Archives available at journals.mriindia.com

International Journal on Advanced Computer Theory and Engineering

ISSN: 2319-2526
Volume 14 Issue 01, 2025

Fraud Detection in Banking using ML

Ms. Deepali Narwade[1], Mukesh Vadgav[2], Om Bambale[3]

[1]Assistant Professor, Department of Artificial Intelligence and Data Science, DYPCOEI, Varale, Pune, Maharashtra, India

[2,3]U.G. Student, Department of Artificial Intelligence and Data Science, DYPCOEI, Varale, Pune, Maharashtra, India

Peer Review Information

Submission: 23 Feb 2025
Revision: 26 March 2025
Acceptance: 30 April 2025

Keywords

Fraud Detection
Machine Learning (ML)
Support Vector Machine (SVM)
Random Forest (RF)

Abstract

Machine learning provides a dynamic and adaptable method of fraud detection by analyzing vast volumes of transactional data, spotting irregularities, and learning from previous fraud incidents. This study looks at how machine learning (ML) methods can be used to identify and stop fraud in the banking industry. Traditional rule-based systems cannot keep up with changing fraud tendencies because of the exponential increase of digital transactions. To ascertain how successfully they detect fraudulent activity, the study also looks at a range of supervised and unsupervised machine learning algorithms, including logistic regression, decision trees, random forests, support vector machines, and clustering techniques. Data asymmetry, feature selection, and real-time detection needs are other difficulties that are addressed in this study. Using state-of-the-art machine learning methods, the initiative aims to improve detection accuracy, reduce false positives, and boost overall banking security. The findings demonstrate that ML-based systems outperform traditional methods by a significant margin, making them essential parts of modern fraud detection systems.

INTRODUCTION

In today's increasingly digitalized financial landscape, the possibility of fraud has grown to be a significant concern for banks and financial organizations worldwide. As the number of online and electronic transactions continues to rise, fraudulent activities are becoming more complex and common. Conventional fraud detection systems find it difficult to keep up with the shifting techniques adopted by scammers because they are typically built on predetermined criteria and manual evaluations. When dealing with enormous volumes of transactional data in real time, these methods are frequently reactive, scope-constrained, and inefficient.

In the banking sector, ML has proven to be a reliable and promising way to improve fraud detection capabilities. ML algorithms can learn from historical data, identify complex patterns, and adapt to new and emerging risks, unlike static rule-based systems. models of ML classify transactions, detect anomalies, and produce precise predictions often in real time by utilizing supervised and unsupervised learning techniques.

.

The implementation of machine learning to identify fraudulent banking transactions is examined in this study. It examines a number of machine learning algorithms, including support vector machines, decision trees, logistic regression, random forests, neural networks, and clustering techniques. Important issues are also highlighted in the paper, such as feature engineering, data imbalance, and the trade-off between false positives and detection accuracy. The study also highlights the significance of real-world implementation considerations and model evaluation measures.

The goal of this research is to demonstrate the effectiveness of ML in building robust, scalable, and adaptive fraud detection systems that can significantly enhance security and trust in modern banking operations.

Paper is organized as follows. Section II describes studies of different research papers in the field of Fraud Detection in Banking. The diagram represents the step of the algorithm used in Machine learning. After different algorithms explained in Section III. Finally, Section IV presents conclusion.

LITERATURE SURVEY

Abdul Khalid et al. suggested to use hybrid ensemble approaches to reduce the gap between independent models such as SVM, KNN and RF and to enhance performance on important fraud detection metrics. Platforms for ensemble learning such as boosting and bagging have been studied to increase accuracy and stability. To reduce the impact of class imbalance, techniques such as SMOTE and under-sampling are used. But since fraudulent behavior is dynamic and ever-changing, a more adaptive approach is needed [1].

Credit card fraud is the fraud category that gets the most evaluations. Samaneh Sorournejad et al. presented a credit card fraud detection model that uses deep neural networks and probabilistic graphical models, highlighting the misuses of supervised and unsupervised techniques and offering advice to novice researchers [2].

J. Karthika et al. studied sequence categorization is where this model was created. This model was created with sequence categorization in mind. The researchers performed a comparison between their model and baseline using data from the actual world. They discovered that taking into account the expected labels and the innate sequential links between transactions produced better outcomes. In addition, a new undersampling algorithm was presented, and when compared to conventional oversampling and undersampling techniques, it produced very good results [3].

Sorin-Ionuț Mihali et al. proposed model uses and extends the Random Forest technique to address the critical problem of credit card fraud detection for unbalanced datasets. To enhance model performance, entropy-based criteria, advanced data processing techniques, and synthetic minority over-sampling techniques (SMOTE) were applied. Extensive analysis using F1-score, precision, accuracy, and recall demonstrates the effectiveness of the model in reducing undetected fraud. By carefully balancing false positives and false negatives, comparative analysis demonstrates increased robustness across a range of data distributions. To further enhance the fraud detection system, the report also recommends future research goals focused on scalability, personalized detection, and field testing [4].

Rishab Pendalwar et al. fouced on Six supervised machine learning models for credit card fraud detection are investigated in this paper using a benchmark dataset: SVM, Random Forest, Naive Bayes, KNN, XGBoost, and LightGBM. It examines the outcomes of dimensionality reduction with

PCA and class imbalance management with GANs. Evaluations of Random Forest's accuracy, precision, recall, and F1-score reveal that it works best when combined with GAN-based data balancing The study highlights the significance of addressing class imbalance and utilizing PCA to increase the accuracy of detection [5]. Wijaya Michael et al. These findings offer crucial guidance on selecting the most appropriate models and preprocessing data to enhance fraud detection systems [6].

Shanshan Jiang et al. recognize credit card fraud, this paper introduces UAAD-FDNet, an unsupervised attentional anomaly detection network. This method combines autoencoders, feature attention approaches, and generative adversarial networks (GANs) to effectively remove fraudulent activity from massive transaction datasets by classifying them as anomalies. Compared to conventional techniques like support vector machines (SVM) and random forests (RF), which usually struggle to detect novel fraud patterns, UAAD-FDNet offers more adaptability and robustness. In addition to enhancing existing financial security systems, this innovative method shows how unsupervised learning may be applied to find novel fraud tactics [7].

Aquino, John Patrick J. et al. developed an improved hybrid credit card fraud detection model by combining Light Gradient Boosting Machine (LGBM) for supervised learning with Kernel Principal Component Analysis (KPCA) for unsupervised dimensionality reduction [8]. Haider, Zeeshan Ali et al. applied advanced preprocessing techniques, including frequency encoding, scaling, and categorical correlation, to the IEEE-CIS dataset in order to enhance model input. Two model versions were evaluated: V1 as a baseline and V2, which had better parameters and techniques. The results demonstrate improved fraud detection efficiency with recommendations for more hyperparameter tuning and dataset balancing [9].

ALGORITHMS USEFUL IN FRAUD DETECTION

Supervised Machine Learning

An algorithm known as supervised machine learning gains the ability to predict or make judgments based on labeled training data. This method involves giving the model input data and the appropriate output (labels) so that it can understand the relationship between the two. In an attempt to lessen the discrepancy between its own forecasts and actual results, the algorithm keeps changing its internal parameters. After training, the model will use the learnt relationship to make predictions about previously unknown data. The most popular applications of supervised learning are in categorization tasks, where labeling data into pre-established categories is the aim, and regression, where continuous values are the aim. NN, k-NN (k-NN), DT, and SVM are a few popular supervised learning methods.The quantity and caliber of labeled training data also have a significant impact on these models' accuracy. Metrics like accuracy, precision, recall, and F1-score are used to measure a model's performance on test data.

Types of Supervised Algorithm

1) Logistics regression
2) K-Nearest Neighbour
3) Decision Tree
4) Random forest
5) Support Vector Machine

1) Logistic Regression

Popular supervised learning techniques like logistic regression are frequently applied to binary classification issues like determining whether an email is spam, spotting fraud, or responding to yes/no queries. Unlike linear regression, which forecasts continuous numerical results, logistic regression predicts the likelihood that an input will fall into a specific category. Following the computation of the weighted sum of the input features, the result is subjected to a logistic (sigmoid) function. The result is transformed into a value between 0 and 1 by the sigmoid function, which indicates the likelihood that the input is a member of the target class. Should this likelihood surpass a specified cutoff point, often 0.5,

Mathematically, logistic regression is represented as:

$$P(y=1|x) =1/ (1+e\^{}- (\beta_0+\beta_1x_1+\beta_2x_2+...\beta nxn)),$$

where β_0 is the intercept, $\beta_1...\beta n$ are the coefficients for the features $x_1...xn$, and e is the base of the natural logarithm.

Advantage of Logistics regression: -

1) Simplicity and Interpretability:

Logistic regression is simple to use and comprehend. The model's coefficients offer clear insight into how each feature affects the result, and it is highly interpretable, particularly in industries like healthcare and finance where comprehending the logic of projections is crucial.

2) Efficient and Fast to Train:

Logistic regression uses less computing power and training time than more complex algorithms like ensemble methods or neural networks. It works well with smaller datasets and in situations where there is a roughly linear relationship between the input and output variables.

3) Regularization Support:

Logistic regression can use L1 and L2 regularization approaches to avoid overfitting, it can be used to high-dimensional data sets or multicollinearity.

Disadvantage of Logistics regression: -

1) Assumes Linear Relationship (in log-odds): -

A linear relationship between the independent variables and the log-odds of the outcome variable is assumed by logistic regression. Unless the characteristics are changed or interactions are included, this assumption makes it unsuitable for modeling complex connections or data with high levels of non-linearity.

2) Sensitive to Outliers: -

Extreme values or outliers in the dataset may have an impact on logistic regression, which could distort the predictions and coefficients. Preprocessing procedures like scaling the data or eliminating outliers are frequently required.

3)Limited to Binary or Multinomial Outcomes

Binary categorization is what logistic regression is used for. Even though it may be extended to multi-class scenarios using strategies like One-vs-Rest, it is not as straightforward or efficient as models that are naturally made for multi-class classification, like Decision Trees or Neural Networks.

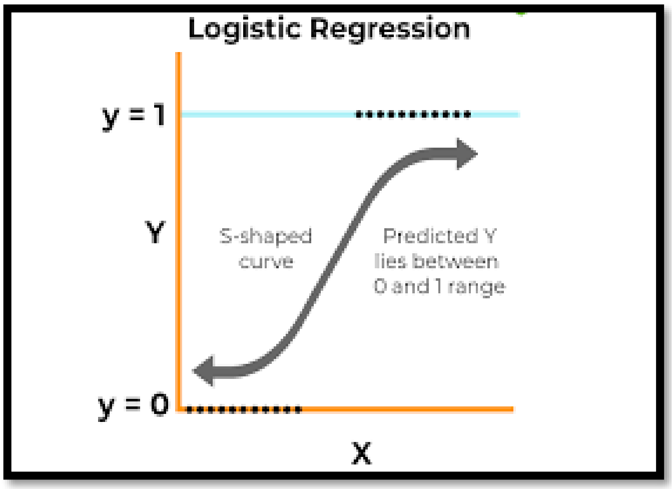

Figure 1: Logistics Regression [10]

2) K-Nearest Neighbour: -

K-Nearest Neighbors (K-NN) is a straightforward yet effective supervised learning method that may be applied to both classification and regression applications, while being more frequently employed for classification tasks. K-NN is also regarded as a non-parametric, instance-based learning method since it never constructs an explicit model during the training phase and makes no assumptions regarding a particular form of the underlying data distribution. Rather, it loads the whole training set into memory and makes predictions based on how similar the stored instances are to the incoming input data.

Advantage of K-Nearest Neighbour: -

1) Efficiency: -

An effective algorithm can improve software performance, particularly in real-time or large-scale applications like search engines, data processing, and online transactions, by reducing the number of steps required and making the most use of memory and storage. Programs run more smoothly and swiftly as a result of problems being resolved more quickly and algorithms using fewer resources.

2) No Training Phase requirement: -

Despite developing a complicated model, the K-Nearest Neighbours algorithm is simple to comprehend and requires no training. It allocates new points to categories by comparing them to known ones based on distance. This makes the premise of the procedure simple. Unlike most machine learning algorithms that require a training phase to learn patterns from data, K-NN just stores the training data and makes judgments when a prediction is needed, making it easy to implement and deploy, especially for novices.

3) Non-Parametric: -

The K-Nearest Neighbours (K-NN) approach is non-parametric and makes no assumptions about the distribution or structure of the data. It is helpful in cases when the underlying data distribution is complicated or unknown because of its versatility, which allows it to deal with the majority of data types without knowing in advance how the data is distributed or behaves.

Disadvantage of K-Nearest Neighbour: -

1)High usage of Memory: -

The K-Nearest Neighbours (K-NN) technique keeps the complete training data set in memory for use in subsequent prediction procedures. Since K-NN lacks a training phase to compute distances and classify additional points as needed, it must store the complete dataset.

2) Computationally expensive: -

The term "computationally expensive" for K-Nearest Neighbours (K-NN) refers to the algorithm's high processing requirements, especially during prediction. In order to classify each new data point, K-NN determines the distance between each new data point and each sample from the training dataset

3) Needs Careful Choice of ‘K’ and Distance Metric:-

The performance of the K-Nearest Neighbours algorithm in terms of accuracy and efficiency might significantly depend on the value of K (the number of the closest neighbours) and the measure of distance (for example, Manhattan or Euclidean). Although a high value of K could smooth forecasts, it might also obscure small patterns. Conversely, a low value could make the model susceptible to noise.

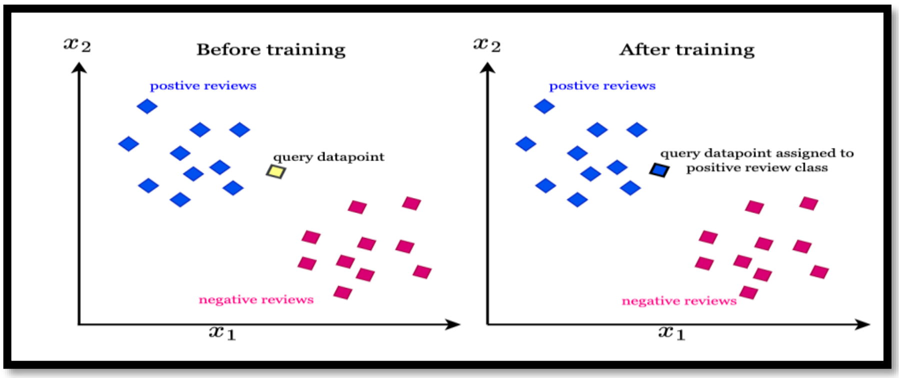

Figure 2: K-Nearest Neighbour [11]

3) Decision Tree

When it comes to solving classification and regression problems, supervised learning algorithms like decision trees are well-known. Outlining several potential outcomes and the costs and factors that go along with them allows it to mimic the decision-making process. Leaf nodes display the final prediction or class label in this architecture, while internal nodes stand in for attributes or features and branching represent decision criteria based on feature values. Before any division, the root node represents the complete dataset. In order to create a branched, tree-like structure that supports precise and logical predictions, the model undergoes a recursive splitting procedure that breaks the data into smaller, easier-to-manage pieces.

Advantage of Decision Tree: -

1) Minimal Data Pre-processing: -

Additionally, decision trees frequently require less data preparation. Without having to go through laborious preprocessing steps like encoding or normalization, they are able to operate with both numerical and categorical data. This feature speeds up the model-building process and streamlines the data preparation procedure.

2) Interpretability and Transparency: -

Perhaps the biggest benefit of decision trees is their interpretability. Decision-making is made simple by the hierarchical structure of a decision tree, which naturally mimics human decision-making processes with its branches and nodes. Such transparency is especially useful in industries like health and finance, where stakeholders need succinct explanations of model forecasts.

3) Robustness to outliers: -

Decision trees are comparatively resistant to outliers. Because outliers divide data according to feature values, they have less of an impact on the tree's overall structure. This robustness guarantees that the model will continue to function well even in the presence of data outliers.

Disadvantage of Decision Tree: -

1)Prone to Overfit: -

Decision trees frequently suffer from overfitting, especially when allowed unfettered growth.

2) Unstable Models: -

Even little changes to the training set can have a significant effect on the structure of a decision tree, making them unstable. This sensitivity could affect the model's reliability and consistency.

3) Computational Complexity: -

It can be computationally taxing to build a decision tree, particularly when working with large amounts of data. The process evaluates each possible split at each node, which might take a lot of time and resources.

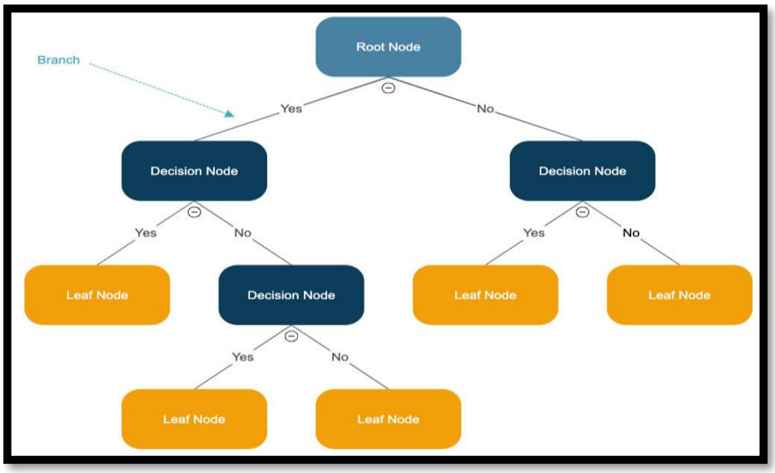

Figure 3: Decision Tree [12]

4) Random forest

Regression, classification, and other tasks can be accomplished with the popular ensemble approach Random Forest. In order to get a final forecast, it builds a number of decision trees during the training phase and combines their results. While it averages all tree predictions for regression, it chooses the most often predicted class (the majority vote) in classification scenarios. The overfitting issue that independent decision trees usually have is successfully mitigated by this ensemble approach, which also improves the model's capacity to generalize across a variety of data sets.

Advantage of Random Forest: -

1) Accurate and robust: -

Through integrating many decision trees' forecasts, Random Forest often achieves higher accuracy than individual trees. The ability of the model to generalize to new data is enhanced by this ensemble technique, which lowers variance.

2) Resistance to Overfitting: -

Random Forest prevents a single decision tree from overfitting its training data by averaging the output of several decision trees, each of which was trained on a distinct sample of the data. As a result, the model performs better on fresh data and is less likely to overfit.

3) Estimating the Importance of Features: -

The approach sheds light on the characteristics that most significantly impact forecast accuracy. It is useful for choosing characteristics and identifying the data's underlying trends.

Disadvantage of Random Forest: -

1) Bias with Multi-Level Categorical Variables: -

Random Forests' predilection for categorical variables with higher values may bias measures of feature significance.

2) Sensitivity to Data Noise:

Although Random Forests treat noise as if it were a significant pattern, they may overfit noisy datasets.

3) Lack of Interpretability:

The decision-making process is difficult to comprehend due to Random Forests' complexity and difficulty in interpretation, in contrast to single decision trees.

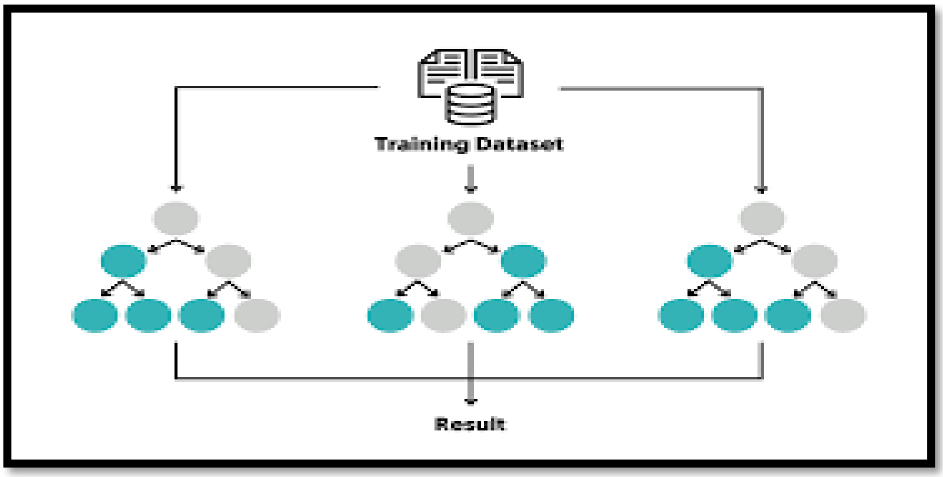

Figure 4: Random Forest [13]

5)Support Vector Machine

In order to ensure the greatest feasible margin between data points of different classes, support vector machines (SVMs) look for the best hyperplane. The support vector, or margin, is the separation between the hyperplane and the nearest data points in each class. A big margin improves the model's capacity to generalize to new data and helps minimize overfitting. SVMs employ kernel functions to project data into a higher-dimensional space when the data is not linearly divided. This allows for the creation of a hyperplane that successfully separates classes. The model can handle complicated and nonlinear datasets with great accuracy thanks to this method.

Advantage of Support Vector Machine:

1) Effective in High-Dimensional Spaces:

SVMs perform exceptionally well with datasets that are rich in characteristics, which qualifies them for applications in text categorization and bioinformatics

2) Versatility Through Kernel Functions:

SVMs may perform both linear and non-linear classification tasks by utilizing kernel functions, which transform the input data into higher-dimensional spaces where a linear separator can be found

3) Robustness to Overfitting:

SVMs tend to focus on maximizing the margin between classes, which reduces the risk of overfitting, especially when there are more features than samples. This allows them to generalize effectively to unknown data.

Disadvantage of Support Vector Machine:

1) Sensitivity to Parameter Selection

The performance of SVM is significantly influenced by the kernel function and its arguments, such as the regularization parameter C and kernel-specific parameters like gamma. Inadequate choice of values may result in models that either overfit or underfit the data, requiring grid search methods or a great deal of cross-validation to find the ideal parameters.

2) Lack of Probabilistic Outputs

SVMs may not be as interpretable or useful in some situations that require probabilistic confidence measures since they do not automatically produce probability estimates for classifications.

3) Complexity in Multi-Class Classification

SVMs are inherently binary classifiers. When applied to multi-class scenarios, strategies such as one-vs-one or one-vs-all are required, which might increase processing demands and complicate modeling.

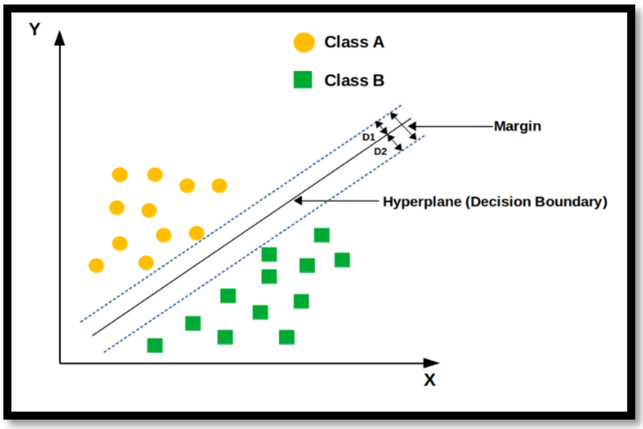

Figure 5: Support Vector Machine [14]

CONCLUSION

In this study, we considered the potential applications of machine learning techniques for detecting fraudulent activities in the banking sector. Unlike traditional rule-based methods, the study shows how machine learning models can significantly increase the accuracy and efficiency of fraud detection systems. These algorithms use patterns in transaction data to identify suspicious activities in real time, reducing money losses and improving security. Our findings show how important it is to choose the right algorithms and pre-processing techniques because feature engineering and the quality of input data have a major impact on how well ML models perform. In addition, the adaptability of machine learning allows for continuous improvement as new data becomes available, making it a valuable tool in the dynamic field of financial crime.

Our findings demonstrate how important it is to choose the right algorithms and pre-processing techniques, as feature engineering and the quality of input data have a major impact on how well ML models perform. In addition, the adaptability of machine learning allows for continuous improvement as new data becomes available, making it a valuable tool in the dynamic field of financial crime.

References

1. Khalid, A. R., Owoh, N., Uthmani, O., Ashawa, M., Osamor, J., & Adejoh, J. (2024). Enhancing credit card fraud detection: an ensemble machine learning approach. *Big Data and Cognitive Computing*, *8*(1), 6.
2. SamanehSorournejad, Z. Z., Atani, R. E., & Monadjemi, A. H. (2016). A survey of credit card fraud detection techniques: Data and technique oriented perspective. *arXiv preprint ArXiv:1611.06439 [Cs]*.
3. Karthika, J., & Senthilselvi, A. (2023). Smart credit card fraud detection system based on dilated convolutional neural network with sampling technique. *Multimedia Tools and Applications*, *82*(20), 31691-31708.
4. Mihali, S. I., & Niţă, Ş. L. (2024, May). Credit card fraud detection based on random forest model. In *2024 International Conference on Development and Application Systems (DAS)* (pp. 111-114). IEEE.
5. Pendalwar, R., Verma, A., & Patil, R. (2024, August). Machine Learning in Action: Supervised Learning Models for Classifying Credit Card Fraud. In *2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT)* (Vol. 1, pp. 1-5). IEEE.

6. Wijaya, M. G., Pinaringgi, M. F., & Zakiyyah, A. Y. (2024). Comparative Analysis of Machine Learning Algorithms and Data Balancing Techniques for Credit Card Fraud Detection. *Procedia Computer Science*, *245*, 677-688.
7. Jiang, S., Dong, R., Wang, J., & Xia, M. (2023). Credit card fraud detection based on unsupervised attentional anomaly detection network. *Systems*, *11*(6), 305.
8. Aquino, J. P. J., De Guia, A. P. G., Cruz, D. C. D., & De Goma, J. C. (2024, April). Fraud Detection in Online Credit Card Transactions Using Deep Learning. In *2024 5th International Conference on Industrial Engineering and Artificial Intelligence (IEAI)* (pp. 85-89). IEEE.
9. Haider, Z. A., Khan, F. M., Zafar, A., & Khan, I. U. (2024). Optimizing Machine Learning Classifiers for Credit Card Fraud Detection on Highly Imbalanced Datasets Using PCA and SMOTE Techniques. *VAWKUM Transactions on Computer Sciences*, *12*(2), 28-49.
10. https://intuitivetutorial.com/2023/04/07/k-nearest-neighbors-algorithm/#google_vignette
11. https://dida.do/what-is-random-forest
12. https://www.upgrad.com/blog/what-is-decision-tree-in-data-mining/
13. https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/
14. https://ashish-mehta.medium.com/support-vector-machine-svm-algorithm-for-machine-learning-350fe9139a52