Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING ALGORITHMS

Ms.JAMPANISRIHARSHINI[1], Ms.GHANTASRAVYA SREE[2], Ms.MOHAMMED VAHIDHA[3], Ms.YENIGALLA JAHNAVI[4], Mrs.CHODISETTI DEEPIKA[5]

1 .BTech, Vijaya Institue of Technology For Women, Enikepadu, Vijayawada, Andhra Pradesh, India.
Email : jampanisriharshini2000@gmail.com
2. BTech, Vijaya Institue of Technology For Women, Enikepadu, Vijayawada, Andhra Pradesh, India.
3. BTech, Vijaya Institue of Technology For Women, Enikepadu, Vijayawada, Andhra Pradesh, India.
4. BTech, Vijaya Institue of Technology For Women, Enikepadu, Vijayawada, Andhra Pradesh, India.
5. Assistant Professor, Computer Science and Engineering, Vijaya Institue of Technology For Women, Enikepadu, Vijayawada, Andhra Pradesh, India.Email : deepika3062@gmail.com

ABSTRACT

During the most recent lockdown brought on by the Covid epidemic, online purchases suddenly increased. They used online purchasing platforms extensively to complete deals and transfer money because the market, food stores, and banks were closed. The use of credit cards was crucial to them. The use of credit cards for payment has grown along with the sharp increase in online purchases. This indicates that there is a higher likelihood of fraudulent transactions, which ultimately results in significant financial losses. As a result, banks and other financial organizations fund the development of software that identify credit card fraud.

The "CREDIT CARD FRAUD DETECTION" project finds the counterfeit card while conducting transactions and notifies the customer of the scam. Reduced false alerts is another goal of this initiative. The notion is a unique genetic algorithm in this application domain. Here, we use ensemble classifiers with bagging and boosting approaches, logistic regression, random forest, and other machine learning algorithms on an unbalanced dateset.

1 INTRODUCTION

People's main concern with data mining in recent years has been the model used to detect credit card fraud. The traditional data mining algorithms are not immediately applicable to our topic because it is handled as a classification problem. In order to create a different strategy, general-purpose meta heuristic methods like genetic algorithms are used. The goal of this study is to suggest a genetic algorithm-based method for detecting credit card fraud. Evolutionary algorithms like genetic algorithms strive to produce better answers over time. A card is typically utilized up to its available maximum when it is cloned, stolen, lost, and discovered by fraudsters. Hence, a solution that reduces the overall allowable limit on cards prone to fraud is more important than the quantity of correctly identified transactions. It uses a genetic algorithm to minimize false alarms by optimism a collection of interval-valued parameters. The technique starts with many populations of chromosomes that are created at random. These chromosomes go through the processes of crossover, mutation, and selection. Using the finest of the present generation, crossover combines the information from two parent chromosomes to create new organisms, whereas mutation, or randomly altering some of the parameters, enables exploration into additional regions of the solution space. Only the most compatible chromosomes will survive in the population to marry and give birth to the next generation through natural selection using a problem-specific cost function. The genetic algorithm eventually reaches a complete solution after several iterations. Evolutionary algorithms like genetic algorithms strive to produce better answers over time. Since their first introduction by Holland, they have been successfully used in a wide range of problem areas, including astronomy, athletics, computer science, and optimisation. They are frequently used with other data mining algorithms and have also been used in data mining, mostly for variable selection. In this study, we solely use a genetic algorithm to try to solve our categorization problem.

2. RELEATED WORK

The goal of fraud detection has typically been viewed as a data mining task, where the classification of transactions as legal or fraudulent must be done correctly. There are numerous performance metrics for classification issues, the most of them are connected to the proper number of instances categorised correctly. Because of the intrinsic structure of credit card transactions, a more suitable approach is required. A card is typically utilised up to its available maximum when it is cloned, stolen, lost, and discovered by

fraudsters. Hence, a solution that reduces the overall allowable limit on cards prone to fraud is more important than the quantity of correctly identified transactions.

Since the fraud detection problem has primarily been framed as a classification problem, numerous data mining methods have been developed to solve it in addition to some statistical approaches. The most well-liked of them are decision trees and artificial neural networks. A fair summary of the literature on issues with fraud detection can be found in the study by Bolton and Hand. However, the classical data mining algorithms are not directly applicable when the problem is viewed as a classification problem with variable misclassification costs, as discussed above. Either some modifications should be made to the existing algorithms, or new algorithms developed specifically for this purpose are required. Using general-purpose meta heuristic methods like genetic algorithms could be an alternate strategy.

Genetic algorithm

Evolutionary algorithms like genetic algorithms strive to produce better answers over time. Since their first introduction by Holland, they have been successfully used in a wide range of problem areas, including astronomy, athletics, computer science, and optimisation. They are frequently used with other data mining algorithms and have also been used in data mining, mostly for variable selection. In this work, we strive to find a single genetic algorithm solution to our categorization problem.

Pseudo code of genetic algorithm

Initialize the population

Evaluate initial population Repeat

Perform competitive selection

Apply genetic operators to generate new

solutions Evaluate solutions in the population Until some convergence criteria is satisfied.

Selection process

The best people, or those chromosomes with the highest fitness values, are chosen by selection. By using the current population, the selection process creates a "mating pool" that contains the individuals that will reproduce. There are various selection techniques, including biassed selection, random selection, selection using a roulette wheel, and selection through a competition. The following selection strategies are employed in this paper.

Tournament Selection

In this, tournament selection has been employed because it chooses the best people from a variety of groupings. The best individual from each group competes in a tournament made up of t uniformly chosen individuals from the current population, and the winner is added to the mating pool for recombination. The number of times this procedure must be carried out in order to reach the specified intermediate population size. The selection strength is governed by the tournament size. The selecting procedure is stronger as the event size increases.

Elitist Selection

This selection operator is designed to ensure that the best members of the solution are passed on to subsequent generations and should not be lost in random selection. Because they have a higher fitness value and are passed on to the population's next generation, we chose a handful of the best chromosomes from each generation.

Reproduction

from the solutions chosen by the genetic operators of crossover (also known as recombination) and/or mutation, to produce a second generation population of solutions. A pair of "parent" solutions are chosen from the pool that was previously chosen to breed in order to produce each new solution. By employing the aforementioned crossover and mutation techniques to construct a "child" solution, a new solution is created that often shares many traits with its "parents." Each new child is given a new set of parents, and this procedure is repeated until a new population of parents has been chosen.

The right size is produced. Although research reveals that more than two "parents" are better to be utilised in order to reproduce a good quality chromosome, methods of reproduction that are based on the use of two parents are more "biology inspired." These activities ultimately lead to a population of chromosomes in the subsequent generation that differs from the first generation. Since only the best creatures from the first generation are chosen for breeding, along with a small percentage of less fit solutions, for the reasons already explained above, the population's average fitness will have grown as a result of this approach. Although the two major genetic operators are recognised as crossover and mutation, genetic algorithms can also make use of regrouping, colonization-extinction, or migration.

Termination

This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria
- Fixed number of generations reached
- Allocated budget (computation time/money) reached
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
- Manual inspection
- Combinations of the above

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

3 Implementation Study

It is the process of putting a newly developed or upgraded system into use. The implementation phase can cause a lot of issues if it is not adequately planned and managed. So, careful implementation is necessary to offer a dependable system to satisfy management needs.

IMPLEMENTATION

The entire application was created in Java. This makes it possible for credit card companies to utilise this application on a wide range of devices, regardless of the manufacturers of such devices. As a back end for storing databases, we use Oracle.

CODING

To make sure that the code is legible, intelligible, and easily editable, standard coding principles are required. Standards and criteria have been established for this project that must be followed when pseudo-coding. These guidelines were followed whilst the programme was being developed in order to provide more consistent code and ease code maintenance.

NOMENCLATURE RULES

Programs are easier to understand because of naming conventions since they are easier to read. They may also include details regarding the identifier's purpose. Every control used in the project had names that were appropriate for that control's type.

COMMENTS The comments are used in programmes to make it easier to understand the code. The code was correctly commented throughout. For each function utilised, appropriate comments that detailed their functioning were provided. Variable names were also given comments to explain what they were used for.

CONSTRUCTION AND INDENTATION OF STATEMENTS

Nested blocks of code, function declarations, header files, etc. have all been properly indented. The project was carefully managed in terms of coding style since it adheres to a coding philosophy that emphasises simplicity and clarity. Creating a computer programme ultimately comes down to writing a series of statements in the language at hand, according to Dernighan and Plauger. The intelligibility of each of these statements depends in significant part on how they are conveyed the whole..."

4 PROPOSED WORK AND ALOGRITHAM

The suggested system effectively resolves the aforementioned problem. A genetic algorithm is used to identify fraud, reduce false alarms, and produce an optimum outcome. Based on client behaviour, the fraud is found. There is a brand-new classification issue with a fluctuating misclassification cost. Here, a set of interval-valued parameters are optimised using a genetic algorithm.

REGRESSION IN LOGISTICS

Another statistical method that machine learning has adopted is logistic regression.

It is the method of choice for issues involving binary categorization (problems with two class values). You will learn about the logistic regression algorithm for machine learning in this post.

LOGISTIC FUNCTION

The logistic function, which is the method's central component, inspired the name of the technique, logistic regression.

Statistics experts created the logistic function, also known as the sigmoid function, to characterise the characteristics of population expansion in ecology, which rise swiftly and peak at the carrying capacity of the ecosystem. Each real-valued number can be transformed into a value between 0 and 1, but never precisely at those ranges, using this S-shaped curve.

Where value is the actual numerical number that you want to alter and e is the base of the natural logarithms used by your spreadsheet's EXP() function. Diagram showing the logistic function's transformation of the numbers between -5 and 5 into the range between 0 and 1.

REPRESENTATION USED FOR LOGISTIC REGRESSION

Like linear regression, logistic regression represents data using an equation.

To anticipate an output value, input values (x) are mixed linearly with weights or coefficient values (y). The main distinction between

The advantage of linear regression is that it may model output values that are binary (0 or 1) rather than numerical.

Here is an illustration of a logistic regression formula:

y = e^(b0 + b1*x) / (1 + e^(b0 + b1*x))

When b0 is the bias or intercept term, b1 is the coefficient for the single input value, and y is the anticipated output (x). The associated b coefficient for each column in your input data must be discovered from your training set.

LOGISTIC REGRESSION PREDICTS PROBABILITIES (TECHNICAL INTERLUDE)

Modeling the probability of the default class using logistic regression (e.g. the first class).

For instance, the logistic regression model could be stated as the probability of male given a person's height if we were modelling people's sex as male or female based on their height. Or, to put it more formally:

P(RESULT=FRAUD|UNFRAUD)

Written another way, we are modeling the probability that an input (X) belongs to the default class (Y=1), we can write this formally as:

P(X) = P(Y=1|X)

We're making probabilities, right? I thought a classification algorithm was logistic regression.

Keep in mind that to make a probability forecast, the prediction must be converted into a binary number (0 or 1). More on this when we discuss making predictions later.

Although logistic regression is a linear technique, the logistic function is used to alter the predictions. The result is that, unlike with linear regression, we can no longer comprehend the predictions as a linear combination of the inputs. For instance, building on what was said earlier, the model can be written as follows:

p(X) = e^(b0 + b1*X) / (1 + e^(b0 + b1*X))

I don't want to dive into the math too much, but we can turn around the above equation as follows (remember we can remove the e from one side by adding a natural logarithm (ln) to the other):

$$\ln(p(X) / 1 - p(X)) = b0 + b1 * X$$

This is helpful because we can see that the input on the left is a log of the likelihood of the default class, and the output on the right is calculated linearly once more (exactly like linear regression).

The odds of the default class are shown by the ratio on the left (it is customary to use odds rather than probability in horse racing). Odds are determined by dividing the chance of an occurrence by the probability that it won't happen, for example, 0.8/(1-0.8) yields odds of 4. Instead, we could say:

$$\ln(odds) = b0 + b1 * X$$

We refer to this left side as the log-odds or the probit because the odds have been log converted. Although it is possible to use alternative functions for the transform (which is outside the focus of this article), the term "link function" is typically used to describe the transform that links the linear regression equation to probabilities, such as the probit link function.

We can write it as follows by repositioning the exponent to the right:

$$odds = e^{(b0 + b1 * X)}$$

All of this helps us understand that indeed the model is still a linear combination of the inputs, but that this linear combination relates to the log-odds of the default class.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

LEARNING THE LOGISTIC REGRESSION MODEL

You must estimate the logistic regression algorithm's coefficients (Beta values b) using the training set of data. Maximum-likelihood estimate is used for this.

Although it does make assumptions about the distribution of your data, the most prevalent learning technique utilised by many machine learning algorithms is maximum-likelihood estimation (more on this when we talk about preparing your data).
For the default class, a model with the best coefficients would predict a value extremely close to 1 (for example, male), and for the other class, a value very close to 0 (for example, female). The idea behind maximum-likelihood logistic regression is that a search process looks for coefficient values (also known as "Beta values") that minimise the difference between the probabilities predicted by the model and those in the data (e.g. probability of 1 if the data is the primary class).

We won't get into the mathematics of greatest likelihood. It suffices to state that the ideal coefficient values for your training data are optimised using a minimization approach. The use of effective numerical optimisation algorithms is frequently put into practise for this (like the Quasi-newton method).

MAKING PREDICTIONS WITH LOGISTIC REGRESSION

Input data into the logistic regression equation, calculate the outcome, and you have made a prediction using a logistic regression model.

PUT DATA INTO LOGISTIC REGRESSION

Similar to the assumptions used in linear regression, logistic regression makes assumptions about the distribution and relationships in your data.

These assumptions have undergone extensive research, and exact probabilistic and statistical language is utilised. My recommendation is to experiment with various data preparation strategies while using these as general recommendations or rules of thumb.

With projects involving predictive modelling and machine learning, your ultimate goal is to make precise predictions rather than to interpret the findings. As long as the model is reliable and effective, you can thus violate some of the assumptions.

Binary Output Variable: Since we've just discussed it, it might be easy to say that logistic regression is designed for binary (two-class) classification issues. It will forecast the likelihood that an instance will belong to the default class, which may be classified as a 0 or 1.

Reduce Noise: While the output variable (y) is assumed to be error-free in logistic regression, you might want to remove outliers and potentially misclassified instances from your training data.

A linear approach called logistic regression uses the Gaussian distribution (with a non-linear transform on output). It does presuppose a linear relationship between the input and output variables. A more accurate model can be produced by applying data transformations to your input variables that better reveal this linear relationship. To better expose this link, you can employ univariate transforms like log, root, Box-Cox, and others.
Eliminate Correlated Inputs: Similar to linear regression, if you have several highly linked inputs, the model may overfit. Think about determining the pairwise correlations between all inputs and eliminating inputs with a strong correlation.
Failure to Converge: It is possible for the process that learns the coefficients for expected likelihood estimation to fail to converge. This may occur if your data contains a large number of highly connected inputs.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

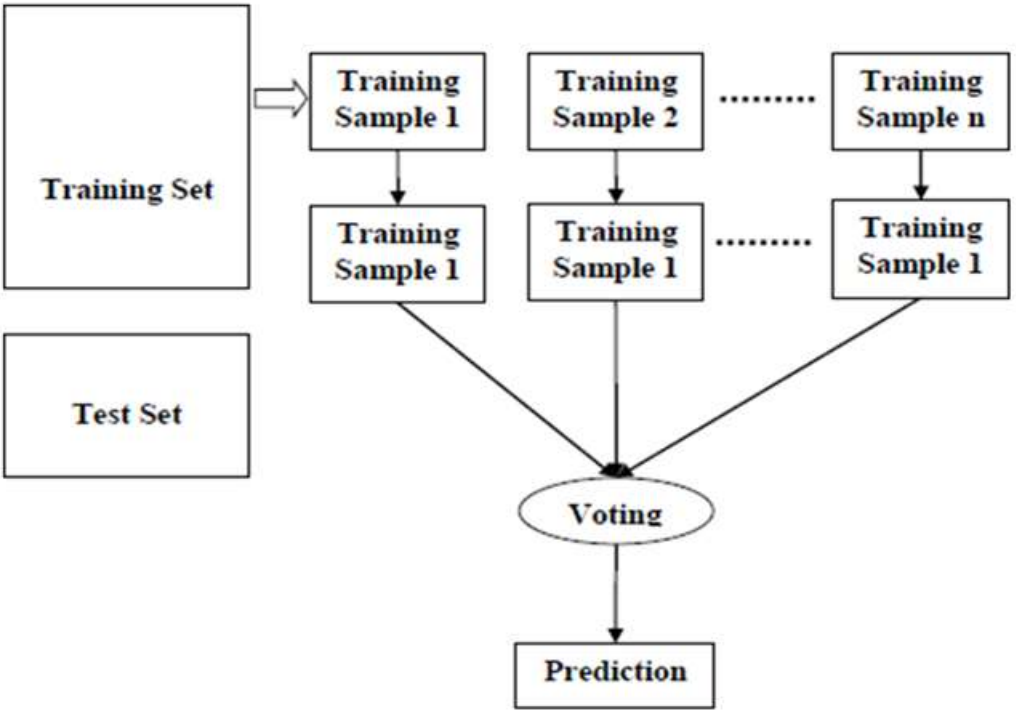
Random Forest

WORKING OF RANDOM FOREST ALGORITHM

We can understand the working of Random Forest algorithm with the help of following steps –

- **Step 1** − First, start with the selection of random samples from a given dataset.
- **Step 2** − Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
- **Step 3** − In this step, voting will be performed for every predicted result.
- **Step 4** − At last, select the most voted prediction result as the final prediction result.

The following diagram will illustrate its working −



XGBOOST

Both XGBoost and Gradient Boosting Machines (GBMs), ensemble tree approaches, use the gradient descent architecture to boost weak learners (CARTs in general). However XGBoost enhances the fundamental GBM architecture with system optimisation and algorithmic improvements.

System Optimization:

1.Parallelization: XGBoost uses a parallelized implementation to approach the sequential tree-building process. This is made possible by the interchangeability of the two inner loops that calculate the features and the outer loop that counts the leaf nodes of a tree when creating base learners. This nesting of loops restricts parallelization because the outer loop cannot be initiated until the inner loop, which is the more computationally intensive of the two, has been finished. As a result, the order of the loops is adjusted utilising initialization, a global scan of all instances, and sorting using parallel threads to save run time. By balancing any parallelization overheads in computation, this choice enhances algorithmic performance.

2. Tree Pruning: In the GBM architecture, the halting criterion for tree splitting is greedy in nature and is dependent on the negative loss criterion at the split point. Instead of starting with the first criterion, XGBoost uses the supplied "max depth" option and prunes the trees backward. Much better computational performance is achieved with this "depth-first" strategy.

3. Hardware Optimization: This algorithm has been created to use hardware resources effectively. By allocating internal buffers in each thread to hold gradient statistics, cache awareness does this. Other improvements, such "out-of-core" computation, maximise disc space while managing large data frames that don't fit in memory.

Enhancements to the algorithms:

1. Regularization: To avoid overfitting, it penalises more complex models using both LASSO (L1) and Ridge (L2) regularisation.

2. Sparsity Awareness: XGBoost handles various forms of sparsity patterns in the data more effectively by automatically "learning" the best missing value based on training loss and naturally admits sparse features for inputs.

3. Weighted Quantile Sketch: To efficiently identify the best split points across weighted datasets, XGBoost uses the distributed weighted Quantile Sketch algorithm.

4. Cross-validation: The algorithm includes a cross-validation approach that is incorporated into each iteration, eliminating the need to explicitly programme this search and to specify the precise number of boosting iterations necessary in a single run.

ADABOOST

AdaBoost stands for adaptive boosting, first and foremost. Ada Boosting was essentially the first truly effective boosting algorithm created for binary classification. It is also the finest place to start when enhancing understanding. Furthermore, contemporary boosting techniques—most notably stochastic gradient boosting machines—build on AdaBoost.
AdaBoost is typically applied on small decision trees. The performance of the tree on each training instance is used after the first tree is formed. Also, we employ it to gauge the importance of the following tree. It should therefore pay attention to each training instance as it is created. As a result, training data with low predictability is given more weight. Yet, cases that are straightforward to foresee are given less weight.
ADABOOST ENSEMBLE

- Basically, weak models are added sequentially, trained using the weighted training data.
- Generally, the process continues until a pre-set number of weak learners have been created.

- Once completed, you are left with a pool of weak learners each with a stage value.

DATA PREPARATION FOR ADABOOST

The best heuristics for preparing your data for AdaBoost are listed in this section. Excellent Data:
The ensemble technique makes an effort to rectify incorrect classifications in the training data. Also, you must take care that the training data is of excellent calibre.
Outliers:
Typically, outliers will drive the group deeper into the work. Therefore, it is quite challenging to adjust for irrational circumstances. The training dataset might be amended to exclude these.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

5 METHODOLOGIES

The two main categories of supervised machine learning algorithms are regression and classification algorithms.

Algorithms for regression: Regression determines whether dependent and independent variables are correlated. In order to forecast continuous variables, such as real estate values, economic trends, climatic patterns, oil and gas prices (a important task in today's world!), etc., regression methods are used.

- Regular Regression
- Decision Tree Regressor
- Random Forest Regressor

Random Forest Regression Algorithm:

Random forest is another well-liked approach in machine learning for non-linear regression. As opposed to decision tree regression, a random forest uses many decision trees to predict the outcome (single tree). This programme creates a decision tree from the input dataset using k randomly selected data points. Then, a number of decision trees are used to forecast the value of any new data point. Due to the large number of decision trees, a random forest method can predict a variety of output values. You must find the average of all the anticipated values in order to ascertain the final outcome for a new data point. This is due to the large number of decision trees that must be mapped when employing this approach. greater processing power. Trees are parallel; this is a bagging strategy rather than a boosting one. In other words, while you build trees, there is no interaction between these trees.

Classification Algorithms:

An algorithm known as classification finds ways to divide the dataset into groups according to various criteria. A computer software separates the data into various groups based on what it has learned from the training dataset.

- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier
- KNN - K Nearest Neighbour

Logistic Regression:

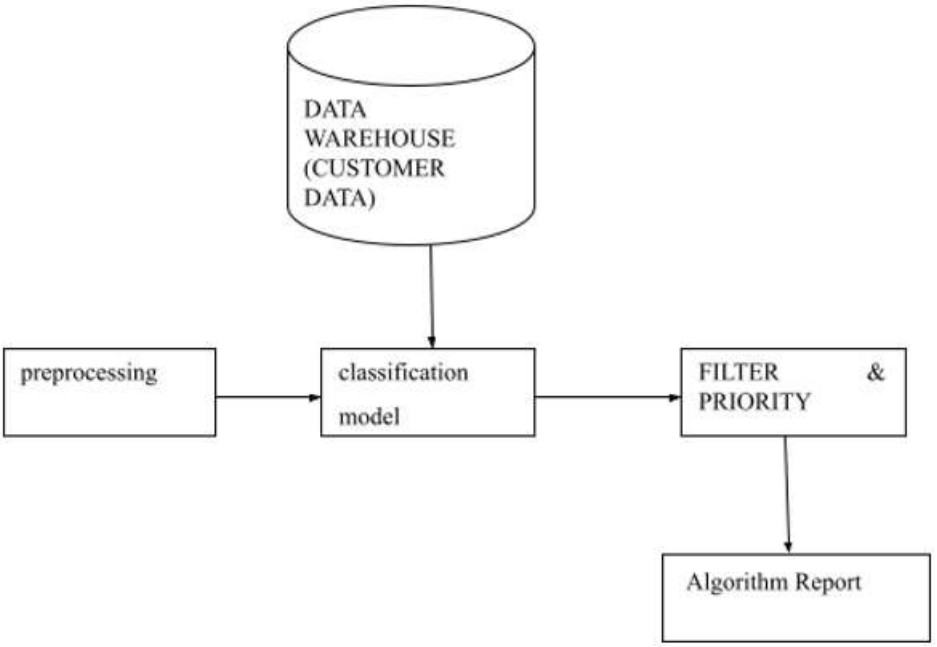
Logistic regression falls within the heading of supervised learning. It is used to forecast the categorical dependent variable using a specified set of independent variables. Logistic regression is used to forecast the result for a categorical dependent variable. The outcome must therefore be a discrete or categorical number. It offers the probabilistic values that lie between 0 and 1 rather than the exact values between 0 and 1. Instead of fitting a regression line in logistic regression, we fit a "S" shaped logistic function that predicts two maximum values. (0 or 1). Logistic regression can quickly identify the most effective parameters that were utilised for classification by employing a variety of data types to classify the observations.

"Conceiving and planning out in mind and making a drawing, pattern, or a sketch" are all parts of the design process. Throughout development, the system design turns a logical representation of what a particular system must do into a physical reality. Consideration should be given to crucial design elements including dependability, response time, system throughput, maintainability, expandability, etc. Constraints on design, such as budget, hardware limits, standard compliance, etc., should also be addressed. The goal of system design is to take the description and attach a specific set of resources to it, such as personnel, equipment (including computing devices), housing, etc., in order to provide detailed specifications for a functional system.

This new system must offer all necessary data processing functions, as well as certain optional features that were uncovered during the analysis effort. It must function within the limitations set and outperform the current system. A decision between the primary approaches must be taken at the beginning of design. While discussing "preliminary design," it is important to identify the primary

design possibilities that can be used in the development and execution of a system. The easiest way to discern between these possibilities is by looking at the actual processing equipment that will be used, as well as who or what will be doing the work.

Defining the requirements and setting the high level of the system are both concerns when describing the general aspects of the software. The different web pages and their connections are recognised and designed during architectural design. The main parts of the software are recognised, broken down into processing modules and conceptual data structures, and their linkages are shown.
The user has excellent screen interaction options thanks to output design. the data made available to users via an information system. To ensure that the information system is used and accepted, useful output is crucial. Users frequently assess a system's worth depending on its results. Only close interaction with users will result in productive productivity. The output is created in an appealing and practical manner so that a user can access them in case of a



difficulty.

Fig. 1: propose Architecture

6 **RESULTS AND DISCUSSION**

SCREENSHOTS

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

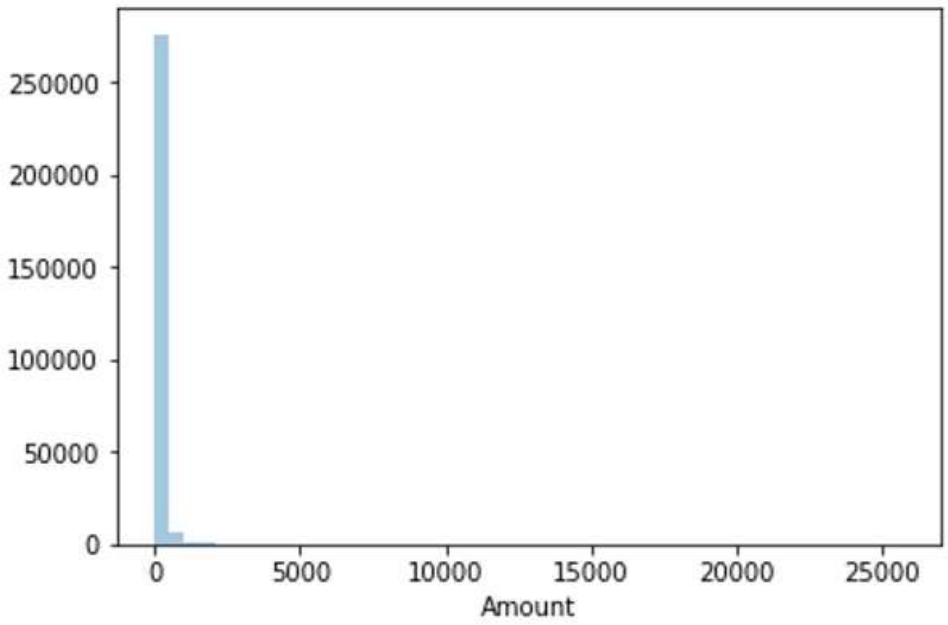


fig 1 :- There are only 7 transactions which are greater than 10,000 dollars and all of these transactions are non-fradulent (Class 0). Therefore, removing these outliers.

evolution metrics

A. Percision = $TPTP+FP$TPTP+FP

B. Recall = $TPTP+FN$TPTP+FN

C. F1 score =

$21Precision+1Recall$

logistic regression

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

	precision	recall	f1-score	support
0	1.00	1.00	1.00	93862
1	0.62	0.84	0.72	122
micro avg	1.00	1.00	1.00	93984
macro avg	0.81	0.92	0.86	93984
weighted avg	1.00	1.00	1.00	93984

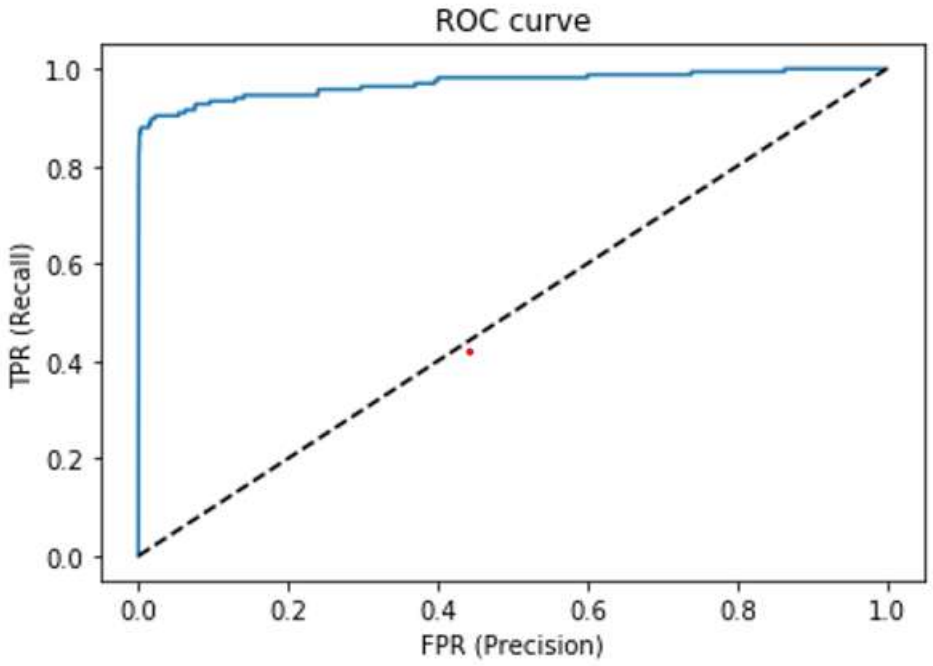
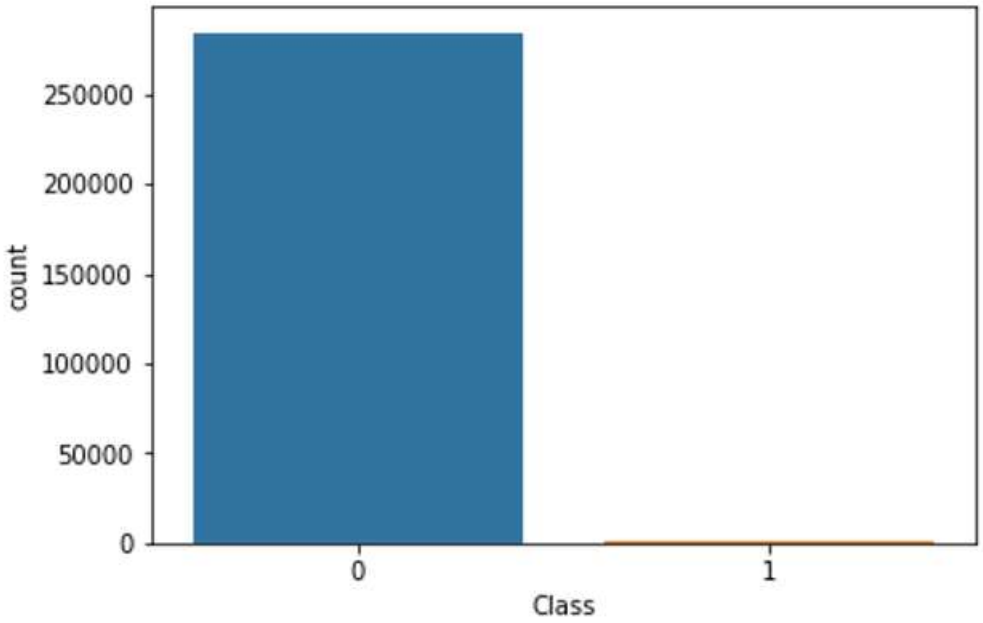


fig 2:- Area under curve (AUC): 0.9705905846413216(logistic regression)

This dataset is quite unbalanced. Because of this, our accuracy and AUC values are exceptionally high. Our model predicts non-fradulent transactions more accurately than fradulent transactions, according to the confusion matrix (F1 score for non-fradulent transactions is 100%, compared to 72% for fradulent transactions). We have 99.9% of non-fraudulent transactions, which is one explanation for this.

just 0.172% of transactions were fraudulent. Correctly identifying the fraudulent transactions is the goal. We can't continue to predict that many fraudulent transactions incorrectly. Just 62% of fraudulent transactions can be predicted by the model

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023



(precision score).

fig 3:- The above distribution makes it evident that there are far less fraudulent transactions (only.17%) than non-fraudulent transactions (99.83%). Every classifier that is performed directly will produce a higher accuracy score since the data heavily favours one of the classes.

accuracy of training and testing dataset(logistic Regression)

accuracy on the training set: 0.9992191594172518 accuracy on the testing set: 0.9991594582229457

confusion matrix of train and test data

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

Confusion Matrix on train data [[190463
122]
[27 208]]
Confusion Matrix on test data [[93811 65]
[14 97]]

descrption

Because the model predicts the "test" dataset nearly identically to the "train" dataset, there is no overfitting. The accuracy simply reflects the underlying class distribution, which in this case is a majority of non-fradulent cases, due to the extreme imbalance in our dataset. In order to identify overfitting, we must thus calculate additional metrics like precision, recall, and F1 score using the confusion matrix.

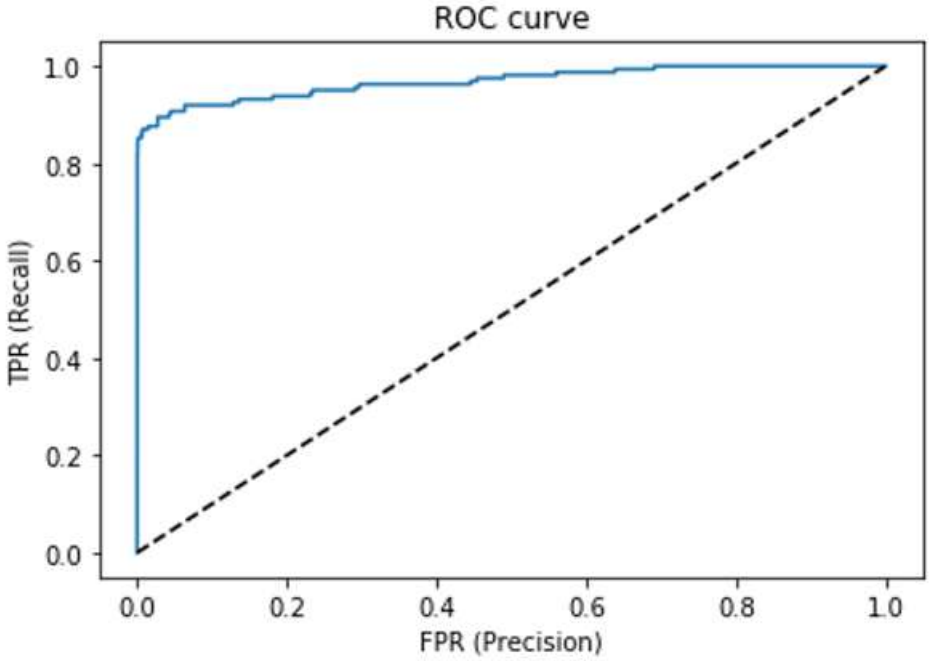
classification report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	93876
1	0.60	0.87	0.71	111
micro avg	1.00	1.00	1.00	93987
macro avg	0.80	0.94	0.86	93987
weighted avg	1.00	1.00	1.00	93987

We found that our model does a great job at predicting non-fraudulent transactions. They are not very good at foreseeing fraudulent transactions, though. Precision is 60%, recall is 87%, and the F1-score is just 71% for fraudulent transactions.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

Precision is a crucial parameter since we need our model to be accurate. Banks would stop legitimate transactions if our model did not anticipate accurately, which would make customers unhappy.
Recall is another crucial statistic; if it's low, it means that our model does a poor job of identifying fraudulent transactions. Due to improper detection of these transactions, the bank would suffer losses.
Thus, F1, the harmonic mean of the precision and recall scores, is what we are interested



in.

fig Area under curve (AUC): 0.96833762619534

Although AUC is high, we are not happy with the results because it does not effectively detect fraudulent transactions. Our approach is more accurate at predicting honest than dishonest transactions. Also, we are aware that the severely unbalanced dataset is the reason of this. So what are we to do?

The following strategies have been tested to address the imbalanced dataset issue.

1. Cross-Validation
2. Oversampling of minority groups (i.e. fradulent transactions)
3. Group modelling (Random Forrest)
4. Bagging
5. Increasing (XGBoost, ADABoost)
6. Modifying the Threshold

Confusion Matrix on train data [[190463
27]
[126 204]]
accuracy on the training set: 0.9991981972539566

Accuracy on Test Data

Confusion Matrix on test data [[93808 17]
[67 95]]
accuracy on the test set: 0.9991062584641602

cross validation- classification report on test data:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	93875
1	0.59	0.85	0.69	112
micro avg	1.00	1.00	1.00	93987
macro avg	0.79	0.92	0.85	93987
weighted avg	1.00	1.00	1.00	93987

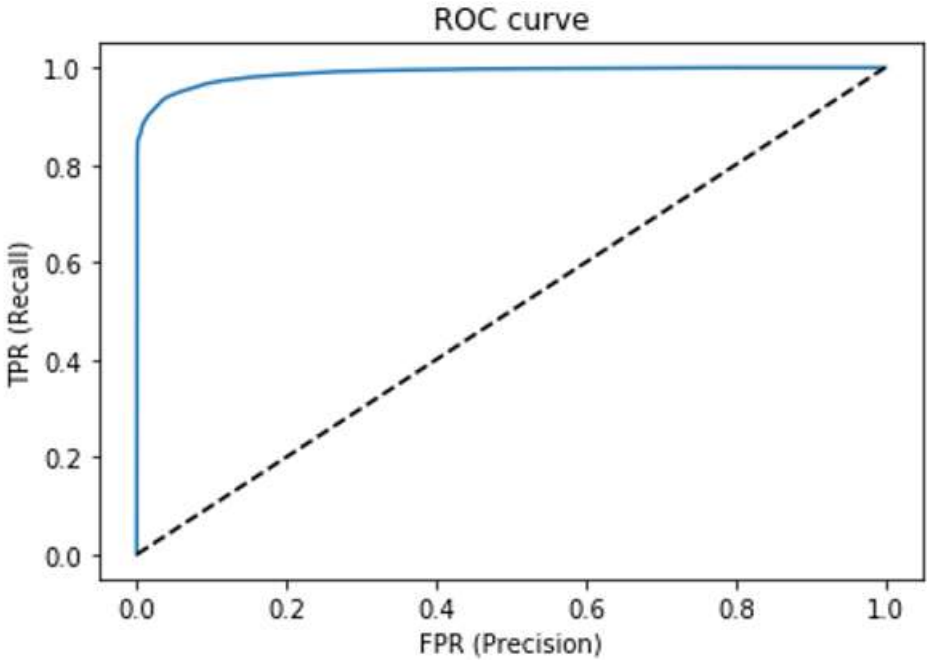
Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

There is not much improvement using cross-validation. For fradulent transactions, precision is 59%, recall is 85% and F1-score of 69%

only. instantiate a logistic regression model, and fit with X_train and y_train

accuracy on the testing set: 0.9464955661664393 [[91344 7691]
[2349 86264]]

	precision	recall	f1-score	support
0	0.97	0.92	0.95	99035
1	0.92	0.97	0.95	88613
micro avg	0.95	0.95	0.95	187648
macro avg	0.95	0.95	0.95	187648
weighted avg	0.95	0.95	0.95	187648



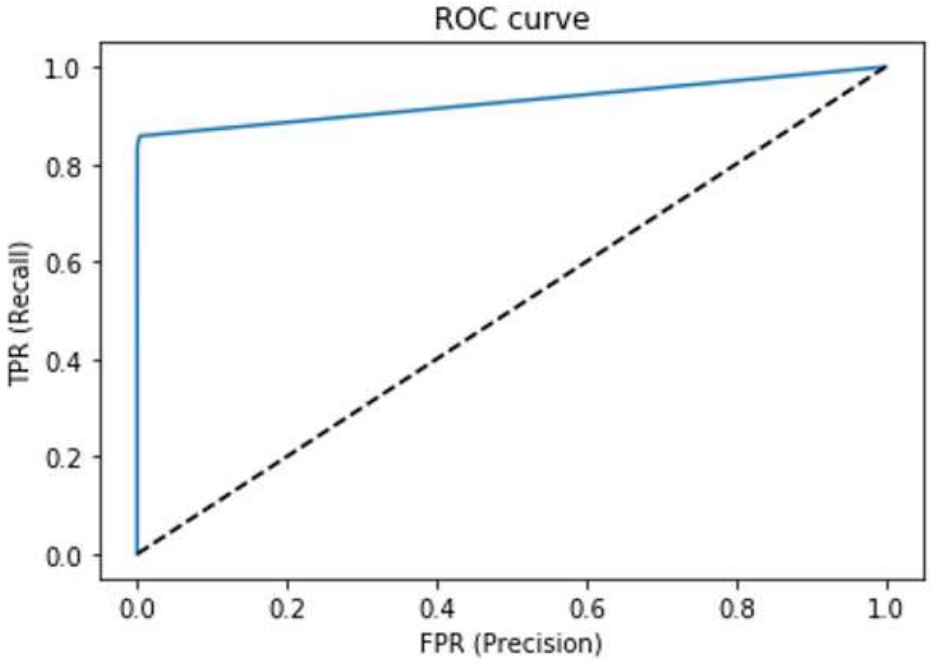
Area under curve (AUC): 0.989053853397718

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

High AUC (98.9%). Also high (95%, 92%, and 97%, respectively) are our F1-score, precison, and recall scores.

With this strategy, we can accurately anticipate 92% of fraudulent cases. Our detection rate dramatically increased as a result of oversampling the minority class.

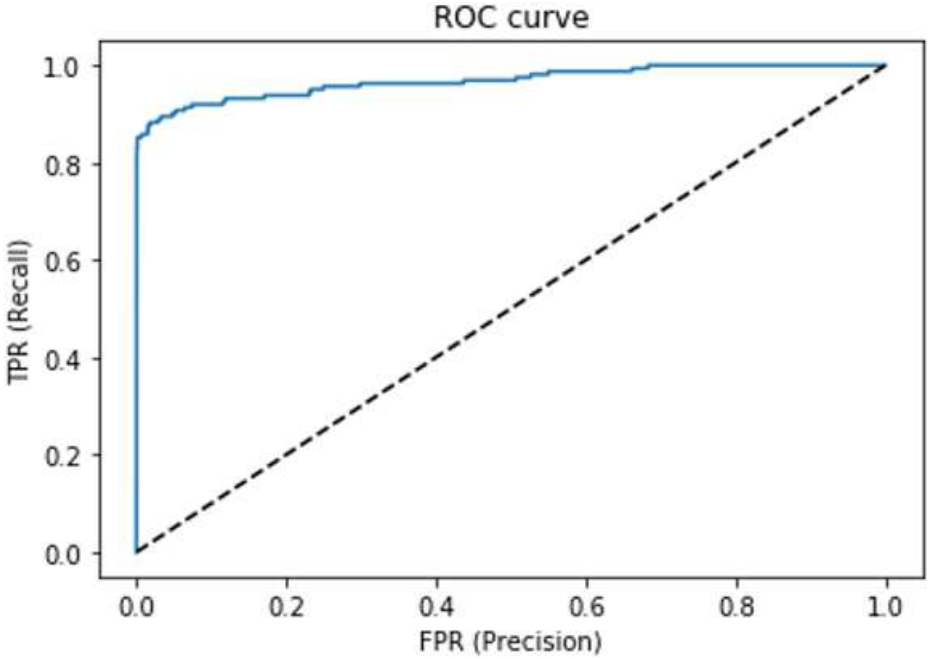
Ensemble modelling (Random Forrest)



Area under curve (AUC): 0.9287169112446668

With Random forrest, our AUC has dropped to 92.2% from 96.8% and recall has also seen a dropped to 75% from 85%. However, precision has been increased from 60% to 94%. If we are only concerned about precision then ensemble method like Random forrest gives better results.

Bagging



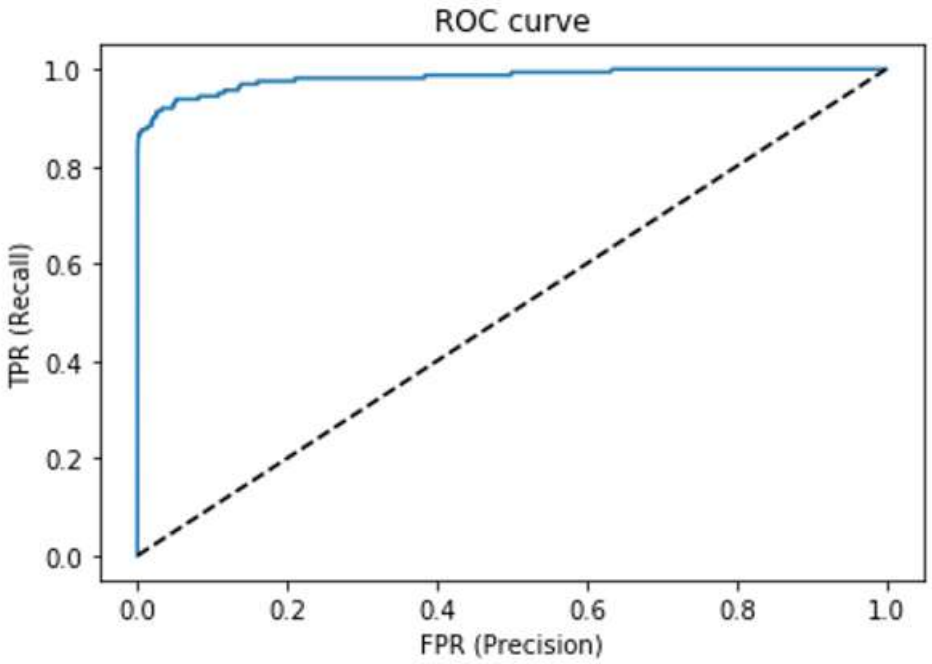
Area under curve (AUC): 0.9680629488179004

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023

Bagging does not play much role in imporvements of AUC, recall or precision scoes. There are marginal improvements.

Boosting (XGBoost, ADABoost)

XG Boost

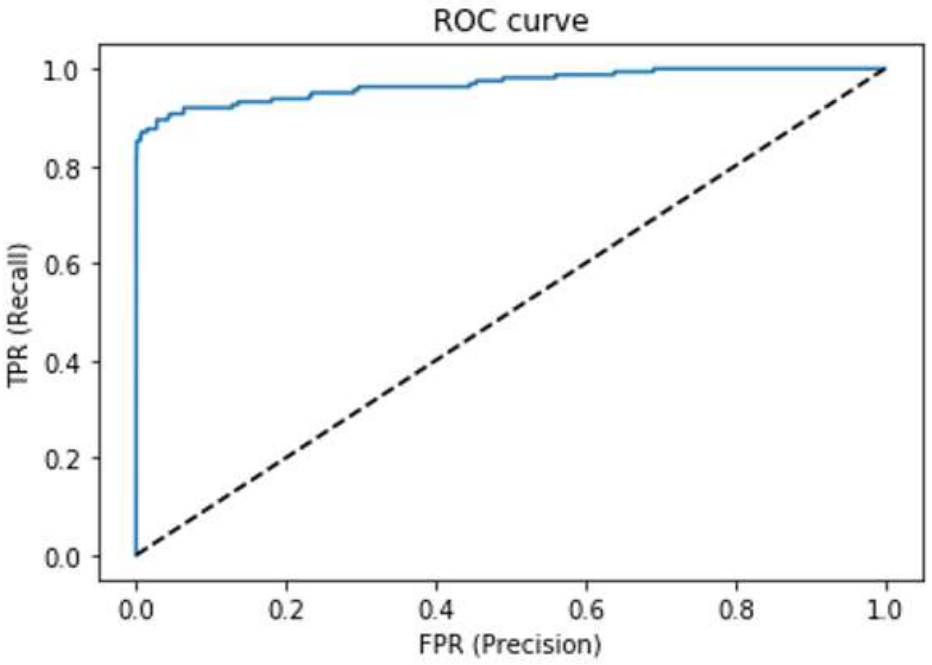


Area under curve (AUC): 0.9828346376396825

Xgboost increased the performance of f1 score significantly to 85% from 71%(Refer section A-4 - original model) and AUC increased from 96.8% to 98.28%.

ADABOOST

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 4, April : 2023



Area under curve (AUC): 0.96833762619534

Our F1-score for detecting fraudulent transactions increased from 71% to 81% after raising the threshold to 10% (see to section A-4 for the original model), even though the AUC did not change.

CONCLUSION

Scores summary of all the models are as follows:

Model	Accuracy	AUC	Precision	Recall	F1-Score
Original Model	0.99	0.96	0.6	0.87	0.71
Cross Validation	0.99		0.59	0.85	0.69
Oversampling (SMOTE)	0.946	0.989	0.92	0.97	0.95
Ensemble (Random Forrest)	0.99	0.928	0.93	0.75	0.83
Bagging (Decision Trees)	0.99	0.97	0.61	0.88	0.72
Boosting (XGBOOST)	0.99	0.98	0.79	0.92	0.85
Boosting (ADABOOST)	0.99	0.969	0.83	0.65	0.73
Threshold change to 10%	0.99	0.968	0.79	0.83	0.81

7. CONCLUSION AND FUTURE WORK

This technique effectively eliminates fraudulent transactions while reducing the amount of false alarms. In this literature, the genetic algorithm is unique in terms of the application domain. The likelihood of fraudulent transactions can be predicted shortly after credit card transactions if this method is used in a bank's credit card fraud detection system. Moreover, a number of anti-fraud methods can be implemented to lower risks and shield banks from significant losses.

Because we had a changeable misclassification cost, the study's goal was interpreted differently than it would be for ordinary classification issues. We choose to employ the multi population genetic algorithm to produce an optimum parameter because the conventional data mining algorithms do not work well in this case.

8. REFRENCES

[1] M. Hamdi Ozcelik, Ekrem Duman, Mine Isik, Tugba Cevik, Improving a credit card fraud detection system using genetic algorithm, International conference on Networking and information technology 2010.

[2] Wen-Fang YU, Na Wang, Research on Credit Card Fraud Detection Model Based on Distance Sum, IEEE International Joint Conference on Artificial Intelligence 2009.

[3] clifton phua, vincent lee1, kate smith & ross gayler, A Comprehensive Survey of Data Mining-based Fraud Detection Research,2005.

[4] Elio Lozano, Edgar Acu˜na, Parallel algorithms for distance-based and density-based outliers,2006.

[5] Credit card fraud detection using hidden markov model – Abinav Srivastava,Amlan Kundu,Shamik Sural,Arun K.majumdar Websites:

[1]http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/tcw2/report.html

[2] http://www.kxcad.net/cae_MATLAB/toolbox/gads/f6691.html

[3] http://java.sun.com/developer/onlineTraining/Programming/BasicJava1/front .html [4]http://www.easywayserver.com/blog/user-login-in-jsp/

[5]http://www.faqs.org/patents/app/20100094765

Textbooks:

[1]Pressman, Roger S. Software engineering: a practitioner's approach / Roger S. Pressman.—5th ed.p. cm (McGraw-Hill series in computer science).

[2]E.Balagurasamy, Programming with java, Tata McGraw-Hill Publication.

[3] Ali Bahrami,Oject Oriented system Development, Tata McGraw-Hill Publication .

[4] Jiwei Han et,al., "Data Mining :Concepts and Techniques", MorganKaufmaan Series,2000