Comparative Analysis of Machine Learning Models for Fraud Detection

CHAN Ho Lam Myles CHONG Tin Tak LAI Chi Shing YANG Huiyan

Hong Kong University of Science and Technology | IEDA3560 Predictive Analytics

Introduction

Fraud detection in financial systems is a vital application of machine learning. This project evaluates multiple supervised algorithms to determine their effectiveness in **detecting fraudulent transactions**, using real-world anonymised data.

Dataset Overview

Feature Type	Feature Name	Description
Numerical	Step	Unit of time (1 step = 1 hour)
Numerical	Amount	Amount of the transaction
Numerical	oldbalanceOrg	Balance before transaction
Numerical	newbalanceOrig	Balance after transaction
Numerical	oldbalanceDest	Recipient's initial balance
Numerical	newbalanceDest	Recipient's new balance
Categorical	Type	Type of online transaction
Categorical	isFraud	Fraud transaction indicator

Table 1. Features Overview with Data Types

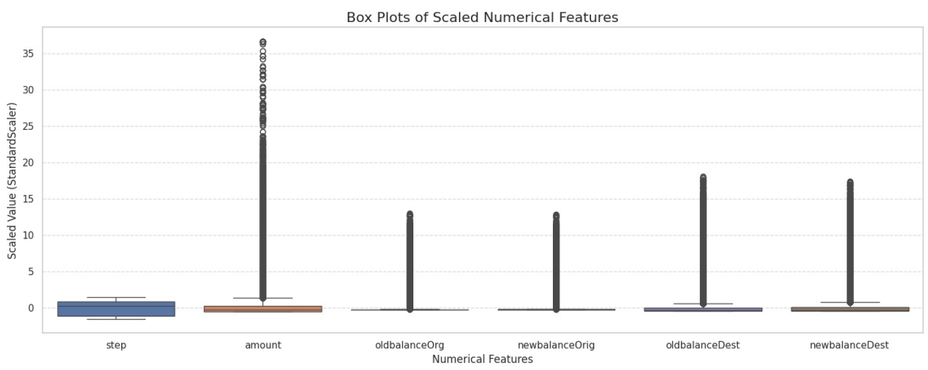

Figure 1. Raw Data Distribution

The dataset [1] contains numerical and categorical features related to online transactions, with a binary target variable, `isFraud`. Figure 1 shows the distribution of numerical features, where long tails and extreme values suggest potential fraud patterns.

Handling Long-Tail Distribution and Class Imbalance

The dataset exhibited a significant class imbalance, with fraud cases constituting only **0.8%** of all transactions. This long-tail distribution challenges model training, leading to biased predictions towards the majority class (non-fraud).

Techniques Used:

- **Outlier Removal:** We applied the **IQR (Interquartile Range)** method to remove extreme outliers from numerical features. This reduced noise and improved model robustness.
- **Resampling:** To address imbalance, we applied **SMOTE (Synthetic Minority Oversampling Technique)** [2] to the training set, generating synthetic fraud samples to balance the class ratio.
- **Train-Test Strategy:** Resampling was performed **only on the training set** to avoid data leakage and maintain a realistic test distribution.

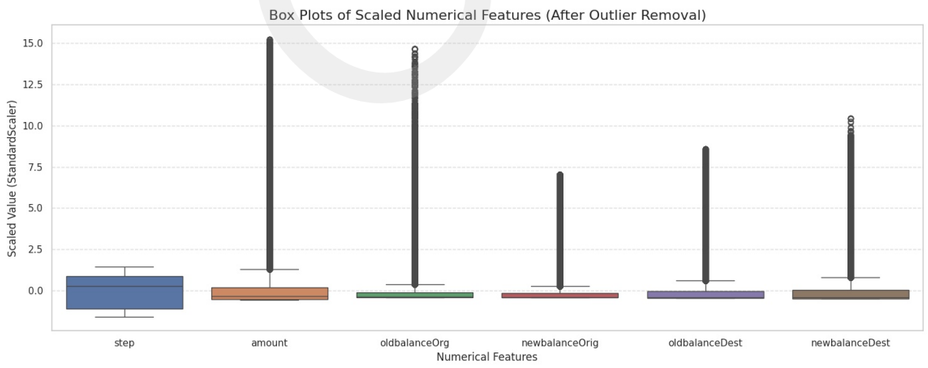

Figure 2. Data Distribution After Outlier Removal

Figure 2 shows the distribution after outlier removal. While scaling improves interpretability, extreme values—often indicative of fraud—**cannot be fully removed** to preserve critical patterns.

Models and Techniques

We tested the following models:

1. Logistic Regression 2. Decision Tree 3. Random Forest
4. XGBoost (Gradient Boosting variant)
5. Neural Network (Multi-layer Perceptron using MLPClassifier)

Hyperparameter Tuning: GridSearchCV with 5-fold cross-validation.

PCA: Applied optionally to explore dimensionality reduction impact.

Performance Metrics

Model	ROC AUC	F1 Score	Recall	Precision	Accuracy
Neural Network	**0.9964**	0.1407	**0.9746**	0.0758	0.9900
XGBoost	0.9938	**0.2171**	0.8961	**0.1235**	0.9945
Random Forest	0.9933	0.2152	0.8591	0.1230	**0.9947**
Decision Tree	0.9669	0.0182	0.9538	0.0092	0.9132
Logistic Regression	0.8203	0.0040	0.7021	0.0020	0.7058

Table 2. Performance Comparison of Models (After Tuning)

This table compares key performance metrics of all tuned models. The Neural Network achieved the highest ROC AUC and recall, while XGBoost had the best F1 score and precision.

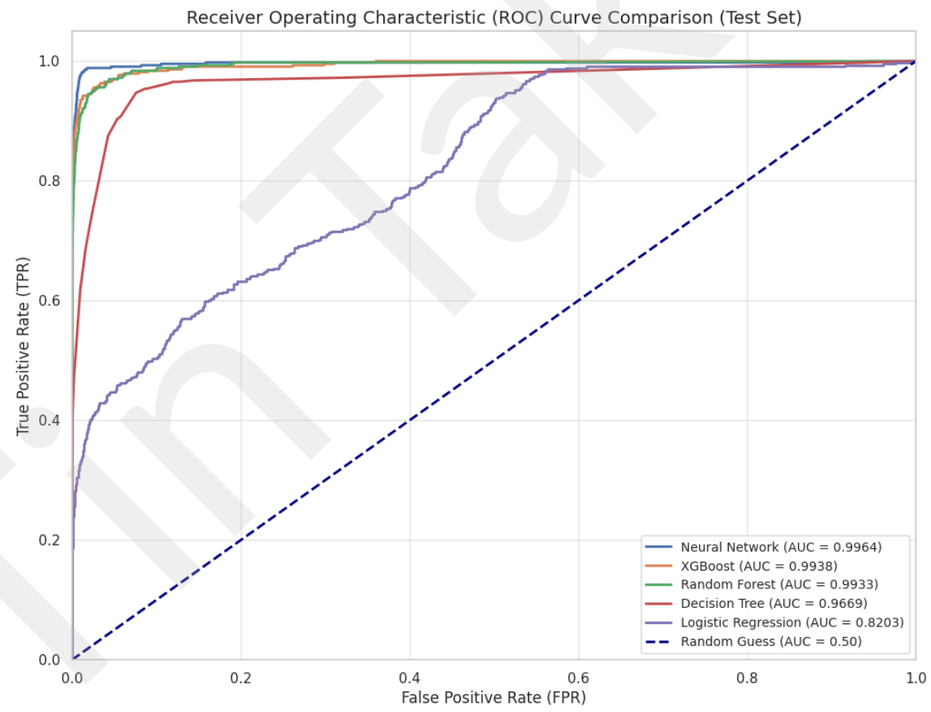

Figure 3. ROC Curves for Each Model

The ROC curves illustrate the classification performance of each model. All tree-based and neural models demonstrate strong discriminatory power, with AUCs above 0.99.

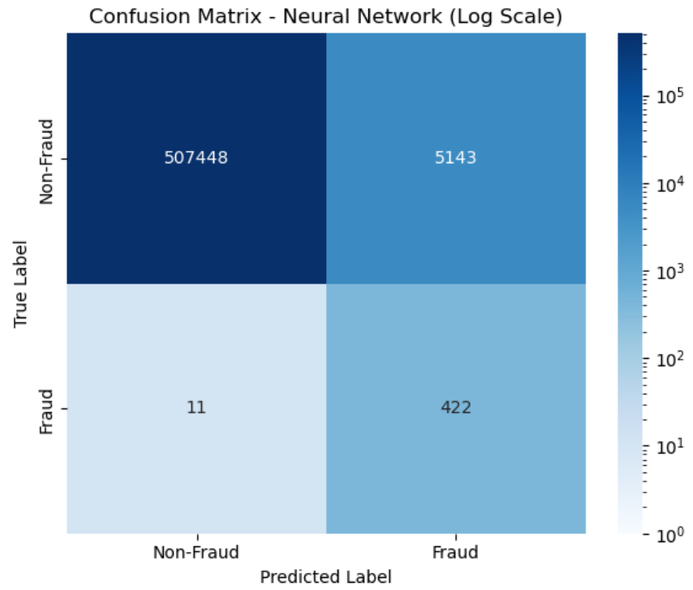

Figure 4. Confusion Matrix of Neural Network

The confusion matrix shows that the Neural Network detects most fraud cases with very few false negatives, demonstrating its effectiveness in identifying rare fraud events.

Conclusion and Future Work

Neural Network and **XGBoost** emerged as top performers, both achieving high ROC AUC and recall. While XGBoost offered better balance in F1 and precision, the Neural Network excelled at identifying rare fraud cases. Future enhancements may include:

- Testing deep models (e.g., LSTM/GRU) for sequential fraud patterns
- Applying cost-sensitive learning to reduce false positives
- Adding explainability (e.g., SHAP, LIME) for trust in deployment

References

[1] J. Shah, "Online Payment Fraud Detection," Kaggle, Oct. 26, 2022. [Online]. Available: `https://www.kaggle.com/datasets/jainilcoder/online-payment-fraud-detection`. [Accessed: Apr 20, 2025].

[2] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.