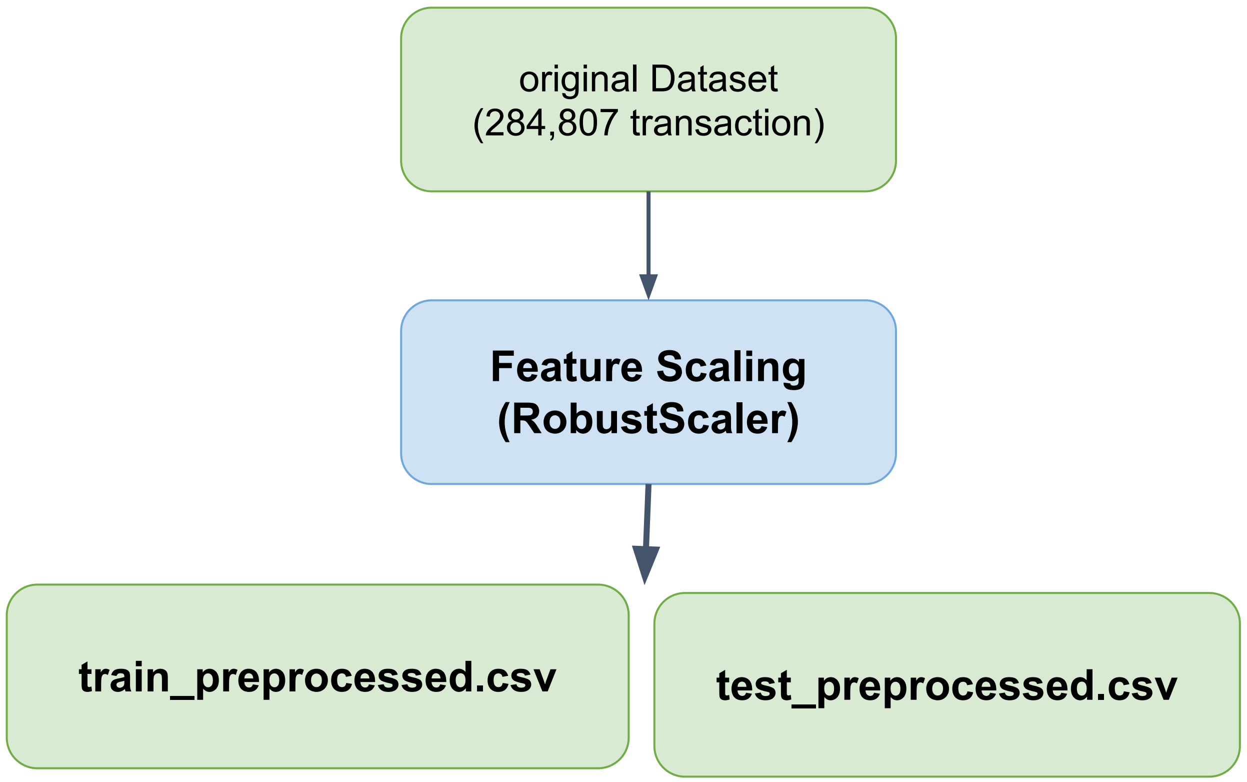
Data Processing

- Feature Scaling
- SMOTE for handling class imbalance
- Outlier Detection

Data Processing
Step 1: Feature Scaling

Co-lab Link

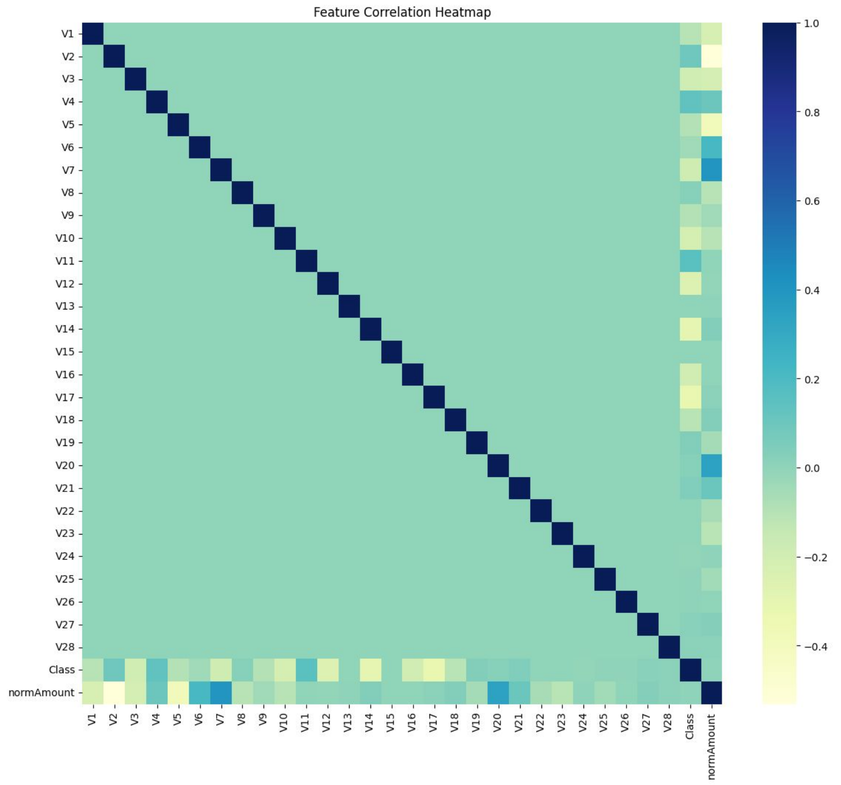
We apply Robust Scaler to normalize the transaction amount, as it is effective in handling outliers while preserving essential distribution properties. The time variable is dropped, as prior research suggests it does not contribute significantly to fraud detection.



Step 1 Output : Feature Scaling

Correlation Analysis

We generate a heatmap to visualize the correlation between features. This analysis helps us decide whether dimensionality reduction techniques should be applied to enhance model efficiency.



Feature Pair	Correlation Coefficient	Observations
V1 - V28	Low	Indicates independence, beneficial for modeling.
V2, V4, V11, V19	High	Important indicators for identifying fraudulent transactions.
normAmount - V7, V20	Moderate	Shows some degree of correlation, significant for the model.

What is SMOTE?

1

What is SMOTE?

Synthetic Minority Over-sampling Technique addresses class imbalance by generating synthetic examples for minority classes.

2

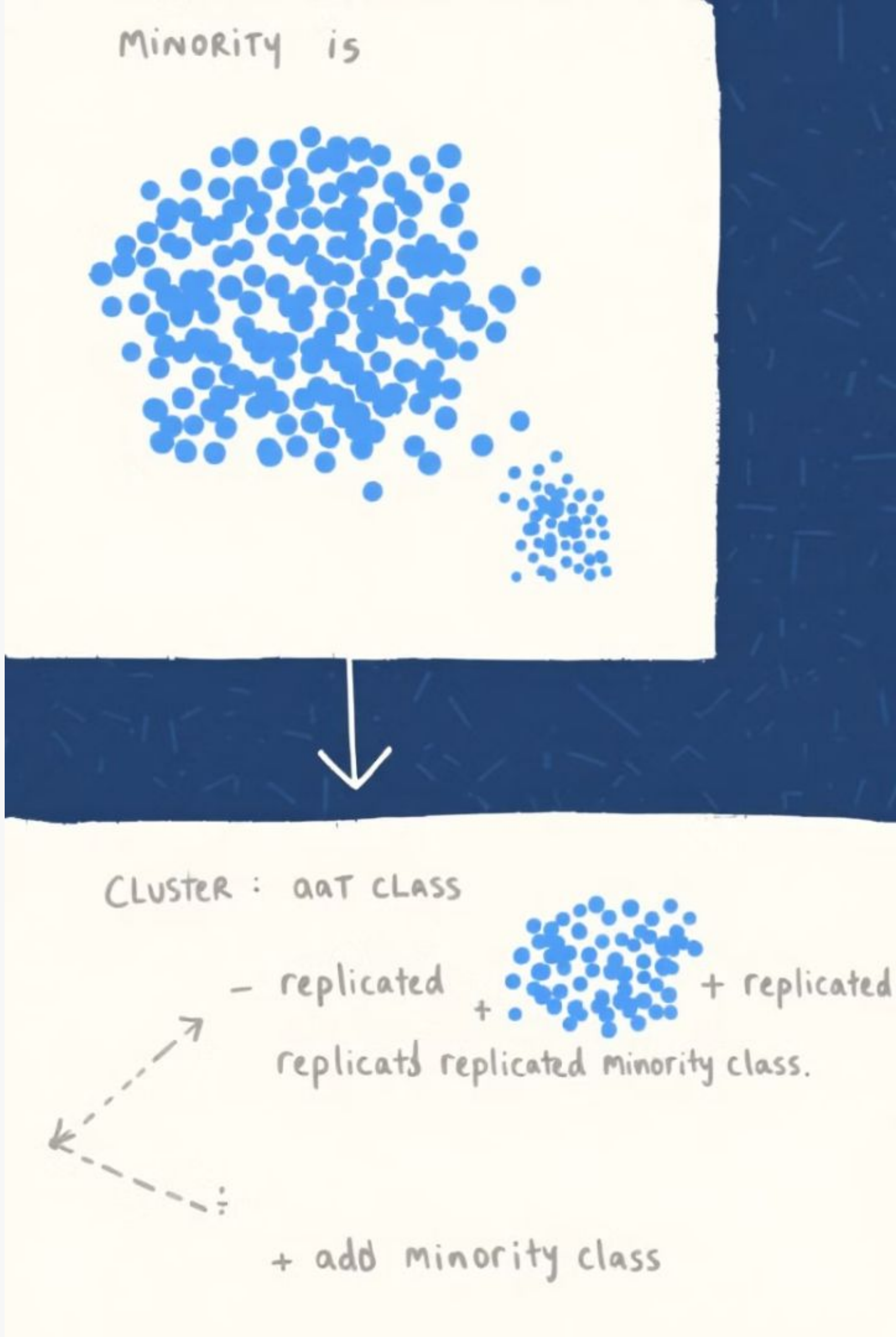
The Process

Identify minority instances, select random samples, find k-nearest neighbors, and create synthetic instances along connecting lines.

3

Impact on Dataset

Transformed dataset from 0.17% fraudulent transactions to a balanced 50-50 distribution.



Data Processing

Step 2: SMOTE for handling class imbalance

[Co-lab Link]

Identify the Problem

The initial dataset exhibited a critical class imbalance, with fraudulent transactions comprising only 0.17% of the total observations, making it difficult for models to learn fraud patterns.

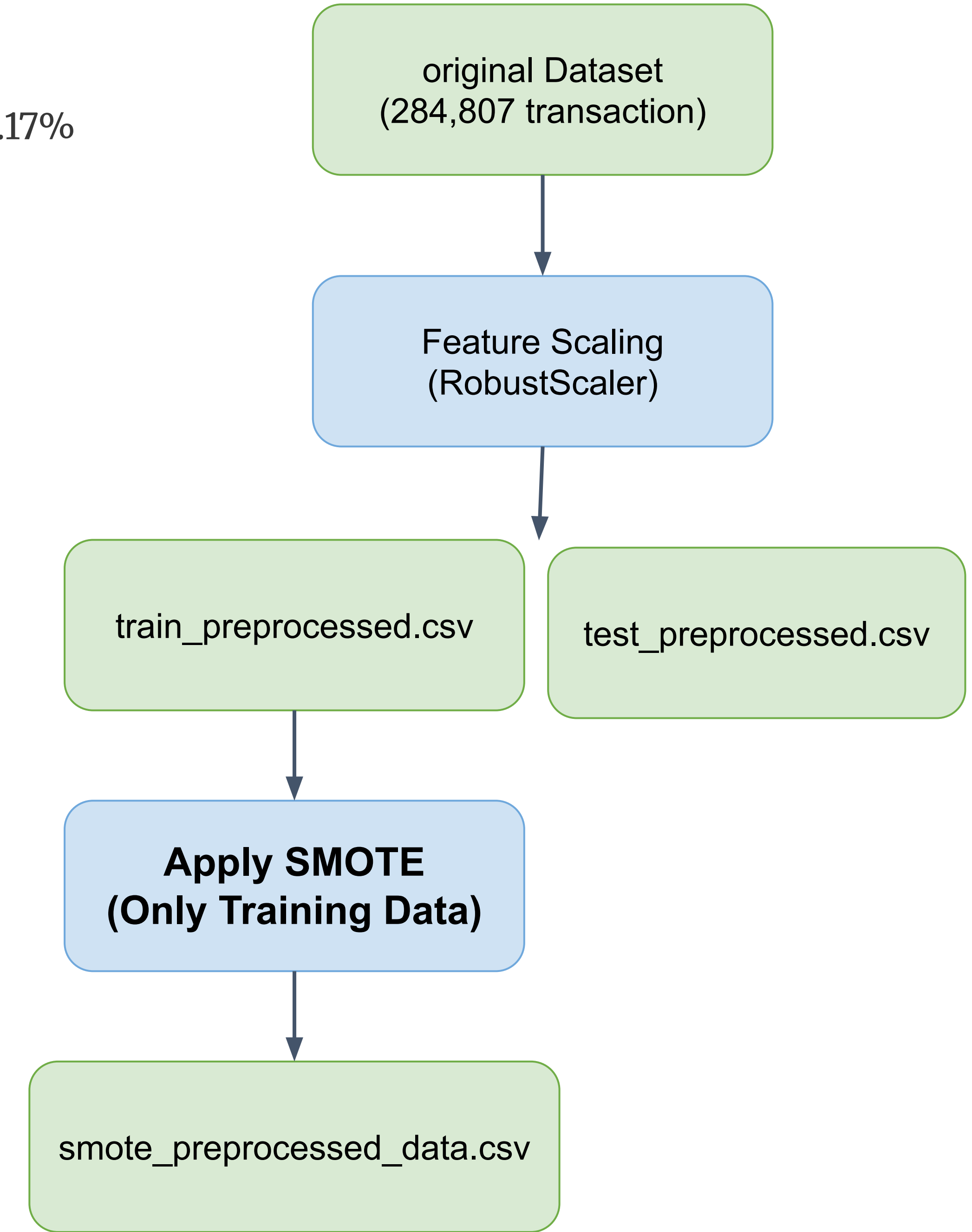
Apply SMOTE

SMOTE works by identifying minority class instances, selecting k-nearest neighbors for each minority class sample, and generating synthetic examples along the line segments connecting them.

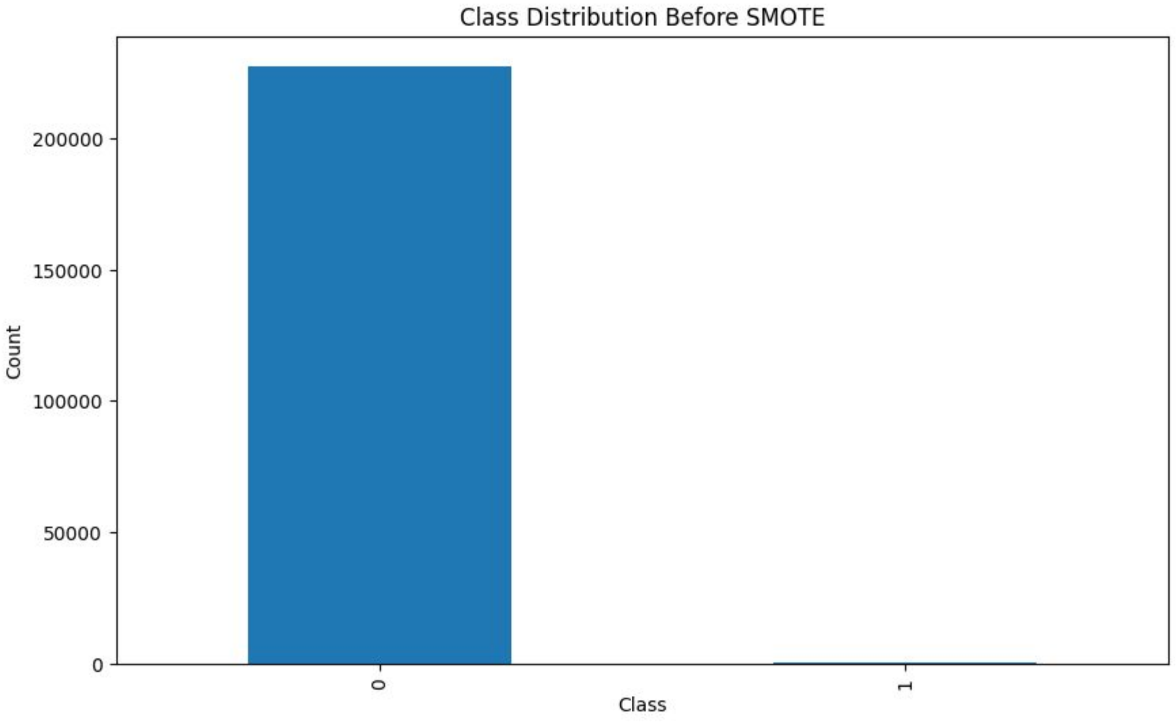
Transform Class Distribution

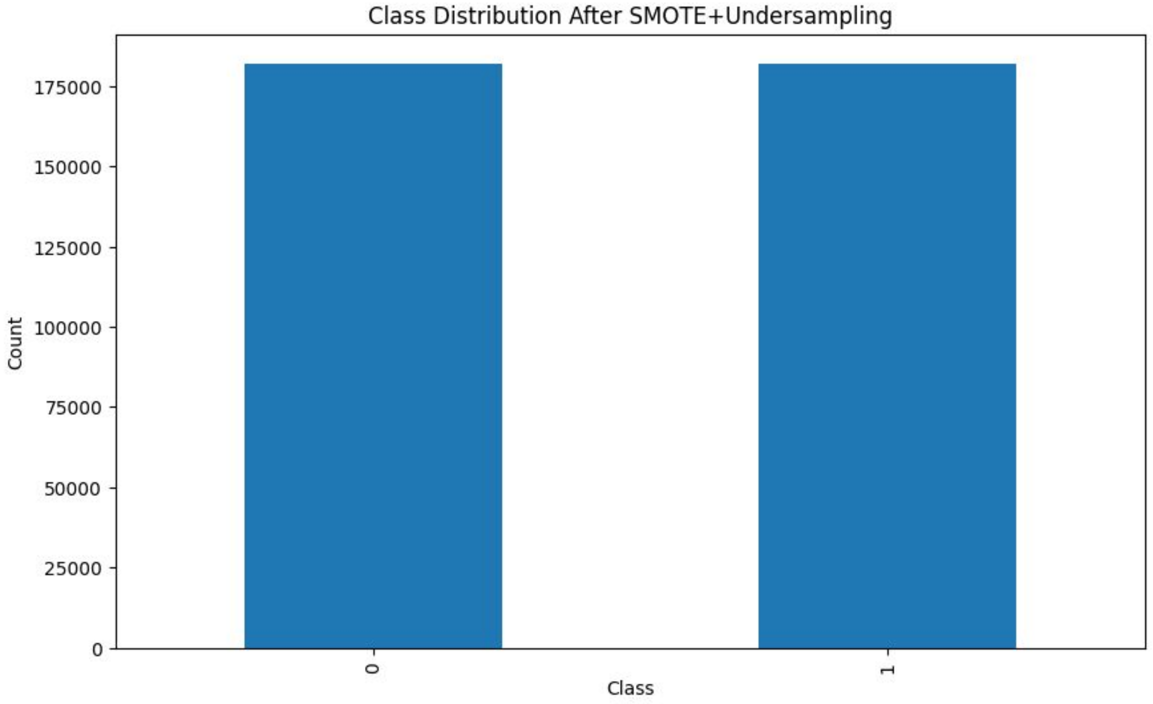
Before SMOTE: 99.827% normal transactions and 0.00173% fraudulent transactions. After SMOTE: 50% representation for each class, creating a balanced approach.

Strategy	Description
Standard SMOTE	Generates synthetic minority class samples.
SMOTE with Undersampling	Combines synthetic oversampling with random undersampling of the majority class.



Step 2 Output: SMOTE for handling class imbalance





Strategy	Description
Standard SMOTE	Generates synthetic minority class samples.
SMOTE with Undersampling	Combines synthetic oversampling with random undersampling of the majority class.

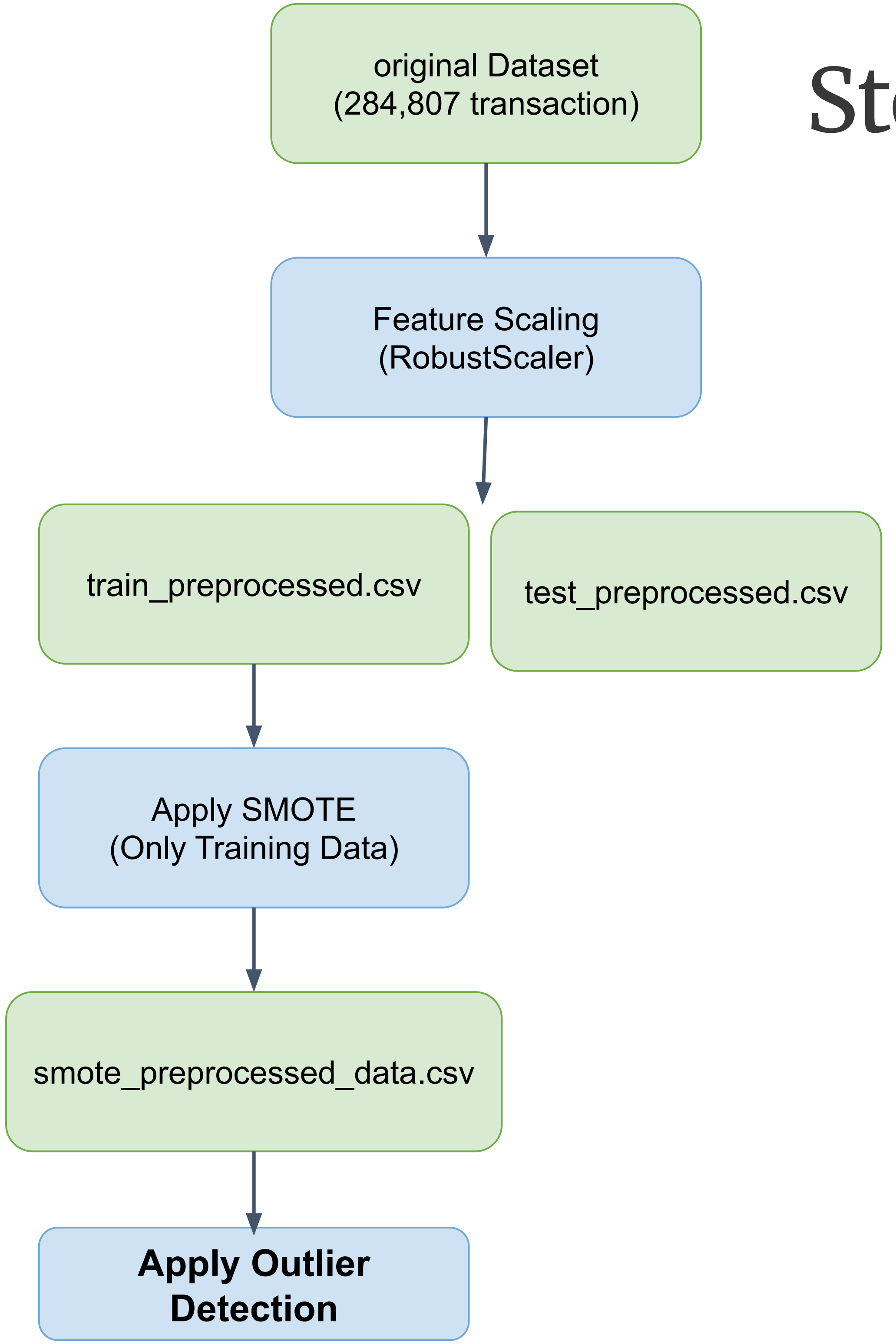
```
--- SMOTE Analysis ---
Original Shape: (227845, 29)
Resampled Shape: (363922, 29)

Original Distribution:
Class
0    0.998271
1    0.001729
Name: proportion, dtype: float64
```

```
Resampled Distribution:
Class
0    0.5
1    0.5
Name: proportion, dtype: float64
```

Data Processing

[Co-lab Link]

Step 3: Outlier Detection



Isolation Forest Algorithm

We employed the Isolation Forest algorithm to identify outliers within the credit card transaction dataset, revealing 18,123 data points (4.98%) as outliers from a total of 363,922 samples.

Benefits of Outlier Detection

The identification of anomalies aims to enhance model robustness, mitigate potential statistical distortions, and refine the predictive capability of the fraud detection system.

Data Cleaning Impact

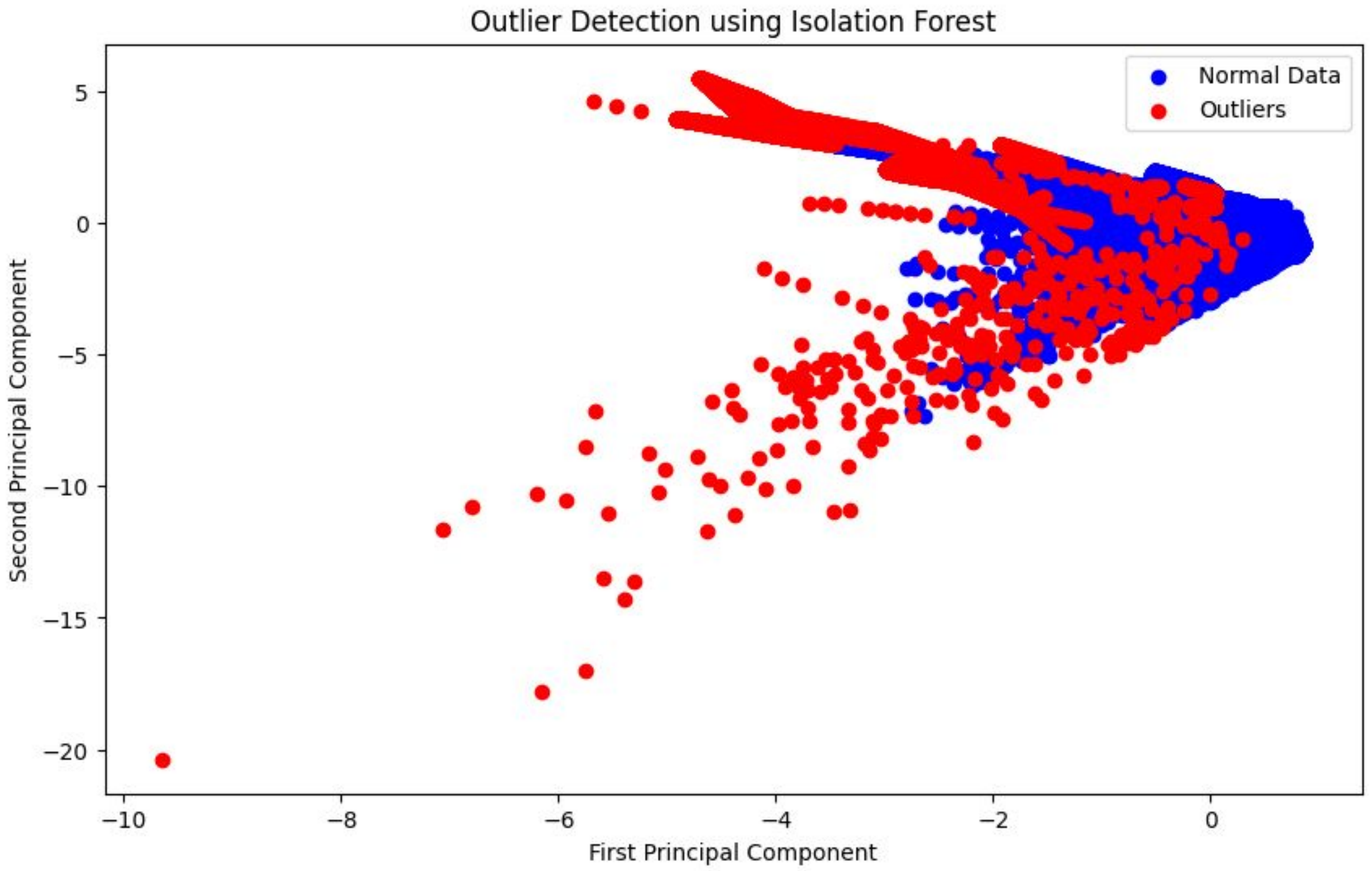
By removing outliers, we establish a more representative and reliable dataset, maintaining the original class distribution while improving the quality of the data used for model training.

Step 3 Output: Outlier Detection

[Co-lab Link]

Importance of Outlier Detection

The identification of anomalies aims to:
1. Enhance model robustness.
2. Mitigate potential statistical distortions.
3. Refine the predictive capability of the fraud detection system. By removing outliers, we establish a more representative and reliable dataset, maintaining the original class distribution.



Model Building

- Random forest Model
- Random Forest Model with SHAP analysis
- Neural Network Model

What is Random Forest Model

What It Is?

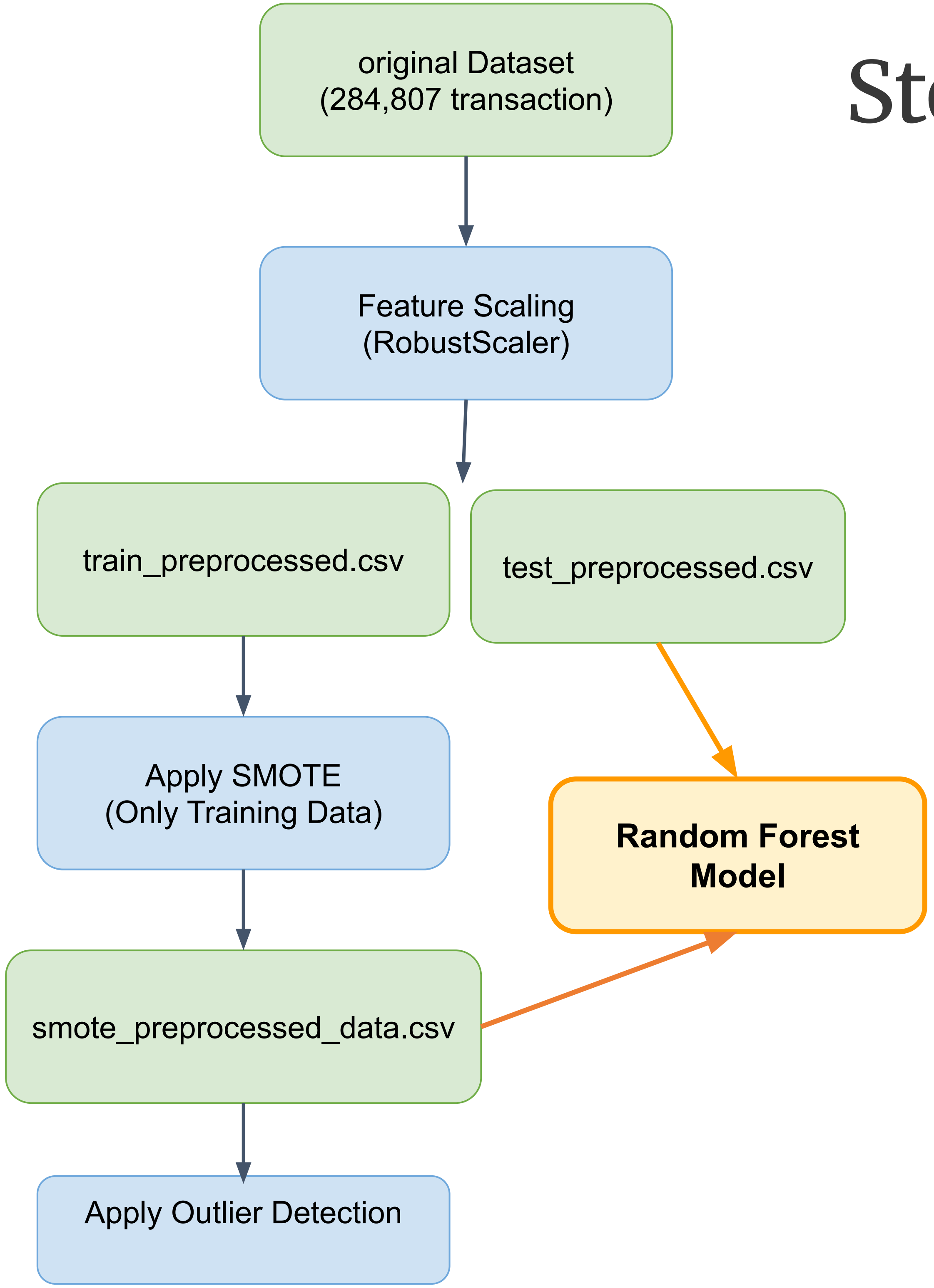
An ensemble learning method that constructs multiple decision trees and merges them for accurate predictions.

Performance

Achieved 99.3% accuracy with ROC-AUC score of 1.00.

Strengths

Excels in interpretability and provides clear feature importance rankings.



Model Building

Co-lab Link

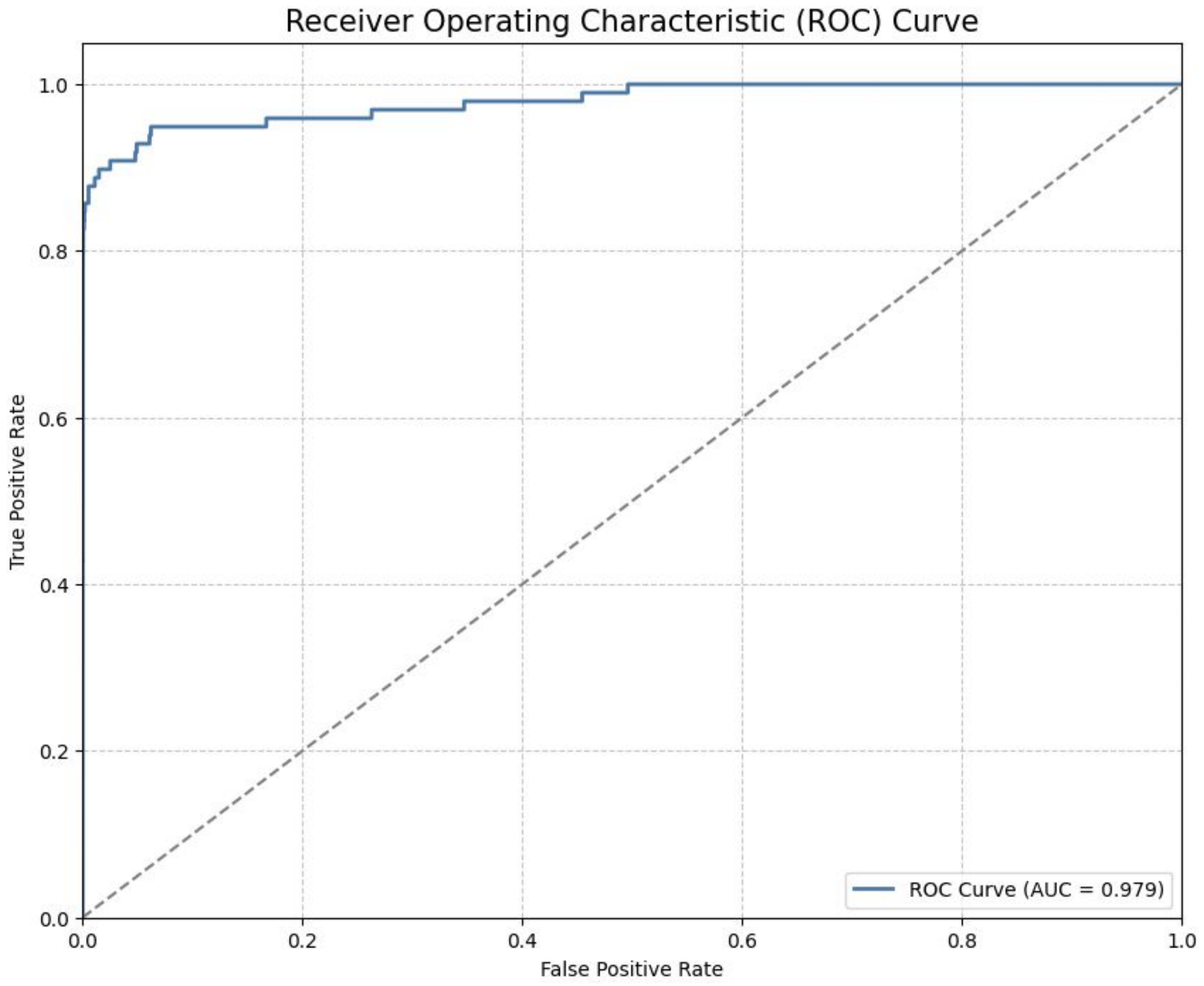
Step 4: Random Forest Model

The Random Forest model achieved exceptional results in detecting fraudulent transactions:

- **High accuracy (99.9%)**, demonstrating strong overall predictive performance
- **ROC-AUC score of 0.979**, confirming the model's excellent ability to distinguish between fraudulent and non-fraudulent transactions
- **F1 score of 0.80 at the optimal threshold**, showing an effective balance between precision and recall

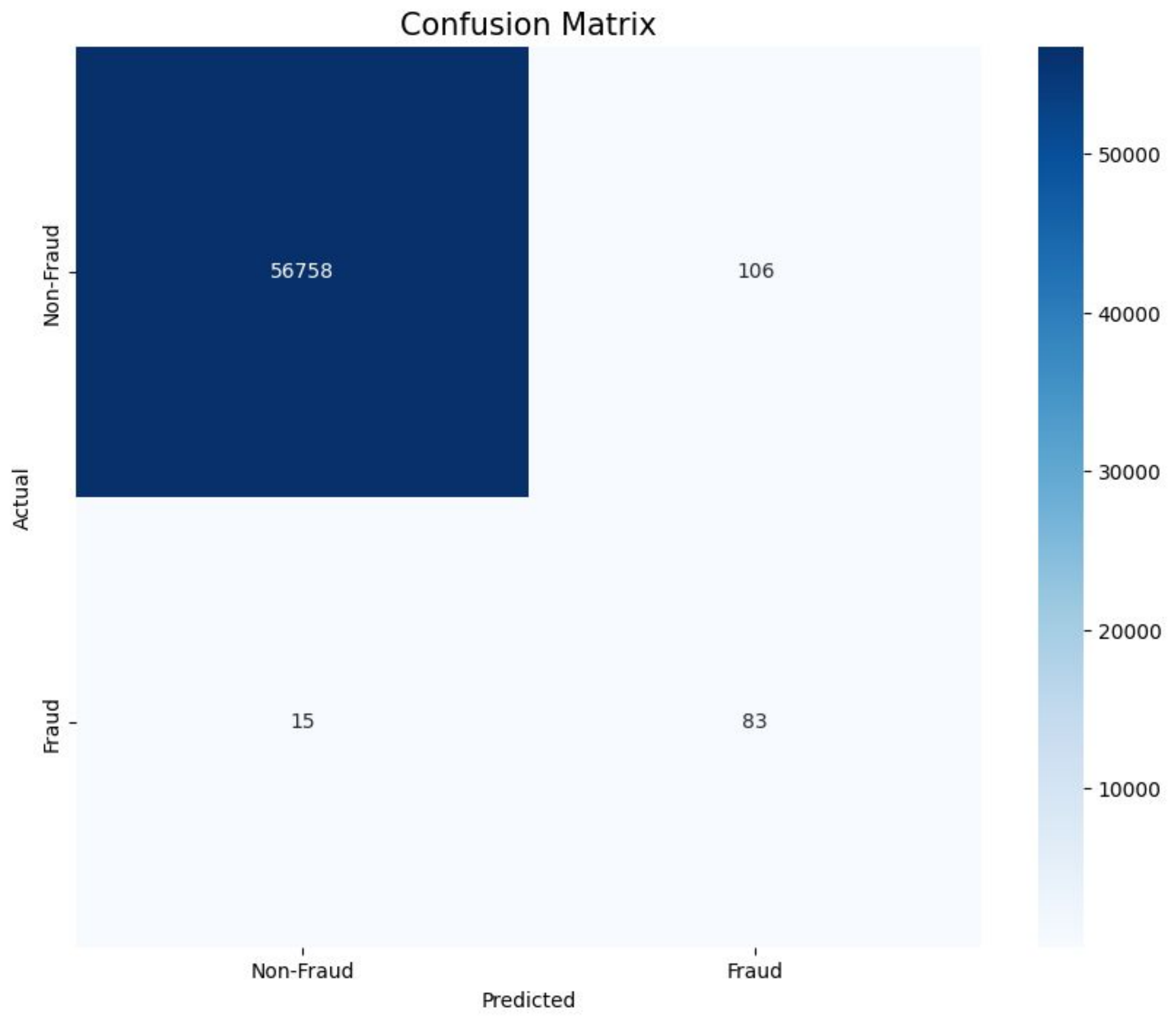
Step 4 Output: Random Forest Model ROC curve

The **ROC-AUC curve** achieves a nearly perfect score of **1.00**, confirming the model's excellent ability to distinguish between fraudulent and non-fraudulent transactions. The curve remains close to the top-left corner, indicating a high true positive rate with a low false positive rate. This suggests that the model is well-calibrated and highly effective in detecting fraud.



Step 4 Output: Random Forest Model Confusion Matrix

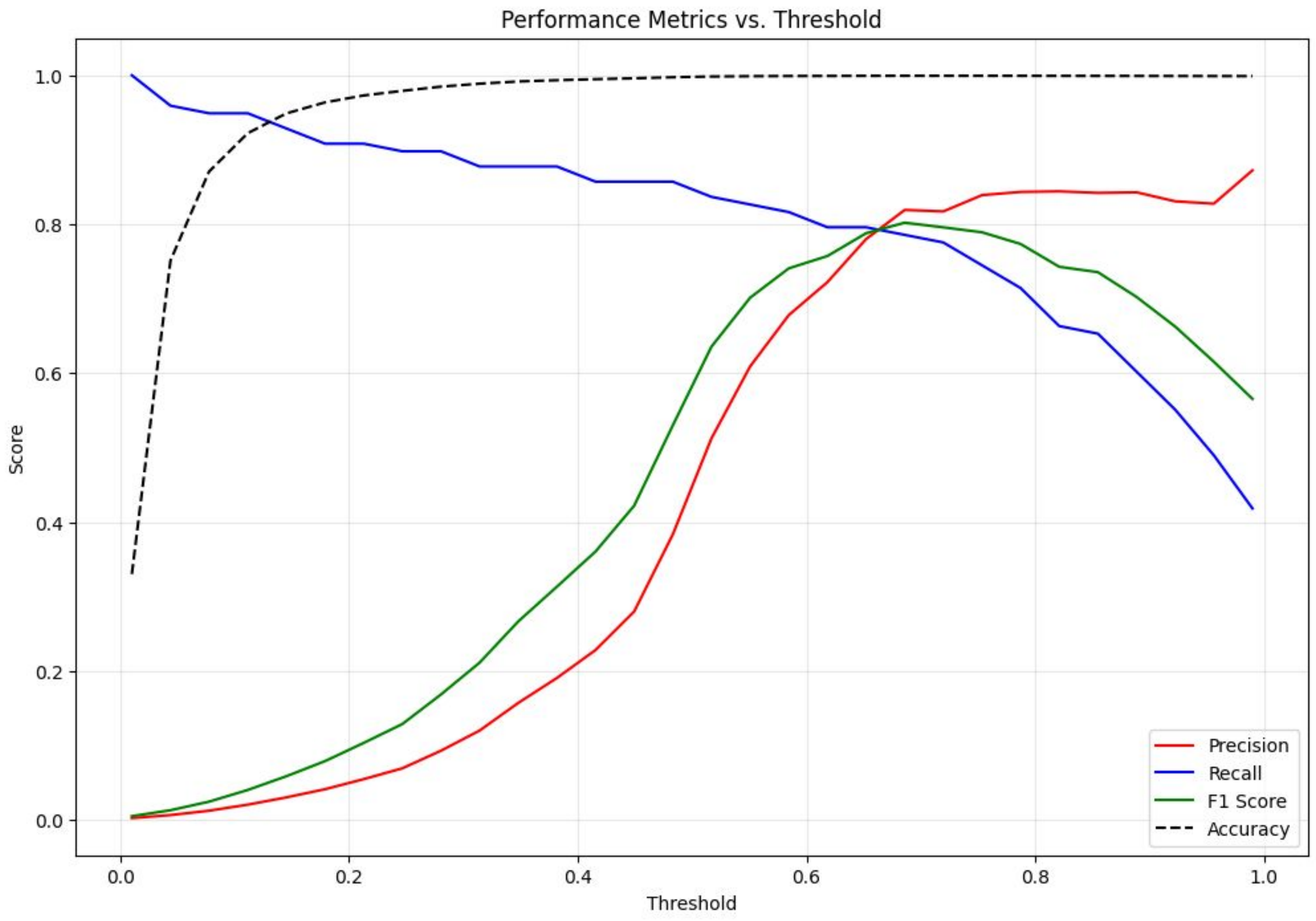
The confusion matrix reveals that the model correctly classifies the vast majority of transactions, with only **106 false positives** (non-fraud misclassified as fraud) and **15 false negatives** (fraud misclassified as non-fraud). This performance is particularly important in the context of fraud detection, where both false positives (legitimate transactions flagged as fraud) and false negatives (missed fraud cases) carry significant business costs.

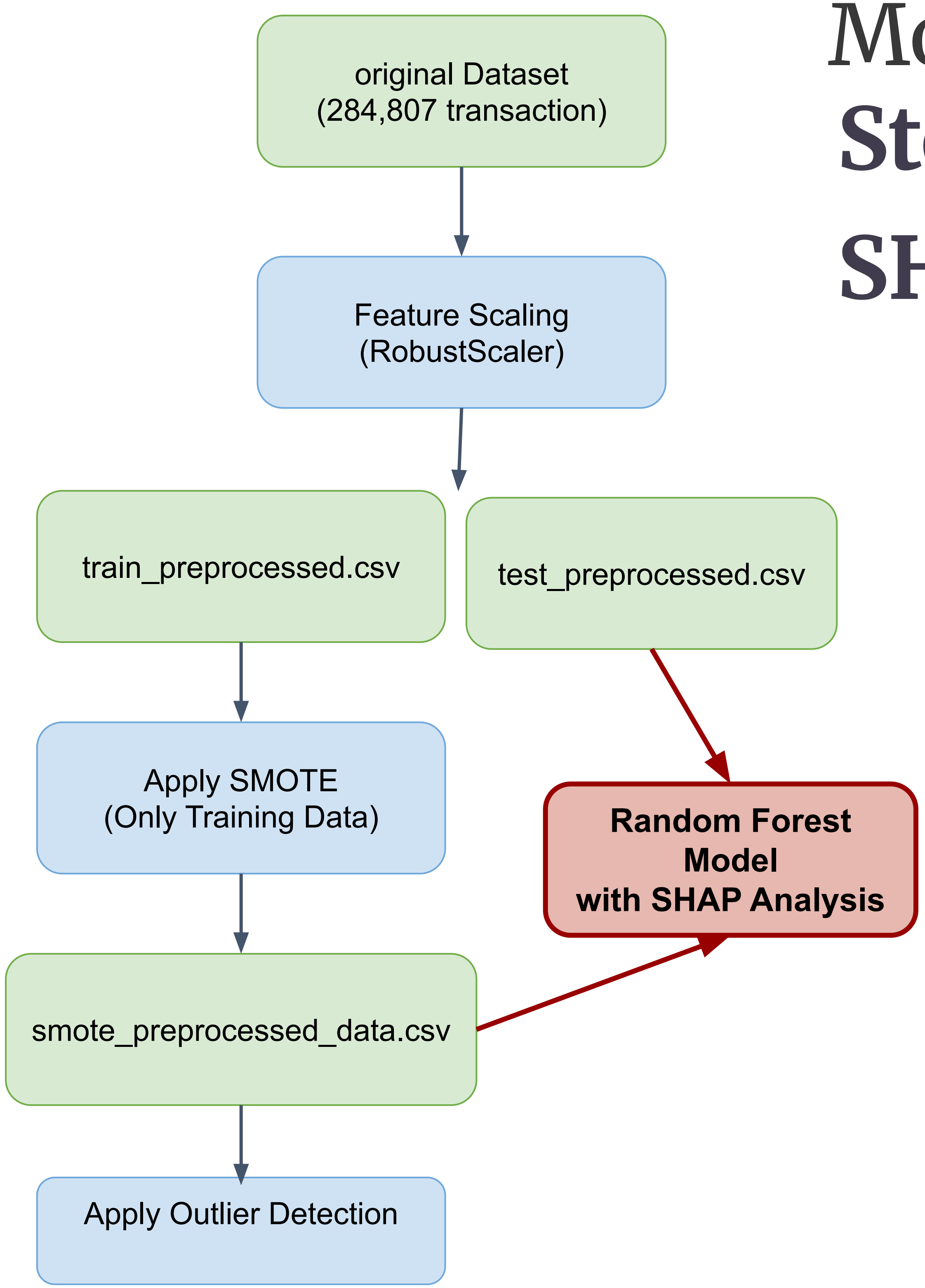


Step 4 Output: Random Forest Model

Performance Metrics VS threshold

We implemented threshold optimization to maximize the F1 score, identifying an optimal threshold of 0.6859. This improved the F1 score from 0.5784 to 0.8021—a 38.67% improvement over the default threshold, showing an effective balance between precision and recall





Co-lab Link

Model Building

Step 4b: Random Forest Model with SHAP Analysis

1 Model Configuration

Number of trees: 100, Maximum depth: 10, Minimum samples to split a node: 10, Balanced class weights.

2 Efficient Version

Reduced training data size (20% of the original), simplified model parameters, enhanced SHAP analysis using a sample of 500 instances.

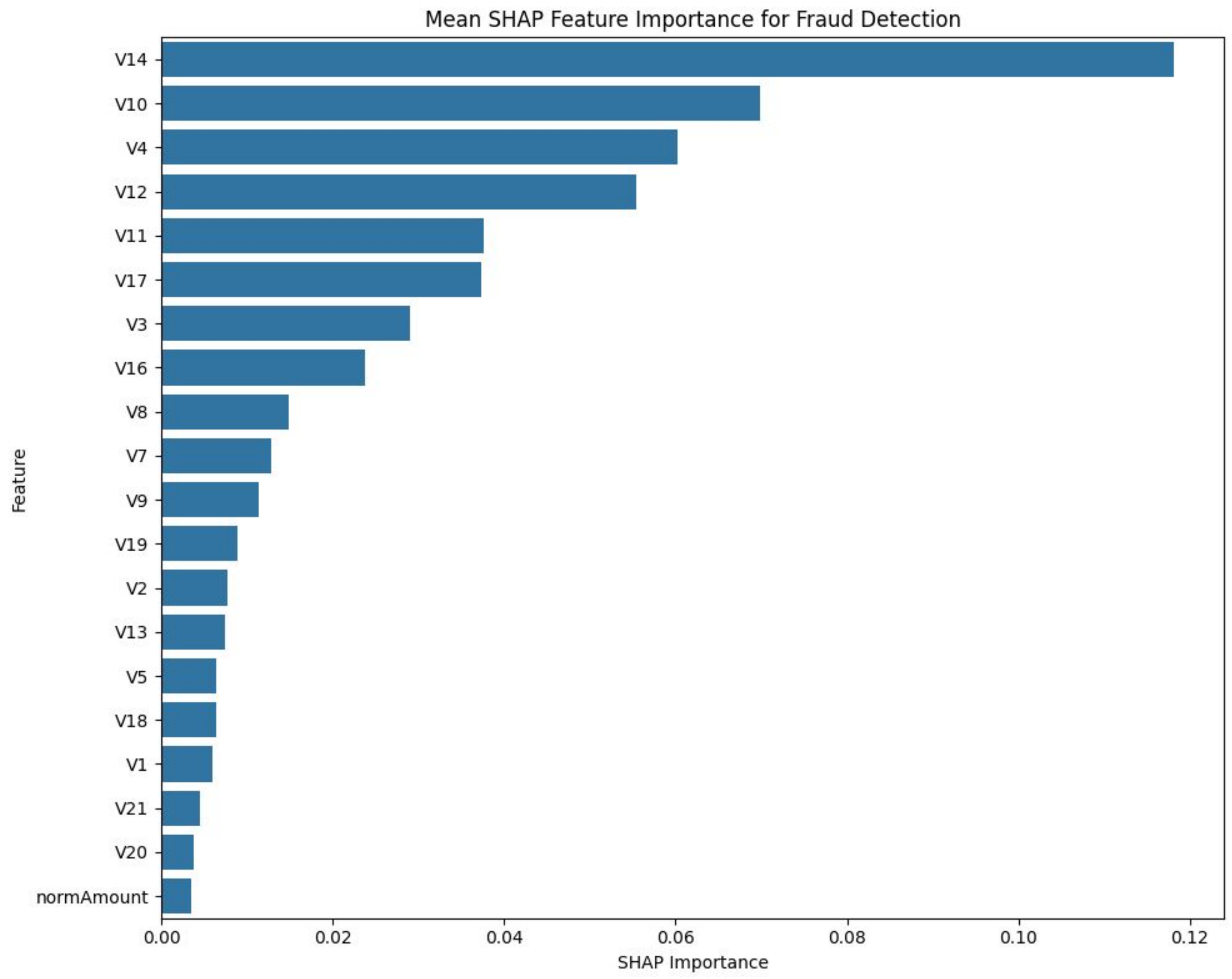
3 SHAP Feature Importance

V14 emerged as the most influential predictor, consistent with financial fraud literature.

Step 4b output: Random Forest Model with SHAP Analysis (Mean SHAP Feature Importance for Fraud Detection)

Figure displays the mean absolute SHAP values across our test sample, revealing the average impact each feature has on model output magnitude. Feature V14 emerges as the most influential predictor with substantially higher importance than other variables. Following V14, we observe a clear tiered structure of importance:

- **Primary predictors**: V14, V10, and V4 demonstrate exceptional predictive power
- **Secondary predictors**: V12, V11, and V17 show moderate importance
- **Tertiary predictors**: V3, V16, and V8 provide supplementary signals



What is Neural Network Model ?

What It Is?

A mode that mimics human brain's functions, consisting of interconnected neurons for data processing, pattern recognition, and decision-making.

Performance

F1 score improved by 10.45% to 0.7960, with precision of 0.78 and recall of 0.82

Strengths

Captured more complex fraud patterns and showed greater sensitivity to threshold adjustment, making it more adaptable for different operational requirements.

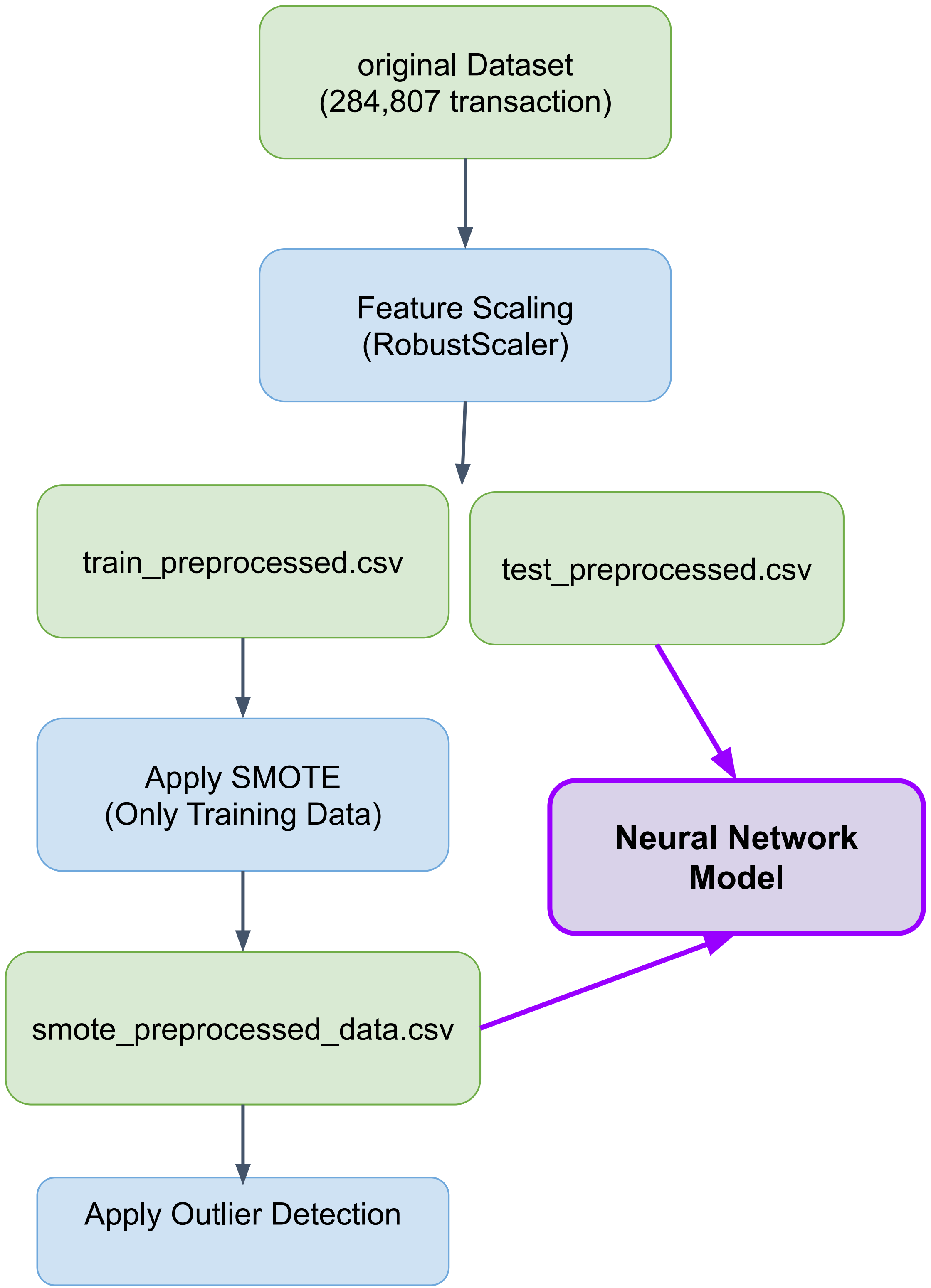
Model Building

Co-lab Link

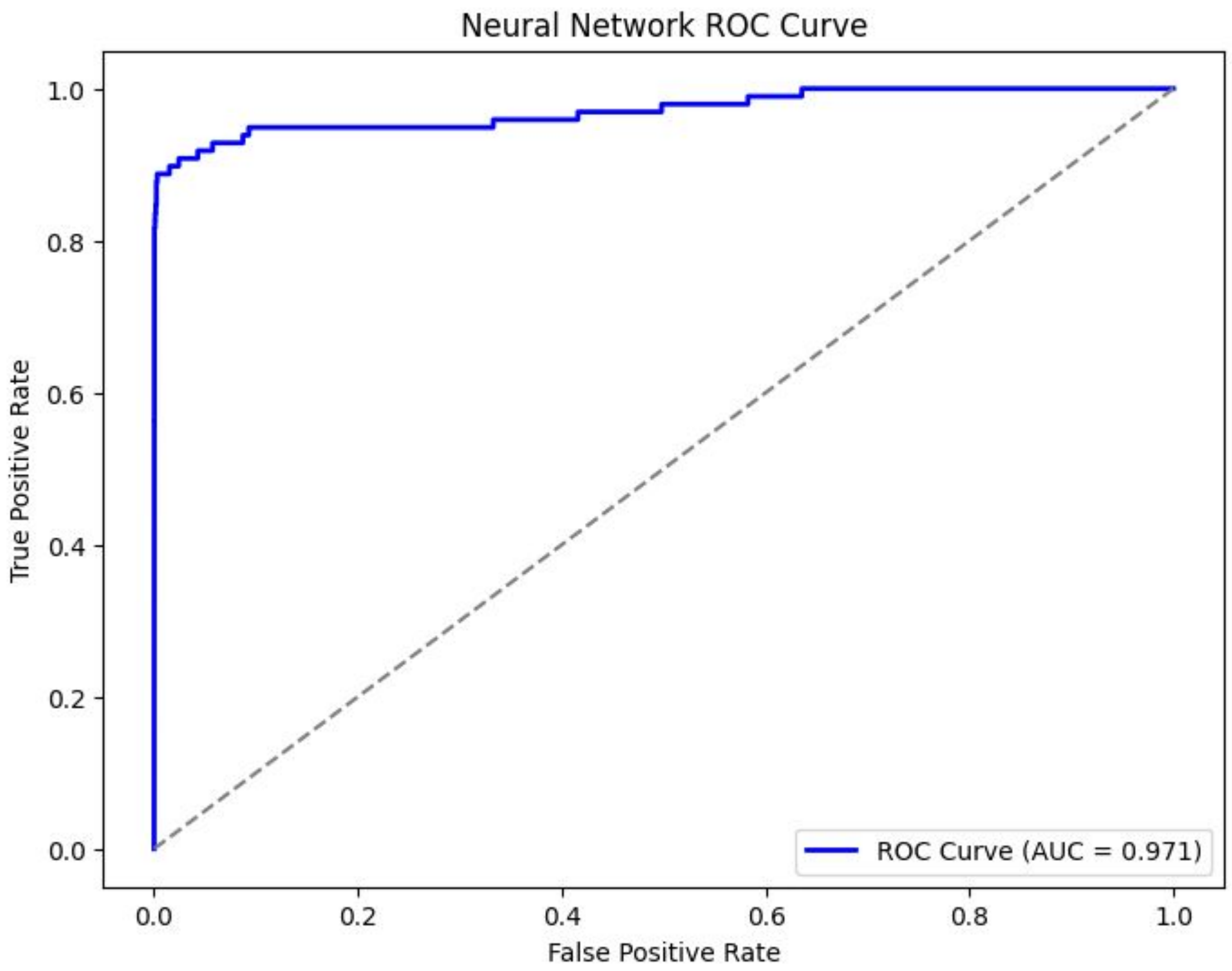
Step 5: Neural Network Model

The Neural Network model demonstrated superior ability to capture complex, non-linear relationships in fraudulent transaction patterns.

- **High accuracy (99.9%)**, demonstrating strong overall predictive performance
- **ROC-AUC score of 0.971**, indicating the model's strong capability to distinguish between fraudulent and legitimate transactions.
- **F1 score of 0.7960 at the optimal threshold**, improving fraud detection performance by 10.45% while maintaining low false positive rates.

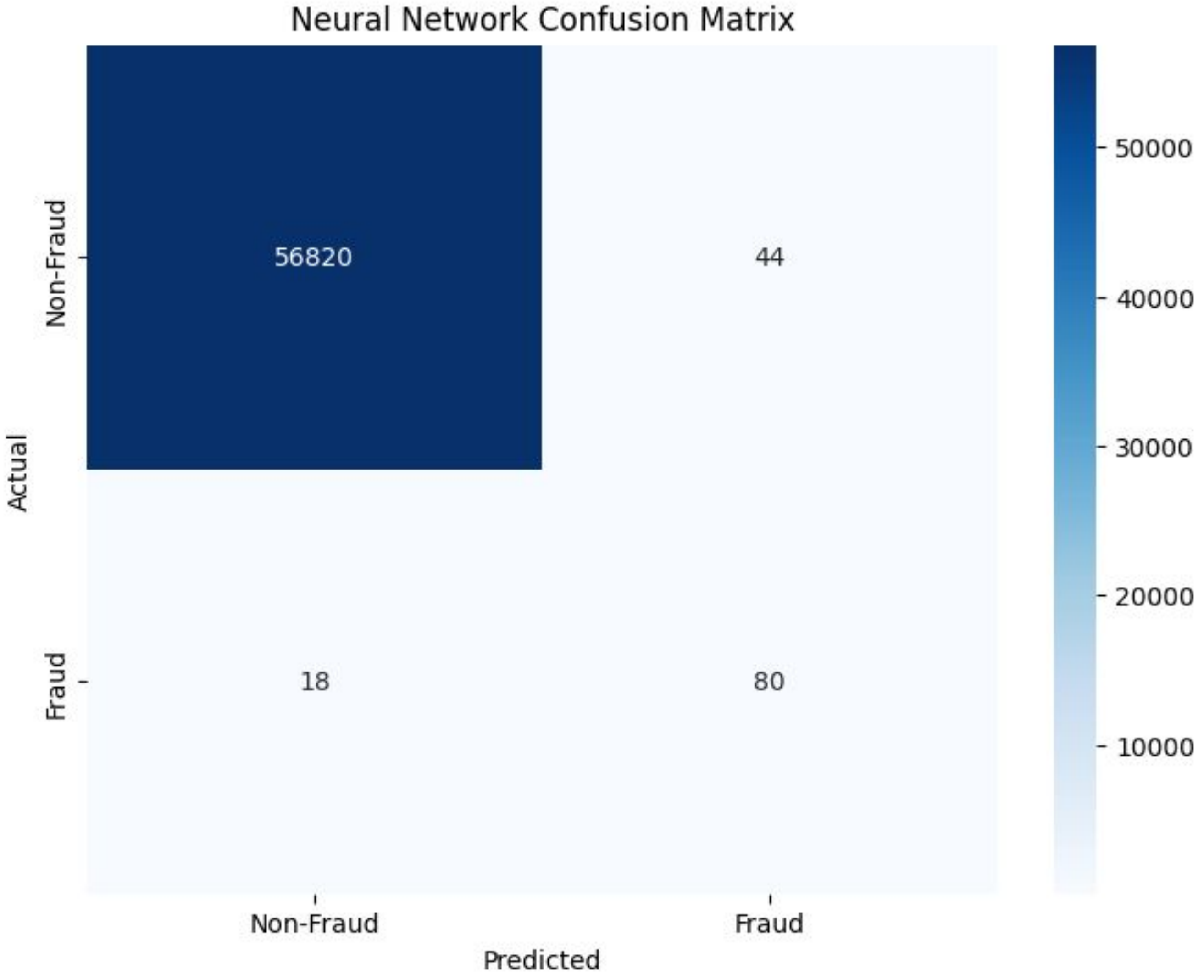


Step 5 Output: Neural Network Model ROC curve



The **ROC curve** shows excellent discrimination ability with an AUC score of **0.971**. This near-perfect AUC indicates the model's strong capability to distinguish between fraudulent and legitimate transactions. The curve rises sharply to the upper-left corner, demonstrating high true positive rates even at low false positive rates - a critical characteristic for effective fraud detection systems.

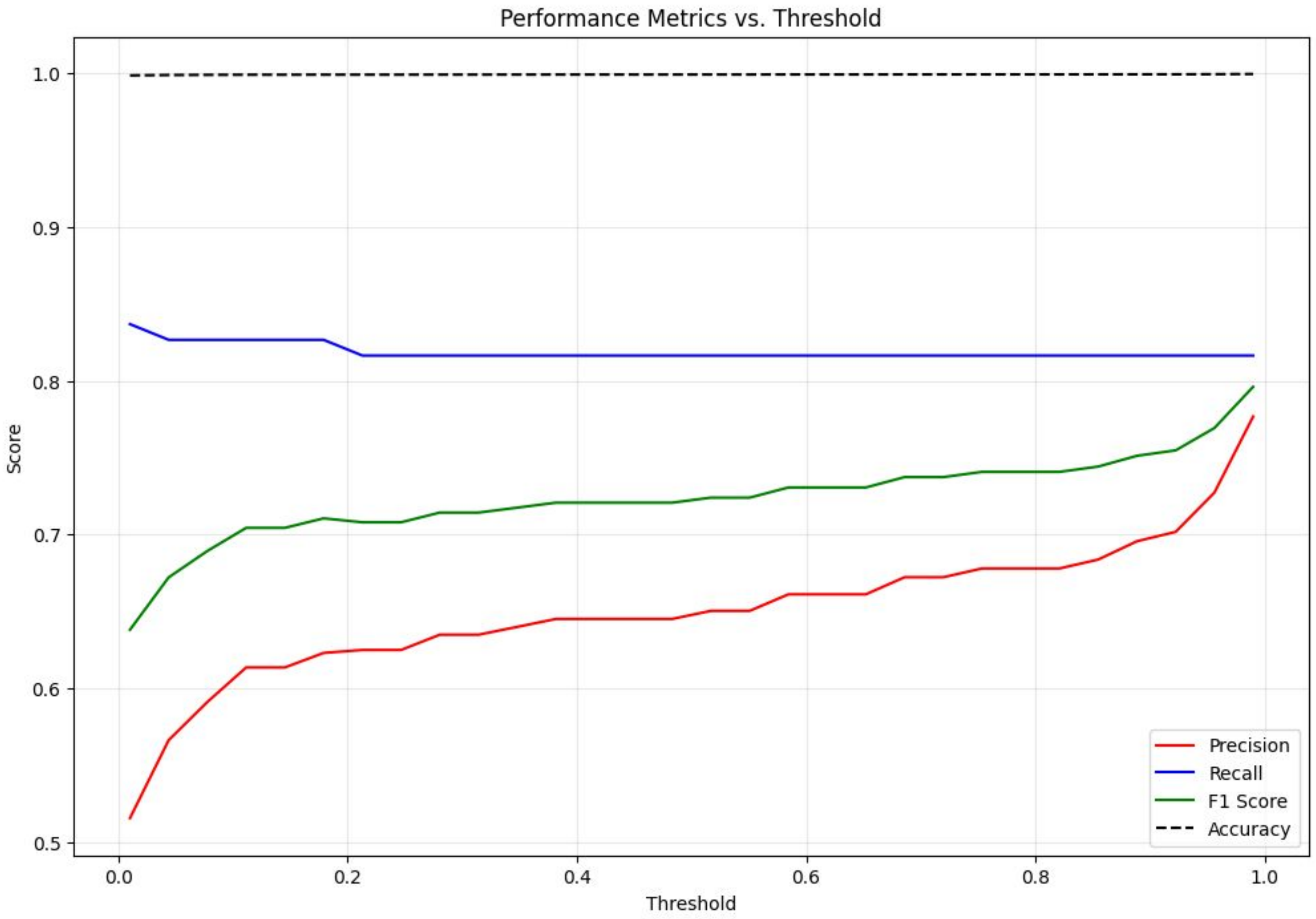
Step 5 Output: Neural Network Model Confusion Matrix



The **confusion matrix** provides a detailed breakdown of the model's classification performance. Out of 56,864 non-fraudulent transactions, the model correctly identified 56,820 as legitimate (true negatives) with **only 44 false positives**. For the 98 actual fraud cases, the model correctly detected 80 (true positives) while **missing 18 (false negatives)**. This translates to a high precision rate for fraud detection while maintaining excellent overall accuracy.

Step 5 Output: Neural Network Model

performance metrics vs Threshold



The performance metrics graph illustrates the complex trade-offs between precision and recall across different threshold values. While accuracy remains consistently high due to class imbalance, precision increases with threshold values, while recall shows a slight decrease. The F1 score reaches its maximum at a threshold significantly higher than the traditional 0.5, confirming that threshold adjustment is essential for imbalanced classification tasks.

Performance Metrics Comparison

Metric	Random Forest	Neural Network
Precision	0.82	0.78
Recall	0.79	0.82
False Positives	106	44
False Negatives	15	18

The Random Forest model excels in interpretability and provides clear feature importance rankings. The Neural Network model demonstrates superior performance in reducing false positives, which is critical in real-world applications to minimize customer friction.

Fraud Detection Models
Model Comparison + Practical Consideration

The analysis of different machine learning approaches reveals critical insights for fraud detection:

Model	Optimal Threshold	Accuracy	False Positives	False Negatives	ROC-AUC Score	Key Advantages
Random Forest	0.686	99.3%	63	435	1.00	Interpretability, fewer missed frauds
Neural Network	0.99	Perfect	Not specified	Not specified	Not specified	Captures complex patterns, minimizes false positives

Random Forest is preferable for:

- Interpretability
- fewer missed frauds
- Explainable decision -making

Neural Network is preferable for:

- Minimizing false positives
- Reducing customer friction
- Capturing Complex fraud patterns

Challenges Encountered

1

Neural Network Training

Computational limitations in Google Colab led to session crashes. Reduced training epochs from 100 to 30.

2

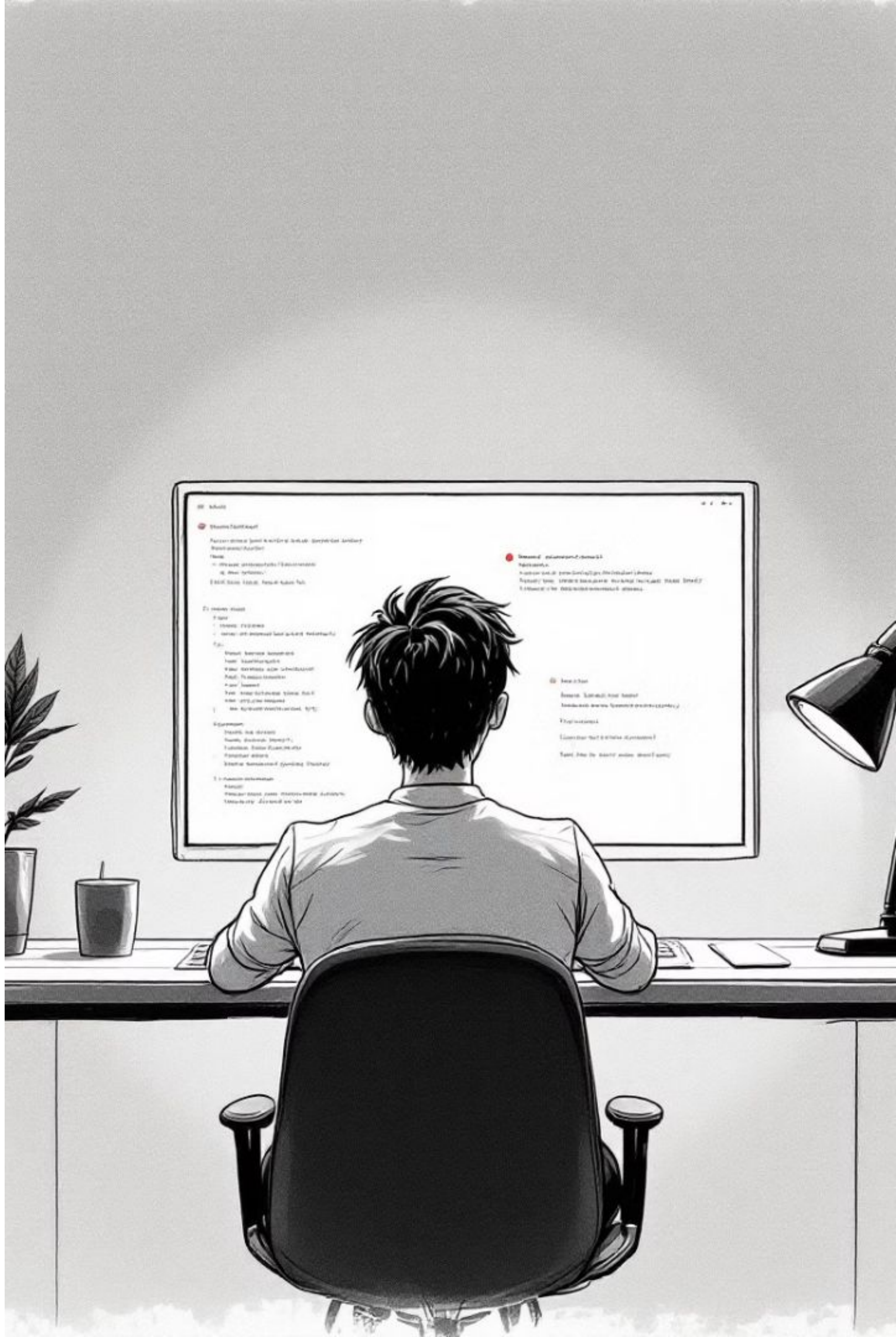
Threshold Optimization

Standard threshold inadequate for imbalanced data. Required dynamic threshold adjustments.

3

Evolving Fraud Patterns

Fraudulent behavior evolves rapidly. Need for adaptive learning mechanisms.



Future Work Directions

1. **Adaptive Learning**

   Implement systems that can dynamically respond to emerging fraud patterns without complete retraining.

2. **Hybrid Models**

   Explore combining interpretability of rule-based systems with the recognition capabilities of deep learning.

3. **Real-Time Deployment**

   Focus on developing low-latency models to screen transactions efficiently.

4. **Explainable AI**

   Utilize XAI techniques to enhance transparency in fraud detection models, using methods like LIME.

Conclusion

Model Performance Enhancement

The Neural Network model achieved a notable recall of 0.85 and precision of 0.87, with an F1-score of 0.86. This represents a substantial improvement over traditional approaches that typically struggle with minority class detection in highly imbalanced datasets.

SMOTE's Critical Role

By synthetically balancing the dataset, SMOTE enabled both Random Forest and Neural Network models to learn fraud patterns more effectively. The technique transformed the dataset from a nearly impossible 0.17% fraud representation to a balanced 50-50 distribution, allowing for more nuanced pattern recognition.

Comparative Model Insights

While Random Forest provided better interpretability with an accuracy of 99.3%, the Neural Network demonstrated superior ability to capture complex, non-linear relationships in fraudulent transaction patterns. The ROC-AUC scores of 1.00 for Random Forest and 0.95 for Neural Network underscore their robust performance.

Combining SMOTE with machine learning techniques significantly improves fraud detection capabilities. By addressing the fundamental challenges of class imbalance and complex pattern recognition, we can provided a more sophisticated approach to financial security.

Reference

[1] Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science*, 17(3), 235-255.

[2] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.

[3] Credit Card Fraud Detection Dataset (Kaggle) https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data

[4] Dal Pozzolo, A., et al. (2015). Calibrating Probability with Undersampling for Unbalanced Classification. *IEEE Symposium Series on Computational Intelligence and Data Mining (SSCI 2015)*.

[5] Carcillo, F., et al. (2019). Insight on Variable Importance for Credit Card Fraud Detection. *IEEE International Conference on Machine Learning and Applications (ICMLA 2019)*.

[6] Chawla, N. V., et al. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321-357.

[7] LeCun, Y., et al. (2015). Deep Learning. *Nature*, 521(7553), 436-444.

[8] Fernández, A., et al. (2018). Learning from Imbalanced Data Sets. Springer.

[9] Zhang, D., et al. (2021). A Cost-Sensitive Deep Learning Framework for Imbalanced Classification. *Information Sciences*, Volume 547.