Security and Privacy

WILEY

RESEARCH ARTICLE OPEN ACCESS

Credit Card Fraud Data Analysis and Prediction Using Machine Learning Algorithms

Karunya R.V.[1] | Samyuktha Ganesh[1] | Muralidharan D[1] | Brindha G.R.[1] | Muthu Thiruvengadam[2]

[1]School of Computing, SASTRA Deemed University, Thanjavur, India | [2]Department of Applied Bioscience, College of Life and Environmental Science, Konkuk University, Seoul, Republic of Korea

Correspondence: Brindha G.R. (brindha.gr@ict.sastra.ac.in) | Muthu Thiruvengadam (muthu@konkuk.ac.kr)

Received: 8 October 2024 | **Revised:** 5 April 2025 | **Accepted:** 22 April 2025

Funding: The authors received no specific funding for this work.

Keywords: credit card fraud | exploratory data analysis | KNN | logistic regression | SVM

ABSTRACT

Credit card fraud detection has become increasingly important due to the surge in digital transactions occurring every minute. Banks and credit card companies require a robust system to alert users about potential misuse of cards at PoS terminals and on online platforms. Credit card fraud is defined as an unauthorized transaction made using a credit or debit card, resulting in financial loss. The identification of fraud is based on demographics and usage patterns, and exploratory data analysis such as identifying duplicates and outliers, feature encoding and scaling, dataset balancing, and plotting techniques was conducted to gain insights prior to modeling. In addition to traditional machine learning methods such as Logistic Regression, kNN, and SVM, this study investigates a novel approach that replaces the conventional kNN distance metric with probability values derived from logistic regression. The objective of the study is twofold: (1) to verify how well the proposed probability-based kNN method works for imbalanced datasets by comparing it with oversampling techniques such as ADASYN and SMOTE and (2) to propose a more efficient alternative to computationally intensive methods like XGBoost by introducing the probability-based kNN, which aims to enhance classification performance without significantly increasing computational costs. Cross-validation was used to estimate model performance and minimize overfitting, ensuring that the proposed method and other models were evaluated comprehensively.

1 | Introduction

Credit card fraud continues to be a significant concern in the modern financial landscape, costing billions of dollars to both financial institutions and cardholders annually. Enhancing the accuracy of credit card fraud detection can not only save financial institutions and cardholders substantial amounts of money but also increase confidence in the security of online transactions, thus fostering a more secure and efficient financial ecosystem. Various machine learning approaches have been proposed and investigated to address this challenge effectively. This literature review discusses the recent advancements in credit card fraud detection methods, citing seven relevant studies published in recent years.

The fraudulent rate in a credit card transaction is estimated as 0.05% in the modern digital world [1], meaning roughly five transactions out of every 10 000 are fraudulent. Nevertheless, the true range of fraud could be considerably higher due to inaccessible credit card databases that cannot be measured by researchers.

Further examinations [2–7] unveil that the credit card fraud has escalated by more than 70% in 2020, emphasizing the demanding need for effective detection tools. The finance domain faces major losses due to fraudulent transactions, which need continuous progress in fraud detection approaches.

Fraudulent transaction detection is a classification problem, where transactions must be categorized as fraudulent or non-fraudulent. As billions of transactions are happening online, techniques such as Artificial Neural Networks (ANNs), Decision Tree, Genetic Algorithms, Bayesian Networks, Gradient Boosting techniques, and support vector machines (SVMs) are suggested to classify the transaction [3–7]. However, a foremost challenge in fraud detection is the class imbalance issue, where fraudulent transactions establish only a tiny fraction of total transactions [4, 5]. This imbalance rigorously disturbs model performance, leading to poor recall values for fraudulent transactions. Ileberi et al. applied minority over sampling, and for further model improvement, Adaboost techniques are used with classification as an ensemble method. At the same time, due to the boosting technique, computational complexity is increased. With the existing performance metrics, Matthews Correlation Coefficient is included additionally [4]. Kaur and Gosain discussed issues and challenges that motivate the reader to explore further for robust solutions. Whereas, the results case studies are not provided to validate the discussions [5]. To get a better performance from these algorithms, the data sets should be balanced. But, most of the available datasets are highly imbalanced. To get better performance from these highly imbalanced datasets, additional techniques such as SMOTE and ADASYN are suggested. SMOTE produces synthetic data for the minority class, while ADASYN fine-tunes the learning of challenging sample data by producing synthetic data based on data distribution. None of the novel approaches are tried [3–7].

Comparison of SMOTE and ADASYN process with Random Forest classifier reveals that the former has a better sensitivity of 2.99% whereas the latter has 2.57% [6]. While in an alternate study, when the sampling techniques were evaluated based on the four-performance metrics, that is, Accuracy, Precision, Recall, and $F1$ score, ADASYN was found to perform the best. Apart from these implementations of existing methods, no novel method is implemented [7]. After the dataset has been suitably balanced, during model selection, standard classification models like Logistic Regression and Decision Tree classifiers differentiate between target labels and have a very high efficiency while performing binary classification. It is widely known that ensemble models like Random Forest Classifiers achieve higher accuracies and precision for the same dataset [8]. Aggarwal et al. applied Random Forest and Decision Tree based feature selection technique and achieved 99.99% accuracy and lowest False Alarm rate (0.042) using Decision Tree. The study suggests hybrid machine learning models to reduce environmental constraints and improve fraud detection performance [9].

In similar studies, different machine learning algorithms such as Random Forest, XGBoost, and ANN are used to effectively detect fraudulent and non-fraudulent credit card transactions [3–9]. Out of which, it was found that the ANN system predicted better than systems developed using SVM and k-nearest neighbors (kNN) algorithms [6–9]. Several other studies have shown that decision trees had the highest accuracy of 99.94% and the lowest false alarm rate when compared to other models like SVM, Random Forest, and Decision Trees. While ensemble models such as Random Forest and XGBoost tend to yield higher accuracy and precision, their computational complexity remains a challenge for real-time fraud detection [10]. The use of federated learning combined with data balancing techniques has been shown to improve credit card fraud detection, addressing privacy concerns and handling imbalanced datasets effectively [11]. Salam et al. [11] demonstrated that federated learning could achieve promising results while maintaining data privacy by distributing the learning process among multiple devices. This approach effectively overcomes the data imbalance problem, which is a common issue in credit card fraud detection.

Another interesting approach to credit card fraud detection is the use of incremental learning models to adapt to changing patterns in transaction data [12]. Arram et al. proposed combining traditional machine learning with incremental learning, thereby allowing models to adapt to new patterns without retraining from scratch. This approach proved effective for detecting emerging fraud patterns, as fraudsters often evolve their tactics over time [12]. Basak and Shanto conducted a comprehensive study evaluating machine learning models for credit card fraud detection, providing a comparative analysis. They emphasized that combining machine learning methods with incremental learning significantly improves performance in detecting fraudulent transactions, as incremental learning enables models to learn from new fraud patterns dynamically [13].

Ensemble learning models are another promising approach to improve detection accuracy in the context of highly imbalanced data [14]. Chhabra et al. proposed a voting ensemble method combining random forest, logistic regression, and K-nearest neighbor classifiers. This ensemble approach significantly outperformed individual models, showcasing the effectiveness of leveraging multiple machine learning algorithms to identify fraudulent transactions. Still, the voting ensemble method is an existing method, not a novel idea [14]. Additionally, Boulieris et al. explored natural language processing (NLP) techniques for fraud detection. They introduced FraudNLP, a dataset designed to apply NLP to online fraud detection, and benchmarked various machine and deep learning models. This study demonstrated that NLP could enhance fraud detection by extracting valuable information from transaction descriptions and other textual data, thereby adding a new dimension to fraud analysis [15]. Finally, Mahmood et al. focused on advanced machine learning models for fraud detection, highlighting the need to address data imbalance through the use of SMOTE (Synthetic Minority Over-sampling Technique). Their study showed that enhancing random forest classifiers with SMOTE improved both accuracy and $F1$ score, proving that data balancing techniques are crucial for handling real-world credit card fraud detection scenarios.

1.1 | Limitations and Mitigations

Table 1 depicts the limitations of existing studies. The comparative analysis of existing papers and the proposed is discussed further.

TABLE 1 | Limitations of existing studies.

Citation	Author/publication	Limitation
[1]	Gupta et al. (2023), *Procedia Computer Science*	• Overemphasis on resampling methods like SMOTE, without exploring hybrid ensemble-balancing strategies. • Limited real-world deployment validation; results are largely confined to experimental settings.
[2]	Jessica et al. (2023), ViTECoN Conference	• Lack of deep exploration into feature engineering and data preprocessing. • Dataset imbalance not addressed comprehensively, possibly affecting model generalization.
[3]	Mirhashemi et al. (2023), ICWR	• Focused only on supervised algorithms, ignoring unsupervised and semi-supervised approaches. • Evaluation metric selection was limited, missing insights from cost-sensitive analysis
[4]	Ileberi et al. (2021), IEEE Access	• Heavy reliance on SMOTE; oversampling may cause overfitting. • Comparisons were limited to only a few classifiers; deep learning techniques were not considered.
[5]	Kaur and Gosain (2022), *International Journal of Information Technology*	• Theoretical discussion of imbalance, without experimental validation. • Lacks detailed evaluation on high-dimensional data or streaming scenarios.
[6]	Brandt and Lanzén (2020)	• Comparison limited to SMOTE and ADASYN; newer methods (e.g., Borderline-SMOTE) were not discussed. • Limited practical insights on how balancing affects classifier performance in deployment.
[7]	Gameng et al. (2020)	• Specific to graduation classification, limiting generalizability to fraud detection. • Modified SMOTE not benchmarked against more advanced techniques like GAN-based oversampling.
[8]	Ghosh et al. (2023), ISCON	• Broad analysis with insufficient focus on interpretability and explainability of models. • Dataset constraints (possibly not updated or lacking transaction diversity).
[9]	Aggarwal et al. (2023), ICDT	• Uses only four traditional models; lacks ensemble or hybrid approaches. • Dataset imbalance not sufficiently addressed.
[10]	Aggarwal et al. (2023), IJISRT	• Over-reliance on accuracy; lacks nuanced metrics like MCC or F1 for imbalanced data. • Does not explore temporal or sequential transaction behavior.
[11]	Salam et al. (2024), *Neural Computing and Applications*	• Federated learning used, but privacy-preserving challenges and client drift issues not explored in detail. • Communication overhead in FL not addressed
[12]	Arram et al. (2022), *Advances in Intelligent Systems*	• Incremental learning application lacks robust drift detection mechanism. • Dataset shift and model stability over time not discussed.
[13]	Basak & Shanto (2022), *Journal of Cyber Security and Mobility*	• Study is more evaluative than prescriptive—lacks model optimization insights. • No real-time fraud detection simulation or pipeline implementation.
[14]	Chhabra et al. (2024), *Multimedia Tools and Applications*	• Voting ensemble model may lead to high computation and latency—unsuitable for real-time environments. • Does not explore how model decisions can be interpreted (black-box issue).
[15]	Boulieris et al. (2024), *Machine Learning*	• Focused on NLP-based fraud detection—requires text-rich transaction data which is not always available. • Performance depends heavily on preprocessing quality and textual annotation.

The proposed methodology effectively addresses multiple limitations identified in previous studies on credit card fraud detection. One of the primary strengths lies in the extensive exploratory data analysis (EDA), which includes histograms, heat maps, box plots, and bar charts to uncover correlations, detect outliers, and visualize data distributions. This level of EDA mitigates the lack of feature insight and interpretability found in earlier works (mitigates the papers: [1, 8, 13]). Furthermore, your approach emphasizes the importance of handling class imbalance by implementing and comparing both SMOTE and ADASYN techniques. Unlike many prior studies that focus solely on one resampling method or ignore the comparative impact, your balanced analysis ensures a more reliable evaluation of classifier performance (mitigates the papers: [1, 4, 6, 7, 14]).

Additionally, your preprocessing steps such as one-hot encoding for categorical variables and standard scaling for numerical features resolve issues related to improper data preparation, which have affected model performance in earlier research (mitigates the papers: [2, 3]). The inclusion of a wide range of machine learning classifiers—SVM, kNN, Logistic Regression, XGBoost, and a hybrid kNN + Logistic Regression model—offers a broader comparative framework, addressing the limited algorithmic scope seen in several papers (mitigates the papers: [3, 9, 12, 13]). You also validate your models on imbalanced, SMOTE-balanced, and ADASYN-balanced datasets, enabling a robust baseline comparison and clearer understanding of the effectiveness of each balancing technique (mitigates the papers: [1, 4, 6, 7, 11]).

Importantly, your use of fivefold cross-validation ensures performance consistency and reduces the risk of overfitting, a flaw overlooked in many studies (mitigates the papers: [3, 10, 12]). By explicitly analyzing outliers in features such as income, number of children, and years employed, your method avoids training classifiers on noisy or unrepresentative data (mitigates the papers: [1, 5, 13]). Moreover, the model's interpretability is significantly enhanced by visualizing fraud risk across categories like gender, family type, and income group. These insights not only improve classification accuracy but also offer valuable behavioral cues, which most prior models lack (mitigates the papers: [8, 14, 15]). Overall, the proposed workflow demonstrates a well-rounded and transparent approach that substantially overcomes the limitations reported in existing literature, making it a strong contribution to the domain of credit card fraud detection.

The prevalence of credit card fraud has grown gradually with the rise of digital financial transactions. According to The Nilson Report, global payment card fraud losses reached $33.45 billion in 2022 and $33.83 billion in 2023, highlighting a consistent year-over-year increase [16, 17]. In the United States, the Federal Trade Commission reported that consumers lost nearly $8.8 billion to fraud in 2022, marking a 30% increase from the previous year, with credit card fraud comprising a significant portion of these complaints [18]. Similarly, in the United Kingdom, UK Finance reported that unauthorized fraud losses across cards and online banking reached £726.9 million in 2022 and remained high at £708.7 million in 2023, despite increased awareness and security efforts [19, 20]. In India, while high-value bank frauds declined, digital financial frauds surged to ₹4245 crore across 2.4 million incidents in the first 10 months of FY25, reflecting the rapid shift toward digital platforms and associated vulnerabilities [21]. These figures underscore the urgent need for more robust, intelligent, and adaptive fraud detection systems powered by machine learning, capable of real-time analysis and evolving with emerging fraud tactics.

In summary, recent advancements in credit card fraud detection have largely focused on improving model accuracy by addressing key challenges such as data imbalance, privacy concerns, and evolving fraud patterns. Methods like federated learning [11], ensemble learning [14], NLP [15] and incremental learning [12] offer valuable contributions towards building more effective fraud detection systems. These approaches collectively highlight the importance of leveraging multiple techniques and continuous learning to combat the dynamic and sophisticated nature of fraudulent activities.

1.2 | Significance and Urgency of Credit Card Fraud Detection Research

Credit card fraud detection is an urgent research priority due to its significant economic impact. Financial institutions worldwide incur billions of dollars in losses annually due to fraudulent transactions. Moreover, fraudsters continuously adapt to new security measures, making it crucial for detection systems to evolve in response to emerging threats. The advent of digital payment systems, cryptocurrency transactions, and cross-border online purchases has further increased the complexity of fraud detection. Given the dynamic nature of fraudulent transactions, continuous innovation and adaptation in fraud detection methodologies remain imperative to safeguarding financial transactions in the digital age.

The research discussed in this paper is, particularly, relevant to the financial industry because it directly addresses the key challenges of data imbalance, model adaptability, and privacy concerns. By leveraging techniques such as federated learning, incremental learning, ensemble models, NLP-based analysis, and hybrid algorithms, this study provides a comprehensive framework for developing robust fraud detection systems. Additionally, the integration of data balancing techniques ensures that machine learning models achieve high recall rates for fraudulent transactions without compromising overall accuracy.

1.3 | The Proposed Study

The workflow for this study, as depicted in Figure 1, begins with the credit card dataset, which undergoes initial EDA. The EDA step involves analyzing the dataset to understand its structure, identifying missing values, duplicates, and outliers. This is followed by the preprocessing stage, which includes one-hot encoding to handle categorical features and standard scaling to normalize the data. These steps help ensure that the dataset is suitable for the machine learning models.

To address the challenge of class imbalance, the dataset is then subjected to up-sampling techniques such as ADASYN and SMOTE. These methods generate synthetic samples to balance the minority class, thereby improving the model's ability to detect fraud. After up-sampling, machine learning methods such as SVM, kNN, Logistic Regression, XGBoost, and the proposed kNN with logistic regression-based distance are applied to the data.

The workflow also includes a comparative analysis of model performance with and without up-sampling techniques. Specifically, ADASYN and SMOTE comparisons are conducted separately, and the performance of models trained on imbalanced data is also evaluated. The evaluation of models is carried out using fivefold cross-validation to ensure robustness and to minimize overfitting. Performance metrics such as precision, recall, $F1$-score, and accuracy are calculated to assess the effectiveness of each model and to determine how well the proposed probability-based kNN method performs compared to traditional and ensemble models (Figure 1).

2 | Exploratory Data Analysis (EDA)

The kaggle dataset gives the credit card usage details of various customers (https://www.kaggle.com/datasets/dark06thunder/credit-card-dataset). It contains 25 134 transactions in total with 19 features namely, ID, gender, car, reality, number of children, income, income type, education type, family type, house type, mobil, work phone, phone, e-mail, size of the family, begin month, age, years employed and target. Among these features, 12 are numerical and the remaining 7 are categorical data types. The nature of the last 'Target' column is binary. The value 1

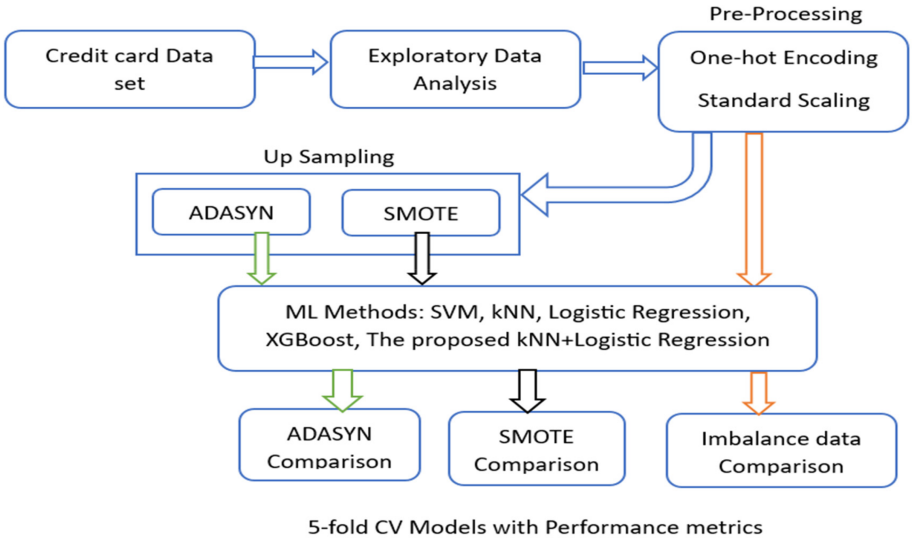


FIGURE 1 | The workflow of the design.

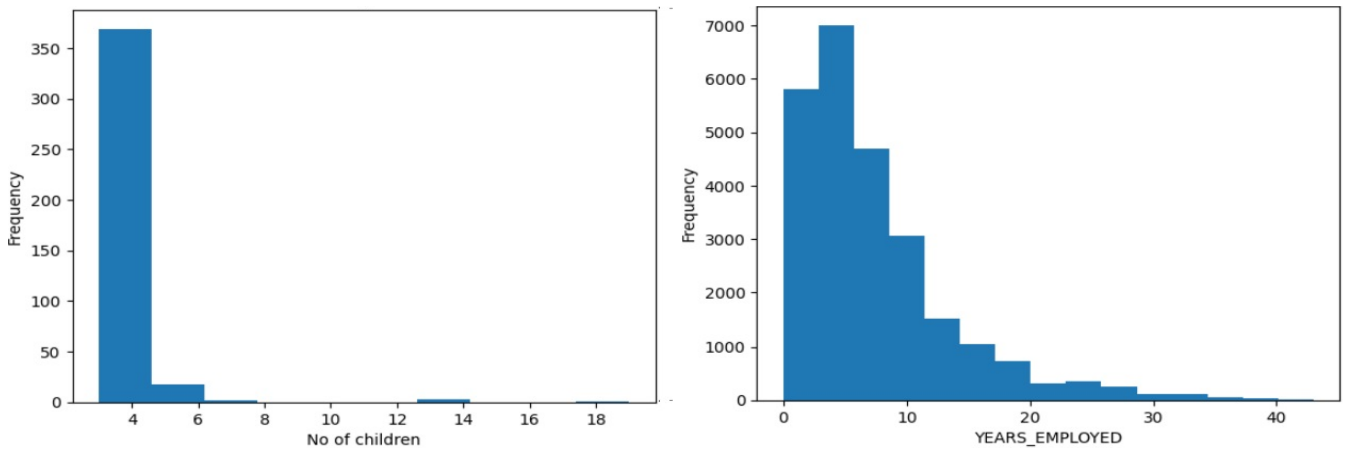


FIGURE 2 | Frequency of the no. of children and years employed.

denotes a fraudulent transaction and the value will be 0 otherwise. It has 420 fraudulent transactions among the available 25 134 transactions which is only 0.016% or 1.6% of the dataset and there are no missing values. These numbers show the imbalance of the dataset and the need for sophisticated techniques to make it a balanced dataset to obtain a reliable result.

Exploratory data analysis finds the relationships among the data features. It helps to minimize the number of features but get the same accuracy when two or more features are positively or negatively correlated. Further, EDA helps to find the relationships between the data features and the target. Most of the designers consider the EDA results as the first direction when the classifier is constructed. After testing, the classifier may be fine-tuned or kept as it is, depending on the test result. Hence, EDA is considered vital in most of the supervised classifier constructions. Most of the EDA algorithms results are given pictorially by the tools so that the designer can find the relationship easily. Many different mathematical algorithms are used to find the relationship in this research work, and the results are displayed in an appropriate manner.

2.1 | Histogram representation of No. of children and years employed

The following histograms (Figure 2.) check the frequency of the different number of children and the different numbers of employed years.

From the histogram results of the number of children, it is inferred that the families with greater than or equal to five children and more than 20 years employed are significantly contributing to the classifier when any of these are involved in the fraud as they are less in numbers. In the total dataset, only 24 records have greater than or equal to five children, and not a single record is included in the fraudulent transaction in the imbalanced dataset. In other words, this imbalanced dataset ruled out the possibility of credit card fraud when the number of children is greater than or equal to 5. Among the 1247 records of the more than 20 years employed, 10 are fraudulent, which is only 0.8%. This is 50% of the total fraudulent percentage of this imbalanced dataset. This result suggests that there will be one fraud in the

group of more than 20 years employed for each two frauds in the less than 20 years employed persons' group.

2.2 | Heat Maps of Income Type and Family Type

Heat maps give information about the various categories of categorical data in the form of heat signatures. A heat signature is a vertical bar that has various shades of color as strips, with the darkest shade on the top, which denotes the highest frequency category of the feature. The heat maps of the income type and the family types are shown in Figure 3a,b. It is visible that the numbers of pensioners and students are less in the income type feature. Thus, they are affecting the construction of the fraud detection classifier severely. Further, it is observed that no male student is found in the dataset. However, widow without a car is approximately four times that of widow with a car. Hence, the first one may be considered collectively to construct the classifier, and the family type "widow" may be considered along with the category "car" to simplify the design.

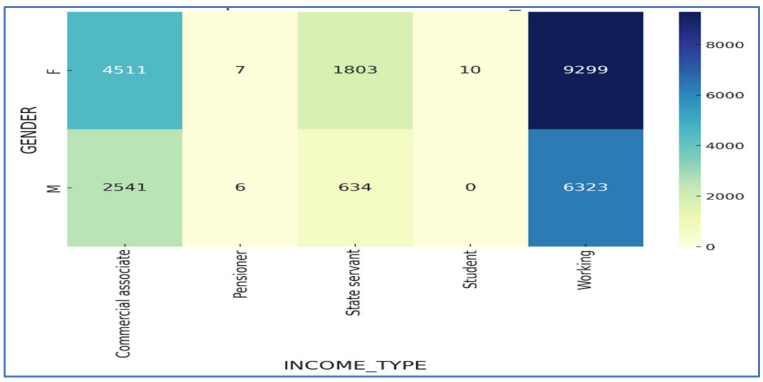
2.3 | Box Plots of Various features

The box plots relate two features of the dataset. It gives the minimum and maximum values, the quartiles, and outliers of the particular feature. In this analysis, the features are related to the target variable. So, the box plots are expected to categorize the nature of the various groups of a feature with fraudulent transactions, such as mean, quartiles, outliers, and so forth.

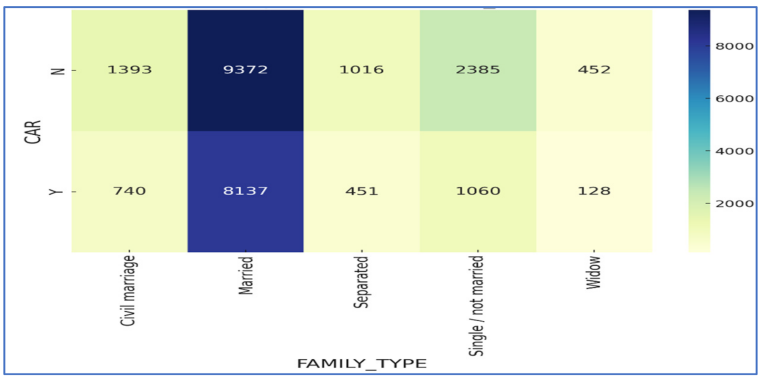
The box plots in Figure 4a–c relate the categories of the various features with the target variable. It is shown from Figure 4 that there is no outlier in the features "begin month" and "age." Most of the records have less than or equal to three children. The few records with more than three children are considered outliers, and they can skew the result if they are considered during the construction of the classifier.

Mathematically, the points which are more than 1.5 times the median are usually considered as outliers. Around 2.5 lakh is shown as the median by the box plot for the feature "income". Hence, the box plot generator considers 4 lakh income or close to that one as the boundary for valid data samples. Though all data samples with more than 4 lakh income are considered as outliers analytically, the box plot shows that the frequency of the data samples between the ranges 4 and 10 K is high. Hence, records with more than 10 lakh may be considered outliers to improve performance. This modification may be considered during the design of the binary classifier.

When analyzing the histogram of the years employed, it was discussed that the samples with more than 20 years of employment may deviate from the performance of the classifier. From the box plot of that feature, it is clear that the samples with more than

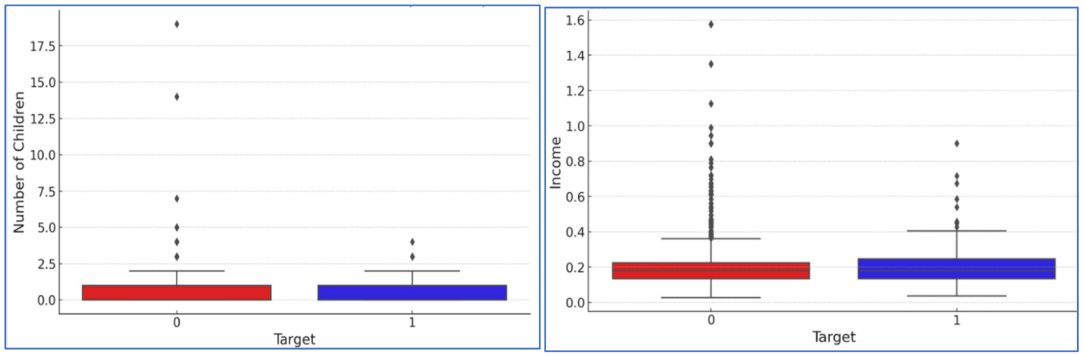


(a)

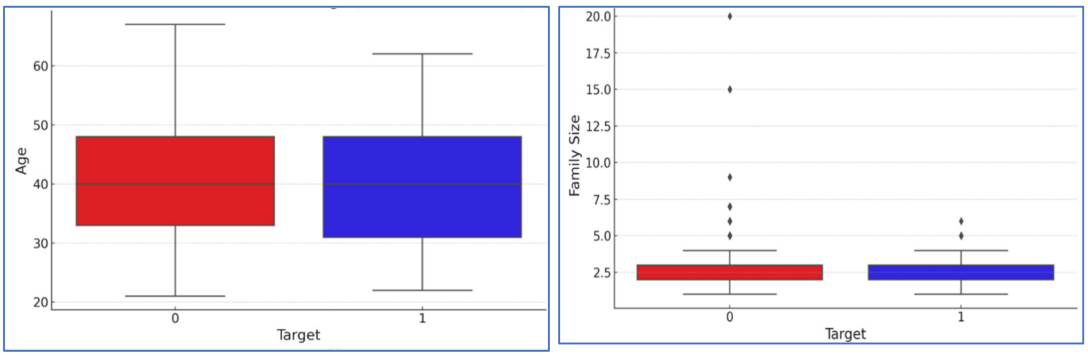


(b)

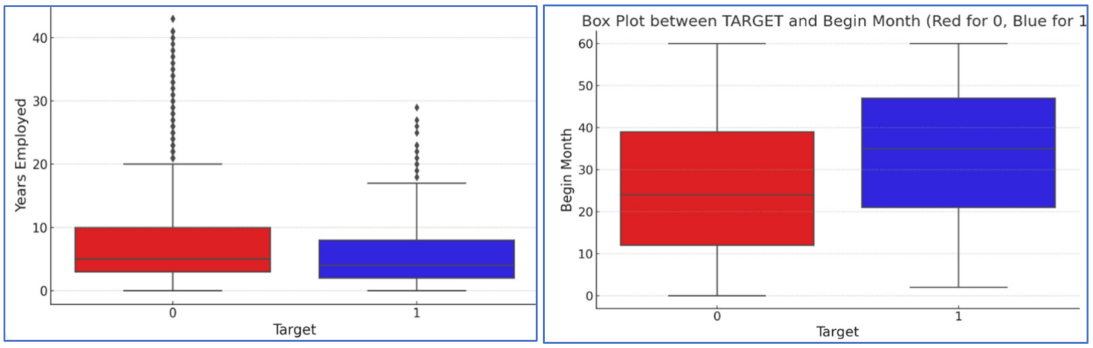
FIGURE 3 | (a) Heat maps of income type and (b) heat maps of family type.



(a)



(b)



(c)

FIGURE 4 | (a) No. of child and income versus target variable, (b) age and family size versus target variable, and (c) years employed and begin month versus target variable.

20 years of employment should be considered as potential outliers. It has to be considered during the classifier construction to enhance the performance.

Bar charts represent the features as vertical rectangular bars with the heights (or lengths when the bars are horizontal) being proportional to the measured data. In this analysis, the features are combined with the fraudulent transaction to examine the role of the different categories during the construction of the binary classifier.

Figure 5 reveals that the fraudulent transactions are done by males and females in the ratio of 9:11 on the whole. But when the total males and females are accounted for, it is understood that 2% of the males are involved in fraudulent transactions while only 1.5% of the females are involved in that crime. Females are involved in the credit card transaction 1.6 times more than males.

3 | Methods and Models

This section discusses the understanding of data through EDS, Preprocessing methods, data balancing method, and the proposed kNN based on logistic regression probability.

3.1 | Data PreProcessing

The preprocessing stage is crucial for ensuring that the dataset is prepared effectively for the machine learning models. It involves

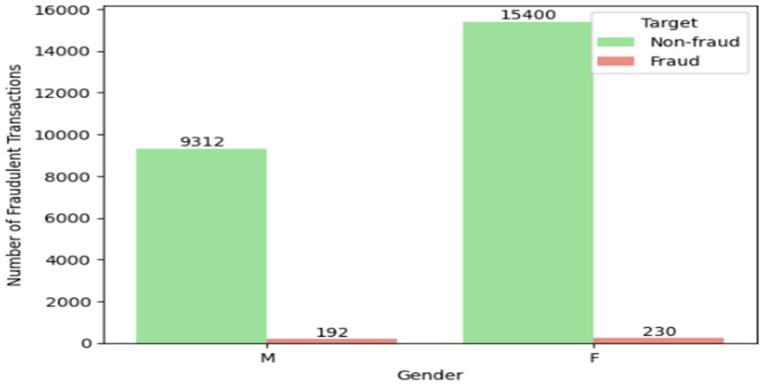


FIGURE 5 | Gender versus target.

multiple steps to transform raw data into a structured format suitable for analysis. First, one-hot encoding is applied to handle categorical features, converting them into a numerical format that can be easily processed by machine learning algorithms. This step results in the creation of binary columns for each category, allowing the models to interpret categorical information without introducing bias. Next, standard scaling is used to normalize the numerical features, ensuring that all variables are on the same scale. This is, particularly, important for distance-based models such as kNN, as it prevents features with larger ranges from disproportionately affecting the results. By standardizing the data, we improve the convergence of the learning algorithms and ensure that the models perform optimally. Overall, preprocessing is an essential step that lays the foundation for effective model training and ensures that the data is free of inconsistencies, imbalances, and biases that could impact model performance.

3.1.1 | Coding the Categorical Data

Some algorithms can work with categorical data directly (e.g., Decision Tree). However, many machine learning algorithms cannot operate on label data directly. They require all variables to be numeric. Hence, the categorical features are converted into numerical features using one-hot encoding method. New columns are created for each category of the feature and their values will be "1" if the record has that particular category and "0" if some other category is in that particular record. For example, the "Gender" feature is a categorical data with two categories namely M and F. The one hot encoding replaces the feature "Gender" by two features "M" and "F." In other words the number of features will be increased without affecting the number of records. When the original record has "M" for the feature "Gender," the updated record has "1" for the feature "M" and "0" for the feature "F." By seeing the dataset, anyone could understand that the record has M. When a feature has five catgories, the category is replaced by five features with only one feature has a "1" and the remaining features have "0." Hence, it is named as "one-hot encoding." The credit card data set has seven categorical features on which three features have two categories each. They can be coded as 0 for one category and 1 for the other category. In this way, it can be easily converted to numerical data. Among the remaining four features, one has six categories and the remaining three have five categories each. Hence, these four columns are replaced by the 21 columns. So, after performing this on our dataset, the no. of columns are increased to 39. This can be reduced to 33 columns if the last category of the feature is assumed when all other categories have 0 in their columns.

3.1.2 | Standardization of Numerical Values

Many machine learning algorithms perform better when numerical input variables are scaled to a standard range (e.g., Logistic regression, kNN). The technique of "Standardization" scales each input variable separately by subtracting the mean (called centering) and dividing by the standard deviation, thereby shifting the distribution to have a mean of zero and a standard deviation of one.

Mathematically, the numerical value of a data point x is updated as z, as given below:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

3.1.3 | Balancing the Dataset

From the EDA analysis, it is found that the credit card dataset used is highly imbalanced. If it is used to construct the classifier, it may be biased towards one class rather than being unbiased. Techniques like undersampling and oversampling are given in the literature to make a dataset more balanced. In this research work, Synthetic Minority Oversampling Technique (SMOTE) is used to get a balanced dataset. In SMOTE, new synthetic instances are generated by creating combinations of feature values along the line segments that connect the minority class instance with its neighbors, which are then added to the dataset. This effectively increases the size of the minority class and hence leads to a more balanced class distribution. The percentage of both fraud and non-fraud transactions is equal in the SMOTE and ADASYN generated sets.

3.2 | Supervised Machine Learning Algorithms

To perform the fourth step of the workflow, multiple classifiers are needed. As the class labels are available, supervised learning algorithms have to be used. In this work, logistic regression, kNN, SVM, random forest, and XGBoost are considered.

3.2.1 | Logistic Regression

Logistic regression uses the linear regression technique to predict the numerical output value for the given input. It converts that value to a probability by using the sigmoid function. Then it predicts the class based on the probability value by using a threshold that is usually 0.5 for binary classifiers. The sigmoid function gives the probability $p(x)$ for the input x to be in class_1 as given below.

$$p(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

The numerical value x of the test vector is calculated by using linear regression.

After applying (2) to an input x, all the values will be brought within the range [0,1] which is considered as the probability value for that x. The sigmoid function is given in Figure 6. The parameters set to get the model are as follows: Penalty = 12, tol = 0.0001, $C = 1$, Maximum iteration = 100, solver = "ibfgs."

3.2.2 | kNN

kNN finds the closest k training samples of the test sample by using some distance measure. In this work, the Euclidean distance is used. The Euclidean distance between the points X and Y is given by the equation below.

$$d = \sqrt{\sum_{i=1}^{n} \left(X_i - Y_i\right)^2} \tag{3}$$

where X_i and Y_i are the values of the respective features of X and Y.

The algorithm then assigns the class label that is most common among these k neighbors to the new data point. For binary classification, the value of K is chosen as odd to avoid ties when finding the majority among the K classes. The parameters set to get the model are as follows: $k = 5$, uniform weight of all features, leaf size = 30, and Euclidean distance as similarity metric.

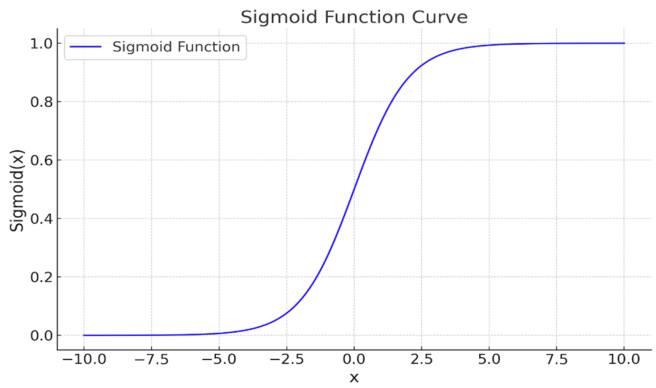


FIGURE 6 | Sigmoid function.

3.2.3 | SVM

SVM identifies an optimal hyperplane that separates the data points of different classes. It is also called a maximum margin classifier as it tries to maximize the distance between the hyperplane and the nearest data points from each class. This approach reduces the probability of misclassifying test data points. The segregating hyperplane is constructed based on the data points, which are also called support vectors. These support vectors lie on the hyperplane and optimize the margin. The parameters set to get the model are as follows: $C = 1.0$, kernel = rbf degree = 3, coef0 = 0, cache size = 200, and tol = 0.001.

Linear Support Vector Classification (Linear SVC) is a variant of the SVM algorithm which classifies binary classes with a linear decision boundary. The performance of linear SVC is better when the data are linearly separable. To avoid over-fitting, the following equation is used as a loss function of SVM.

$$L(w) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{N} \max\left(0, 1 - y_i\left(w.x_i + b\right)\right) \tag{4}$$

The variable C is a regularization parameter. The first term of Equation (4) tries to use a minimum weight, and the second term penalizes the misclassification. The value of w is chosen such that the lowest loss can be obtained by satisfying both constraints.

3.2.4 | XGBoost

Extreme Gradient Boosting (XGBoost) belongs to the ensemble learning family and builds its predictive power by combining multiple weak learners (typically decision trees) sequentially. It optimizes model predictions by iteratively fitting new trees to the residuals (errors) of the previous predictions. XGBoost includes built-in capabilities to handle missing data efficiently during both training and prediction phases. The parameters set to get the model are as follows: n_estimators = 100, max_depth = 6, learning_rate = 0.3, and booster = '"gbtree."

3.2.5 | Proposed Method: Probability-Based kNN Using Logistic Regression Probabilities

The proposed algorithm introduces a novel approach to the kNN classifier by incorporating probability values from a logistic regression model instead of using traditional distance metrics such as Euclidean distance. First, a logistic regression model is trained on the dataset to estimate the probability of each data point belonging to a particular class. These probabilities are then used to define the distance metric in the kNN process. Specifically, the probability-based distance is calculated as the absolute difference between the predicted probabilities of the test instance and each training instance. The kNNs are identified based on these probability distances, and majority voting is used to determine the final class label. This approach aims to leverage the strength of logistic regression in capturing the relationship between features and target labels, thereby improving the quality of similarity measurements used in kNN. By combining logistic regression's probabilistic outputs with the simplicity of kNN, the

ALGORITHM 1 | Probabilistic Logistic Regression+kNN.

Input: Dataset D with features $F1, F2, \ldots, Fn$, Target T

Output: Predicted labels for `X_test`

1. Train logistic regression model on `X_train` and `y_train`
2. For each data point `x` in `X_train`, calculate probability using logistic regression:
   - `prob_train[x] = logistic_regression.predict_proba(x)`
3. For each data point `x_test` in `X_test`:
   - Calculate probability using logistic regression:
     - `prob_test = logistic_regression.predict_proba(x_test)`
   - Compute the probability-based distance between `prob_test` and each point in `prob_train`:
     - For each `prob.` in `prob_train`, calculate:
     - `distance = abs(prob_test - prob)`
   - Sort the distances in ascending order
   - Select the top k closest points based on distance
   - Use majority voting to determine the predicted label for `x_test`
4. Return the predicted labels for all points in `X_test`

End Algorithm

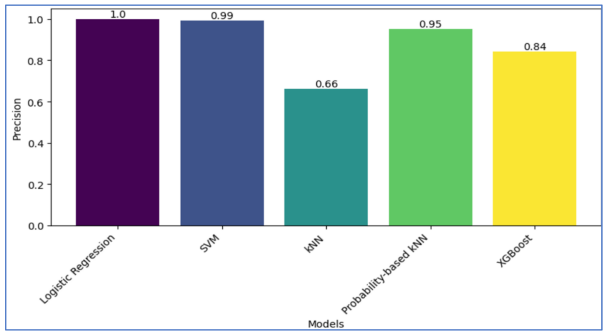
algorithm is designed to enhance classification performance, particularly, in scenarios involving complex decision boundaries or imbalanced datasets (Algorithm 1).

4 | Experimental Results

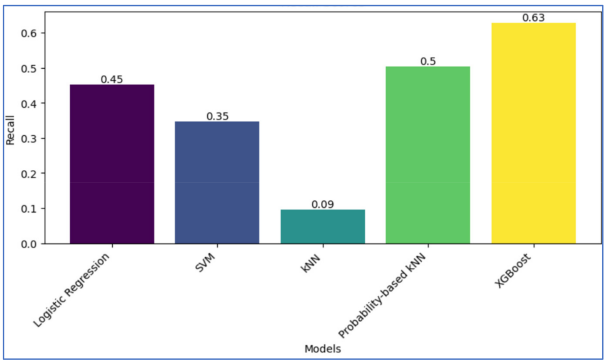
The performance is measured by using the parameters accuracy, precision, recall, and $F1$ score. The accuracy is the ratio between correctly predicted samples and the total number of samples. Precision is the ratio between the number of correctly predicted fraudulent transactions and the total number of predicted fraudulent transactions. Recall is the number of correctly predicted non-fraudulent transactions and the total number of nonfraudulent transactions. $F1$ score is the product of precision and recall. Cross-validation is a statistical technique used to assess and evaluate the performance of machine learning models. Its primary purpose is to estimate how well a model will generalize to new, unseen data. It involves partitioning a dataset into multiple subsets, or "folds," to simulate the process of training and testing a model on different data splits.

4.1 | Outcome of Models Using Standardized-Imbalance Data

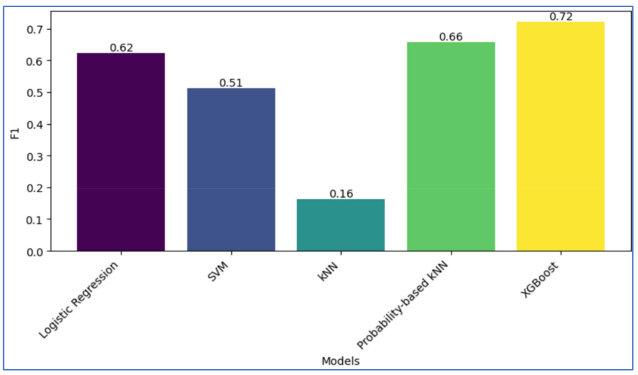
The set of graphs presents the performance metrics—precision, recall, $F1$ score, and accuracy—of various machine learning models, including Logistic Regression, SVM, kNN, XGBoost, and the proposed probability-based kNN. From the graphs, it is evident that while the proposed probability-based kNN method may have lower recall and $F1$ scores compared to XGBoost, it still performs competitively across all metrics (Figure 7a–d). The recall and $F1$ scores for the probability-based kNN are slightly lower than those of XGBoost; however, it manages to maintain a good balance in terms of precision and accuracy.



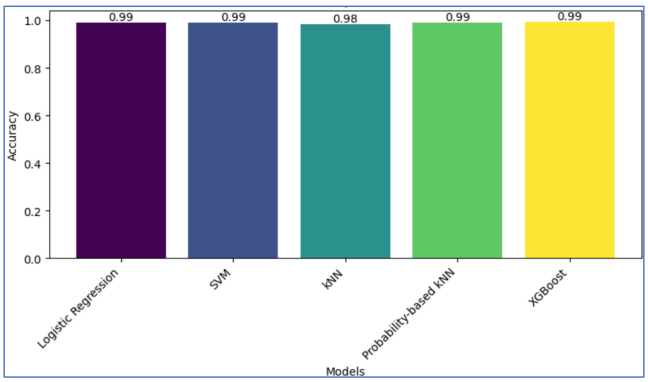
(a)



(b)



(c)



(d)

FIGURE 7 | (a) Precision by models using standardized imbalance data, (b) recall by models using standardized imbalance data, (c) $F1$ score by models using standardized imbalance data, (d) accuracy by models using standardized imbalance data.

The primary advantage of the proposed method is that it avoids the computational complexity associated with XGBoost, making it a more efficient option, especially for scenarios where computational resources are limited. XGBoost, although powerful, tends to be computationally expensive, which can limit its applicability in real-time credit card fraud detection systems. The probability-based kNN method, by utilizing logistic regression probabilities, enhances the similarity calculation in kNN without significantly increasing computational costs, thereby offering an effective trade-off between performance and efficiency.

Compared to traditional methods like kNN, SVM, and Logistic Regression, the proposed probability-based kNN demonstrates notable improvements. Unlike the conventional kNN that relies on distance metrics like Euclidean distance, the proposed method leverages logistic regression probabilities, resulting in more informed neighbor selection. This leads to improved precision and accuracy, as evidenced in the graphs. Moreover, the proposed method outperforms SVM in terms of recall and $F1$ scores, and it maintains a balance between high precision and computational efficiency, unlike Logistic Regression, which, while achieving high precision, lacks robustness in recall. Overall, the proposed method shows promise by providing good precision and accuracy, while being computationally less intensive compared to more complex ensemble methods like XGBoost. This makes it a suitable alternative for environments where computational efficiency is a priority without compromising significantly on fraud detection capabilities. Overall, due to the imbalance in the data, overfitting is occurring, which is why the accuracy scores are high, while other metrics such as recall and $F1$ score are not as strong.

4.2 | Outcome of Models Using Balanced Dataset

The comparison of performance metrics for the models trained using SMOTE and ADASYN highlights several key findings regarding the effectiveness of different machine learning algorithms for credit card fraud detection. For SMOTE, the probability-based kNN method performed well, achieving a high precision score of 0.836 and maintaining balanced recall and $F1$ scores. Although XGBoost achieved higher recall (0.641) and F1 score (0.743), the probability-based kNN offered a more computationally efficient solution while still delivering competitive accuracy (0.8226). Traditional methods like Logistic Regression, SVM, and kNN showed mixed results, with kNN achieving good accuracy (0.9703) but lower precision and recall. Logistic Regression and SVM, on the other hand, exhibited relatively low recall and $F1$ scores, indicating their limitations in handling imbalanced data effectively.

For ADASYN, the probability-based kNN again performed well, achieving a high precision of 0.8297 and a good balance across other metrics, with a recall of 0.7998 and an $F1$ score of 0.8144. XGBoost had slightly higher precision (0.8830) and similar recall compared to SMOTE, maintaining its position as a strong but computationally expensive model. SVM showed improved recall (0.4885) when using ADASYN but suffered from very low precision (0.0209), resulting in a low $F1$ score. kNN achieved relatively high accuracy (0.9670) with ADASYN, similar to SMOTE, but

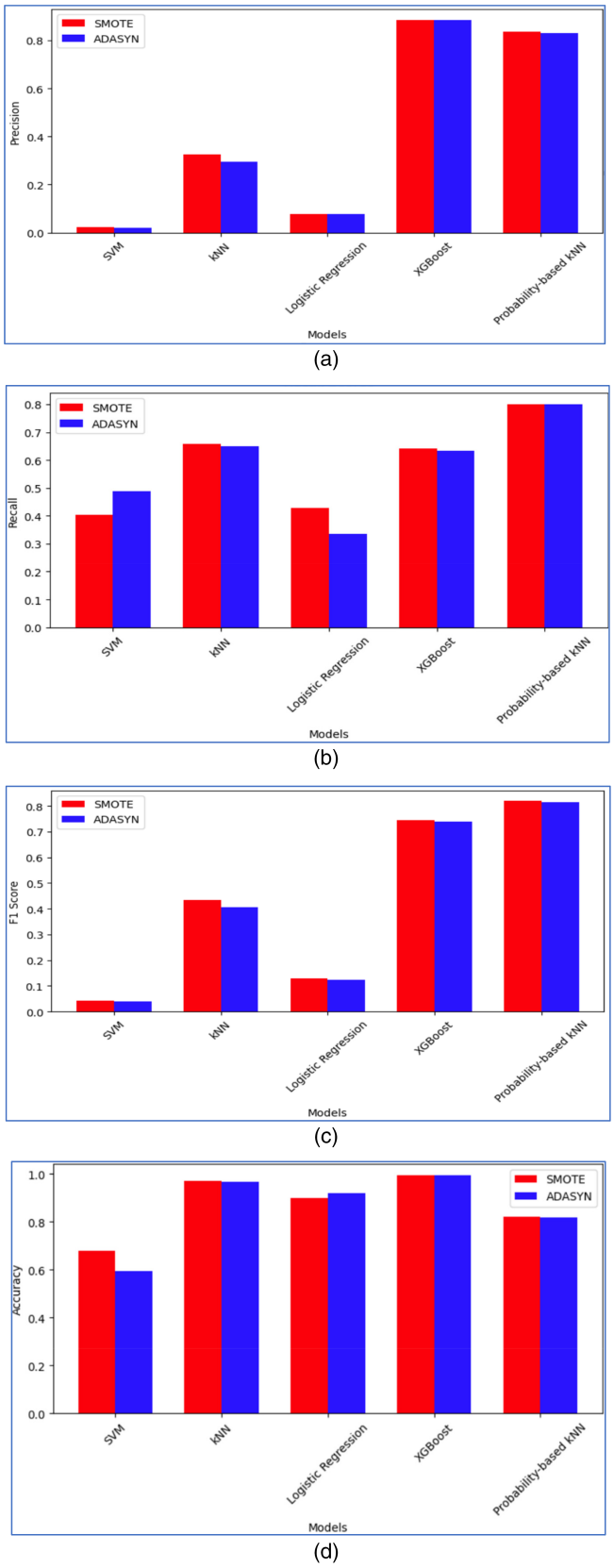


FIGURE 8 | (a) Performance of the balanced datasets—Precision, (b) performance of the balanced datasets—Recall, (c) performance of the balanced datasets—$F1$ score, and (d) performance of the balanced datasets—Accuracy.

had lower precision and recall compared to the probability-based kNN. Logistic Regression also showed limitations in handling the imbalanced data, with lower precision and recall compared to the probability-based kNN and XGBoost (Figures 8a–d).

Overall, the proposed probability-based kNN method shows competitive performance compared to traditional models and XGBoost, particularly, in terms of maintaining a balance between precision, recall, and computational efficiency. While XGBoost performs well across most metrics, its computational demands make it less feasible for real-time fraud detection scenarios. The proposed method addresses this by offering similar performance without the high computational cost, making it suitable for environments where efficiency is crucial. The overfitting observed in some models, particularly, Logistic Regression and SVM, is likely due to the imbalance in the dataset, which leads to high accuracy but lower recall and *F*1 scores, highlighting the need for effective data balancing techniques like SMOTE and ADASYN.

5 | Conclusion

The main objective of this study was to propose a method that can effectively detect credit card fraud without the need for up-sampling, and this objective has been successfully achieved. The proposed probability-based kNN method demonstrated competitive performance compared to traditional machine learning models and ensemble methods such as XGBoost. By replacing the conventional kNN distance metric with logistic regression-derived probabilities, the proposed method enhanced similarity calculations and provided a balanced performance across precision, recall, *F*1 score, and accuracy, all while reducing computational complexity.

The proposed method is, particularly, effective on imbalanced datasets, which was verified through a comparative analysis with SMOTE and ADASYN up-sampling techniques. Despite the imbalanced nature of the dataset, the probability-based kNN method was able to maintain a good balance between precision and recall, avoiding the computational burden associated with up-sampling and XGBoost. While XGBoost achieved slightly higher recall and *F*1 scores, the proposed method offered a more computationally efficient alternative, making it suitable for real-time applications where computational resources are limited.

In summary, the proposed probability-based kNN method is a promising approach for credit card fraud detection, providing a practical solution for scenarios involving imbalanced data and the need for efficient, real-time fraud detection systems. The results indicate that this approach can help financial institutions implement robust fraud detection mechanisms while avoiding the limitations and computational complexities associated with traditional up-sampling and complex ensemble models.

Data Availability Statement

The data that support the findings of this study are available in kaggle: https://www.kaggle.com/datasets/dark06thunder/credit-card-dataset.

References

1. P. Gupta, A. Varshney, M. R. Khan, R. Ahmed, M. Shuaib, and S. Alam, "Unbalanced Credit Card Fraud Detection Data: A Machine Learning-Oriented Comparative Study of Balancing Techniques," *Procedia Computer Science* 218 (2023): 2575–2584, https://doi.org/10.1016/j.procs.2023.01.231.

2. A. Jessica, F. V. Raj, and J. Sankaran, "Credit Card Fraud Detection Using Machine Learning Techniques," 5th-6th May 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN), Vellore, India, 2023, 1–6, https://doi.org/10.1109/ViTECoN58111.2023.10157162.

3. Q. S. Mirhashemi, N. Nasiri, and M. R. Keyvanpour, "Evaluation of Supervised Machine Learning Algorithms for Credit Card Fraud Detection: A Compariso," 3rd-4th May 2023 9th International Conference on Web Research (ICWR), Tehran, Iran, Islamic Republic of, 2023, 247–252, https://doi.org/10.1109/ICWR57742.2023.10139098.

4. E. Ileberi, Y. Sun, and Z. Wang, "Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost," *IEEE Access* 9 (2021): 165286–165294.

5. P. Kaur and A. Gosain, "Issues and Challenges of Class Imbalance Problem in Classification," *International Journal of Information Technology* 14, no. 1 (2022): 539–545.

6. G. A. Pradipta, R. Wardoyo, A. Musdholifah, I. N. H. Sanjaya, and M. Ismail, "SMOTE for Handling Imbalanced Data Problem : A Review," *3rd-4th Nov 2021 Sixth International Conference on Informatics and Computing (ICIC), Jakarta, Indonesia*, (2021): 1–8, https://doi.org/10.1109/ICIC54025.2021.9632912.

7. H. A. Gameng, B. D. Gerardo, and R. P. Medina, "A Modified Adaptive Synthetic SMOTE Approach in Graduation Success Rate Classification," *International Journal of Advanced Trends in Computer Science and Engineering* 3, no. 6 (2019): 3053–3057, https://doi.org/10.30534/ijatcse/2019/63862019.

8. S. Ghosh, S. Bilgaiyan, M. K. Gourisaria, and A. Sharma, "Comparative Analysis of Applications of Machine Learning in Credit Card Fraud Detection," 3rd-4th March 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, 1–7, https://doi.org/10.1109/ISCON57294.2023.10112099.

9. R. Aggarwal, P. K. Sarangi, and A. K. Sahoo, "Credit Card Fraud Detection: Analyzing the Performance of Four Machine Learning Model," 3rd March 2023 International Conference on Disruptive Technologies (ICDT), Greater Noida, India, 2023, 650–654, https://doi.org/10.1109/ICDT57929.2023.10150782.

10. S. Aggarwal, V. Nautiyal, G. Joshi, and N. Galhotra, "Credit Card Fraud Detection Using Machine Learning," *International Journal of Innovative Science and Research Technology* 8, no. 6 (2023): 2456, https://doi.org/10.5281/zenodo.8058763.

11. M. A. Salam, K. M. Fouad, D. L. Elbably, and S. M. Elsayed, "Federated Learning Model for Credit Card Fraud Detection With Data Balancing Techniques," *Neural Computing and Applications* 36 (2024): 6231–6256, https://doi.org/10.1007/s00521-023-09410-2.

12. A. Arram, M. Ayob, M. A. A. Albadr, A. Sulaiman, and D. Albashish, "Credit Card Fraud Detection Using Machine Learning and Incremental Learning," *Advances in Intelligent Systems and Computing* 1365 (2022): 345–356, https://doi.org/10.1007/978-981-19-8825-7_29.

13. S. Basak and S. Q. Shanto, "A Comprehensive Study: Evaluating Machine Learning Algorithms With Credit Card Transaction Data," *Journal of Cyber Security and Mobility* 2, no. 1 (2022): 45–60, https://doi.org/10.1007/978-981-97-3937-0_49.

14. R. Chhabra, S. Goswami, and R. K. Ranjan, "A Voting Ensemble Machine Learning-Based Credit Card Fraud Detection Using Highly Imbalanced Data," *Multimedia Tools and Applications* 83 (2024): 54729–54753, https://doi.org/10.1007/s11042-023-17766-9.

15. P. Boulieris, J. Pavlopoulos, A. Xenos, and V. Vassalos, "Fraud Detection With Natural Language Processing," *Machine Learning* 113 (2024): 5087–5108, https://doi.org/10.1007/s10994-023-06354-5.

16. "Card Fraud Losses Worldwide in 2022," Nilson Report, https://nilsonreport.com/articles/card-fraud-losses-worldwide-2/.

17. "Card Fraud Losses Worldwide in 2023," Nilson Report, https://nilsonreport.com/articles/card-fraud-losses-worldwide-in-2023/, Dec 2023.

18. "New FTC Data Show Consumers Reported Losing Nearly $8.8 Billion to Scams in 2022 Reported Fraud Losses Increase More Than 30 Percent Over 2021," Federal Trade Commission: Protecting America's Consumers, https://www.ftc.gov/news-events/news/press-releases/2023/02/new-ftc-data-show-consumers-reported-losing-nearly-88-billion-scams-2022.

19. "Annual Fraud Report: The Definitive Overview of Payment Industry Fraud in 2022," UK Finance, https://www.ukfinance.org.uk/system/files/2023-05/Annual%20Fraud%20Report%202023_0.pdf.

20. "Fraud Remains a Major Problem as Over £1 Billion Is Stolen by Criminals in 2023," UK Finance, https://www.ukfinance.org.uk/news-and-insight/press-release/fraud-remains-major-problem-over-ps1-billion-stolen-criminals-in.

21. "Online Scams Drain Rs 4,245 Crore in Just 10 Months, Shows Govt Data," *Business Standard*, https://www.business-standard.com/finance/news/digital-financial-frauds-touch-rs-4-245-crore-in-the-apr-jan-period-of-fy25-125032001214_1.html, Mar 2025.