# *Multi-Algorithm Data Mining Model for Credit Card Fraud Pattern Discovery*

*Md. Shakib Ahamed*

*Id. 2215151130*

*Dept. Of Computer science and engineering (CSE)*

*University of information technology and sciences (UITS),Dhaka, Bangladesh*

*2215151130@uits.edu.bd*

*Rawnak Tasnim Ruku*

*Id. 2215151152*

*Dept. Of Computer science and engineering (CSE)*

*University of information technology and sciences (UITS),Dhaka, Bangladesh*

*tasnim_ruku1152@uits.edu.bd*

## Abstract

This project uses several data mining algorithms to analyze the well-known credit card fraud detection dataset from Kaggle. The goal is to identify fraudulent transactions and uncover patterns and structures in the data. The process includes outlier detection using Isolation Forest and Local Outlier Factor, association rule mining with Apriori, and graph-based ranking methods like PageRank and HITS. Experimental results show that Isolation Forest can achieve good recall for the minority fraud class, while Local Outlier Factor tends to label almost all transactions as normal. Association rules based on discretized features show consistent patterns in fraudulent transactions. Graph-based analysis reveals that fraud instances tend to occupy less central and less authoritative positions in a similarity network. This study illustrates how using different data mining approaches can provide a deeper understanding of fraud behavior beyond what a single classifier can offer.

## 1. Introduction

Credit card fraud is a serious problem in financial systems because of the high volume of transactions and the uneven nature of fraudulent behavior. Traditional supervised learning models can detect fraud well when there is enough labeled data. However, in many real-world situations, fraud patterns change and labeled data is limited or delayed.

This project looks at data mining methods that don’t depend only on standard classification. The goal is to create and implement a multi-algorithm data mining model that:

- Detects suspicious transactions as outliers.

- Extracts common patterns and rules from fraudulent behavior.

- Analyzes the network structure of transactions using graph-based ranking.

The dataset used is the “Credit Card Fraud Detection” dataset from Kaggle. It contains anonymized transaction features and a binary fraud label.

## 2. Dataset and Preprocessing

The dataset contains:

- 284,807 transactions
- 31 features:
    - Time, Amount
    - V1–V28: PCA-transformed, anonymized numerical features
    - Class: target label (0 = normal, 1 = fraud)

Key preprocessing and exploratory data analysis (EDA) steps:

- Missing values: Checked for all columns; none were found.
- Duplicates: Duplicated rows were checked and considered negligible.
- Class imbalance: The fraud class (Class = 1) represents a tiny fraction of the data (typical for real-world fraud datasets), confirming a strong class imbalance.
- Visualization:
    - Distribution of Amount to understand transaction size behaviour.
    - Count plot of Class to visualize imbalance between normal and fraudulent transactions.

For modelling:

- Features V1–V28, Time, and Amount were used as predictors.
- Data were split into training (80%) and test (20%) sets with stratification to preserve fraud ratio.
- A StandardScaler was applied to all input features before feeding them into the algorithms, since most methods are distance-based or sensitive to scale.

## 3. Methodology

The project is divided into three major methodological components:

### 3.1 Outlier Analysis

Two unsupervised anomaly detection algorithms were used:

1. Isolation Forest
    - Intuition: Anomalies are easier to isolate in random partitioning trees.
    - Setup:
        - Number of estimators: 200
        - Contamination parameter set approximately to the empirical fraud rate in the training data
    - Predictions:
        - Model outputs -1 for outliers and 1 for inliers, which were mapped to 1 = fraud and 0 = normal.

2. Local Outlier Factor (LOF)
    - Intuition: An observation is an outlier if its local density is significantly lower than that of its neighbours.
    - Setup:
        - n_neighbors = 20
        - Contamination approximately equal to the fraud rate
    - LOF was fitted on the training set and evaluated on the test set using its novelty detection mode.

Both methods were evaluated using the known labels on the test set with standard metrics: precision, recall, F1-score, and confusion matrix.

### 3.2 Association Rule Mining (Apriori)

Association rule mining was used to understand which feature patterns are common among fraudulent transactions.

Steps:

1. A 10% random sample of the dataset was drawn for computational efficiency.
2. Continuous variables were discretized:
    - The amount was binned into categories: very low, low, medium, high.
    - Time was split into four quartiles (Q1–Q4) to approximate different time-of-day segments.
    - Selected PCA features (V1, V2, V3) were converted into sign-based categories such as V1_positive vs V1_negative.
3. Only rows with Class = 1 (fraudulent transactions) were retained for rule mining to focus on fraud-specific patterns.
4. The resulting categorical data were one-hot encoded and passed to the Apriori algorithm:
    - Minimum support: 0.05
5. Association rules were then generated using lift as the main metric, with lift ≥ 1 indicating rules stronger than chance.

This approach reveals which combinations of discretized attributes frequently co-occur in fraudulent transactions.

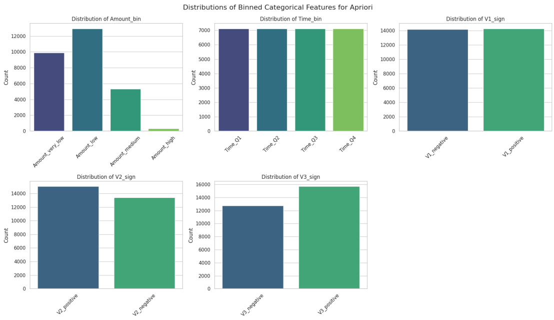

figure:

### 3.3 Graph-Based Mining: PageRank and HITS

To study the structural role of fraud in the transaction space, a graph-based analysis was conducted:

1. A subset of 2,000 transactions was sampled to keep the graph manageable.
2. A k-nearest neighbours (kNN) graph was constructed:
    - Nodes: individual transactions
    - Edges: directed edges from each transaction to its 10 nearest neighbours in feature space, using Euclidean distance on the scaled data.

3. This yielded a directed graph with 2,000 nodes and a substantial number of edges.

On this graph:

- PageRank was applied to measure the global "importance" or centrality of each transaction node.
- HITS (Hyperlink-Induced Topic Search) was applied to compute:
  - Hub scores – nodes that point to many important nodes
  - Authority scores – nodes that are pointed to by many important nodes

The resulting scores were then aggregated by class (Class = 0 vs Class = 1) to understand how fraud vs non-fraud transactions differ in terms of graph centrality.

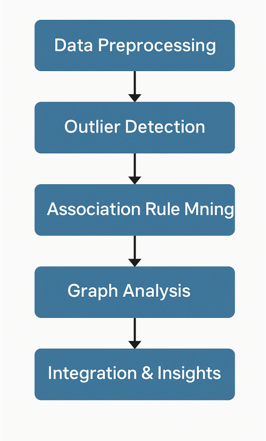

Figure 1: Methodology

## 4. Experimental Results

### 4.1 Outlier Detection Performance

**Isolation Forest**

On the test set, Isolation Forest achieved:

- Normal class (0):
  - Precision ≈ 0.9988
  - Recall ≈ 0.9986
  - F1-score ≈ 0.9987
- Fraud class (1):
  - Precision ≈ 0.28
  - Recall ≈ 0.31
  - F1-score ≈ 0.29

Overall accuracy was about 99.8%, but due to class imbalance, accuracy is not very informative. The confusion matrix showed that the model correctly identified the majority of normal transactions, and it was able to detect a non-trivial portion of fraud cases but still missed many.

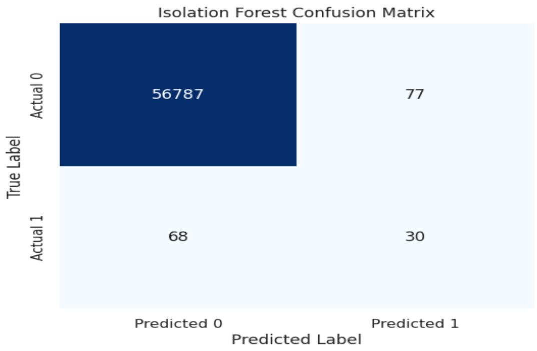

Figure 2: Isolation Forest Confusion Matrix

**Local Outlier Factor (LOF)**

For LOF:

- Normal class (0):
  - Precision ≈ 0.9983
  - Recall ≈ 0.9983
- Fraud class (1):
  - Precision = 0.0
  - Recall = 0.0

LOF effectively predicted all test samples as normal, resulting in a recall and F1-score of zero for fraud. This highlights how sensitive such methods are to parameter settings and extreme imbalance.

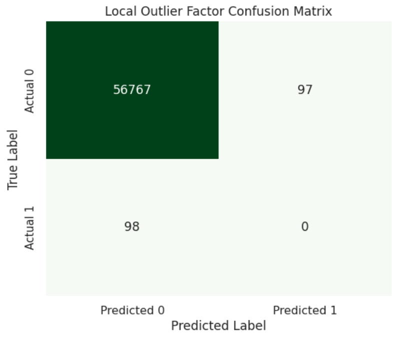

Figure 3: LOF Confusion Matrix

Comparison

- Isolation Forest, despite limited recall on the fraud class, performed substantially better than LOF for anomaly detection in this setting.
- LOF in novelty mode with default parameters is not suitable here without careful tuning or rebalancing strategies.

### 4.2 Association Rule Mining Findings

From the sampled fraud-only subset, frequent itemsets revealed clear tendencies:

- A very high proportion of fraudulent transactions had certain PCA features with negative signs (e.g., negative V1 and negative V3).
- Another feature (V2) tended to be positive for the majority of fraud cases.
- Two-attribute combinations like:
    - (V1_negative, V3_negative)
    - (V3_negative, V2_positive)

    showed relatively high support within the fraud subset.

The association rules built from these itemsets, especially those with high lift, suggest consistent structural patterns in how the PCA-transformed features align during fraudulent behaviour. Although the PCA features are not directly interpretable, these rules confirm that frauds cluster in specific regions of the transformed feature space.

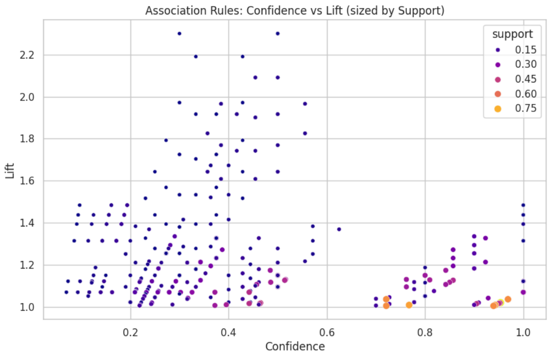

Figure 4: Association Rule Confidence Graph

### 4.3 Graph-Based Ranking (PageRank and HITS)

On the kNN transaction graph:

- PageRank averages by class:
    - Non-fraud (Class = 0): average PageRank ≈ 0.00050
    - Fraud (Class = 1): average PageRank ≈ 0.000078

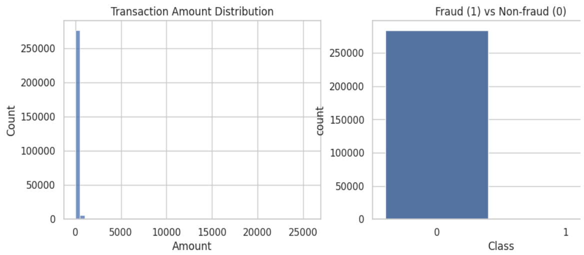

Figure 5 : Fraud vs Non-fraud ratio

This indicates that fraudulent transactions tend to be less central in the similarity network compared to normal ones, which dominate the graph structure due to their sheer number.

- HITS averages:
    - For non-fraud:
        - Mean hub score ≈ 0.00050
        - Mean authority score ≈ 0.00050
    - For fraud:
        - Mean hub score ≈ 0.000149
        - Mean authority score ≈ 0.000002

Fraud nodes appear with lower hubs and especially lower authority scores, suggesting they are less well-connected and less “trusted” in terms of in-links within the similarity network. This supports the idea that frauds occupy the fringes or less dense regions of the transaction manifold.

## 5. Discussion

The combination of methods provides complementary views:

- Outlier detection shows that a tree-based anomaly detector like Isolation Forest can find some types of fraud fairly well, but the performance is limited by the extreme imbalance. LOF, on the other hand, has trouble without a lot of tuning, which shows how risky it is to rely on just one unsupervised method.
- Association rule mining does not directly classify transactions, but it reveals recurrent patterns within fraudulent behaviour in the PCA feature space. Even though the features are anonymized, the presence of stable sign and range patterns can guide further feature engineering or supervised model design.

Graph-based algorithms (PageRank, HITS) show that fraud cases tend to be less central, less authoritative, and less hub-like in the similarity graph. This observational insight could be incorporated in future models, for example, by using graph centrality features as additional inputs to supervised classifiers.

Overall, the project demonstrates that exploring the data through multiple data mining algorithms provides a more nuanced understanding of fraud beyond simple supervised learning.

## 6. Conclusion and Future Work

- Using the Kaggle credit card fraud dataset, this project applied a multi-algorithm data mining model that combined graph-based ranking, association rule mining, and outlier detection:

- Isolation Forest offered a reasonable compromise between capturing fraud and avoiding false alarms.

- Local Outlier Factor failed to detect fraud in its default setup, underscoring the need for careful calibration.

- Apriori-based rules highlighted recurring structural patterns in fraudulent transactions.

- PageRank and HITS showed that fraud transactions typically lie at the periphery of the transaction similarity graph.

Future work may include:

1. Combining these unsupervised and graph-derived features with a supervised classifier (e.g., Gradient Boosting, XGBoost) to improve fraud detection performance.

2. Applying resampling techniques (SMOTE, undersampling) or cost-sensitive learning to better handle the extreme class imbalance.

3. Extending the graph modelling to include temporal edges (e.g., linking transactions within short time windows) for dynamic fraud pattern analysis.

4. Evaluating additional data mining algorithms, such as clustering or sequence mining, to capture further hidden structures.

This multi-algorithm approach illustrates how classical data mining techniques can be systematically combined to study financial fraud from several complementary perspectives.

## 7. Reference

[1] A. Dal Pozzolo, O. Caelen, R. A. Johnson and G. Bontempi,“Calibrating probability with undersampling for unbalanced classification,” 2015 IEEE Symposium Series on Computational Intelligence (SSCI), Cape Town, South Africa, 2015, pp. 159–166.

[2] N. Dua and G. Singh,“Credit Card Fraud Detection Using Machine Learning: A Systematic Literature Review,” International Journal of Information Security and Privacy, vol. 13, no. 1, pp. 22–45, 2019.

[3] F. T. Liu, K. M. Ting and Z.-H. Zhou, “Isolation Forest,” 2008 IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 413–422.

[4] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules,” Proc. 20th Int’l Conf. Very Large Data Bases (VLDB), pp. 487–499, 1994.

[5] S. Brin, R. Motwani and T. Winograd, “The PageRank citation ranking: Bringing order to the web,”Technical Report, Stanford InfoLab, 1998.

[6] J. Kleinberg, “Authoritative sources in a hyperlinked environment,” Journal of the ACM, vol. 46, no. 5, pp. 604–632, 1999.

### 7.1 Dataset reference

[7] M. L. G. ULB Machine Learning Group,

“Credit Card Fraud Detection Dataset,”

Kaggle, 2018. [Online]. Available: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud