ISSN No:-2456-2165

A Synergistic Approach for Enhancing Credit Card Fraud Detection using Random Forest and Naïve Bayes Models

Joe Essien
Department of Computer Science and Information Technology Veritas University, Abuja, Nigeria

Abstract:- Due to the rapid advancement of electronic commerce technologies, the use of credit cards has increased significantly. Given that credit cards are the most common form of payment, the incidence of credit card fraud has also risen. With the rise of online money transfers in the cashless economy and the migration of businesses to the internet, fraud detection has become a crucial aspect of transaction security. With the advent of technological advancement and the emergence of new e-service payment options, such as e-commerce and mobile payments, credit card transactions have become more common. To prevent credit card fraud, a robust and reliable fraud detection system is necessary. Several approaches, including predictive approaches and algorithms, have been proposed to detect credit card fraud. These algorithms establish a set of logically sound principles that permit the classification of data as either normal or dubious. However, credit card fraud has persisted despite the adoption of more sophisticated techniques. This study presents an approach for detecting credit card fraud using random forests. The dataset and the user's current dataset are analysed using the random forest technique. Before analysing a subset of given data to detect fraud, the method increases the precision of the outcome data. In addition, a comprehensive comparison and analysis of current and future fraud detection measures is presented. The dataset is applied Random Forest-based classification models, and the model's performance is evaluated using graphical representations of precision and classification accuracy.

Keywords:- *Hidden Markov Model, Artificial Neural Networks, Random Forest, Machine Learning, Data Visualization.*

I. INTRODUCTION

Credit and debit card fraud is one of the main sources of financial losses in the finance sector. In recent years, as data volume has increased, so has the number of payment card transactions, leading to the automation of credit card fraud detection [1]. Credit card fraud refers to the unauthorized use of another person's credit card to make purchases or obtain monetary advances without their consent. One of the most prevalent forms of identity theft is credit card fraud [2]. With the emergence of online money transfers in the cashless economy and the migration of businesses to the internet, fraud detection has become an essential component of transaction security. Internet-based fraudulent transactions are the most straightforward and straightforward to execute. The expansion of transnational, economic, and political spaces has resulted in the emergence of the Internet as a new global marketplace, drawing together sellers and buyers from all regions and countries [6]. This widespread approval of cashless transactions encourages fraudsters to conduct frequent fraudulent attacks and modify their methods to avoid detection [1]. Detection of credit card fraud in the payment sector attempts to determine whether a transaction is fraudulent based on historical data [3]. Application and Behavior fraud are two card fraud types [2]. Application fraud occurs when criminals register for a credit card using stolen or forged identification. Even though this is detectable through background checks, it allows criminals to use a legitimate credit card with a fake written record. As a result, criminals may be able to use a legitimate credit card with a false written history, despite the fact that background checks can detect this situation. A similar type of fraud entails obtaining a Credit Card account by assuming the identity of the customer and forging a similar counterfeit written account. Numerous studies have been conducted on the identification and prevention of credit card fraud [4]. Additionally, numerous technologies have been utilized to detect counterfeit credit cards. Credit card segmentation and fraud protection continue to be exceedingly challenging to implement. Credit card fraud detection and prevention is a multi-step process that may include Decision Tree, Genetic Algorithms, Clustering Techniques, Neural Networks, Naive Bayes Classifiers, K-Nearest Neighbor Algorithms, Support Vector Machines (SVMs), and Bagging Ensemble Classifier [5]. Link analysis, Bayesian networks, decision theory, and sequence matching are some additional methods for detecting fraud [6]. Nevertheless, fraud detection and prevention remain essentially unsolved. The primary objective of this study is to distinguish between legitimate and fraudulent Credit Card transactions using supervised machine learning for classification and regression and random forest machine learning. This enables sequential neural-based networks to autonomously prioritize the most important data items throughout the segmentation procedure, resulting in improved acquisition performance. The rationale for employing the Random Forest method is that its output is dependent on a large number of decision trees, making it impartial and resulting in more reliable findings [3]. It is applicable to classification and regression problems and is considered a robust method because minor modifications to the dataset have no effect on the output [2]. The Random Forest method addresses the issue of

ISSN No:-2456-2165

intransigence in data streams through the generation of a new model when new data becomes available [6].

II. REVIEW OF LITERATURE

Several strategies have been developed for the detection of credit card fraud in electronic finance and banking. Some of the approaches closely related to Random Forestare Bayesian Networks, Hidden Markov Model (HMM), Artificial Neural Networks and K-Nearest Neighbor.Random forest is an algorithm based on Machine Language (ML) that is derived from a decision tree (DT) method and is typically applied to a wide range of regression and classification problems. [7]. It helps precisely predict the output of massive datasets. Random Forest employs multiple classifiers to solve a variety of complex problems. For estimating the average output of other trees, the random forest is valuable. The conclusion tends to become more precise as the number of trees increases [9]. The random forest decision tree method [8] is a supervised machine learning technique used to solve classification and regression problems. Random forest is based on the premise that a group of "weaker learners" can be combined to form a "stronger learner." [7] As random forests provide multiple decision trees; a single decision tree is a "weaker learner" compared to a collection of decision trees [10]. In order to classify a novel object, each forest tree is examined [11]. Each classification tree generates a classification output. The forest places the new item in the class that maximizes output [18]. Random forests are effective and can manage large, unbalanced datasets with multiple attributes [7]. It has been demonstrated that it provides a precise estimate of the generalization error and resists overfitting [9]. Random forest eliminates a number of the disadvantages of the decision tree algorithm [1]. It also decreases the elevation of datasets, thereby increasing precision [11]. However, random forest has drawbacks when training multiple datasets, specifically regression issues [12]. Many authors have delved into method of detecting credit card fraud using Bayesian Networks Algorithm. In this approach, each variable in a given domain is represented as a graph node using this method. To detect fraud, two Bayesian networks describing user behavior are constructed [11]. First, a Bayesian network is built to anticipate user behavior under the assumption that the user is fraudulent. (F). Then, a second model is constructed (L) assuming the user is authentic. (non-fraudulent, NF). A fraud net requires specialized knowledge to construct. The data from legitimate users is used to construct a "user net." Depending on current data, the user net is tailored to a specific user throughout operation [12]. By injecting evidence into these networks and distributing it throughout the network, it is possible to reduce the probability that the measurement is inaccurate [11]. This value indicates the degree to which observed user behavior should comply with F or NF standards. It enables the incorporation of expert data used for model configuration's inception. In contrast, the user model is retrained without supervision using data. Most researchers agree that the Bayesian approach incorporates both expert knowledge and learning [13]. Another contemporary approach that has been adopted in credit card fraud detection has been the Hidden Markov Model (HMM). The Hidden Markov Model is a random process with two phases, one of which is hidden and the other of which is accessible to everyone [10]; [14]. This is possibly the simplest model available for modelling sequential data. Unlike the Markov model, in which the state is plainly visible to the observer, the state is not immediately apparent in HMM. In HMM, however, the state-dependent output is evident. The HMM is a restricted set of states to which a probability distribution [15] is assigned. The probabilities governing state transitions are referred to as transition probabilities. In accordance with the corresponding probability distribution, a given condition can produce a result or observation [10]. To an external observer, only the outcome is visible; consequently, states are "hidden" externally [14]. HMM reduces significantly the number of legitimate transactions that are incorrectly identified as fraudulent by a fraud detection system [10]. Other researchers have explored the use of Artificial Neural Networks (ANN) in solving the problem of credit card fraud detection. Neural networks have the capacity to address complex problems that demand a significant level of precision, such as the identification of patterns within vast datasets.An artificial neural network that has been trained using simulated annealing has been affirmed as capable of significantly identifying fraudulent credit card transactions by determining the optimal configuration weight for a neural network [7]. The stimulation annealing method optimizes performance [16] as it is a model with regularly interconnected neuronal layers or structures [1]. ANN is constructed by layering nodes and connecting them with weighted interconnections that can be modified. According to [17], a neural network is a collection of "processing nodes" that exchange information via connections. A node receives input from interconnected nodes and calculates output values using the weights of the connected nodes and a straightforward function [1]. ANN is frequently based on supervised and unsupervised approaches [18]. Unsupervised Neural Networks are frequently used to detect fraud [16]. The unsupervised neural network attempts to recognize patterns between current credit cardholders and past transactions. ANN approaches are extremely error tolerant and are regarded as an effective solution for the CCFD [1] due to their high processing speed and efficiency. Despite the fact that ANN and clustering are effective at detecting fraudulent transactions, ANN's structure, which requires progressive trial and error, is hardly accounted for [16]. For credit card fraud detection, an artificial neural network model [10] and back propagation [10] have been proposed. The procedure continues by retrieving the customer dataset, which contains the customer's name, transaction identifier, and time. In terms of detecting fraudulent transactions in real-time data, the proposed method has yielded significant, albeit limited, results [10]. Compared to the Artificial Neural Networks. Though ANNs are known to perform well on high-dimensional datasets and can learn complex nonlinear relationships between inputs and outputs, however they require careful tuning of hyperparameters and a large amount of data to prevent overfitting when compared to the K-Nearest Neighbor (KNN). KNN can perform well on small datasets with few input features. However, its performance can degrade when dealing with high-

dimensional data or when the class distributions are imbalanced. This method maintains all extant cases and classifies any new instances based on a measure of similarity. Several authors [9, 15] have referred to K-Nearest Neighbor (KNN) as an instance-based learner. In this method, each new instance is compared to existing instances using a distance metric, and the class of the new instance is determined using the nearest existing instance, also known as the nearest neighbor[12]. In some instances, the majority class of the closest K-neighbors is allocated to the new instance despite the deployment of multiple of the closest existing instances [15]. According to [4], the KNN method provides consistently excellent performance across a variety of credit card fraud detection algorithms with no prior assumptions regarding the distributions from which the training instances are generated. KNN is a supervised machine learning technique beneficial for classifying problems and performing regression analysis [18]. KNN utilizes a supervised method to determine the existence of fraudulent credit card transactions [24] and includes two estimates: transaction correlation and the distance between occurrences of transaction in the data. KNN is suitable for identifying fraudulent activities at the time of the transaction [18]. Due to the absence of false-positives during classification, KNN has been shown to be effective with respect to all employed metrics. Using KNN in a second investigation, the accuracy of CCFD was determined to be 72%. [19]. Although the authors utilized KNN to conduct progressive testing, its limitations must be acknowledged. The KNN algorithm is a memory-intensive algorithm that emphasizes irrelevant data properties. As a large quantity of data is fed to the KNN algorithm, its performance declines. Consequently, these limits affect the accuracy and recall matrix of the CCFD procedure.

III. METHOD

The present study employs the random forest classifier technique to construct machine learning models. The classifier is a machine learning methodology that employs training data to assign a trained classifier to a given data point. This supervised learning ensemble method employs a set of empirical data that is amenable to observation for the purposes of training. The values of the target variable can manifest as either a categorical variable with two possible outcomes, namely fraud or non-fraud, or as a binary variable represented by the integers 0 and 1. The dataset's 'Class' column comprises solely of two distinct values, namely 0, which denotes valid transactions, and 1, which represents illegal transactions. The present study employs the Random Forest Algorithm. This method demonstrates efficacy in analyzing datasets of varying sizes, from small to large. The random forest algorithm is an expansion of the bagging methodology, as it integrates bagging with randomization in order to create a collection of decision trees that are not correlated with one another. Decision trees have the potential to address the issue of overflow in the training set. The forecast will be computed based on the level of complexity. When performing regression, the mean of individual decision trees is computed, whereas in classification, the most common categorical variable is determined through a majority vote to predict the class. The process of cross-validation is utilized on the given sample in order to arrive at a conclusive prediction. The illustration highlights the distinction between the two, enabling a multitude of parameters to concomitantly contribute to prognostication. The efficacy of the technique surpasses that of decision trees due to its underlying principle of combining independent trees. To enhance the robustness of a random forest, it is advisable to generate a considerable number of sample trees and subsequently compute their average. The Random Forest classifier presents several benefits: The reduction of overfitting is known to improve the effectiveness of decision trees. The remarkable efficiency of this system is evident when processing large datasets. The software is capable of handling data that is both categorical and continuous. The utilization of substitution techniques is facilitated by the presence of missing data values. The implementation of a rule-based approach is deemed appropriate as there exists no necessity for data uniformity.

A. Naïve Bayes Model and Algorithm Design Approach

The Naïve Bayes is a classification algorithm that is based on Bayes' theorem, which states that the probability of a hypothesis (in this case, a class label) given evidence (in this case, feature values) is proportional to the probability of the evidence given the hypothesis, multiplied by the prior probability of the hypothesis[19]. The approach adopted in applying the Naïve Bayesalgorithm in this work assumes that the features are conditionally independent of each other given the class label, which means that the presence or absence of one feature does not affect the likelihood of another feature being present or absent. This assumption simplifies the probability calculation and makes the algorithm computationally efficient. The Naïve Bayes is represented as in Eq. 1.

$$P\left(\frac{C}{X}\right) = P(\frac{X}{C})/P(X) \quad \text{Eq. 1}$$

Where;

$$X = \{X_1 \dots X_J\} \quad \text{Eq. 2}$$

The efficacy of the naïve Bayes algorithm is optimized when the features exhibit independence or weak correlation, and when there exists an adequate number of training samples for each class label [20]. The algorithm is capable of accommodating both categorical and continuous features. However, it necessitates certain adjustments to address the management of missing values and skewed distributions. The methodology involves formatting the data in a manner conducive to allow classification, wherein every sample is denoted by a collection of features and an associated class label.The prior probability of each class label is calculated as the proportion of samples in the training data that belong to that class.The likelihood probability of each feature given the class label is calculated as the proportion of samples in the training data that belong to that class and have the feature.The posterior probability of each class label given the evidence (i.e., the feature values) is calculated using Bayes' theorem, with the prior and likelihood probabilities as inputs. To make a prediction for a new sample, the posterior probabilities are calculated for each class label,

and the class label with the highest probability is assigned to the sample.

B. Smoothing and Ensemble Approach

The research adopts the Laplace smoothing technique for smoothing. The Laplace smoothing technique is a technique used in probabilistic models to avoid zero probabilities when estimating probabilities from a limited sample [21]. It adds a small constant value to each count to ensure that every possible outcome has a non-zero probability estimate. The ensemble approach is combined with the Laplace smoothing technique in this work.Ensemble learning is an approach to machine learning where multiple models are combined to improve the overall performance of the system. The idea is that by combining the predictions of multiple models, the errors and biases of each model can be mitigated, resulting in better accuracy and robustness.In the context of Naïve Bayes, one way to apply ensemble learning is to use multiple Naïve Bayes models with different subsets of features or different smoothing parameters. The output of each model is then combined using a voting or averaging scheme to make the final prediction. This approach is called ensemble Naïve Bayes or Bayesian averaging.

C. Dataset Sampling, Balancing and Classification

In machine learning, it is essential to have a good training dataset that is representative of the population you want to make predictions for. However, often datasets may suffer from class imbalance, where the number of instances in one class is much smaller than the other. This can lead to poor performance of the machine learning models as they tend to favor the majority class.One approach to overcome class imbalance is to balance the dataset using sampling techniques. There are several sampling techniques, including oversampling, under sampling, and a combination of both.In this approach, both oversampling and undersampling are used to balance the dataset. The combination technique involves oversampling the minority class and undersampling the majority class until both classes have the same number of instances.Once the dataset is balanced, the dataset was split for training, validation, and test sets. The training set is used to fit the model, the validation set is used to tune hyperparameters, and the test set is used to evaluate the model's performance. The dataset used for this work is obtained from Kaggle [23].

D. The Mathematical Theory of Naïve Bayesfor Fraud Detection

A total of 30 characteristics from the dataset are used to determine whether or not a class is fraudulent. Several libraries were created for Nave Bayes classification and the mathematical model used were factored as Transact SQL. Time, V1–V28, and Amount are used to determine whether or not fraud has occurred. The first logical approach was to determine the total number of rows in the dataset using the following models;

Count (Total number of rows)

The second stage, which is the second logical step, was to calculate the likelihood of fraud and non-fraud occurring in the dataset;

$$Probability(NotFraud) = \frac{count(NotFraud)}{count(Total\ Number\ of\ Rows)} \quad \text{Eq.3}$$

$$Probability(Fraud) = \frac{count(Fraud)}{count(Total\ Number\ of\ Rows)} \quad \text{Eq. 4}$$

The third phase was to determine the likelihood of the attributes occurring for Fraud and Not Fraud. For instance, if we only use the Time attribute, the likelihood of Fraud occurrence is:

$$Probability(Time \mid NotFraud) = \frac{count(NotFraud\ and\ Time)}{count(NotFraud)} \quad \text{Eq. 5}$$

$$Probability(Time \mid Fraud) = \frac{count(Fraud\ and\ Time)}{count(Fraud)} \quad \text{Eq. 6}$$

Same is carried out for all the given attributes of the dataset.

Multiplying each attribute's conditional probability by its class is the next step. The classes include Fraud and Not Fraud. Using the attributes Amount, V1, V2, V3, and Amount as an example, the calculation would be as follows:

$$P\left(\frac{X}{Fraud}\right) = p\left(\frac{Amount}{Fraud}\right) * P\left(\frac{V1}{Fraud}\right) * P\left(\frac{V2}{Fraud}\right) * P\left(\frac{V3}{Fraud}\right) * P\left(\frac{Amount}{Fraud}\right) * P(Fraud) \quad \text{Eq. 7}$$

$$P\left(\frac{X}{NotFraud}\right) = p\left(\frac{Amount}{NotFraud}\right) * P\left(\frac{V1}{NotFraud}\right) * P\left(\frac{V2}{NotFraud}\right) * P\left(\frac{V3}{NotFraud}\right) * P\left(\frac{Amount}{NotFraud}\right) * P(NotFraud) \quad Eq.8$$

And the final step was to get the probability of the Total Fraud and the Total Not Fraud which is

For Fraud: P (x |Fraud)/P (x| NotFraud) + P (x |Fraud)
For Not Fraud: P (x |NotFraud)/P (x| Fraud) + P (x |NotFraud)
Prediction: Max (Fraud, NotFraud)

ISSN No:-2456-2165

IV. DATA VISUALIZATION, IMPLEMENTATION AND RESULTS

The dataset for this study was obtained from Kaggle [23] and generated using the GitHub utility Sparkov Data Generation. The dataset consists of simulated credit card transactions, both legitimate and fraudulent. It encompasses the credit cards of 2,000 customers who conduct business with 142 merchants. This dataset contains a total of 2,84,807 transactions, and 2,100 fraudulent transactions were identified out of the total number of transactions. The dataset is extremely unbalanced, with the positive class (frauds) comprising only 0.5727 percent of all transactions. The dataset contains 22 distinct data types, such as "Amount," "Category," and "is fraud" among others. It also contains numerical and categorical characteristics. The "trans_date_transtime" column contains the date and time associated with each recorded transaction. The 'Amount' feature column includes the amount of the transaction, while the "is Fraud" feature is the response variable that indicates whether a transaction is fraudulent or not. If it is fraud, the value is 1, and otherwise it is 0. This dataset can be accessed at [23]. Figure 1 depicts the dataset dispersion between fraudulent applied in Eq. 4 and legitimate transactions applied in Eq. 3 for the amount attribute. The output reveals that the minimal and maximum values of the amount feature for non-fraudulent distribution are 1.00 and 28948.9, respectively, whereas they are 1.18 and 1371.81 for fraudulent distribution. Moreover, we can see from the output.

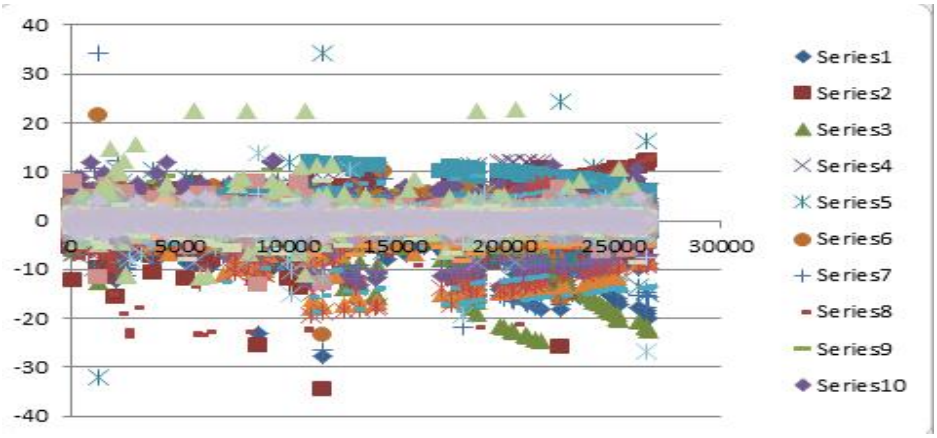

Fig. 1: Dataset Series Scatter Plot

The dataset consists of two-day card transactions performed by European cardholders in September 2018. Due to the fact that some of the input variables contain sensitive financial information, the PCA transformation of these input variables was performed in order to safeguard their privacy. Three of the specified attributes remained unchanged. The "Time" function depicts the amount of time between the first and subsequent transactions in the dataset. The "Amount" function displays the sum of all credit card transactions. Class represents a label and only admits two possible values: 1 for fraudulent transactions and 0 otherwise. Sampling: The data set is then reduced to 560 transactions, 228 of which are fraudulent and 332 are valid. Separated from the dataset are training and test data sets. Sixty percent of the data set is trained, while forty percent is tested. In this case, supervised machine learning techniques are employed. The algorithms Naive Bayes, Logistic Regression, and Random Forest use boosting. Ignorant Bayes: The Bayes theorem computes the probability of an event based on the probability of an event that has already occurred. The Nave Bayes algorithm is simple and quick. This algorithm requires fewer training data and is extremely scalable. P (A/B) = (P (B/A) P (A)) / P (B) Where P (A) is the priority of A, P (B) is the priority of B, and P (A/B) is the posterior priority of B. This method is comparable to the linear regression procedure. In contrast, linear regression is employed to predict or forecast values, whereas logistic regression is used to classify data. This method simplifies both binary and multivariable classification assignments. Binary has only two possible varieties. (0 or 1). Multinomial includes at least three potential categories that are not ordered, whereas Ordinal is ordered. The Random Forest technique begins with the selection of random samples from a given dataset. The algorithm then constructs a decision tree for each sample in order to derive the prediction result from each decision tree. Then, for each predicted result, a Parse is initiated to determine the most iterated path of the prediction parse as the final prediction result. The 'moment' feature does not indicate the actual moment of a transaction, but rather lists the information in chronological order. On the basis of the preceding data visualization, we conclude that the 'Time' feature has little or no significance in correctly classifying a fraudulent transaction; therefore, we will not analyze this column further. In this study, we will eliminate the Time variable before converting the Class variable to a factor.

A. Dataset Sampling and Classification

The Credit Card Analysis displays the conditional likelihood of fraud based on the Kaggle dataset[23]. Figure2 depicts the Classification of Credit Card Datasets by Type. It demonstrates that fraud Class has just 0.172 percent fraud transactions, making it imbalanced, and if we use the Nave Bayes model to the data set, this will result in a biased forecast. The result of time classification chart using the cluster sampling approach defined in (Eq. 5) for not fraud and (Eq. 6) for not fraud is shown in Figure 3.Cluster samplingtechnique is used because of the large size of the

ISSN No:-2456-2165

dataset to divided it into clusters, and a sample is taken from each cluster. This technique is useful when the dataset is too large to be processed in one go and is spatially distributed[6].

The Credit Card Input Test Page enables the researcher to enter input data for testing of the Naïve Bayes algorithm to see if the prediction is fraud or not fraud using the Mathematical Theory of Naïve Bayes defined in (Eq. 7) and (Eq. 8).

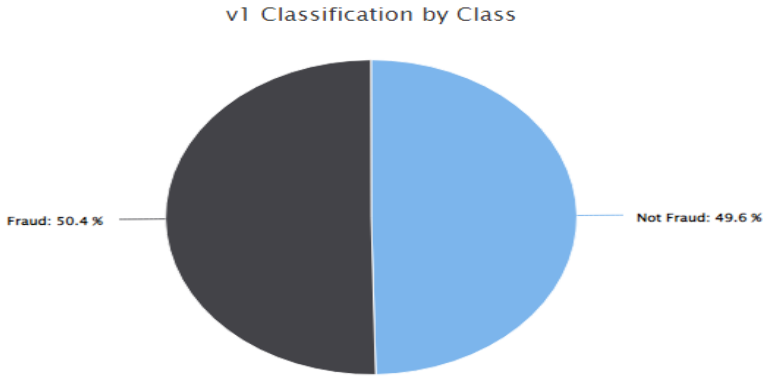

Fig. 2: V1 Classification by Type

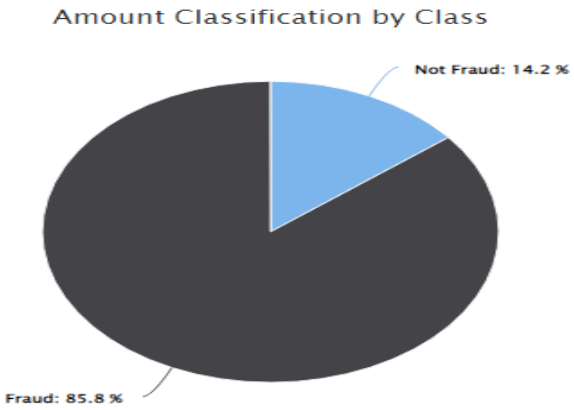

Fig. 3: Amount Classification by Type

B. Dataset Distribution

Figure 3 demonstrates the extreme imbalance of the dataset used for this study, with 99.8% of the cases being non-fraudulent transactions. A straightforward metric like accuracy is inappropriate in this situation because even a classifier that classifies all transactions as legitimate will be over 99 percent accurate. Area Under the Curve would be a suitable indicator of model performance in this case, Area under the Precision-Recall Curve.The "Time" feature exhibits a comparable appearance for both transaction types, as depicted in Figure 4. It can be argued that the distribution of fraudulent transactions is more consistent compared to legitimate transactions, which exhibit a cyclic pattern.

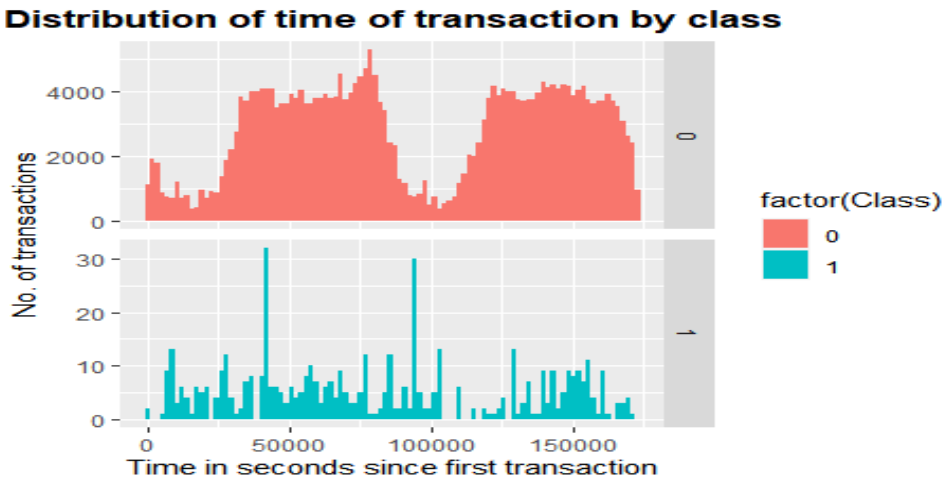

Fig. 4: Distribution of time of Transaction by Class

ISSN No:-2456-2165

In Figure 4, there is clearly a lot more variability in the transaction values for non-fraudulent transactions. The Distribution of time of Transaction by Class indicates that the majority of data attributes are not related. Prior to publication, a Principal Component Analysis (PCA) method was provided with the vast majority of the data. The features V1 through V28 are the Principal Components that were most likely generated after propagating the actual features using PCA. We are uncertain if the numbering of the features reflects the importance of the Principal Components.

V. DISCUSSION

To enhance our understanding of the data, the t-Distributed Stochastic Neighbor embedding technique is utilized for dimensionality reduction, in order to generate visual representations. A graphical representation is a visual tool used to present data in a way that enables the identification of patterns and trends. It allows for a quick and intuitive understanding of the underlying data by leveraging visual elements such as points, lines, bars, or shapes.When a graphical representation demonstrates the presence of identifiable patterns within the data, it means that the visual representation reveals recurring structures, relationships, or behaviors that can be observed and interpreted. The patterns include trends, clusters, outliers, correlations, or other significant characteristics of the dataset.

```
Creditcard = creditcard[,-1]
creditcard$Class<-as.factor(creditcard$Class)
levels(creditcard$Class) =c("Not_Fraud", "Fraud")
creditcard[,-30] <- scale(creditcard[,-30])
head(df)
Algorithm for using the under-sampling technique
tables(train$Class)
Not_Fraud     Fraud
  199020       344
set.seed(9560)
down_train<-downSample(x = train[, -ncol(train)],
                       y =train$Class)
table(down_train$Class)
Not_Fraud     Fraud
    344       344
```

Fig. 5: Algorithm for identifiable patterns within the dataset assimilated

The graphical representation demonstrates the presence of identifiable patterns within the data assimilated by the algorithm in Figure 5. A perplexity value of 20 was designated during the training of the model. In cases where the data exhibits an absence of a clear structure, it is probable that the model will exhibit suboptimal performance. Figure 4 displays a noticeable division between the two categories, with a significant concentration of deceitful transactions located in close proximity to the periphery of the data cluster. The challenges of learning from unbalanced data are encountered by standard Machine Learning (ML) algorithms due to various factors. ML algorithms encounter challenges in acquiring knowledge from dependent variables that exhibit uneven distributions. As a result, the classifiers currently in use exhibit a bias towards the dominant class, leading to a performance that is skewed. The algorithms prioritize accuracy and aim to minimize overall error, with minimal contribution from the minority class. In order for machine learning algorithms to operate effectively, it is imperative that the class distributions within the dataset are balanced. The term commonly used to describe these solutions is "Sampling Methods." Commonly, these methodologies utilize a mechanism for converting an imbalanced data distribution into a balanced one. The process of modification involves the manipulation of the size of the initial dataset while preserving its equilibrium. Several research studies have indicated that classification accuracy is improved by utilizing balanced data sets as opposed to unbalanced ones. Consequently, these methodologies have gained significant significance. Therefore, it is imperative to comprehend them.

Prior to implementing sampling techniques, it is advisable to examine the efficacy of Naive Bayes in the context of imbalanced data. The receiver operating characteristic (ROC) curve function from the Random Over-Sampling Examplestool is utilized to assess the model's performance on the test set. The effectiveness of the model on the test dataset is assessed through the computation of the Area Under the Curve(AUC) score, as illustrated in Figure 4. The area under the receiver operating characteristic curve AUC for the initial dataset was observed to be 0.912. The under-sampling technique will be implemented on the data, followed by an assessment of the test set's performance. The present study utilized under-sampling techniques to address the issue of imbalanced datasets, exemplified by the fraud credit card transaction dataset, wherein the frequency of fraudulent cases is considerably lower than that of normal transactions. The inadequacy of accuracy as a performance

ISSN No:-2456-2165

measure for models has been discussed and the effectiveness of under sampling the response variable in enhancing model training has been assessed through the utilization of the area under the ROC curve metric. The findings suggest that the utilization of under-sampling technique yielded favorable outcomes on the dataset, resulting in a noteworthy enhancement of the model's performance in comparison to the imbalanced data. The maximum attainable score was 0.942. However, this study highlights the significance of sampling, modelling, and predicting data when dealing with an imbalanced dataset.

VI. CONCLUSION

The Random Forest machine learning approach was utilized to detect credit card fraud through a predictive methodology. The algorithms establish a set of logically consistent principles that enable the categorization of data as either typical or dubious. The present study proposes a methodology for detecting credit card usage through the utilization of random forests. The random forest technique is employed to analysis both the dataset and the user's current dataset. Prior to conducting an analysis on a specific subset of provided attributes with the aim of detecting fraudulent activities, this approach enhances the accuracy of the resulting data. Furthermore, a thorough evaluation and examination of existing and prospective fraud detection methodologies has been undertaken. Consequently, the Random Forest technique is utilized to construct classification models for the given data, and the efficacy of the model is assessed by means of precision and classification accuracy graphical illustrations. The dataset procured from Kaggle [23] exhibited a significant degree of imbalance. The resolution of this matter was achieved through the implementation of training and resampling procedures utilizing both under sampling and over sampling methodologies on the dataset. The Random Forest algorithm's classification and regression techniques were employed to identify instances of credit card fraud through machine learning. The classification of fraudulent card transactions was performed offline using a Random Forest supervised learning algorithm. This study successfully predicted the occurrence of fraudulent credit card transactions by utilizing transaction-related data such as location and transaction amount. In contrast to Random Forest and rule-based approaches, the aforementioned technique exhibits superior accuracy and relevance in its responses. This is attributed to its ability to incorporate multiple parameters, thereby facilitating a more comprehensive analysis of a larger volume of data points, including intricate patterns of account activity. The extension of this work to incorporate racing with incremental data, whereby the data fed into the race is derived from novel segments of the stream, would be a logical progression.

DECLARATIONS

- **Manuscript title: A Synergistic approach for Enhancing Credit Card Fraud Detection using Random Forest and Naïve Bayes Models**
- **Declaration of Originality:** I declare that the article titled “A Synergistic approach for Enhancing Credit Card Fraud Detection using Random Forest and Naïve Bayes Models” is an original work, and all information, data, figures, and content presented in the article are the result of our own research and analysis. Any references to the work of others have been appropriately cited and acknowledged.
- **Conflict of Interest:** I declare that there is no conflict of interest that could influence the objectivity or integrity of the information presented in the article. In the case of any potential conflicts, they have been disclosed in the acknowledgments or funding sections of the article.
- **Data and Source Attribution:** I declare that any data, figures, tables, or other information presented in the article from external sources have been properly attributed. The sources of such data or information are accurately cited, and permissions for the use of copyrighted material have been obtained, where applicable.
- **Ethical Considerations:** I declare that the research presented in the article has been conducted in accordance with ethical guidelines and regulations. If the research involved human participants, proper informed consent was obtained, and all necessary ethical approvals were obtained from relevant institutional review boards or ethics committees.
- **Acknowledgment of Contributions:** I/we acknowledge the contributions of individuals or organizations who have assisted in the research, writing, or editing of the article. Their names, affiliations, and specific contributions, if any, are duly recognized in the acknowledgments section of the article.
- **Funding and Support:** I declare that no financial support, grants, or sponsorships have been received for the research or publication of the article.
- **Compliance with Publication Guidelines:** I affirm that the article complies with the guidelines, formatting, and submission requirements of the target journal or publication. All necessary permissions, licenses, or releases for previously published material or copyrighted content have been obtained, and all necessary disclosures have been made.

By proving and affirming this information content declaration, I confirm that the above statements are true and accurate to the best of my knowledge.

ISSN No:-2456-2165

REFERENCES

[1.] Asha, R.B. and KR, S.K., 2021. Credit card fraud detection using artificial neural network. Global Transitions Proceedings, 2(1), pp.35-41.

[2.] Delamaire, L., Abdou, H. and Pointon, J., 2009. Credit card fraud and detection techniques: a review. Banks and Bank systems, 4(2), pp.57-68.

[3.] Paasch, C.A., 2008. Credit card fraud detection using artificial neural networks tuned by genetic algorithms. Hong Kong University of Science and Technology (Hong Kong).

[4.] Bhattacharyya, S., Jha, S., Tharakunnel, K. and Westland, J.C., 2011. Data mining for credit card fraud: A comparative study. Decision support systems, 50(3), pp.602-613.

[5.] Xuan, S., Liu, G. and Li, Z., 2018. Refined weighted random forest and its application to credit card fraud detection. In Computational Data and Social Networks: 7th International Conference, CSoNet 2018, Shanghai, China, December 18–20, 2018, Proceedings 7 (pp. 343-355). Springer International Publishing.

[6.] Xuan, S., Liu, G., Li, Z., Zheng, L., Wang, S. and Jiang, C., 2018, March. Random forest for credit card fraud detection. In 2018 IEEE 15th international conference on networking, sensing and control (ICNSC) (pp. 1-6). IEEE.

[7.] Bhatla, T.P., Prabhu, V. and Dua, A., 2003. Understanding credit card frauds. Cards business review, 1(6), pp.1-15.

[8.] Tiwari, P., Mehta, S., Sakhuja, N., Kumar, J. and Singh, A.K., 2021. Credit card fraud detection using machine learning: a study. arXiv preprint arXiv:2108.10005.

[9.] Rai, A.K. and Dwivedi, R.K., 2020, July. Fraud detection in credit card data using unsupervised machine learning based scheme. In 2020 international conference on electronics and sustainable communication systems (ICESC) (pp. 421-426). IEEE.

[10.] Tripathi, K.K. and Pavaskar, M.A., 2012. Survey on credit card fraud detection methods. International Journal of Emerging Technology and Advanced Engineering, 2(11), pp.721-726.

[11.] Jain, Y., Tiwari, N., Dubey, S. and Jain, S., 2019. A comparative analysis of various credit card fraud detection techniques. Int J Recent Technol Eng, 7(5S2), pp.402-407.

[12.] Sethi, N. and Gera, A., 2014. A revived survey of various credit card fraud detection techniques. International Journal of Computer Science and Mobile Computing, 3(4), pp.780-791.

[13.] Sobanadevi, V. and Ravi, G., 2020. Multi-Level Credit Card Fraud Detection.

[14.] Mor, B., Garhwal, S. and Kumar, A., 2021. A systematic review of hidden Markov models and their applications. Archives of computational methods in engineering, 28, pp.1429-1448.

[15.] Georgieva, S., Markova, M. and Pavlov, V., 2019, October. Using neural network for credit card fraud detection. In AIP Conference Proceedings (Vol. 2159, No. 1, p. 030013). AIP Publishing LLC.

[16.] Mathur, N. and Jain, M., COMPARISONS OF MACHINE LEARNING ALGORITHMS FOR FRAUDULENT ANALYSIS IN FINANCIAL SECTOR.

[17.] Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A. and Aljaaf, A.J., 2020. A systematic review on supervised and unsupervised machine learning algorithms for data science. Supervised and unsupervised learning for data science, pp.3-21.

[18.] Niu, X., Wang, L. and Yang, X., 2019. A comparison study of credit card fraud detection: Supervised versus unsupervised. arXiv preprint arXiv:1904.10604.

[19.] Sadineni, P.K., 2020, October. Detection of fraudulent transactions in credit card using machine learning algorithms. In 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 659-660). IEEE.

[20.] Poojitha, S. and Malathi, K., 2022, October. An Original Approach to Identify the Better Accuracy in Credit Card Fraud Transaction by Comparing Logistic Regression with K-Nearest Neighbours Algorithm. In 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS) (pp. 6-11). IEEE.

[21.] Aung, M.H., Seluka, P.T., Fuata, J.T.R., Tikoisuva, M.J., Cabealawa, M.S. and Nand, R., 2020, December. Random Forest Classifier for Detecting Credit Card Fraud based on Performance Metrics. In 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE) (pp. 1-6). IEEE.

[22.] Jones, Z. and Linder, F., 2015, April. Exploratory data analysis using random forests. In Prepared for the 73rd annual MPSA conference.

[23.] https://www.kaggle.com/. Available at available at https://www.kaggle.com/kartik2112/fraud-detection. Accessed March 2023.