HIGH SCHOOL EDITION
Journal of Student Research

How can Artificial Intelligence Techniques Effectively Enhance Credit Card Fraud Detection Systems?

Naitik Gupta

Los Gatos High School, USA

ABSTRACT

The boom in Artificial Intelligence (AI) revolution is significantly transforming our everyday lives. Every aspect of technology, process, and system implements AI to provide a superior user experience, enhanced machine capabilities and advanced problem-solving and research capabilities. However, due to this technological revolution, rise in fraud has become an enormous challenge in the digital economy. This research paper aims to test different Machine Learning (ML) models to explore real-time fraud detection capabilities accurately, particularly for credit card fraud prevention systems. Technologies like online bank transfers and smartphone payments based on credit accounts are major contributors to fraudulent transactions. AI/ML models have proven to be industry-disruptors, robust, significantly faster and produce more accurate results. These advantages have led to launch of successful companies like OpenAI. This paper uses model metrics such as Supervised, Unsupervised, and Ensemble methods to improve the detection of unauthorized transactions for fraud detection. Models will be ranked for performance using accuracy metrics, F-1, and Area Under Curve (AUC) scores. While zero false rates are not yet achievable, this study aims to reach a reasonably low level by selecting an appropriate model.

Introduction

At the turn of the century, e-commerce paved a huge path for digital transactions, resulting in the exponential use of online payment systems. An increasing number of companies are experiencing continuous astronomical growth in credit card transactions. In parallel with this digital growth, fraud rates have also increased exponentially due to increased number of transactions, opportunities, and adoption. Previously, the leading method of protection against fraud was encryption (Huling, 2023). According to the Federal Trade Commission (FTC), $8.8 billion was lost to fraudulent transactions in 2022—more than a 30% increase from 2021 (Ritchie et al., 2023). A practical fraud detection system demands the most promising models to deliver near real-time accuracy. The most prominent problem for this research is imbalanced data (Johnson & Khoshgoftaar, 2019). Because most transactions are legitimate, the ‘TRUE POSITIVE’ (fraudulent) rate is meager. Therefore, the model can simply memorize that most data is “TRUE NEGATIVE” (non-fraudulent) and still achieve a low fraudulent rate. To define the process, the paper is segmented into chronological components. Starting with Methodology section, the overall research steps are explained that includes Exploratory Data analysis (EDA), data pre-processing, and models. The paper is further structured with sections on model results and selection, implications and limitations, future work, literature review, and conclusion.

Methodology

This paper employs a multi-step methodology which involves interpretation of data through Exploratory Data analysis (EDA), pre-processing, modeling, and results. Figure 1. shows these steps chronologically. The first step, detailed in subsequent sections involves EDA for data analysis. Through EDA, outliers, trends, and clusters can be easily identified. This allows for better interpretability of the data. Following EDA is data pre-processing step, involving outlier removal, feature ranking by importance, standardization, and data splitting for precise results. The subsequent section defines the modeling step, where we experiment with different types of models to better predict fraudulent transactions: KNN, SVM, ANN, Logistic Regression, Bagging, Random Forest, and Boosting.

The results section comprehensively presents the outcome of each model, comparing them to the rest. The optimal model is selected through discussion and output metrics analysis. This section also outlines the implications of different models used in research and explores the limitations with the chosen model. The paper's future work section calls for extended research on the topic while the literature review analyses research already done to build upon their work. Finally, the conclusion summarizes the discussion and study results in a coherent format.

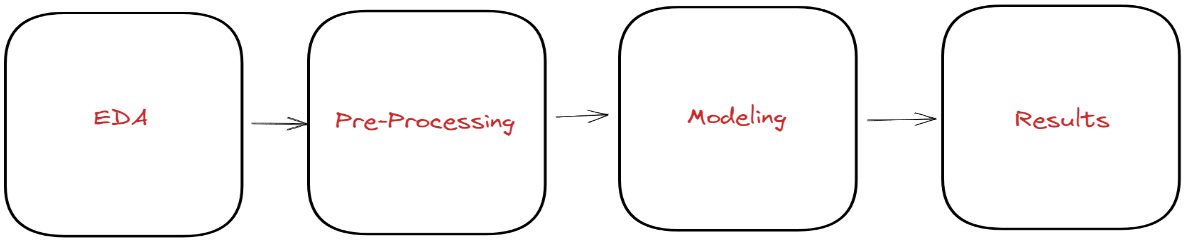


Figure 1. Experiment Sequence. The figure showcases the experimental process in which experiments are conducted.

EDA Experiment

EDA is a comprehensive data analysis technique to identify outliers, trends, and clusters. The production of graphs better allows this research to study the different relations between features. The following subsections define data collection, histogram, scatter plot, box plot, and correlation heatmap.

Data Collection

Data is fundamental to any AI/ML research. A high-quality, feature-rich, high-fidelity data model is essential to analyze credit card transactions to determine fraud. When researching credit card fraud, it is imperative to ensure the collection and organization of data with high-quality standards. Without accurate and well-organized data, it will be harder to identify patterns and trends that skew results in our model's predictions. The data utilized in this examination is from the Kaggle database “Abstract data set for Credit card fraud detection” (Joshi, 2018). Kaggle provides a publicly accessible dataset designed explicitly for credit card fraud detection, which is well-suited for research for the above reasons. This specific dataset only incorporates contents on transactions (3075 rows) that are not necessarily fraudulent.

Below, Table 1 represents each feature, feature type, and description for all its features included in the dataset. Outside a binary measure of fraud, every variable collected is either related to the transaction's amount (monetary amount) or its chargeback (number of disputed transactions won), which was used. However, the data must be interpreted in the context of a machine-learning model, as discussed in the next section.

Table 1. Features information description and feature type. Explains what each feature incorporates and the type.

Feature Name	Feature Description	Feature Type
Merchant_id	A unique sequence ID attached to a merchant	Numerical
Average Amount/transaction/day	The average amount of transactions per day	Numerical
Transaction_amount	The monetary amount per transaction	Numerical
Is declined	If the transaction is declined	String
Total Number of declines/day	The number of declines per day	Numerical
isForeignTransaction	If the transaction is foreign	String
isHighRiskCountry	If the transaction is high-risk	String
Daily_chargeback_avg_amt	The daily average chargeback amount	Numerical
6_month_avg_chbk_amt	6 Month average chargeback amount	Numerical

Histogram – Transaction Amount vs. Occurrence

Histograms are a precious tool in EDA for credit card fraud detection. By plotting the transaction amount (fraud amount) against the number of times it occurred (Occurrence), a histogram can show the frequency of fraudulent transactions at different transaction amounts, for which we can identify patterns and anomalies. For example, the histogram in Figure 2 shows that most fraudulent transactions occur at lower transaction amounts or that specific transaction amounts are more likely to be fraudulent. Many low-amount transactions may be passed off as hidden fees or forgotten purchases. Problematically, this can amount to enormous sums of money over a period and can reach thousands of dollars per person. To prevent this, we can analyze these transactions and have models figure out patterns to help predict more actual frauds. Moreover, recovering money from fraud artists is not commercially viable for every fraud. Therefore, the most significant prevention is not allowing it to happen and conducting real-time fraud analysis. To avoid the skewing of the graph with the new dataset, we must use a logarithmic function to approximate normality and reduce the impact of outliers.

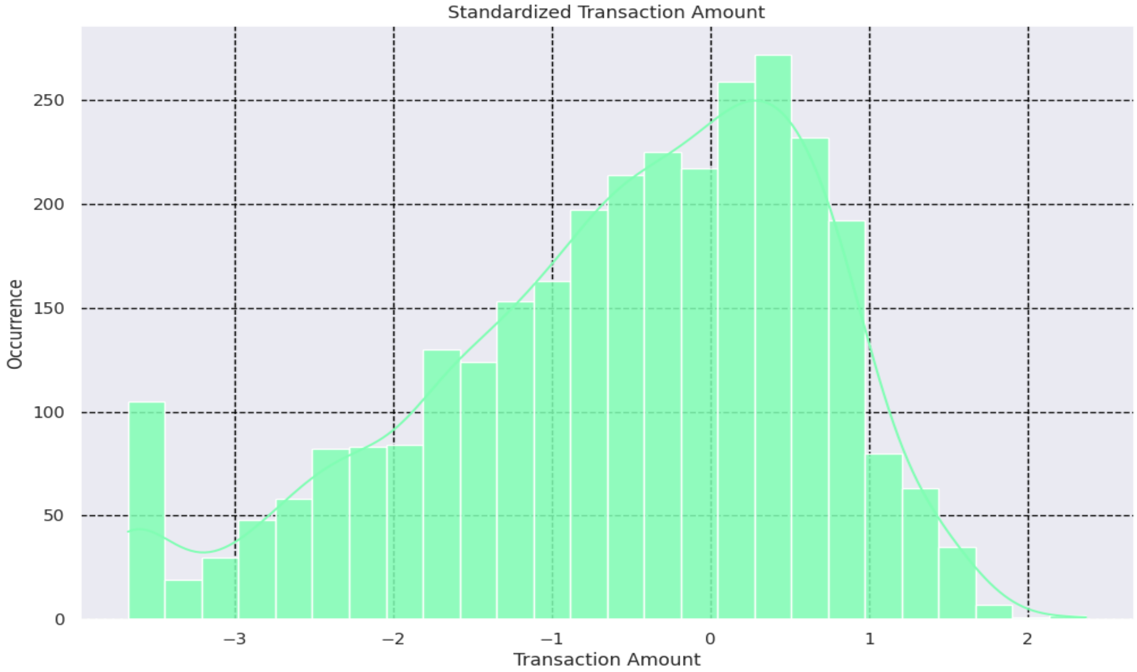


Figure 2. Standardized Transaction Amount vs Occurrence Histogram. The figure pertains to information on a logarithmically transformed data set matching the transaction amount to the occurrence.

Scatter Plot – Transaction Amount vs. Daily Chargeback Average Amount

As shown in Figure 3, a scatter plot is generated to comprehend the relationship between the transaction amounts and daily chargeback in credit card fraud (the measure of which disputed transactions were won). The scatter plot was created using data from fraudulent transactions and plotted the transaction amount (how much the value of the transaction was). The scatter plot showed a negative correlation between transaction amount and daily chargeback, suggesting that higher transaction amounts align with lower daily chargebacks. When the amount of a fraudulent transaction is low, the number of chargebacks significantly increases in density, but when the amount is higher, the density is very low. As mentioned, this is only logical as banks, credit card issuers, and financial institutions are considerably hesitant to refund transactions that pertain to higher amounts, as that can be two-sided. Inside the graph, several clusters are grouped. These clusters show the relationship that the numbers pose in our data. Furthermore, this illustrates the great importance of models predicting fraud with higher accuracy for low amounts and high amounts because if there is a false positive, then the vendor is not only losing money but is essentially the target of groups looking for ideal conditions for false positives.

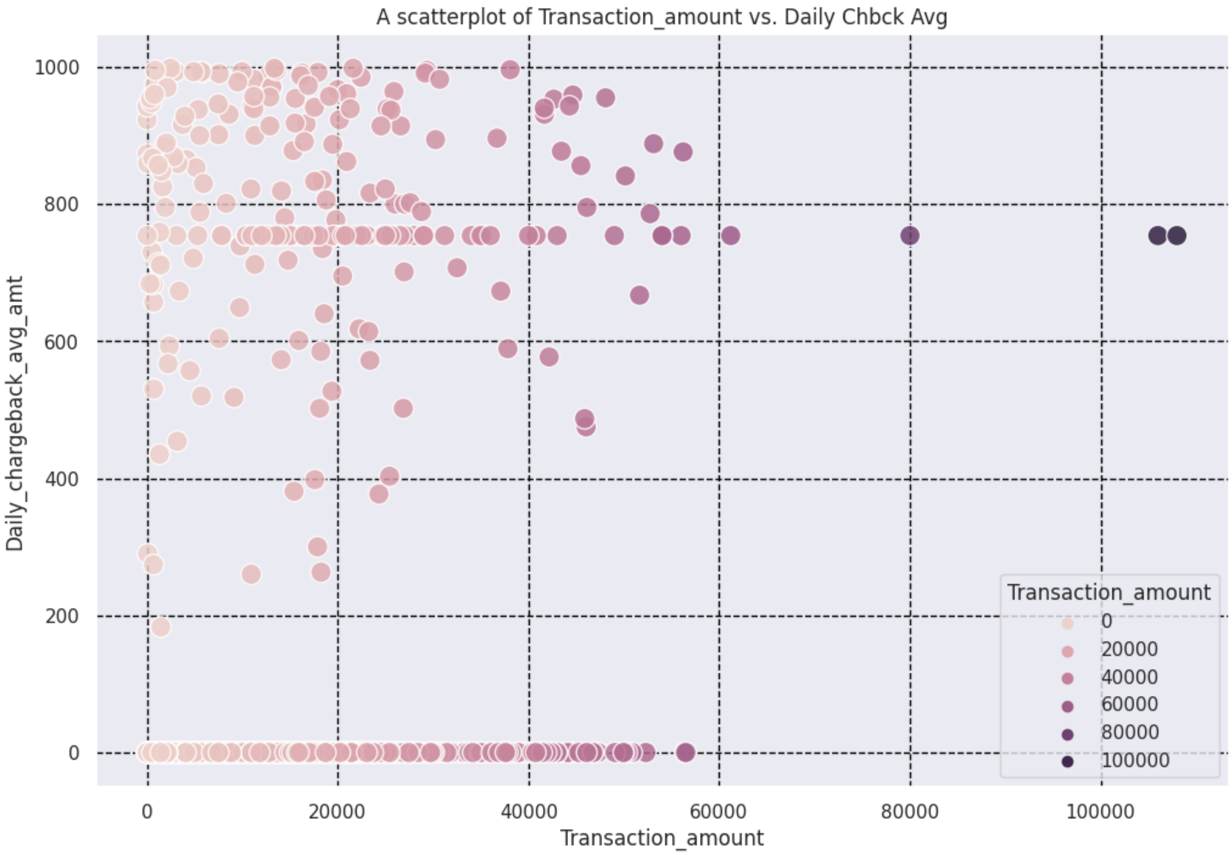


Figure 3. Average Daily Chargeback vs Transaction Amount Scatterplot. The legend is contained in the figure. The figure pertains to information on the amount of the transaction vs the daily average chargeback amount received. The central cluster is towards a lower transaction amount to a higher daily chargeback amount of occurrence.

Box Plot – Transaction Amount vs. Number of Cases

As shown in Figure 4, box plots can quickly summarize data and make it possible to visualize metric points. The plot shows the median, mode, extrema, and spread. The relation graphed is the transaction amount to the number of cases. As previously explained, the “TRUE NEGATIVE” rate is undersized, causing the median (Quartile 2) to be shallow. Figure 4 shows that the extrema is also towards the lower end. This plot can also warn about skewness. Skewness can be indicated if the main box is leaning towards one side, like here. Moreover, the box represents the interquartile range (IQR), the change in the middle 50% of the data. IQR is the amount of spread in the middle 50% of a dataset.

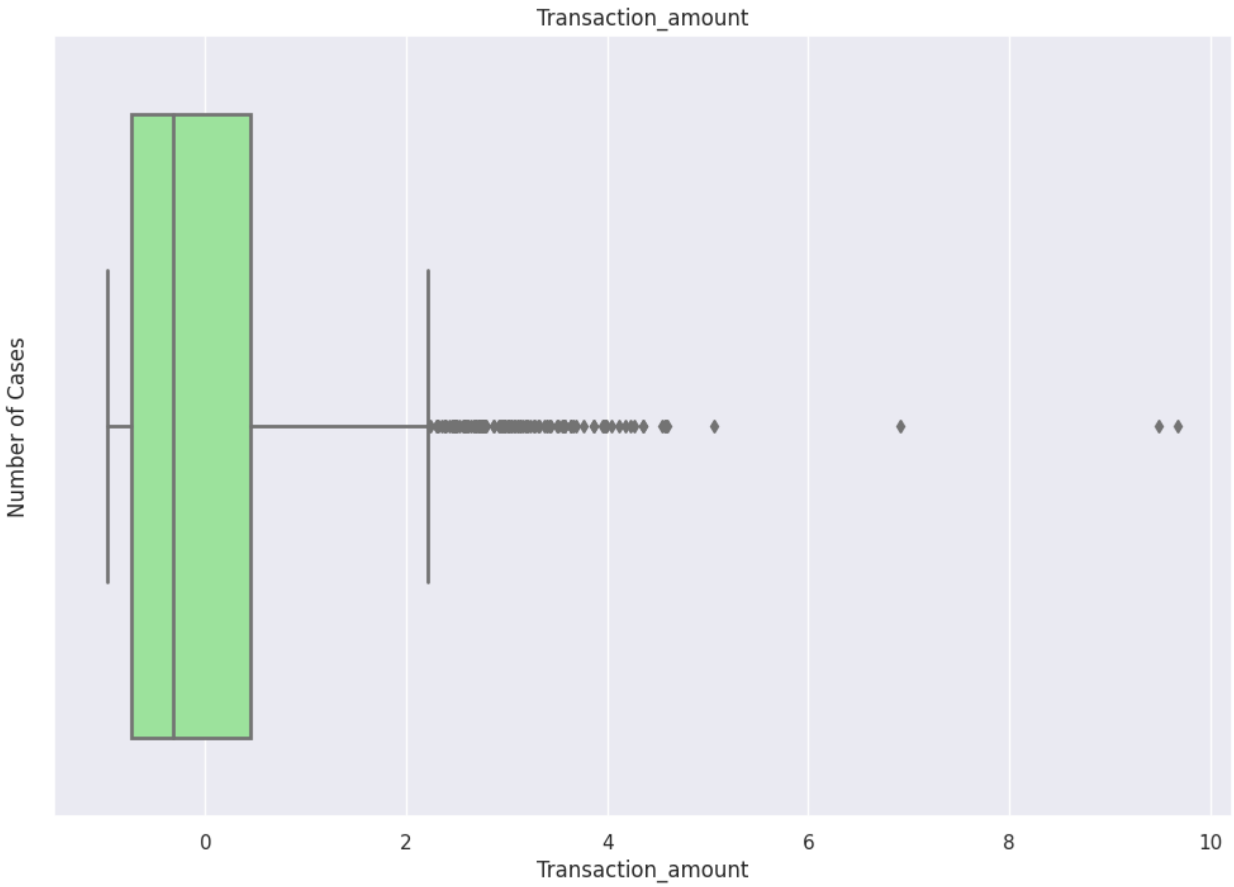


Figure 4. Transaction amount vs Number of cases box plot (LogTransformed). The figure pertains to information on the quartile ranges of the data set and provides valuable insight into the metrics of data. Aside from a few outliers, most of the data is in the lower range.

Correlation Heatmap – All Features

A correlation heatmap, as shown in Figure 5, represents the correlation between features. The heatmap is the standard way in EDA to find correlation. High correlational data is unsuitable for a model because it requires many resources. Since it is similar data, the efficiency is also reduced. Moreover, patterns are easily spotted depending on the value of correlation. Then, there is also the dimensionality problem. This is where a low-dimensional setting meets with high-dimensional data, causing computational inefficiencies. The values with a "1" correlation are identical, the lower the color and value, the lower the correlation.

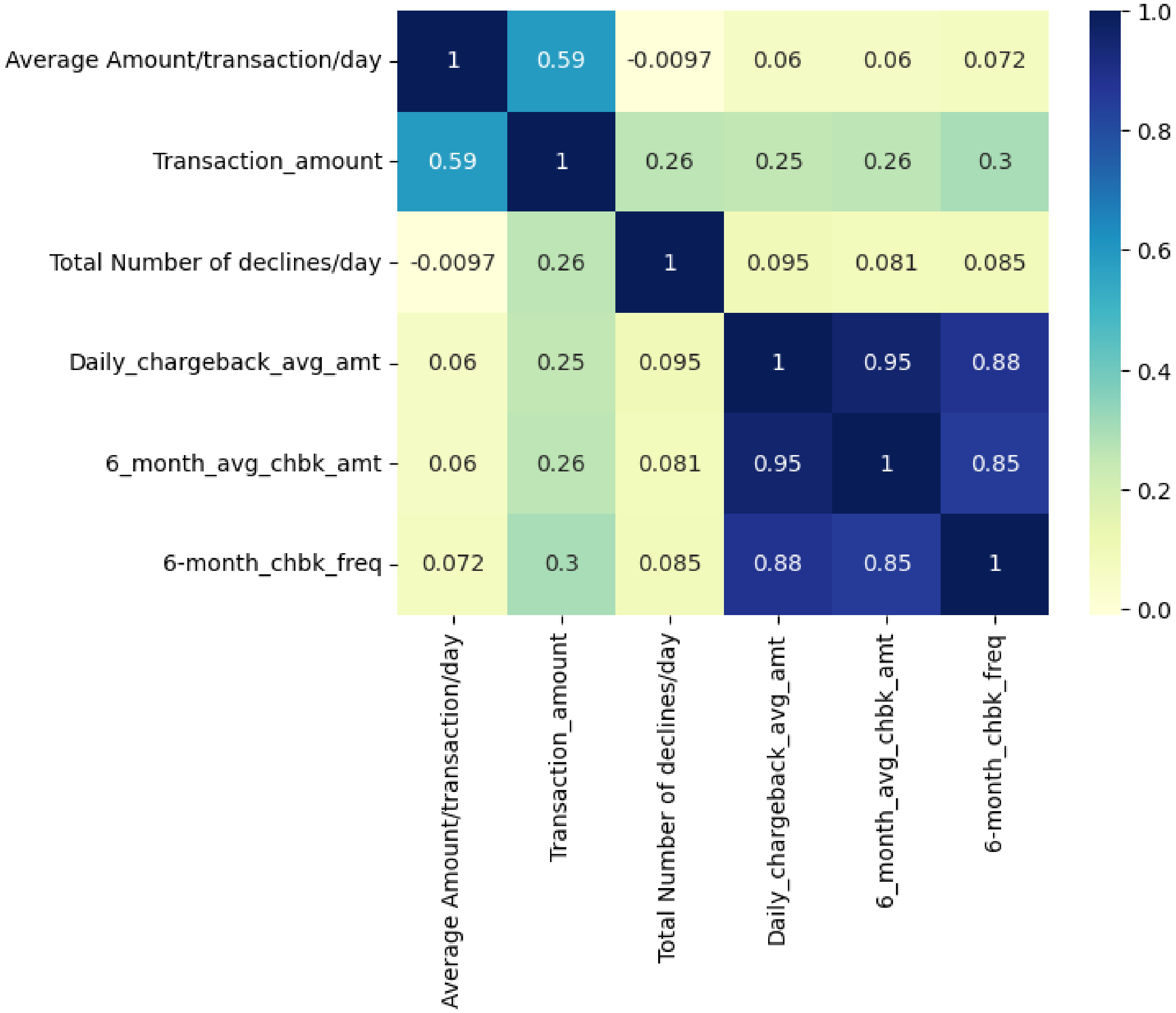


Figure 5. All features correlation matrix. The legend pertains to the figure. The correlation matrix explains the correlation levels between certain features in a matrix format. The darkest squares have the highest correlation with a maximum of 1 (equivalence property).

Data Preprocessing

The data preprocessing section defines post-EDA steps. After patterns and numerical data are observed visually, it must be incorporated into the dataset. The following subsections discuss how we removed outliers, scored feature importance, standardized data, and split the data. Then, we discuss all the models chosen to train the data on and the specifics of each.

Remove Outliers

The outliers in this study were removed with quartile ranges. The IQR is the spread in the data so that the following equation can be used: $Q1 - 1.5IQR = Outlier\ and\ Q3 + 1.5IQR = Outlier$. This removes any data points 1.5 IQR below quartile 1 and 1.5 IQR above Q3. If the outliers were left, the model would make very different predictions because the outlying data is highly different.

Feature Importance

Feature importance was discovered through the “RandomForestClassifier,” which labels each feature with a score (NVIDIA, n.d.). Features with a score below 0.05 importance are dropped because they are irrelevant or not crucial to the measure we are predicting. The accuracy score will be one of the most significant measures to score the feature's importance (KDNuggets, 2020).

Standardized Data

Data standardization is the process of making everything into a consistent and readable format. Inconsistent data leads to incorrect analysis, which will result in incorrect predictions. If the data is standardized, models can compare and process the data accurately (Hale, 2021) (ScienceDirect, n.d.).

Split Test vs. Train Data

The data was split into training and test sets. This ensures the model is not memorizing the training and overfitting the data. Another issue that might happen is that the model can predict “NOT” every time because of the low fraud rates. Most transactions are typical, and this would skew results from real-life implications. Our research used 80% of the data for training and 20% for testing. The following section explains each model that was fitted with the data.

Models

The following subsections will define the various models used in research to find fraud in credit card transactions. Every sub-section will describe the model and include the analysis of credit card data under consideration.

KNN Model

KNN (k-nearest neighbors) is a prevalent non-parametric classification algorithm that designates class labels based on the majority vote of its neighbors. For now, implementing the KNN model on the processed dataset and evaluating its performance on the accuracy score is the best way to assess it. This KNN model is classified based on whether the transaction is fraudulent (IBM, n.d.). Some real-life examples for KNN might be finding patterns in fraudulent cases based on the majority votes of its neighbors (data points). The majority class label determines the label with the most vital data points among its k-nearest neighbors. As seen in the Confusion Matrix in Figure 6, the model incorrectly labeled seven “TRUE” class data points belonging to the False Positive (FP) class. Similarly, the model incorrectly classified six “FALSE” class data points in the False Negative (FN) class.

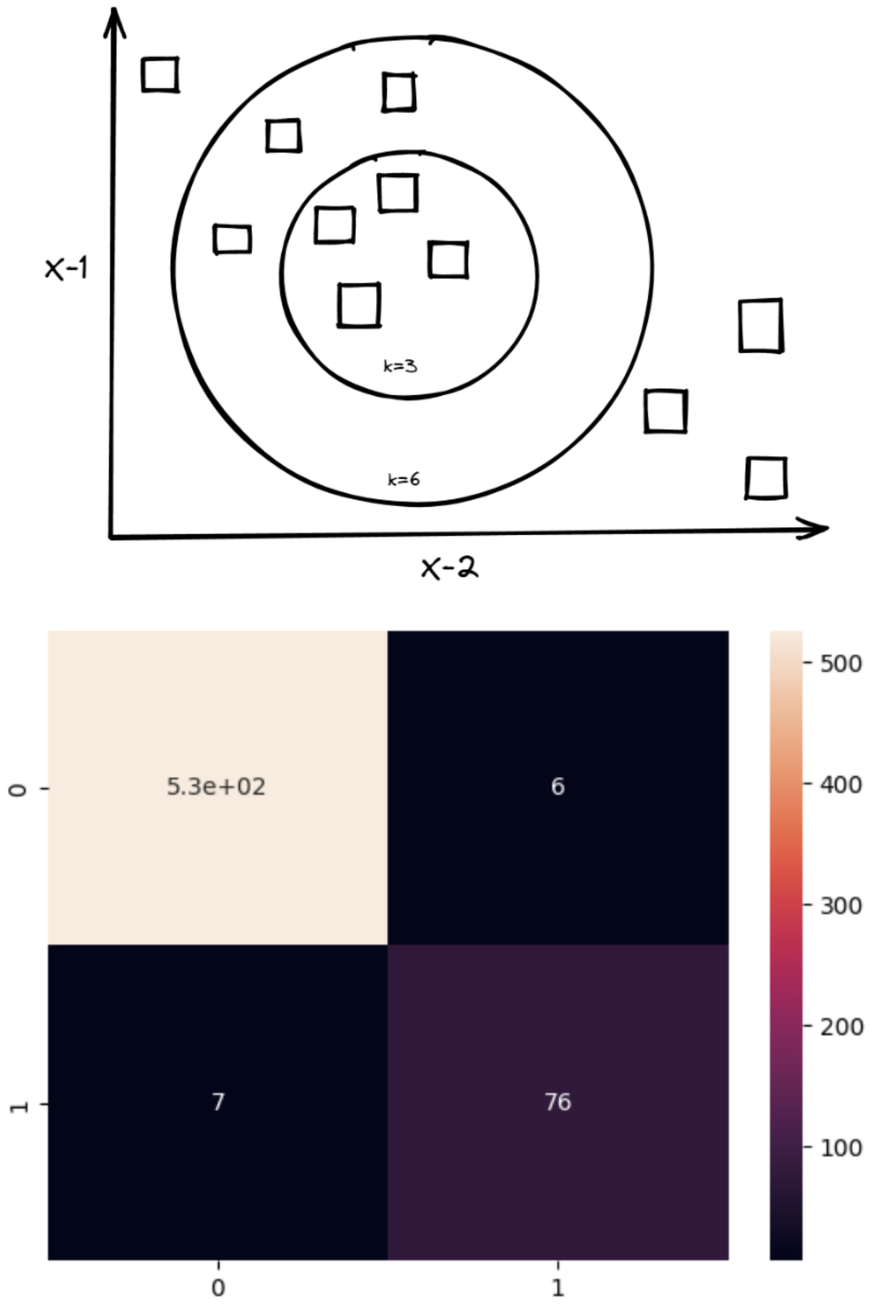


Figure 6. KNN Diagram and Confusion Matrix. The KNN model diagram explains how the K-nearest neighbor algorithm works, and the confusion matrix represents the results of how the model is performed.

SVM Model

Support vector machines (SVMs) are classified as a type of supervised ML model. The model works by making two classes divided by a hyperplane. This plane must be the most significant distance away from any point. The accuracy score is the best way to evaluate the model. In this case, our model separates whether each transaction is fraudulent. Problems with a binary outcome are usually the best because the model is divided into two classes.

This model also suits our dataset because fraud is a binary concept. This also increases the accuracy that the model might present. As seen in the Confusion Matrix in Figure 7, the model incorrectly labeled five transactions to the False Negative (FN) class, and the model classified five transactions to the False Positive (FP) class.

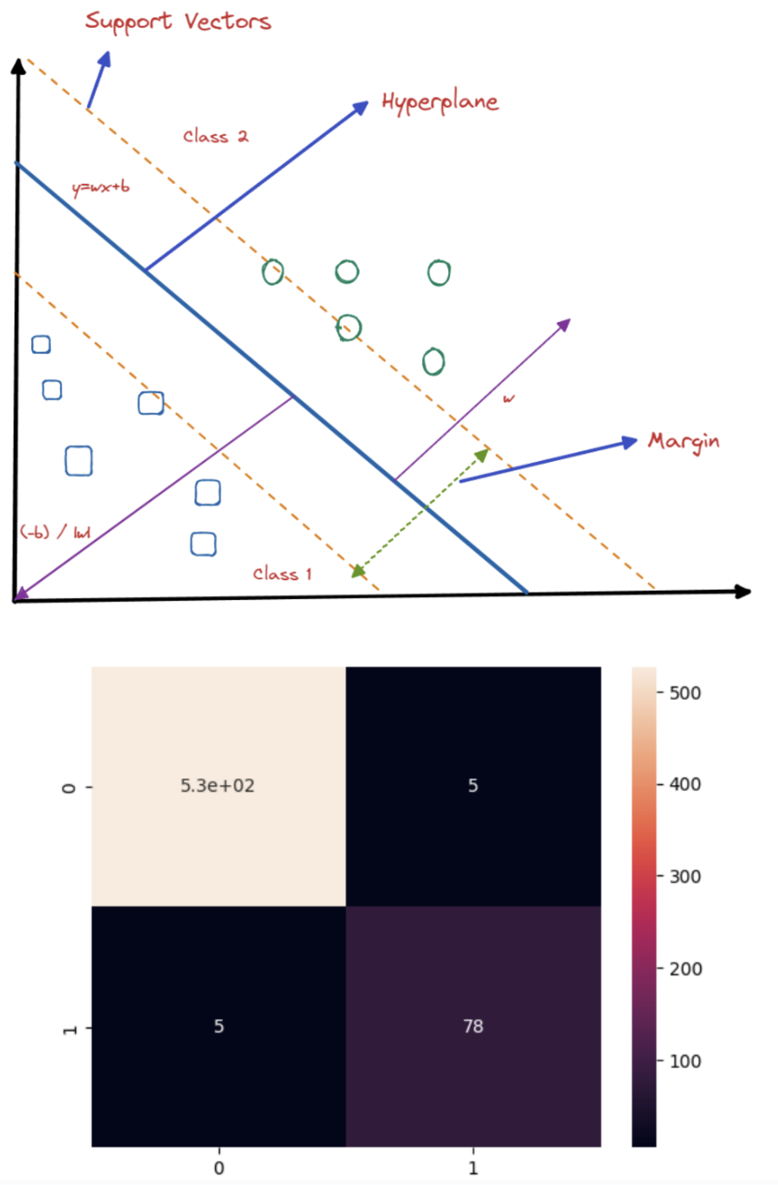


Figure 7. SVM Diagram and Confusion Matrix. The SVM diagram explains how the model works, and the confusion matrix shows the results of how the model performed.

ANN Model

Artificial Neural Networks (ANNs) is a machine learning model related to the human brain. They are made up of interconnected nodes called neurons, which learn to recognize patterns in data. One key advantage of ANNs is that they learn from data without being programmed to do so. This makes them perfect for tasks that are challenging to program, such as recognizing objects in images or understanding the meaning of text. ANNs can learn from large amounts of data, which gives them an advantage over other machine-learning models because of their vast scale (Editors ScienceDirect, n.d.). Figure 8's confusion matrix shows three instances of False Negatives and four instances of False Positives.

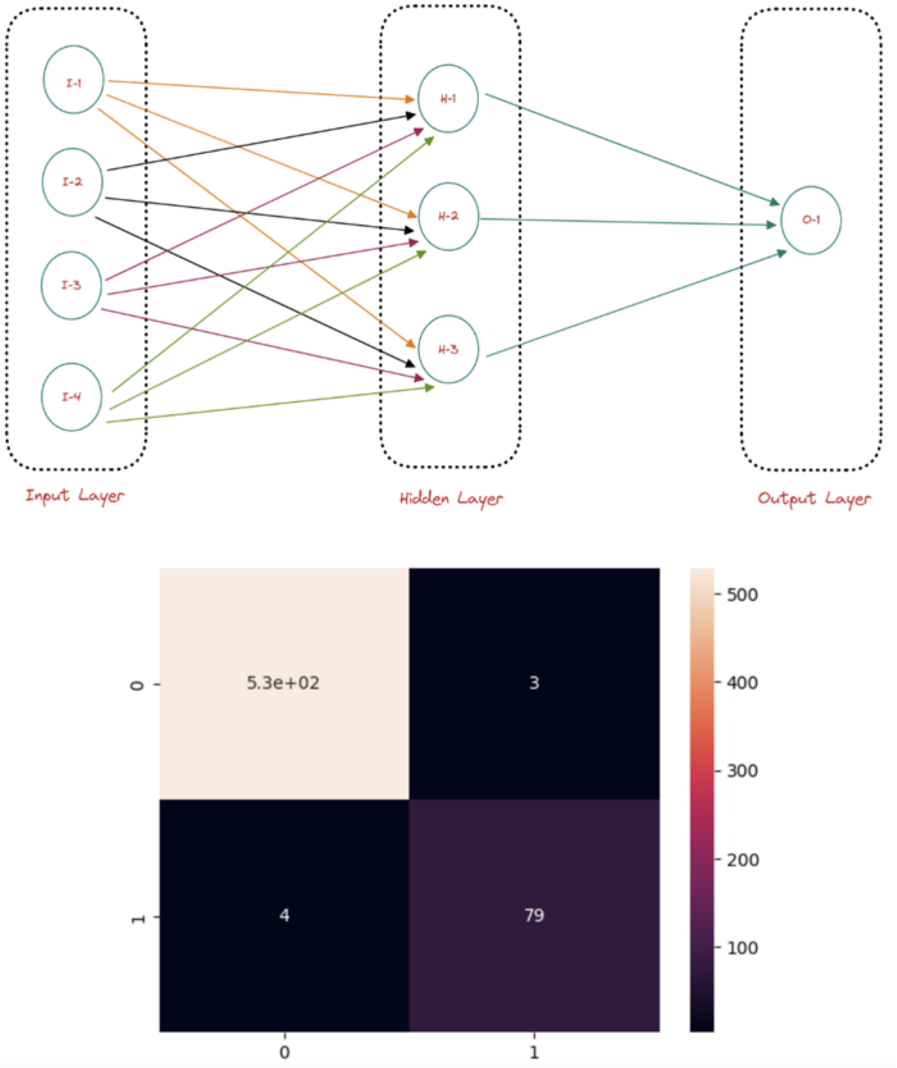


Figure 8. ANN Diagram and Confusion Matrix. The diagram explains how the ANN model works, and the confusion matrix explains how the model is performed. In this case, the ANN model scores slightly better than the other models.

Logistic Regression

Logistic regression is a statistical method used to predict the probability of an outcome. This study employed logistic regression to determine if a transaction was fraudulent or not. The logistic function is a nonlinear function that maps the input values to a probability value between 0 and 1 (Editors IBM, n.d). In Figure 9's confusion matrix, eight transactions are classified as False Negatives, and twenty-eight are classified as False Positives. These scores indicate the inefficiencies in the regression model.

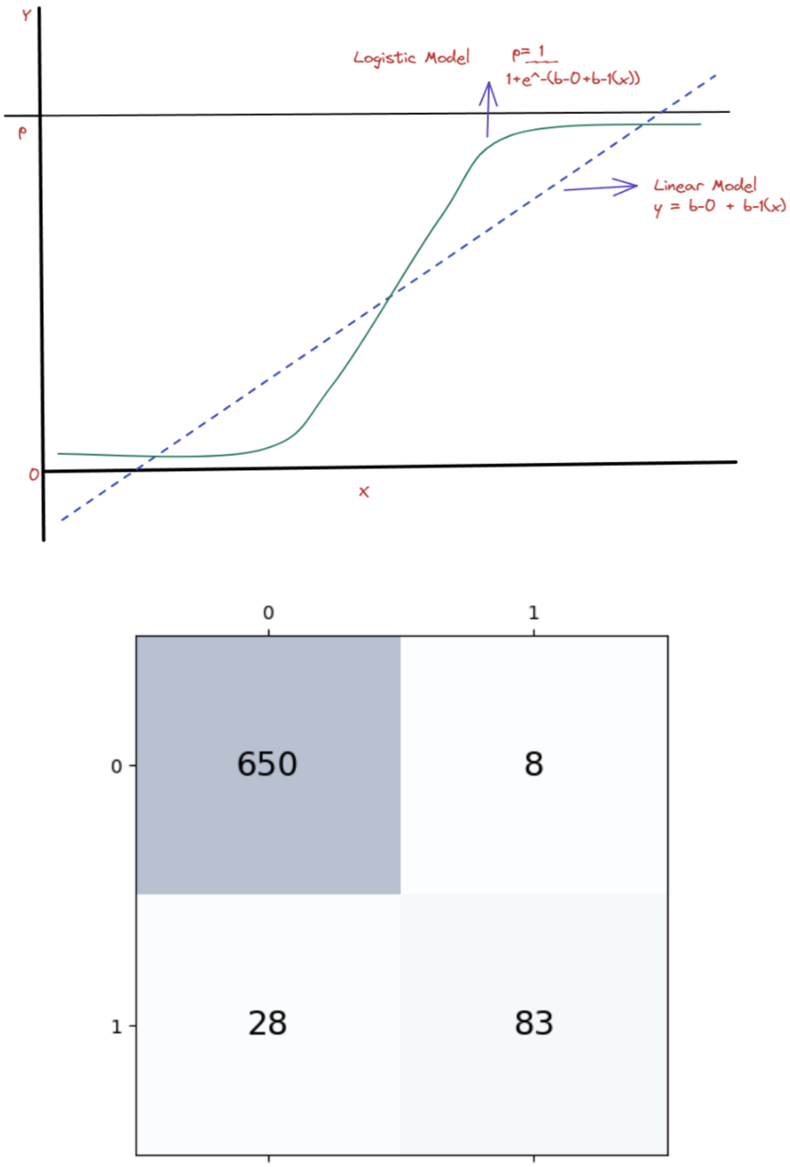
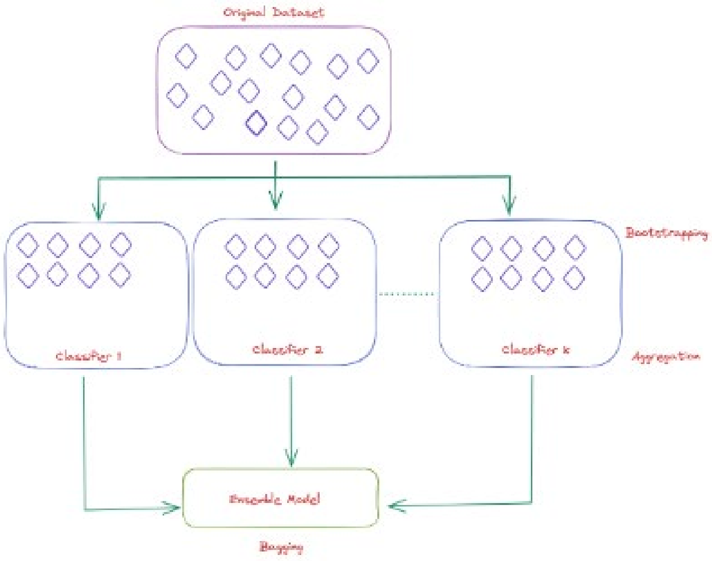


Figure 9. Logistic Regression Diagram and Confusion Matrix. The diagram explains how the model works and performs, and the confusion matrix shows how the model performs. Here the model performance is slightly worse than the others.

Bagging

Bagging creates multiple models from a single dataset and then averages their predictions. This means that the model runs multiple simulations, with the average prediction constituting the output. This method helps to decrease variance in a tree method. A tree method is a supervised model used to categorize predictions based on previous questions answered (IBM, n.d.). In this model, there are forty-five False Negatives and fifty-nine False Positives as shown in Figure 10.



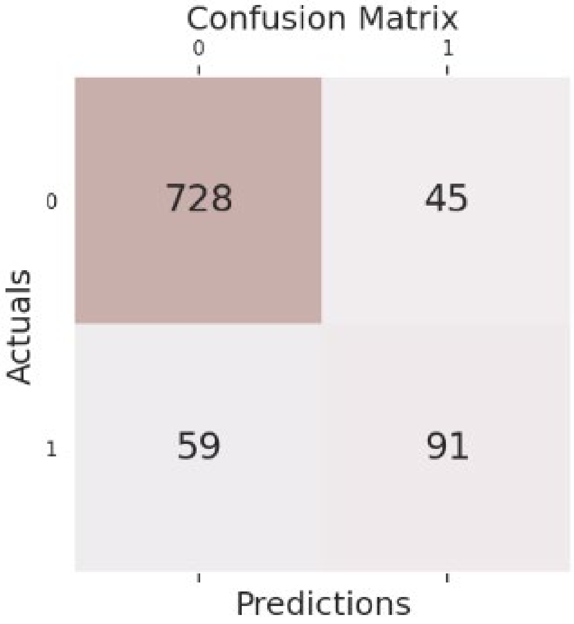
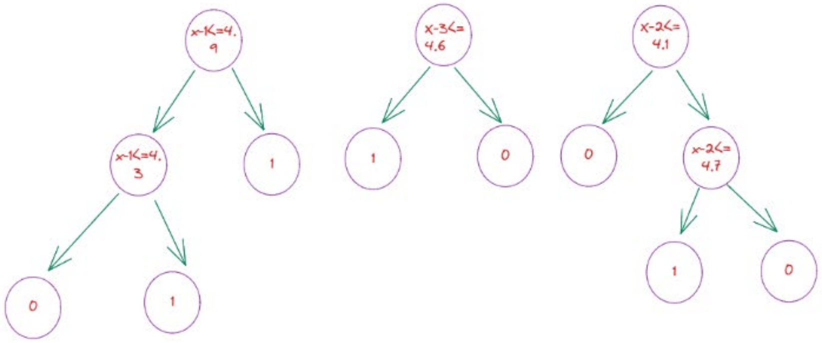


Figure 10. Bagging Diagram. The diagram shows how the bagging method works. Bagging is an ensemble method and goes through many classifiers before the final ensemble model.

Random Forest

Random forests are a type of bagging method that creates multiple decision trees from a single dataset. Each tree is trained on a randomly selected subset of the data, which is then subsequently averaged. Random forest is an ensemble method used to solve regression and classification tasks. (NVIDIA, n.d.). This model has nineteen False Negatives while having twenty-three False Positives.



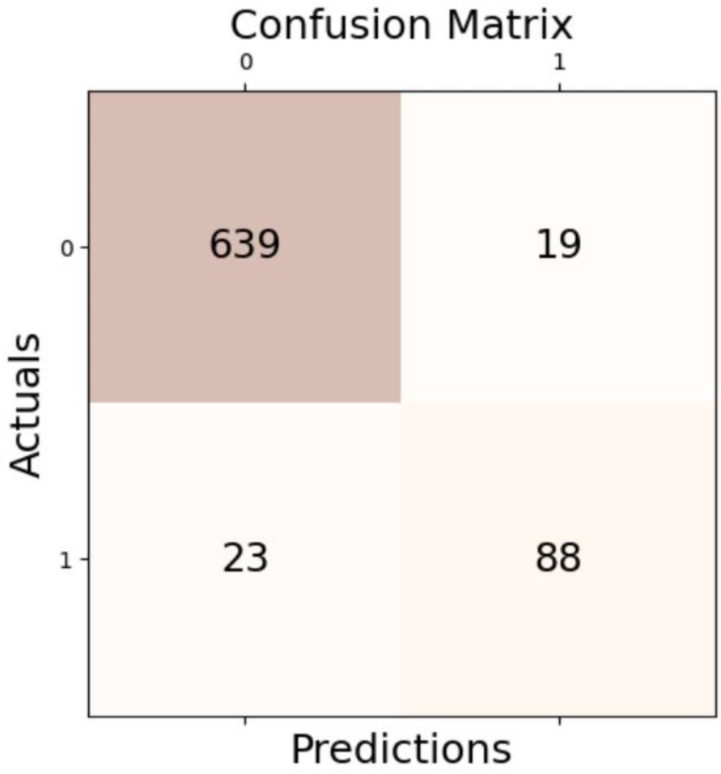
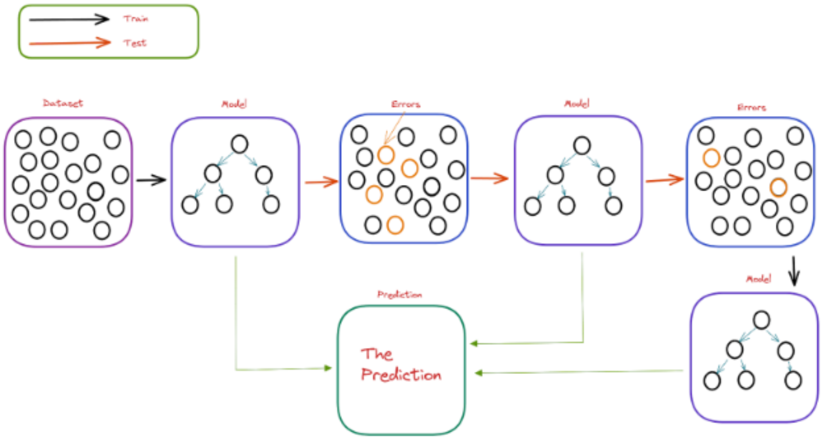


Figure 11. Random Forest Diagram. The diagram shows how random forest, a tree method, works and shows an example.

Boosting & XG Boost

Boosting is another type of ensemble method that creates multiple models from a single dataset. However, unlike bagging, boosting models are created sequentially, with each subsequent model trained to correct the errors of the previous models. This represents a try-check-improve cycle to increase performance with each run of the model. One example of this is XGBoost, a machine-learning library that uses gradient-boosting prediction. Designed to be efficient and scalable, it has proved to be a notable model as it learns by adding weak learners to a model. Each weak learner is trained to predict the residuals of the previous learners, so the predictions of all the learners are then combined to produce the final prediction. This process is repeated once the accuracy goal is achieved or efficiency is maxed out (Editors, n.d.). There are two False Negatives in this confusion matrix, while there are thirty False Positives.



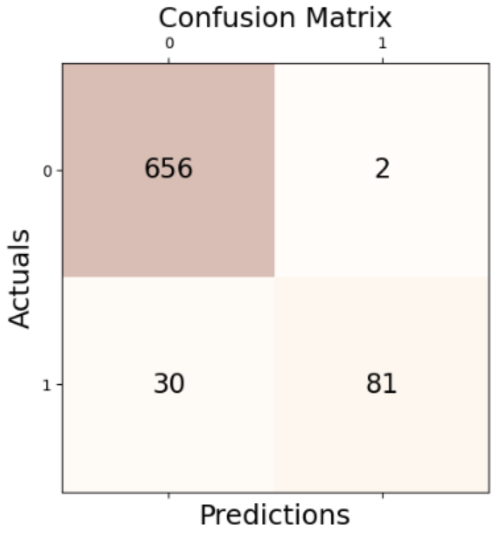


Figure 12. Boosting and XG Boost Diagram. Boosting is another ensemble method that creates multiple models from a single dataset. However, boosting is sequentially based, unlike bagging, with each model trained on the last.

Results

After applying GridSearchCV to each model, we evaluated the performance of each model using three metrics: accuracy, F-1 score, and area under the curve (AUC). Here, accuracy is the number of correct predictions from total predictions. The F-1 scores are the harmonic means of precision and recall for each model listed in Table 2. AUC, or area under curve, is the probability of a random chosen positive case will induce a higher prediction rate than a negative case. The metrics listed above are going to help us classify each model's performance.

AUC is the probability that a randomly chosen positive case will have a higher predicted probability than a randomly chosen negative case. These metrics help us assess how well each model can classify the credit card fraud cases and avoid false positives and false negatives.

Model Results

The table below shows all the model's accuracy scores post-GridSearchCV (Editors, n.d.) and is linked to a confusion matrix graph. GridSearchCV is a method used to adjust hyperparameters in ML models by researching through a specified parameter grid to find the optimal assemblage of hyperparameters that results in the best performance.

Table 2. The table explains each model's accuracy, F-1, and AUC scores. The figure also explains each model's advantages and disadvantages, simplifying the model selection process.

Models	Accuracy (Rounded)	F-1 Scores (ACC)	AUC Score	Advantages	Disadvantages
KNN	0.98	0.99, 0.94	0.96 (Rounded)	Relies on neighbor proximity for decision-making	Sensitive to distance metric and neighbor count
SVM	0.97	0.98, 0.97	0.968	Relies on hyperplane separation for decision-making	Sensitive to kernel choice and hyperparameters
ANN	0.99 (0.989)	1, 0.99	0.99	Utilizes artificial neural networks for complex pattern recognition	Computationally Extensive, prone to overfitting
Logistic Regression	0.95	0.97, 0.92	0.96	Uses linear decision boundaries for classification	Limited to linear boundaries
Bagging	0.89	0.88, 0.8	0.84	Enhances model robustness through ensemble methods	Ineffective with strong base models
Random Forest	0.95	0.9, 0.87	0.94	Uses multiple trees for better predictions	Lack of interpretability
Boosting / XGBoost	0.94	0.95, 0.93	0.967	Strengthens weak models iteratively	Sensitive to noise and outliers

Model Selection and Discussion

The above-listed models were trained and tested using the standard metrics defined earlier. The ANN model had the highest accuracy, including for the F-1 scores. (0.99 and 0.97) Based on multiple tests, the ANN model consistently outperformed the other models in terms of comparison. Disadvantages to using an ANN model included difficulty with training and being computationally expensive. For larger institutions, the scale of their application requires an adaptable and easily maintainable system to carry out their daily tasks. Even if the ANN model has the highest accuracy, other metrics must be considered. For example, the ANN model takes longer to compute than others in this case. The ANN model also has the highest F-1 scores as well as the best AUC scores. There is, however, a detail to keep in mind as ANN models are comparatively newer than the others and have multiple layers in which calculations are performed.

Model Implications

KNN, SVM, ANN, Bagging, Random Forest, and Boosting are all machine-learning algorithms with real-world implementations. These models work not only in theoretical situations but also in real-life implementations of many use cases. KNN is used for classification and regression tasks. It is used in spam filtering, fraud detection, and medical diagnosis, all implying intensive, transaction-based filtering. SVM is used for classification tasks, which also implies classifying transactions based on data. ANN is used for classification and regression tasks in natural language processing, speech recognition, and image recognition. This also directly applies as we process complicated transactions through ANN's node-based systems. Bagging is used for classification and regression tasks. It is used to improve the accuracy of machine learning models. Random Forest is used for classification and regression tasks. Boosting is used for classification and regression tasks. Random forest is also used to determine the importance of features. Finally, each method has pros and cons to consider when implementing it in fraud detection. We aim to minimize false positives, and picking a suitable model is essential. Thus, our final selection remains the ANN model because it has the highest metrics and performance among the others. However, it is essential to note that ANN models are prone to overfitting. This is because of their memorization aspects, as the model may simply memorize one aspect of the answers. For example, in the case of fraud detection, most cases are actual transactions (class imbalance), which causes the model to simply predict a 'NO' for fraud every time, resulting in high accuracy.

Limitations

To explore the limitations of the ANN model, we must not only look at the theoretical limitations (overfitting, etc.) but also the physical limitations (computational expense, etc.): ANN models can be challenging to train and can be computationally expensive. They can also be sensitive to the size and quality of the training data. ANN models can be predisposed to overfitting, leading to poor performance on unseen data. They are also challenging to interpret, which may cause unnecessary problems in the situation at the point.

In addition, this study found that the data sample size needs to be large for the ANN model to train and test the data. However, the application we are researching, banking, has millions of data points and will be a suitable fit. Furthermore, there may be limited access to data as banking involves secure and confidential data. Factoring other limitations, including missing or null values, higher quality datasets, and feature importance, these components also incur limitations as they directly contribute to the model.

Future Work

Furthermore, several steps can be implemented in this study to delve deeper into fraud detection. First, trying out more ML models would broaden the study and different model suggestions. Moreover, Principal Component Analysis (PCA) can be utilized as a statistical method to reduce the dimensionality of a dataset. PCA increases the interpretability of a dataset while preserving most of the critical data. Cluster analysis is a statistical technique that groups similar observations into 'clusters.' These can help describe trends and patterns in our data, which can effectively help the prediction models.

Recently, Large language models (LLM) have changed the landscape of AI/ML; with this method, the output contains more promising results. LLMs are being adopted to predict the next transaction of a customer, which can help payment firms preemptively assess risks and block fraudulent transactions. Generative AI also helps combat transaction fraud by improving accuracy, generating reports, reducing investigations, and mitigating compliance risk. Generating synthetic data is another important application of generative AI for fraud prevention (Levitt, 2023). Synthetic data can improve the number of data records used to train fraud detection models and increase the variety and sophistication of examples to teach the AI to recognize the latest techniques employed by fraudsters.

Literature Review

Fraud detection is a paramount problem in the financial sector, posing substantial risks to both financial organizations and consumers. In accordance with this challenge, implementing AI techniques has appeared as a new approach to enhance fraud detection capabilities. In today's systems, fraud detection relies on rule-based systems. These methods are losing their advantage; even if they are protective to some extent, they cannot adapt to the changing fraud detection methods. As highlighted by Ileberi, Sun, and Wang, rule-based systems cannot compare with emerging fraud schemes, resulting in higher fraud cases. Recent breakthroughs in deep learning, especially the popularity of deep neural networks, have taken attention in the field (Ileberi et al., 2022). Research by Abakarim, Lahby, and Attioui illustrated the effectiveness of convolutional neural networks (CNNs) in image-based fraud detection and recurrent neural networks (RNNs) in interpreting sequential trade data (Abakarim et al., 2023). New studies are emerging daily due to the changing world of fraud detection, which can help further increase the perspective on the study.

Conclusion

The primary focus of this research paper is to study how various model(s) can be used in fraud detection using AI/ML methods. In this research on credit card fraud detection, the ANN model has proved to be the most optimal model because it has the highest accuracy and F-1 scores. It will, therefore, be most effective in detecting credit card fraud. However, there are some limitations to the ANN model like computational expense and training difficulty.

To advance this study further, some next steps may be adopted like – adding more data types, increasing the complexity of data, adding more models like Large Language Models (LLMs). As technology rapidly advances and adapts, so will the number of fraudulent cases. Therefore, it is imperative to keep developing and refining these models to lower the occurrences of fraudulent transactions. Overall, this paper has provided a comprehensive survey of the different machine learning models that can be used for fraud detection and for elucidating the integration of AI to mitigate false positives.

References

Huling, N. (2023, September 7). What is fraud prevention, and why is it important? ComplyAdvantage. https://complyadvantage.com/insights/what-is-fraud-prevention/

Ritchie, J. N., & Staff in the Bureau of Competition & Office of Technology. (2023, February 23). New FTC data show consumers reported losing nearly $8.8 billion to scams in 2022. Federal Trade Commission. https://www.ftc.gov/news-events/news/press-releases/2023/02/new-ftc-data-show-consumers-reported-losing-nearly-88-billion-scams-2022

Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, *6*(1), 1-54. https://doi.org/10.1186/s40537-019-0192-5

Joshi, S. (2018, April 9). Abstract data set for credit card fraud detection. Kaggle. https://www.kaggle.com/datasets/shubhamjoshi2130of/abstract-data-set-for-credit-card-fraud-detection/data

NVIDIA.com. (n.d.). What is a random forest? NVIDIA Data Science Glossary. https://www.nvidia.com/en-us/glossary/data-science/random-forest/

KDnuggets.com. (n.d.). Model Evaluation Metrics in Machine Learning. KDnuggets. https://www.kdnuggets.com/2020/05/model-evaluation-metrics-machine-learning.html

Hale, J. (2021, December 13). Scale, standardize, or normalize with Scikit-Learn. Medium. https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02

ScienceDirect. (n.d.). Data standardization. ScienceDirect. https://www.sciencedirect.com/topics/computer-science/data-standardization

IBM.com. (n.d.). What is the K-nearest neighbors algorithm? IBM. https://www.ibm.com/topics/knn

Editors, S. (n.d.). Artificial Neural Network Model. ScienceDirect. https://www.sciencedirect.com/topics/engineering/artificial-neural-network-model

Editors, I. (n.d.). What is logistic regression? IBM. https://www.ibm.com/topics/logistic-regression

IBM.com. (n.d.-a). What is bagging? IBM. https://www.ibm.com/topics/bagging

Editors, N. (n.d.). What is XGBoost? NVIDIA Data Science Glossary. https://www.nvidia.com/en-us/glossary/data-science/xgboost/

Editors, R. (n.d.). GRIDSEARCHCV: Grid Search CV. RDocumentation. https://www.rdocumentation.org/packages/superml/versions/0.5.6/topics/GridSearchCV

Levitt, K. (2023, December 16). How is AI used in fraud detection? NVIDIA Blog. https://blogs.nvidia.com/blog/ai-fraud-detection-rapids-triton-tensorrt-nemo/#:~:text=LLMs%20are%20being%20adopted%20to,investigations%20and%20mitigating%20compliance%20risk

Ileberi, E., Sun, Y. & Wang, Z. (2022). A machine learning-based credit card fraud detection using the GA algorithm for feature selection. J Big Data, 9, 24. https://doi.org/10.1186/s40537-022-00573-8

Abakarim, Y., Lahby, M., & Attioui, A. (2023). A Bagged Ensemble Convolutional Neural Networks Approach to Recognize Insurance Claim Frauds. Applied System Innovation, 6(1), 20. https://doi.org/10.3390/asi6010020