data

MDPI

Article

Machine Learning in FinTech for Financial Fraud Data Detection

Sanjaikanth E. Vadakkethil Somanathan Pillai and Wen-Chen Hu *

School of Electrical Engineering and Computer Science, University of North Dakota, Grand Forks, ND 58202, USA; s.evadakkethil@und.edu

* Correspondence: wenchen@cs.und.edu; Tel.: +1-(0021)-701-777-2413

Abstract

Financial fraud keeps rising these days. Organizations attempt to stop this trend by using various methods, such as distributing guides on how to avoid scams and frauds and automatically generating alerts when suspicious activities occur. However, this passive approach does not mitigate the problem, as the trend is worsening, and it is usually too late when victims realize they have been scammed. Therefore, active approaches must be employed before scams reach the victims. A wide variety of preventive methods, such as neural networks and data mining, have been used to detect financial fraud data, but none have proven entirely effective in combating scams. Each method has its pros and cons. This research takes advantage of multiple machine learning techniques, such as k-nearest neighbors (kNN) and decision trees, by utilizing data fusion to detect financial fraud accurately. The data fusion function used here is self-adjusting through learning. During the training phase, the system is repeatedly applied to the dataset until an optimal detection rate is achieved. Experimental results from credit card transactions show that the proposed method outperforms each individual method. Parameter or threshold values for the data fusion are set heuristically. Future research will focus on developing reconfigurable data fusion by automatically adjusting the values.

Keywords: fintech; fraud; financial fraud data detection; machine learning; financial technology

1. Introduction

According to the Federal Trade Commission [1], U.S. consumers lost more than 10 billion dollars to various kinds of fraud in 2023, marking a 14% increase over losses in 2022. Many of the victims are elderly individuals who are particularly vulnerable to financial fraud, as noted in several reports [2–4]. Here are some key statistics:

- A total of 24% of financial losses occur to seniors who hold more than 50% of the household net worth.
- U.S. seniors lost about 1 billion dollars to scammers in 2023.
- A total of 18% of U.S. scam victims are over 60 years old.
- Over one-third of bank account takeover victims in the U.S. are over 60.

The financial impact on seniors is especially severe because many do not have active income. When their savings are insufficient to sustain their living standards, it creates significant challenges for society. Various methods, such as education [5] and sending alerts, have been used to combat scams. However, these approaches are passive and do not adequately address the problem. Instead, this research proposes a proactive method using machine-learning techniques to identify potential financial fraud before it causes harm.

Academic Editor: Kani Chen

Received: 1 April 2026
Revised: 16 June 2026
Accepted: 25 June 2026
Published: 6 August 2026

A variety of innovative methods, such as data classification and knowledge discovery, have been applied to financial fraud detection. However, none of them can solve the problem satisfactorily, as each method has its own advantages and disadvantages. This research integrates the detection results from seven machine-learning methods (kNN, Gaussian Naïve Bayes, decision trees, random forest, AdaBoost, bagging, and logistic regression) to produce more consistent and accurate detection outcomes. A dataset of credit-card transactions is used for system training. Before that, seven methods (Z-score, modified Z-score, Tukey's IQR, isolation forest, DBSCAN, standard deviation, and Winsorization) are used to detect outliers, which are the data points significantly different from the rest of data in a dataset. Utilizing multiple data sources yields better results compared to relying on a single source. The data fusion function Hybrid FSAF (Smoothed Fraud-Sensitive Adaptive Fusion) employed here is self-adjusting through learning. Each of the seven machine-learning methods is assigned a weight based on its accuracy when compared to the expected results. More accurate methods are given higher weights, which play a greater role in the final result. During the training phase, the system is repeatedly applied to the dataset until an optimal detection rate is achieved. These final weights are then used in the testing phase.

This paper is organized as follows: Section 2 discusses the related methods of financial fraud data detection. Section 3 introduces the proposed system by detailing the system structure and the data fusion used to integrate seven machine-learning methods. Section 4 describes the dataset used in this research, including data pre-processing and analysis. The methods employed in this research are explained in Section 5, and the experimental results are presented in Section 6. Finally, the conclusion and future research directions are provided, followed by references.

2. Background Information and Related Research

A wide variety of methods are used for financial fraud detection, which can be classified into two themes: neural networks and other ad hoc methods. This research combines a couple of the detection methods and provides a better result compared to the use of individual methods. These themes are discussed in this section.

2.1. Neural Networks for Financial Fraud Data Detection

Neural networks have been extensively used for financial fraud detection. Besides the papers discussed below, related research utilizing neural networks can also be found in References [6–8]. Zhang et al. [9] propose the use of eFraudCom, a fraud detection system based on competitive graph neural networks (CGNNs). The three major tasks accomplished by eFraudCom are: (1) classifying user behavior by modeling the distributions of normal and fraudulent behaviors separately; (2) detecting fraudulent behaviors in the presence of new fraud patterns; and (3) maximizing the separability between normal and fraudulent behaviors to further improve CGNN. Experiments on two Taobao datasets (one of the largest e-commerce platforms) and two public datasets demonstrate that the proposed deep framework, CGNN, outperforms other baselines in detecting fraudulent behaviors.

A spatial–temporal attention-based graph network (STAGN) for credit card fraud detection is proposed by Cheng, Wang, Zhang, and Zhang [10]. They employ spatial–temporal attention on top of learned tensor representations, which are then fed into a 3D convolution network. The attentional weights are jointly learned in an end-to-end manner with 3D convolution and detection networks. The results show that STAGN outperforms other state-of-the-art baselines in both AUC (Area Under the Curve) and precision–recall curves. It also demonstrates superiority in detecting suspicious transactions, mining spatial and temporal fraud hotspots, and uncovering fraud patterns. Additionally, they integrate

the proposed STAGN into the fraud detection system as the predictive model and present the implementation details of each module in the system.

Tong and Shen [11] propose a graph neural network-based fraud detection model named HHLN-GNN that utilizes subgraph generators and neighborhood samplers to handle category imbalance. Furthermore, the model employs a self-attentive module and distinguishes between homogeneous and heterogeneous connections to reduce the artificiality of fraudulent nodes and more fully exploit the hidden information in transaction data. Experiments conducted on three real-world public benchmark datasets—YelpChi, Amazon, and Elliptic—demonstrate that their model significantly outperforms existing benchmark methods on various performance metrics, such as F1-macro, AUC (Area Under the Curve), and GMean, achieving improvements of 10%, 12.5%, and 17.3% on YelpChi, and 0.7%, 2.5%, and 4% on Amazon, respectively.

2.2. Other Ad Hoc Methods for Financial Fraud Data Detection

Other than the neural networks, various methods have been applied to financial fraud detection [12]. Some of them are discussed below. Li et al. [13] propose TA-Struc2Vec, a cutting-edge graph-learning algorithm specifically designed for detecting financial fraud over the Internet. This algorithm captures both the topological features and transaction amount details within a financial transaction network graph. By converting these features into low-dimensional dense vectors, it facilitates intelligent and efficient classification and prediction through the training of classifier models. Our proposed method significantly enhances the efficiency of financial fraud detection online, delivering improvements in precision, F1-score, and AUC (Area Under the Curve).

Khodabandehlou and Golpayegani [14] propose a system named FiFrauD, an advanced and scalable solution designed to identify manipulative behaviors in transaction streams using an unsupervised approach. This method transforms real-time transactions between traders into a sequence of graphs. By utilizing the density signals within these graphs, FiFrauD accurately identifies fraudulent traders, eliminating the need for supervised and semi-supervised learning techniques. Theoretical analysis establishes upper bounds on FiFrauD's effectiveness in detecting suspicious trades. Comprehensive experiments on five real-world datasets, containing both actual and synthetic labels, show significant accuracy improvements over existing fraud detection methods.

A system, named CoDetect, using both network and feature information for financial fraud detection is proposed by Huang, Mu, Yang, and Cai [15]. CoDetect introduces an approach to analyzing financial activities, uncovering fraud patterns and identifying suspicious characteristics. The framework also offers a more interpretable method for detecting fraud within sparse matrices, simultaneously identifying fraudulent activities and associated feature patterns. The experimental results, utilizing both synthetic and real-world datasets, demonstrate that CoDetect effectively detects fraud patterns and suspicious features.

Cui, Yan, and Wang [16] transform the traditional fraud detection method to a pseudo-recommender model. In this model, each individual is considered a pseudo-user, their behavior a pseudo-item, and the corresponding label a pseudo-rating. Leveraging the concept of collaborative filtering, information from similar individuals is used to create a behavior profile for each person. To uniformly handle heterogeneous attributes, the authors utilize an embedding-based method that simultaneously learns both attribute embeddings and individuals' behavior profiles within a shared latent space. To effectively utilize label information, the model is designed to align with pseudo-users' correct preference rankings for pseudo-items. This approach enables the model to clearly distinguish between fraudulent and legitimate activities. Extensive experiments on a real-world online banking

transaction dataset demonstrate that the proposed model not only surpasses benchmarks across all metrics but also can be combined with them to achieve even greater performance. Related research can also be found in References [17–19].

3. The Proposed System

This section introduces the proposed system, detailing (i) the system structure and (ii) the data fusion techniques used to integrate the results from seven machine-learning methods. Details of each system component will be provided in the subsequent sections. The final system can be found on GitHub [20].

3.1. The System Structure

Figure 1 shows the workflow of the proposed system, which combines results from seven machine-learning methods to enhance performance. It includes the following components:

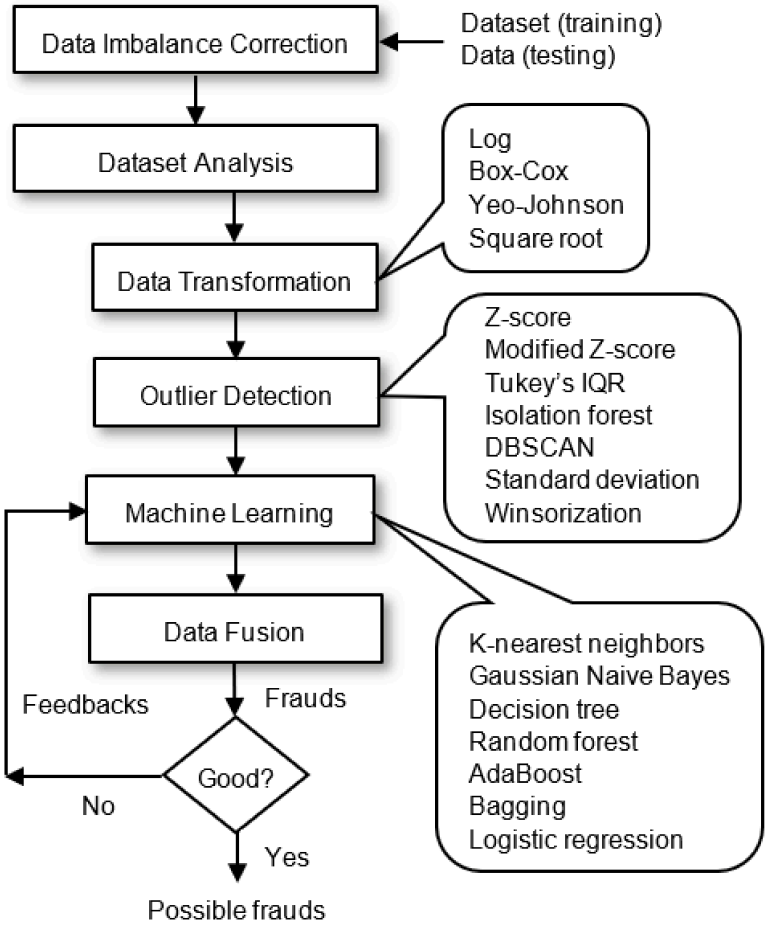

Figure 1. A workflow of the proposed system.

- Dataset: The data used in this research is sourced from Kaggle [21], encompassing legitimate and fraud transactions from 1 January 2019 to 31 December 2020.
- Data Imbalance Correction: Various methods are employed to balance the data before applying machine learning. In this research, the synthetic oversampling technique is used.
- Data Analysis: This component examines the dataset used, as the quality of the data significantly impacts machine-learning results.
- Data Transformation: Three methods are utilized to transform non-normal dependent variables into a more normal distribution.

- Outlier Detection: Seven methods (Z-score, modified Z-score, Tukey's IQR, isolation forest, DBSCAN, standard deviation, and Winsorization) are used to detect outliers.
- Machine Learning: Seven machine-learning methods (*k*NN, Gaussian Naïve Bayes, decision trees, random forest, AdaBoost, bagging, and logistic regression) are applied to detect financial fraud.
- Data Fusion: Data fusion (Hybrid FSAF) combines multiple data sources, leading to more consistent, accurate, and useful information, unlike relying on a single data source.

3.2. Data Fusion for Machine Learning

Instead of relying on a single data source, data fusion integrates multiple data sources to produce more consistent, accurate, and useful information. The data fusion function, Hybrid FSAF (Smoothed Fraud-Sensitive Adaptive Fusion), used in this research is self-adjusting through learning. Each of the seven machine-learning methods applied is associated with a weight, determined by comparing the results from each method to the expected outcomes. More accurate methods receive higher weights, and contribute more significantly to the final result. During the training phase, the final system is repeatedly applied to the dataset until an optimal detection rate is achieved. These final weights are then used in the testing phase. Additionally, users can provide feedback to the data fusion function, allowing it to adjust the weights accordingly.

4. Dataset Analysis

The dataset used in this research is discussed in this section, which includes three subsections: (i) the dataset used, (ii) data imbalance correction, and (iii) data analysis.

4.1. Dataset Used

This study uses a dataset available on Kaggle [21]. The dataset is a simulated credit card transaction dataset containing legitimate and fraudulent transactions for the period from 1 January 2019 to 31 December 2020. It covers the credit card transactions of 1000 customers interacting with a pool of 800 merchants. The data was generated using the Sparkov Data Generation tool created by Brandon Harris [22]. This simulation was run from 1 January 2019 to 31 December 2020. The files were then combined and converted into a standard format. The dataset includes columns such as Transaction Date, Transaction Time, Credit Card Number, Merchant, Category, Amount, First Name, Last Name, Gender, Street, City, State, Zip Code, Latitude, Longitude, City Population, Job, Date of Birth, Transaction Number, Unix Time, Merchant Latitude, Merchant Longitude, and Is Fraudulent or Not. This dataset contains the basic information typically collected by any payment processing company or financial institution. The training dataset consists of 1,296,675 records. The percentage distribution of legitimate and fraudulent transactions is shown in the pie chart in Figure 2.

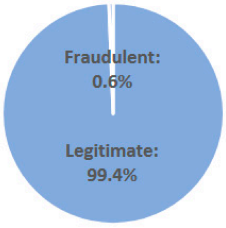

Figure 2. Distribution of legitimate and fraudulent transactions.

When creating machine learning models using this dataset, the high probability of accuracy, around 99%, can be attributed to the large number of legitimate transactions.

Even if machine learning models predict all transactions as legitimate, the accuracy would be 99.4% due to the dataset's imbalance. This imbalance is common in financial institutions, as fraudulent transactions are usually much fewer than legitimate ones. To enable machine learning models to train effectively, the dataset needs to be balanced to ensure accurate detection of both classes.

4.2. Data Imbalance Correction

Balancing an imbalanced dataset is crucial for accurate machine learning model training. Several methods can be employed to achieve this balance. Data-level correction methods involve manipulating the data to create a more balanced dataset. These methods are categorized into the following different types:

- Resampling Methods
  - Oversampling: This technique increases the minority class instances by replicating them to balance the dataset.
  - Undersampling: This technique reduces the number of majority class instances to match the minority class instances.
- Algorithm-Level Methods

Algorithm-level methods focus on modifying the learning algorithm rather than manipulating the dataset. These methods make models more sensitive to the minority class, improving their effectiveness in learning from imbalanced datasets.

This research uses the synthetic oversampling technique. An example of synthetic oversampling is SMOTE (Synthetic Minority Over-sampling Technique) [23], which creates synthetic instances of the minority class. Equation (1) presents the synthetic sampling.

$$x_{new} = x_i + \lambda \times (x_{i,knn} - x_i) \quad (1)$$

where x_i is a minority class instance, $x_{i,knn}$ is a k-nearest neighbors of x_i and λ is a random number between 0 and 1 that determines how far the new synthetic sample should be between x_i and $x_{i,knn}$.

4.3. Data Analysis

In this research, we will explore various machine learning techniques to identify fraudulent financial transactions, aiding financial institutions in managing those tasks that are otherwise nearly impossible to check manually. As with any machine learning model, it is essential to analyze the dataset to understand its properties. Preprocessing the dataset is necessary to eliminate unnecessary records and remove outliers. Dataset analysis is a crucial step in machine learning to avoid anomalies and ensure data quality. The dataset contains the latitude and longitude of transactions made by users. Figure 3 shows the scatter plot for legitimate transactions (red dots) and fraudulent transactions (blue dots) in the US.

The scatter plot analysis indicates that fraudulent transactions do not cluster in any specific geographical area; rather, they occur across various locations. To gain further insights, we analyzed the time of day when fraudulent transactions are most frequent, as shown in Figure 4. This information is crucial for financial institutions to enhance security measures and monitoring during high-risk periods. Our analysis reveals those fraudulent transactions peak during the middle of the night, specifically from 10 PM to 3 AM. This visualization suggests that banking institutions (BIs) should implement increased scrutiny and monitoring during late-night hours to mitigate the risk of fraud effectively.

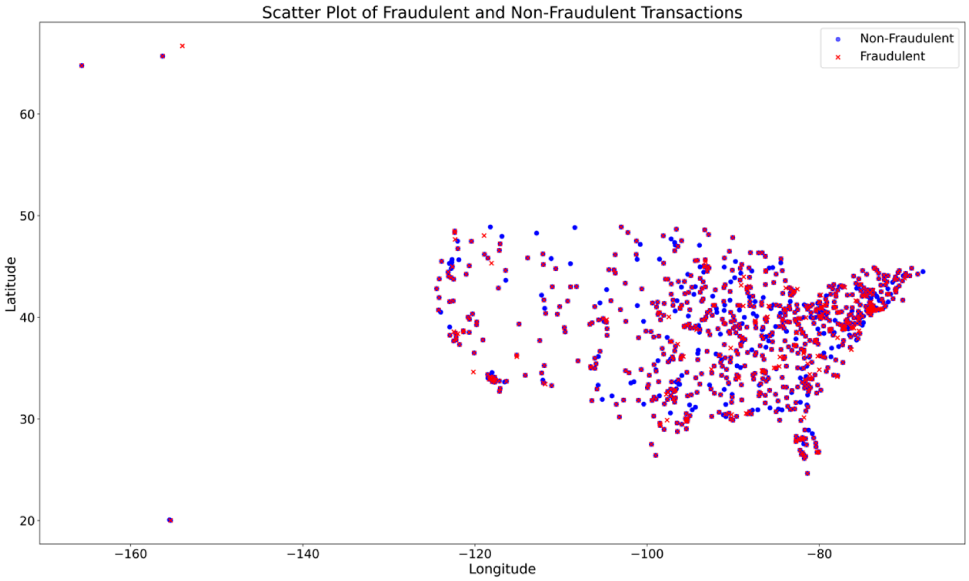

Figure 3. Scatter plot for legitimate transactions (red dots) and fraudulent transactions (blue dots) in the US.

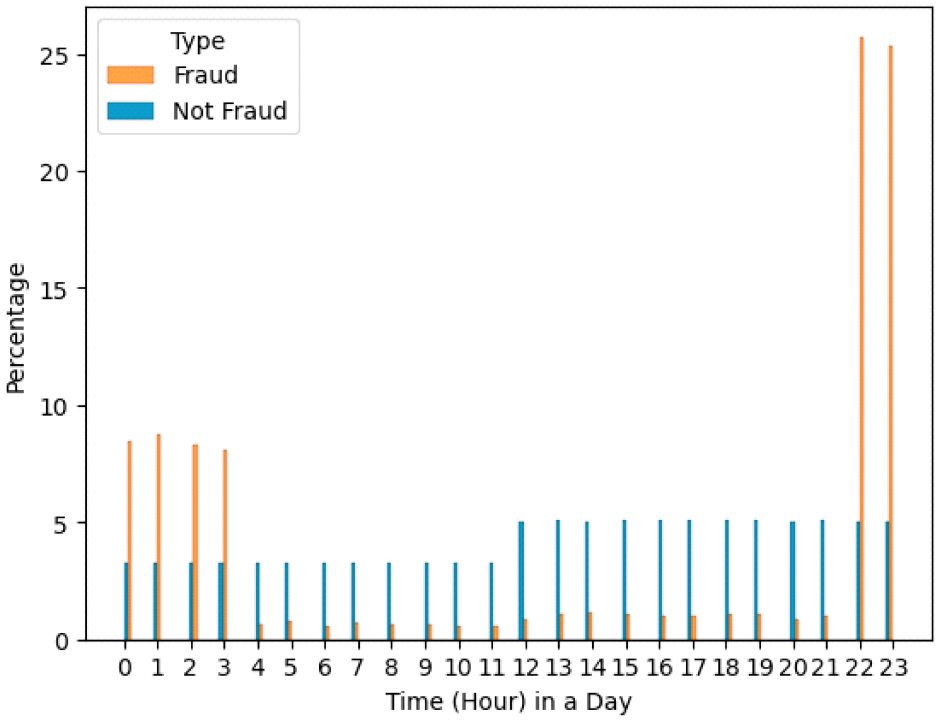

Figure 4. Hourly distribution of transactions.

Furthermore, it is also helpful to identify which days are more prone to fraudulent transactions, as shown in Figure 5. Analyzing the data reveals that weekends see a higher incidence of fraudulent transactions. This indicates that financial institutions should strengthen their security protocols and monitoring efforts during weekends to better safeguard against fraud.

Other analyses were conducted on the percentage of fraudulent transactions for each category of transactions, as shown in Figures 6 and 7. Point of sale (POS) transactions at grocery stores, online shopping, and miscellaneous internet activities tend to generate more fraudulent transactions compared to other categories.

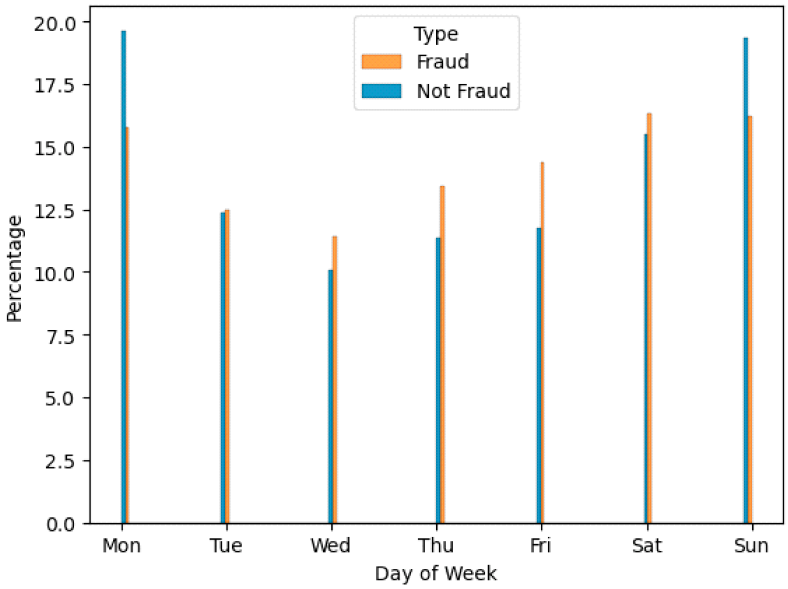

Figure 5. Daily distribution of transactions.

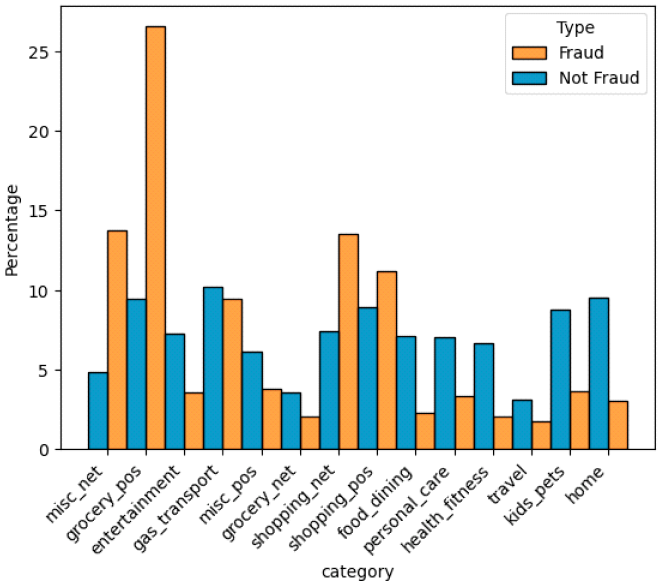

Figure 6. Categorical distribution of transactions.

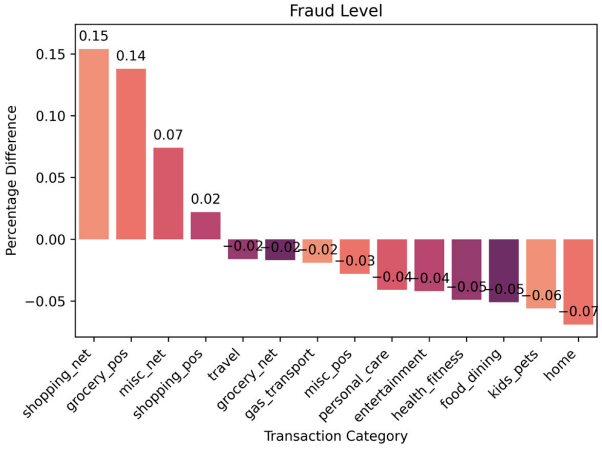

Figure 7. Fraud levels of transactions based on category.

When analyzing the correlation between transaction amounts and fraudulent activity, Figure 8 shows that higher transaction amounts are more likely to be associated with fraudulent transactions.

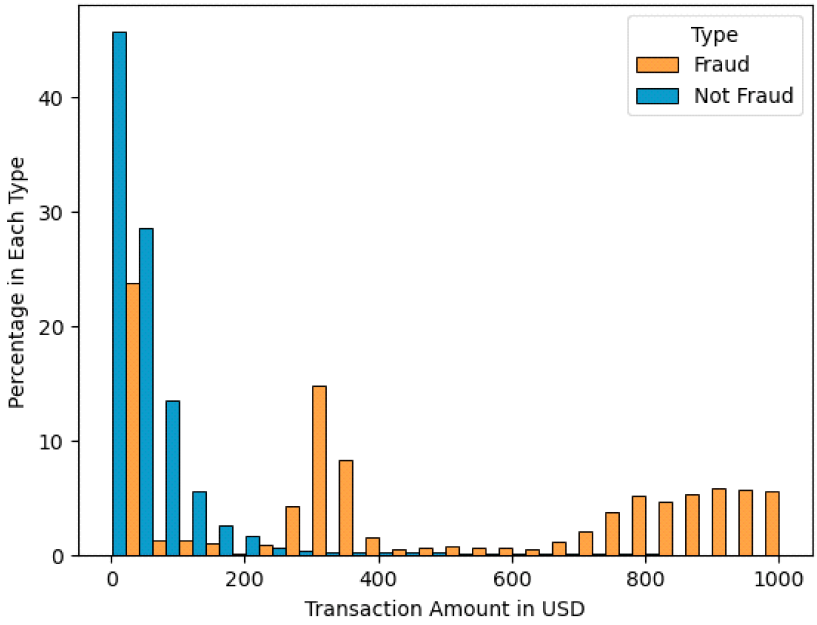

Figure 8. Distribution of transactions based on amount.

Additionally, Figure 9 shows an analysis of the monthly distribution of fraudulent transactions revealed that the period from January to June experiences a higher incidence of fraudulent activity compared to other months.

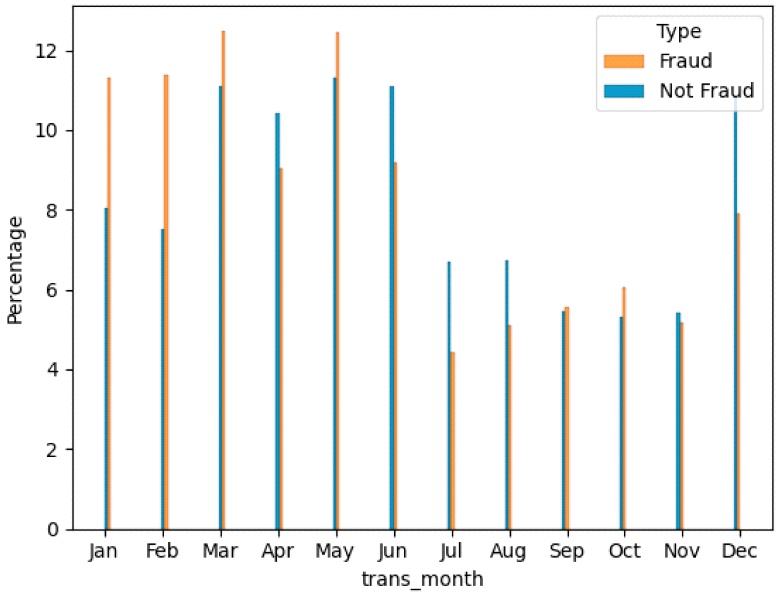

Figure 9. Monthly distribution of transactions.

Figure 10 shows the correlation between credit card holders' age and the occurrence of fraudulent transactions. The data shows that the density of both fraudulent and legitimate transactions is equal up to the age of 30. However, fraudulent transactions become more prevalent after the age of 50. This indicates that fraudulent activity tends to target both younger and older credit card holders.

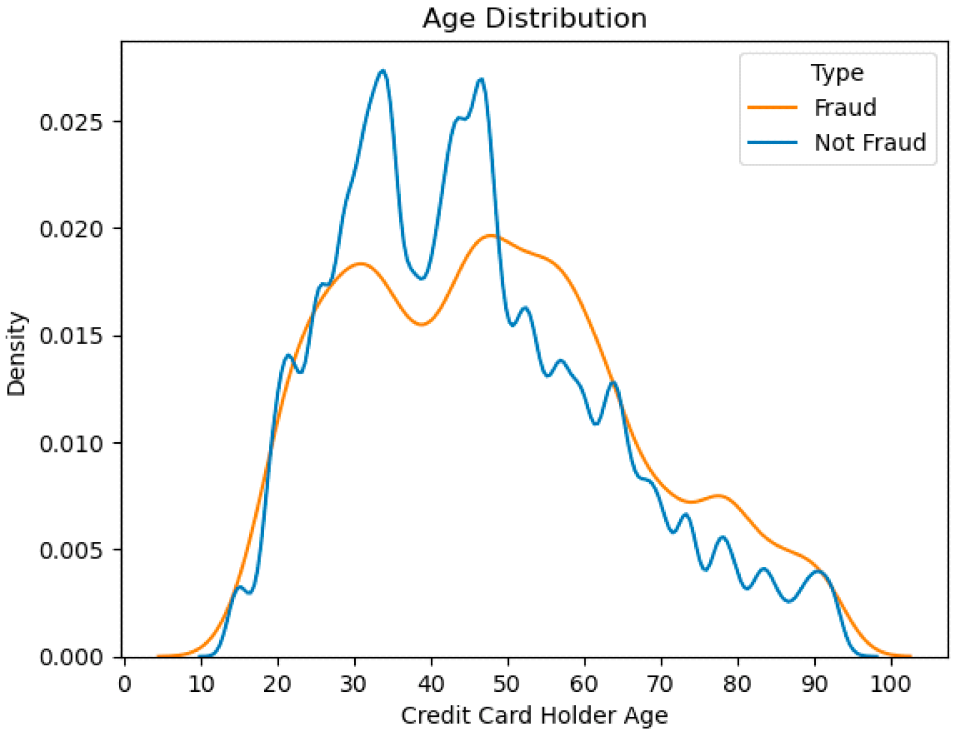

Figure 10. Credit card holder ages and transaction types.

Feature selection is performed on the dataset to identify which features significantly impact the classification of transactions as fraudulent or legitimate. This process determines which columns in the dataset have the most weight in distinguishing between fraudulent and real transactions. Table 1 and Figure 11 show the feature importance used in this research.

Table 1. Feature importance.

Feature	Importance
Amount	56.54%
Category	16.48%
City population	4.81%
Age	4.41%
Distance between user and merchant	3.67%
Job name of user	3.65%
Longitudinal distance between merchant and user	3.56%
Latitudinal distance between merchant and user	3.35%
Month	2.14%
Gender	0.93%
Transaction year	0.46%

The insights from analyzing the dataset are summarized in Table 2. It is important to note that the dataset used in this study differs from those typically utilized by banking institutions. Banking institutions often use private datasets that are not accessible for external exploration. However, most features examined in this study are generic and

applicable across different datasets. While the specific action items for financial institutions (FIs) may vary based on their unique datasets, the findings and methodologies presented here offer a general roadmap. Table 2 provides a comprehensive guide to the recommended action items, which FIs can adapt according to their specific dataset characteristics and requirements.

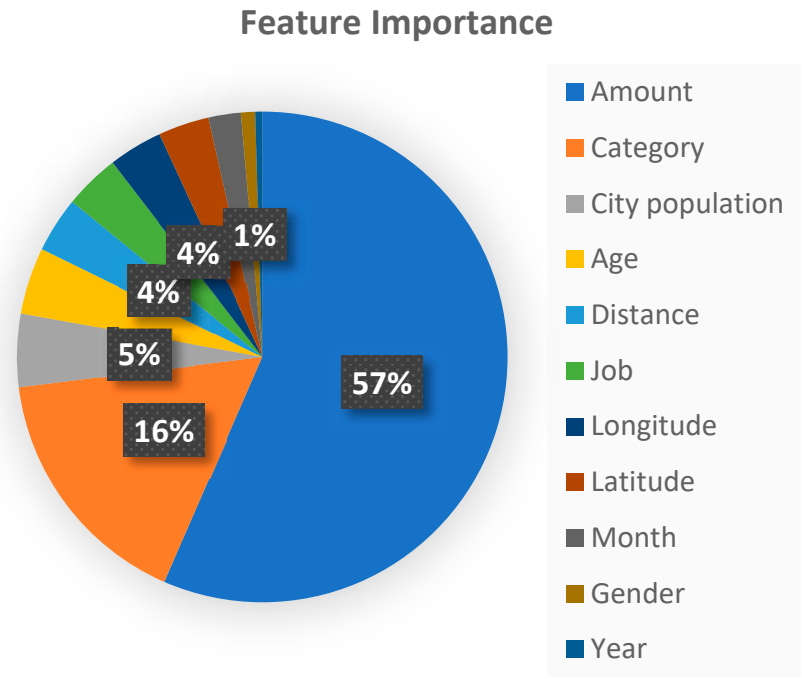

Figure 11. Feature importance with sample percentages.

Table 2. Dataset evaluation and financial institution action items.

Figure 3	
Data analyzed	Scatter plot for fraudulent and real transactions based on latitude and longitude.
Interpretation	There are no specific areas in the geographical region analyzed that are more prone to fraudulent transactions.
Action items for financial institutions	Financial institutions should check their own datasets to identify areas more prone to fraudulent transactions and implement increased monitoring in those areas.
Figure 4	
Data analyzed	Hourly distribution of transactions.
Interpretation	Fraudulent transactions are more frequent late at night, particularly from 10 PM to 3 AM.
Action items for financial institutions	Financial institutions should increase scrutiny during late-night hours.

Table 2. *Cont.*

Figure 5	
Data analyzed	Daily distribution of transactions.
Interpretation	Weekends are more prone to fraudulent transactions.
Action items for financial institutions	Financial institutions should add extra monitoring on weekends.
Figures 6 and 7	
Data analyzed	Categorical distribution of transactions.
Interpretation	Point of sale (POS) at grocery stores, online shopping, and miscellaneous internet transactions are more prone to fraud compared to others.
Action items for financial institutions	Financial institutions should add extra monitoring on grocery POS purchases and internet purchases.
Figure 8	
Data analyzed	Distribution of transactions based on amount.
Interpretation	Higher transaction amounts are more likely to be fraudulent.
Action items for financial institutions	Financial institutions should add extra monitoring for transactions involving amounts beyond a certain threshold.
Figure 9	
Data analyzed	Monthly distribution of transactions.
Interpretation	The period from January to June is more prone to fraud compared to other months.
Action items for financial institutions	Financial institutions should increase monitoring during the mentioned months.
Figure 10	
Data analyzed	Credit card holder age and transaction type.
Interpretation	The density of fraudulent transactions is equal to real ones up to the age of 30 and increases after the age of 50, indicating that young and very old credit card holders are more vulnerable to fraud.
Action items for financial institutions	Financial institutions should add extra monitoring for minor and elderly banking users.

Table 2. *Cont.*

Figure 11	
Data analyzed	Feature importance of the dataset calculation.
Interpretation	The transaction amount has a significant impact on determining if a transaction is fraudulent.
Action items for financial institutions	Financial institutions should analyze their own datasets to identify which features are most impactful in determining transaction legitimacy. In this case, it is the amount, but it could vary for different in-house datasets used by banking institutions.

Figure 12 demonstrates that the transaction amount has the maximum impact on determining whether a transaction is fraudulent or not. This finding is corroborated by Figure 8, which shows that higher transaction amounts are associated with higher levels of fraudulent transactions. To further explore the diversity of transaction types based on amount, we employed a UMAP (Uniform Manifold Approximation and Projection) plot, as shown in Figure 12. UMAP is a technique that converts multi-dimensional data into 2D or 3D visualizations for easier interpretation. In Figure 12, two distinct colors represent real and fraudulent transactions. The UMAP plot uses the transaction amount and its correlation with transaction type. The clear separation between the purple (real transactions) and red (fraudulent transactions) clusters indicates that UMAP effectively distinguishes transaction types based on the selected features, particularly the transaction amount. The elongated shape of the real transaction cluster suggests greater variability in their features, while the densely packed red cluster indicates more consistent feature patterns among fraudulent transactions. This consistency supports the observation that higher transaction amounts are more likely to be fraudulent. The minimal overlap between the purple and red clusters highlights the strong discriminatory power of the features used in the UMAP plot, confirming their effectiveness in distinguishing between fraudulent and real transactions.

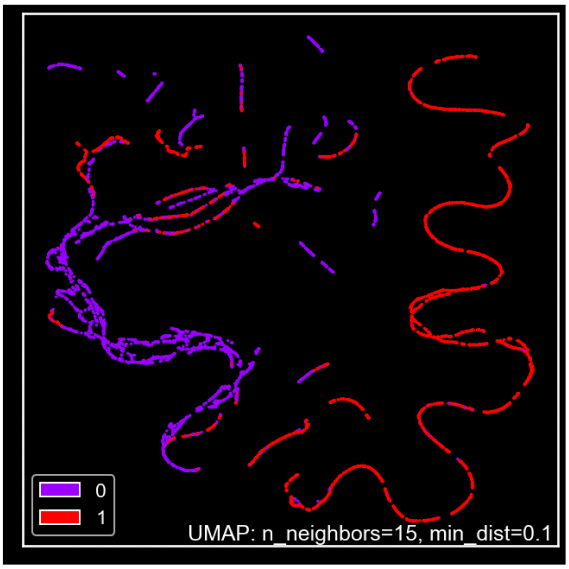

Figure 12. UMAP plot of transaction types.

5. The Proposed Methods

Outliers are the data points significantly different from the rest of data in a dataset. They can negatively impact model performance and accuracy, so removing outliers is a crucial step in developing any machine-learning model. Each of these methods helps identify and handle outliers to ensure the dataset is clean and reliable for training accurate machine learning models.

5.1. Data Transformations

However, the outlier detection methods may require the data to follow a Gaussian distribution. The dataset used for financial fraud detection does not follow a Gaussian distribution. Below are the methods used to transform non-normal dependent variables into a more normal distribution [24]:

- Log Transformation

This transformation applies to positive data only and aims to make the data more normally distributed.

$$\mathrm{Y}' = \log(\mathrm{y}) \tag{2}$$

- Box–Cox Transformation

This transformation [25] is used to make the data more normally distributed and is only applicable to positive values. The data transformation methods are as follows:

$$\mathrm{y}(\lambda) = \begin{cases} \frac{\mathrm{y}^{\lambda-1}}{\lambda} & \text{if } \lambda \neq 0 \\ ln(\mathrm{y}) & \text{if } \lambda = 0 \end{cases} \tag{3}$$

where λ is the transformation parameter.

- Yeo–Johnson Transformation

This transformation can apply to both positive and negative values. The transformation methods are as follows:

$$y(\lambda) = \begin{cases} \frac{((y+1)^{\lambda}-1)}{\lambda}, & if\ y \geq 0,\ \lambda \neq 0 \\ \ln(y+1) & if\ y \geq 0\,, \quad \lambda = 0 \\ \frac{-(|y|+1)^{2-\lambda}-1}{2-\lambda} & if\ y < 0, \quad \lambda \neq 2 \\ -\ln(|y|+1) & if\ y < 0, \quad \lambda = 2 \end{cases} \tag{4}$$

where λ is the transformation parameter.

- Square Root Transformation

This transformation applies to non-negative values. The transformation methods are as follows:

$$\mathrm{y}' = \sqrt[2]{\mathrm{y}} \tag{5}$$

5.2. Outlier Detection Methods

The following methods [26] are applied to outlier detection in this research: (1) Z-score method, (2) modified Z-score method, (3) Tukey's IQR method, (4) isolation forest method, (5) DBSCAN (Density-Based Spatial Clustering of Applications with Noise), (6) standard deviation method, and (7) Winsorization (Percentile Capping) method. A brief explanation of each method is provided as follows:

- Z-score Method

The Z-score method [27] identifies outliers in a dataset by determining how many standard deviations a data point is from the mean. This can be applied to any numerical

data. While this method is useful for normally distributed data, it is less effective for non-normally distributed data. The Z-score for each data point can be calculated as follows:

$$Z_i = \frac{y_i - \mu}{\sigma} \tag{6}$$

where y_i is the ith data point and μ is the mean and is the standard deviation:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{7}$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \mu)^2} \tag{8}$$

A common threshold for the Z-score is ± 3 for normally distributed data, and any data point outside this range is considered an outlier.

- Modified Z-score Method

The modified Z-score is used for any numerical data and is particularly useful for data that does not follow a normal distribution. The modified Z-score for each data point can be calculated as follows:

$$M_i = \frac{0.6745 \times (y_i - M)}{MAD} \tag{9}$$

where M is the median and MAD is the Mean Absolute Deviation, which is calculated as

$$MAD = \text{median}(|y_i - M|) \tag{10}$$

The constant 0.6745 makes the modified Z-score comparable to the Z-score for normally distributed data.

- Tukey's IQR Method

This method is useful for skewed distributions and datasets with outliers and can be applied to any numerical data. It identifies outliers using the interquartile range (IQR) and is effective for datasets that do not follow a normal distribution:

$$IQR = Q_3 - Q_1 \tag{11}$$

where Q_3 is the 75th percentile of the data and Q_1 is the 25th percentile of the data. The method to calculate outliers using these boundaries is given below:

$$\text{Lower bound: } Q_1 - 1.5 \times IQR \tag{12}$$

$$\text{Upper bound: } Q_3 + 1.5 \times IQR \tag{13}$$

- Isolation Forest Method

This method is effective in high-dimensional datasets where traditional methods may have issues. An anomaly score can be calculated using the following formula:

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{14}$$

where $E(h(x))$ is the average path length of x across the trees, and $c(n)$ is the average path length in a binary search tree of n points. The anomaly score $s(x,n)$ ranges between 0 and 1, where a score close to 1 indicates an outlier and a score less than 0.5 indicates a normal point.

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

This method identifies clusters and outliers in a dataset based on density. It is applicable to any numerical data.

- Standard Deviation Method

This method identifies outliers in a dataset based on how many standard deviations a data point is away from the mean.

$$\text{Lower Bound}: \ \mu - k\,\sigma \tag{15}$$

$$\text{Upper Bound}: \ \mu + k\,\sigma \tag{16}$$

where k is a threshold, and μ and σ are the mean and standard deviation, respectively.

- Winsorization Method (Percentile Capping)

This method clips the extreme values of the dataset at certain percentiles. Instead of removing the outliers, it replaces them with the lower and upper percentile values.

The outlier detection methods using four transformations are shown in Table 3, where the symbol * means that, for the isolation forest method, all four transformations return the same number of outliers because the percentage of impurities set in the model is 5%.

Table 3. Summary of the outlier detection methods.

Outlier Detection Method	Transformation	Number of Outliers Detected
Z-score Method	Log Transformation	22,502
Z-score Method	Box-Cox Transformation	4399
Z-score Method	Yeo-Johnson Transformation	4080
Z-score Method	Square Root Transformation	64,534
Modified Z-score Method	Log Transformation	21,161
Modified Z-score Method	Box-Cox Transformation	1064
Modified Z-score Method	Yeo-Johnson Transformation	868
Modified Z-score Method	Square Root Transformation	78,256
Tukey's IQR Method	Log Transformation	57,737
Tukey's IQR Method	Box-Cox Transformation	6140
Tukey's IQR Method	Yeo-Johnson Transformation	5496
Tukey's IQR Method	Square Root Transformation	214,118
Standard Deviation Method	Log Transformation	22,502
Standard Deviation Method	Box-Cox Transformation	4399
Standard Deviation Method	Yeo-Johnson Transformation	4080
Standard Deviation Method	Square Root Transformation	64,534
Isolation Forest Method	Log Transformation	64,459 *
Isolation Forest Method	Box-Cox Transformation	64,459 *
Isolation Forest Method	Yeo-Johnson Transformation	64,459 *
Isolation Forest Method	Square Root Transformation	64,458 *
DBSCAN Method	Log Transformation	2589
DBSCAN Method	Box-Cox Transformation	102,758
DBSCAN Method	Yeo-Johnson Transformation	80,890
DBSCAN Method	Square Root Transformation	411,144
Winsorization Method	Log Transformation	347,351
Winsorization Method	Box-Cox Transformation	347,351
Winsorization Method	Yeo-Johnson Transformation	347,351
Winsorization Method	Square Root Transformation	347,351

One example illustrating the outliers is given in Figure 13, which shows the outliers in a logarithmic plotting between distance and amount using DBSCAN.

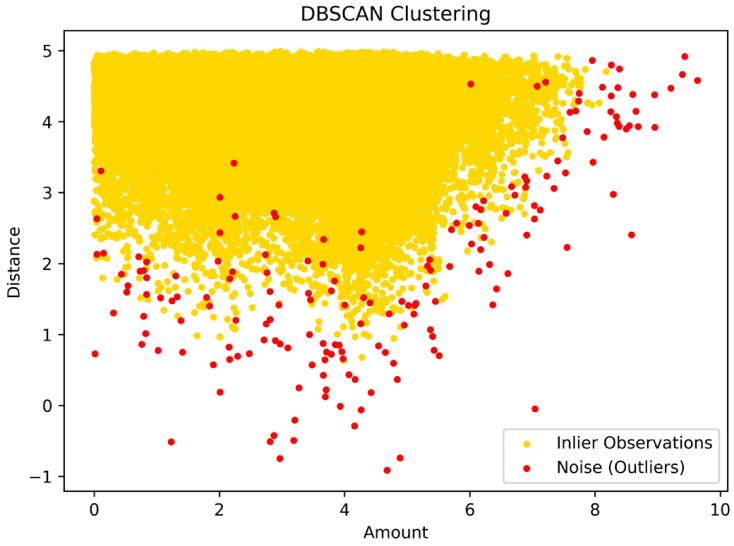

Figure 13. Outliers in logarithmic plotting between distance and amount using DBSCAN.

5.3. Fusion Method (Hybrid Fraud-Sensitive Adaptive Fusion)

Beyond individual classifier outputs, this research proposes a composite fusion mechanism that integrates soft voting with a Smoothed Fraud-Sensitive Adaptive Fusion (FSAF) component. Conventional soft voting computes a simple mean of fraud probability scores across all participating classifiers. However, this equal-weighted averaging fails to account for the varying degrees of fraud detection capability among classifiers. The proposed Hybrid FSAF framework overcomes this shortcoming by computing transaction-level weights that reflect both a classifier's historical reliability on fraud cases and its prediction confidence for the specific transaction under evaluation. Financial fraud datasets present a fundamental class imbalance challenge, with genuine transactions vastly outnumbering fraudulent ones. A robust fusion strategy must therefore be specifically designed to enhance recall of the minority fraud class rather than optimizing for overall accuracy. The Hybrid FSAF approach is motivated by precisely this requirement.

Consider an ensemble comprising base classifiers. For a given transaction, the probability of fraud as estimated by the classifier is expressed as

$$pk(x_i) = \mathrm{P}(y_i = 1|x_i) \tag{17}$$

Here, $y_i = 1$ indicates a fraudulent transaction, while corresponds to a legitimate one. As baseline aggregation, soft voting computes the arithmetic mean of individual classifier fraud probabilities:

$$S_{\text{soft}}(x_i) = \frac{1}{K}\sum_{k=1}^{K} p_k(x_i) \tag{18}$$

This score provides a stable but undifferentiated estimate of fraud likelihood. To quantify how well each classifier performs specifically on the fraud class, a per-classifier reliability score is defined as its fraud-class F1 score:

$$R_k = F1_{\text{fraud},\,k} \tag{19}$$

Collectively, these scores form a reliability vector, which captures the relative fraud detection strength of each model. Rather than applying a hard probability cutoff, a sigmoid

smoothing function is used to produce a continuous confidence measure for each classifier's fraud estimate:

$$\Phi_k(x_i) = \frac{1}{1 + e^{-\lambda(p_k(x_i)-c)}} \tag{20}$$

This parameter governs how sharply the function responds to deviations from the center point. Multiplying this factor by raw probability yields a smoothed fraud estimate:

$$\widetilde{p}_k(x_i) = p_k(x_i) \cdot \Phi_k(x_i) \tag{21}$$

This transformation retains probabilistic ordering while suppressing low-confidence predictions. Transaction-specific weights are derived by jointly considering each classifier's reliability score and its smoothed probability output:

$$w_k(x_i) = \frac{R_k \cdot \widetilde{p}_k(x_i)}{\sum_{j=1}^{K} R_j \cdot \widetilde{p}_j(x_i) + \varepsilon} \tag{22}$$

The term is a small stabilizing constant that prevents numerical instability from division by zero. Through this formulation, classifiers that are both fraud-reliable and highly confident for a given transaction naturally receive larger weights. The smoothed FSAF score is obtained by taking the weighted combination of raw fraud probabilities using the adaptive weights derived above:

$$S_{\text{FSAF}}(x_i) = \sum_{k=1}^{K} w_k(x_i) \cdot p_k(x_i) \tag{23}$$

The two scoring components—soft voting and FSAF—are merged through a tunable linear combination:

$$S_{\text{hybrid}}(x_i) = \alpha \cdot S_{\text{soft}}(x_i) + (1 - \alpha) \cdot S_{\text{FSAF}}(x_i) \tag{24}$$

The mixing coefficient governs the relative contribution of each component. Setting, for instance, places greater emphasis on the stable soft voting estimate while still incorporating the fraud-sensitive FSAF adjustment.

The Hybrid FSAF approach offers several advantages over conventional ensemble fusion techniques. First, classifier contributions are not fixed but instead vary per transaction, enabling the model to leverage whichever classifiers are most suitable for each individual case. Second, the sigmoid smoothing function eliminates sharp probability cutoffs, ensuring that borderline predictions are handled in a continuous and well-calibrated manner. Third, by explicitly incorporating fraud-class F1-scores into the weighting scheme, the method prioritizes fraud detection performance rather than overall classification accuracy—a crucial distinction when dealing with severely imbalanced data. These characteristics collectively make the Hybrid FSAF method a principled and effective fusion strategy for real-world financial fraud detection scenarios.

6. Experimental Results

This section shows the experimental results from this research, including confusion and performance matrices.

6.1. Confusion Matrices

Various machine learning (ML) methods can be employed to identify fraudulent transactions. The ML models used in this research are presented below:

- K-nearest neighbors;
- Gaussian Naive Bayes;
- Decision tree classifier;
- Random forest classifier;
- AdaBoost classifier;
- Bagging classifier;
- Logistic regression.

Since the dataset is highly imbalanced, the synthetic oversampling technique SMOTE (Synthetic Minority Over-sampling Technique) [23] is used to make the dataset more balanced. The model is trained on the training dataset mentioned above and tested on the testing dataset. Figure 14 shows the confusion matrices, where the values of quadrant 1 (top-left), quadrant 2 (top-right), quadrant 3 (bottom-left), and quadrant 4 (bottom-right) are FN (false negative), TN (true negative), FP (false positive), and (true positive).

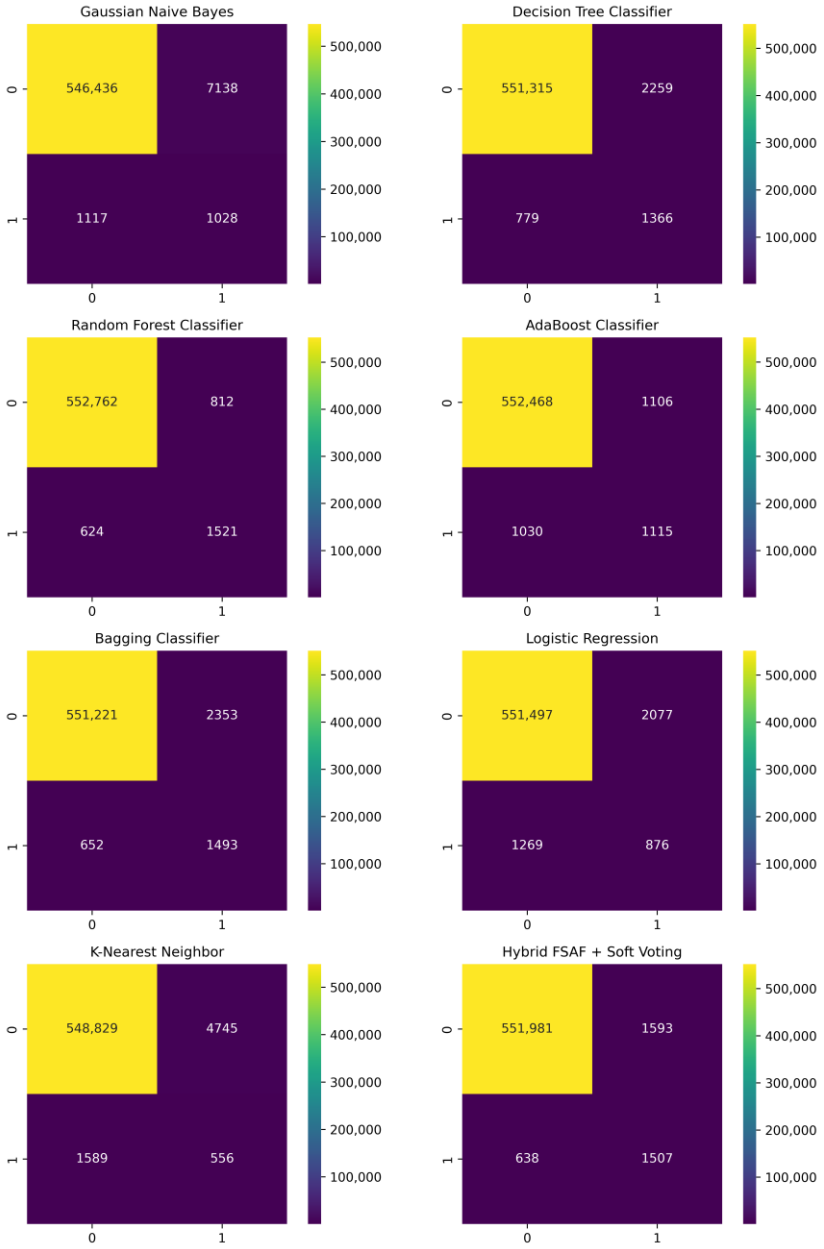

Figure 14. Confusion matrices from the eight ML models used.

Figure 15 shows the curve of ROC (receiver operating characteristic), where the AUC is Area Under the Curve, a graphical representation of the performance of a binary classification model at various classification thresholds. The orange dashed diagonal marks random performance (AUC = 0.50). Any curve above this baseline outperforms random guessing.

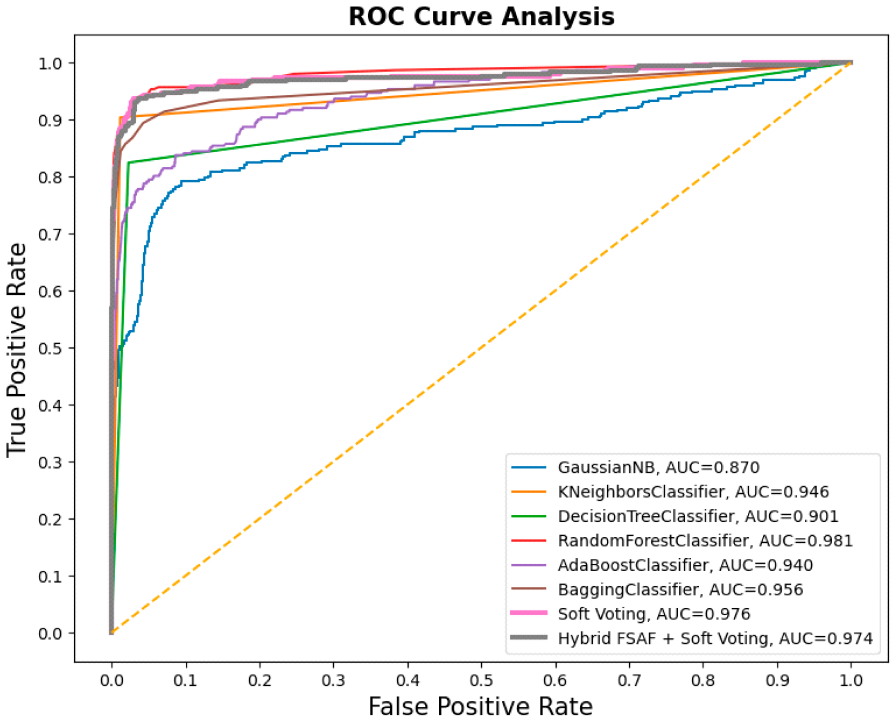

Figure 15. ROC curve of the eight ML used.

6.2. Performance Matrices

Table 4 shows the performance matrices, and Equations (25), (26), and (27) are used to calculate the *precision*, *recall*, and *F1-score*, respectively.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{25}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{26}$$

$$\text{F1} - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{27}$$

All the machine learning models exhibit an accuracy of 99% or higher. However, this high accuracy is misleading due to the imbalanced nature of the dataset, where the testing dataset contains a higher number of true transactions. All models show near-perfect performance for real transactions, with precision, recall, and F1-score close to or at 1. The detection accuracy of the models for fraudulent transactions varies.

The Random Forest classifier stands out as the highest performer in detecting fraudulent transactions, with precision, recall, and F1-score of 0.65, 0.71, and 0.68, respectively. This indicates that it not only identifies a good number of fraudulent transactions but also limits false positives effectively. AdaBoost and decision tree classifiers also perform well in detecting fraudulent transactions. K-nearest neighbor and Gaussian Naive Bayes show lower performance in detecting fraudulent transactions and may not be suitable for this task. Bagging Classifier and Logistic Regression demonstrate moderate performance in de-

tecting fraudulent transactions. Hyperparameter tuning for the Random Forest, AdaBoost, and decision tree models can be considered to improve accuracy further.

For the Random Forest classifier, a precision value of 0.65 indicates that 65% of the transactions predicted as fraudulent were indeed fraudulent, highlighting the model's ability to minimize false positives effectively. The recall value of 0.71 demonstrates that 71% of actual fraudulent transactions were correctly identified, showcasing the model's effectiveness in detecting fraud. An F1-score of 0.68 reflects a good balance between precision and recall, indicating that the model maintains an effective equilibrium between accurately identifying fraudulent transactions and minimizing false positives. This balance is crucial in the context of fraud detection, where both false positives and false negatives can have significant repercussions.

Table 4. Performance matrices of the seven ML models used.

Model	Accuracy	Precision (Real)	Recall (Real)	F1 (Real)	Precision (Fraud)	Recall (Fraud)	F1 (Fraud)	ROC-AUC	PR-AUC
Gaussian Naive Base	~0.96	~0.99	~0.96	~0.97	~0.15–0.20	~0.50	~0.25	0.87	0.249
K-Nearest Neighbor	~0.98	~0.99	~0.98	~0.98	~0.35–0.45	~0.60–0.70	~0.45–0.50	0.962	0.436
Decision Tree	~0.97	~0.99	~0.97	~0.98	~0.20–0.30	~0.70	~0.30–0.40	0.894	0.211
Random Forest	~0.99	~1.00	~0.99	~0.99	~0.80	~0.80	~0.80	0.983	0.842
AdaBoost	~0.98	~0.99	~0.98	~0.98	~0.50	~0.75	~0.60	0.941	0.508
Bagging	~0.99	~1.00	~0.99	~0.99	~0.60–0.70	~0.75–0.80	~0.65–0.70	0.959	0.645
Hybrid	0.989	~1.00	~0.99	~0.99	0.467	0.844	0.601	0.975	0.772

6.3. Data Fusion

During the training phase of this research, seven different machine-learning methods are applied to a dataset for financial fraud detection and each method has its own pros and cons. In order to produce more accurate results, data fusion is subsequently used to integrate the results from the methods. Each method has a weight, which is proportional to its accuracy. During the testing phase, the weights are then used to check the incoming data to see whether they include any financial frauds. Experiments show the integrated method outperforms each individual one.

Figure 16 shows the proposed Hybrid FSAF model demonstrates strength in two critical metrics: fraud recall and precision–recall Area Under the Curve (PR-AUC). These gains are especially meaningful in the context of heavily imbalanced financial datasets, where fraudulent transactions represent only a small fraction of all records. A high recall ensures that fewer fraud cases go undetected, while a strong PR-AUC reflects a reliable ranking of suspicious transactions even under severe class imbalance. This makes the Hybrid approach well-suited for high-stakes fraud detection environments where the cost of missing a fraudulent transaction far outweighs the inconvenience of a false alarm. Concretely, the Hybrid FSAF model attains a fraud recall of 0.84 and the highest PR-AUC of 0.772 among all evaluated approaches. While Random Forest achieves a competitive PR-AUC as a standalone classifier, the Hybrid model delivers a more favorable trade-off between precision and recall. This balance is essential in operational fraud detection, where neither excessive false negatives nor an unmanageable volume of false positives is acceptable. The results therefore support the Hybrid fusion strategy as the most practically viable solution for real-world deployment.

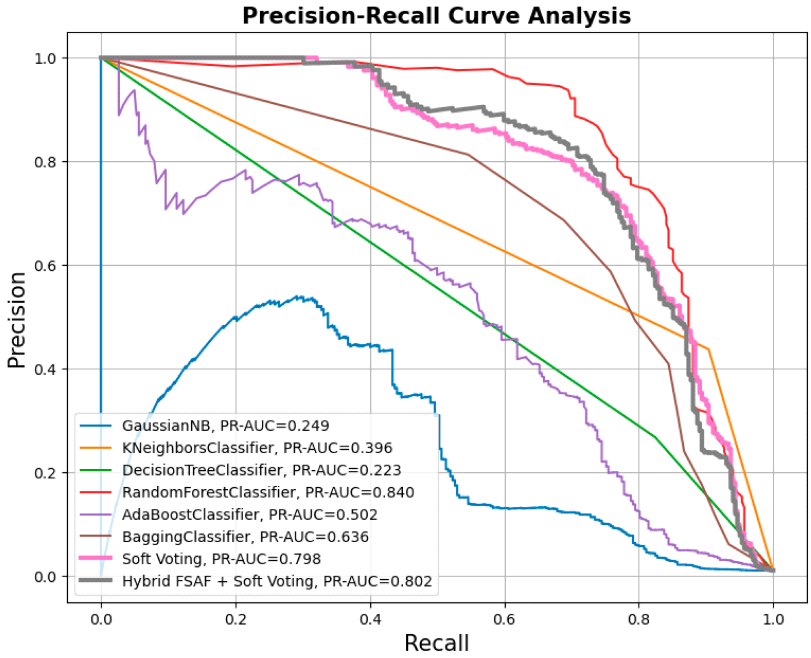

Figure 16. Precision–recall curve of the eight ML used.

The findings from the experimental evaluation confirm that the proposed Hybrid FSAF framework delivers meaningful gains in fraud detection performance relative to standalone machine learning classifiers. Conventional models—including Gaussian Naïve Bayes, k-nearest neighbors, decision tree, Random Forest, AdaBoost, and Bagging—record high overall accuracy figures, yet this apparent strength is largely a product of the dataset's class distribution, where legitimate transactions heavily outnumber fraudulent ones. Consequently, overall accuracy provides a misleading picture of how well these models actually detect fraud. When evaluated specifically on fraud identification, the Hybrid model records a notably higher recall than most competing classifiers, reflecting its greater capacity to flag genuine fraud cases rather than overlooking them. Models such as Random Forest perform well in isolation and demonstrate strong PR-AUC scores, but their decision boundaries remain fixed throughout inference and do not adapt to the characteristics of individual transactions. The Hybrid approach, by contrast, recalibrates each classifier's influence on a per-transaction basis, weighing both its track record on fraud cases and its confidence level for the specific instance being evaluated. This flexibility allows the model to capture subtle and varied fraud patterns that rigid, static classifiers tend to miss.

The precision–recall curve analysis reinforces these findings. Across the moderate-to-high recall range—the operating region most relevant to practical fraud detection systems—the Hybrid model sustains competitive precision, demonstrating that improved fraud sensitivity does not come at the cost of a sharp drop in precision within this critical zone. As is characteristic of severely imbalanced classification problems, precision naturally declines as recall is pushed toward its upper limit across all models. Nevertheless, the Hybrid approach maintains a more favorable precision–recall trade-off throughout the region where fraud detection systems are most commonly deployed. The performance advantage of the Hybrid model stems directly from its adaptive fusion design. Whereas individual classifiers depend entirely on their own internal learning mechanisms, the Hybrid method consolidates the probabilistic outputs of multiple models through a fraud-aware weighting scheme augmented by smooth confidence modulation. This architecture allows the ensemble to amplify contributions from classifiers with stronger fraud detection credentials while using probabilistic aggregation to maintain overall prediction stability. The result is a model that is both sensitive to fraudulent activity and robust against noise—

qualities that are essential in real-world financial fraud detection, where the consequences of undetected fraud are severe.

7. Conclusions

Financial fraud is rampant these days. The population of baby boomers in the US exceeded 70 million [28] as of 2023. Having worked hard all their lives, they have accumulated significant wealth, making them prime targets for scammers. Baby boomers are often popular and easy targets due to their kindness and trust in others. Criminals employ various elder fraud schemes, such as government impersonation and sweepstake scams, to deceive seniors [4]. Despite numerous efforts to mitigate elder fraud, the problem persists as no method has proven entirely successful. Detecting fraudulent transactions is challenging, even for humans, who struggle to differentiate between legitimate and fraudulent activities. This is where machine learning can make a significant impact, as it excels at addressing such complex problems.

In this research, seven machine-learning methods are employed to detect fraudulent credit-card transactions. Instead of relying on a single method, the data fusion (Hybrid FSAF Smoothed Fraud-Sensitive Adaptive Fusion) is used to integrate multiple data sources, producing more consistent, accurate, and useful information. The data fusion function is self-adjusted through learning. Each of the seven machine-learning methods is assigned a weight based on its accuracy, determined by comparing the results from each method to the expected outcomes. More accurate methods receive higher weights, and play a greater role in the final result. During the training phase, the system is applied repeatedly to the dataset until an optimal detection rate is achieved. These final weights are then used in the testing phase. The experimental results demonstrate that these methods can be effectively employed for detecting fraudulent credit-card transactions.

The experimental results indicate that the proposed method effectively detects financial fraud but has room for improvement. In this research, a simple data fusion technique was employed. Future work will involve adopting reconfigurable data fusion by sending new parameters to the machine-learning and data-fusion algorithms. Additionally, ensemble learning, which often yields higher overall accuracy than individual methods, may be utilized instead of data fusion. Ensemble learning combines predictions from multiple machine-learning models to improve predictive performance. Federated learning, which involves multiple clients collaboratively training a model, will also be explored. Each client model is updated locally, and the parameters from these models are used to collaboratively update the model parameters on a central server.

Author Contributions: Conceptualization, S.E.V.S.P. and W.-C.H.; methodology, S.E.V.S.P.; software, S.E.V.S.P.; validation, S.E.V.S.P.; formal analysis, S.E.V.S.P.; investigation, S.E.V.S.P.; resources, S.E.V.S.P.; data curation, S.E.V.S.P.; writing—original draft preparation, S.E.V.S.P.; writing—review and editing, W.-C.H.; visualization, S.E.V.S.P.; supervision, W.-C.H.; project administration, W.-C.H.; funding acquisition, W.-C.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding. The APC was funded by W.-C.H.

Data Availability Statement: The original data presented in the study are openly available in GitHub at https://github.com/sanjaikanth/FinancialFraudDetection (accessed on 24 June 2026).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Federal Trade Commission. Consumer Sentinel Network, Data Book 2023. February 2024. Available online: https://www.ftc.gov/system/files/ftc_gov/pdf/CSN-Annual-Data-Book-2023.pdf (accessed on 11 January 2025).
2. Morgan, R.E.; Tapp, S.N. Examining financial fraud against older adults. *Natl. Inst. Justice J.* **2024**, *20*, 307792. Available online: https://nij.ojp.gov/topics/articles/examining-financial-fraud-against-older-adults (accessed on 12 December 2025).
3. GCUFCU, Gulf Coast Educators Federal Credit Union. Top Scams That Identity Thieves Use Against Senior Citizens. 2024. Available online: https://www.gcefcu.org/top-scams-that-identity-thieves-use-against-senior-citizens (accessed on 5 January 2026).
4. Nordwall, M.D. Elder Fraud Report 2023. Federal Bureau of Investigation. 2023. Available online: https://www.ic3.gov/AnnualReport/Reports/2023_IC3ElderFraudReport.pdf (accessed on 13 June 2025).
5. Burke, J.; Kieffer, C.; Mottola, G.; Perez-Arce, F. Can educational interventions reduce susceptibility to financial fraud? *J. Econ. Behav. Organ.* **2022**, *198*, 250–266. [CrossRef]
6. Pozzolo, A.D.; Boracchi, G.; Caelen, O.; Alippi, C.; Bontempi, G. Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 3784–3797. [CrossRef] [PubMed]
7. Motie, S.; Raahemi, B. Financial fraud detection using graph neural networks: A systematic review. *Expert Syst. Appl.* **2024**, *240*, 122156. [CrossRef]
8. Thilagavathi, M.; Saranyadevi, R.; Vijayakumar, N.; Selvi, K.; Anitha, L.; Sudharson, K. AI-driven fraud detection in financial transactions with graph neural networks and anomaly detection. In Proceedings of the 2024 International Conference on Science Technology Engineering and Management (ICSTEM), Coimbatore, India, 26–27 April 2024; pp. 1–6. [CrossRef]
9. Zhang, G.; Li, Z.; Huang, J.; Wu, J.; Zhou, C.; Yang, J.; Gao, J. eFraudCom: An e-commerce fraud detection system via competitive graph neural networks. *ACM Trans. Inf. Syst. (TOIS)* **2022**, *40*, 47. [CrossRef]
10. Cheng, D.; Wang, X.; Zhang, Y.; Zhang, L. Graph neural network for fraud detection via spatial-temporal attention. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 3800–3813. [CrossRef]
11. Tong, G.; Shen, J. Financial transaction fraud detector based on imbalance learning and graph neural network. *Appl. Soft Comput.* **2023**, *149*, 110984. [CrossRef]
12. Aros, L.H.; Bustamante Molano, L.X.; Gutierrez-Portela, F.; Hernandez, J.J.M.; Barrero, M.S.R. Financial fraud detection through the application of machine learning techniques: A literature review. *Humanit. Soc. Sci. Commun.* **2024**, *11*, 1130. [CrossRef]
13. Li, R.; Liu, Z.; Ma, Y.; Yang, D.; Sun, S. Internet financial fraud detection based on graph learning. *IEEE Trans. Comput. Soc. Syst.* **2023**, *10*, 1394–1401. [CrossRef]
14. Khodabandehlou, S.; Golpayegani, A.H. FiFrauD: Unsupervised financial fraud detection in dynamic graph streams. *ACM Trans. Knowl. Discov. Data* **2024**, *18*, 111. [CrossRef]
15. Huang, D.; Mu, D.; Yang, L.; Cai, X. CoDetect: Financial fraud detection with anomaly feature detection. *IEEE Access* **2018**, *6*, 19161–19174. [CrossRef]
16. Cui, R.J.; Yan, C.; Wang, C. ReMEMBeR: Ranking metric embedding-based multicontextual behavior profiling for online banking fraud detection. *IEEE Trans. Comput. Soc. Syst.* **2021**, *8*, 643–654. [CrossRef]
17. Engels, C.; Kumar, K.; Philip, D. Financial literacy and fraud detection. *Eur. J. Financ.* **2020**, *26*, 420–442. [CrossRef]
18. Isaia, E.; Oggero, N.; Sandretto, D. Is financial literacy a protection tool from online fraud in the digital era? *J. Behav. Exp. Financ.* **2024**, *44*, 100977. [CrossRef]
19. Mugerman, Y.; Steinberg, N.; Wiener, Z. The exclamation mark of Cain: Risk salience and mutual fund flows. *J. Bank. Financ.* **2022**, *134*, 106332. [CrossRef]
20. Vadakkethil Somanathan Pillai, S.E. Financial Fraud Detection. GitHub. 2024. Available online: https://github.com/sanjaikanth/FinancialFraudDetection (accessed on 2 December 2025).
21. Shenoy, K. Credit Card Transactions Fraud Detection Dataset. Kaggle. 2020. Available online: https://www.kaggle.com/datasets/kartik2112/fraud-detection (accessed on 10 April 2025).
22. Harris, B. Sparkov Data Generation. GitHub. 2015. Available online: https://github.com/namebrandon/Sparkov_Data_Generation (accessed on 16 January 2025).
23. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
24. Riani, M.; Atkinson, A.C.; Corbellini, A. Automatic robust Box–Cox and extended Yeo–Johnson transformations in regression. *Stat. Methods Appl.* **2023**, *32*, 75–102. Available online: https://link.springer.com/article/10.1007/s10260-022-00640-7#citeas (accessed on 5 June 2025). [CrossRef]
25. Atkinson, A.C.; Riani, M.; Corbellini, A. The Box–Cox transformation: Review and extensions. *Stat. Sci.* **2021**, *36*, 239–255. [CrossRef]
26. Boukerche, A.; Zheng, L.; Alfandi, O. Outlier detection: Methods, models, and classification. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 55. [CrossRef]

27. Sayah, F. Outlier Detection Using PDF and Z-Score. Kaggle. 2022. Available online: https://www.kaggle.com/code/faressayah/outlier-detection-using-pdf-and-z-score (accessed on 21 December 2025).
28. Korhonen, V. Resident Population in the United States in 2023, by Generation (in Millions). Statista. 6 August 2024. Available online: https://www.statista.com/statistics/797321/us-population-by-generation (accessed on 10 February 2026).