Predicting Loan Repayment Using a Hybrid of Genetic Algorithms, Logistic Regression, and Artificial Neural Networks

Pham Thanh Binh and Nguyen Dinh Thuan(✉)

University of Information Technology, VNU-HCM, Ho Chi Minh City, Vietnam
binhpt.15@grad.uit.edu.vn, thuannd@uit.edu.vn

Abstract. Loans are important products of financial institutions and banks. All institutions are trying to find effective business strategies to convince more customers to apply for a loan. However, some customers are unable to repay the loan after their application is approved. Therefore, many financial institutions and banks have considered some events when approving a loan. Determining whether a borrower can repay a loan is difficult. If the Financial institution, the Bank is too strict, there will be fewer approved loans, which means less profit. But if the approval is too loose, they will approve loans that default. The Machine learning classification algorithms are applied to predict loan default: Logistic Regression, Decision Tree, and Artificial Neural Networks. Accuracy, precision, recall, and ROC curve are used to evaluate the models and the results compared. We use feature selection techniques and propose models of Ensemble learning that are Logistic Regression with Decision Tree, and Logistic Regression with Decision Tree. We achieve the highest accuracy of 84.68% using the Logistic Regression with Decision Tree ensemble learning model.

Keywords: Loan repayment prediction · Logistic regression · Decision tree · Genetic algorithm · Ensemble learning

1 Introduction

Financial institutions and banks use credit scoring models to assess the risk of default. The models generate a score that predicts default, making lending decisions easier. The development of a loan repayment prediction model is time-consuming. These models are also fixed and are not easily developed with changing customer behaviour to predict default more accurately. Machine Learning approaches can help improve customer default prediction accuracy. The dataset used in this paper was from www.lendingclub.com. The dataset includes 37,066 loans between January 2018 and September 2020. In this study, we consider the problem of choosing predictor variables in the classification problem. Based on the factors gathered during the loan repayment process, the classification objective is to predict whether the borrower will be able to repay the loan or not. The method includes steps of data collection, data preprocessing, data analysis, and model building by applying Logistic Regression, Decision Tree, and Genetic Algorithm,

T. K. Dang et al. (Eds.): FDSE 2022, CCIS 1688, pp. 161–175, 2022.
https://doi.org/10.1007/978-981-19-8069-5_11

and we propose the method of feature selection and ensemble learning model is Logistic Regression with Decision Tree, Logistic Regression with Artificial Neural Networks Loan repayment prediction.

2 Related Works

In this paper, we look at different papers that have been used to predict defaults.

Wang et al. present a study that uses 4000 records and 21 attributes to build and evaluate a classifier predictive model. Four algorithms are used in this paper: Classic SVM, Backpropagation Neural Network, C4.5, and R SVM. The results show that the prediction accuracy of R SVM is better than other methods [1].

Reddy and Kavitha [2] use a neural network through attribute relevance analysis in the test class defaulter. Hassan and Abraham [3] used a bank dataset with 1000 cases with 24 numeric attributes to develop and compare models generated from different training algorithms, and conjugate gradient backpropagation. Scaling, and the Levenberg-Marquardt algorithm and secure one-step reverse propagation (SCG, LM, and OSS). Research shows that the slowest algorithm is OSS and the best algorithm is LM because it has the largest R.

Hamid and Ahmed [4] propose a classification model for the application of loans using three algorithms; J48, Bayesian network, and Naive Bayes classifier. They use the Weka app for deployment and testing. The results show that J48 has the best accuracy of 78.378%. Turkson et al. [5] applied 15 different types of machine learning algorithms to predict customer creditworthiness. Testing shows that, in addition to the nearest Centroid and Gaussian Naive Bayes. Each of these algorithms achieves accuracy rates from 76% to over 80%.

The Odegua recommends using an Extreme Gradient Bo boost algorithm called XGBoost to predict loan defaults. The prediction is based on loan data from a bank with a dataset containing 4368 samples and 10 attributes from both the loan application and applicant demographics. The location and age of the customer are the two most important characteristics that affect a loan's likelihood of default. The XGBoost model has an accuracy of 79%, accuracy (97%), recall (79%), and an F1 score (87%) [6]. Hybrid classifier and default prediction using a real data set of 132,029 cases from an international bank using AdaBoost, XGBoost, random forest, multilayer perception, and K-Nearest neighbours.

Mohammad et al. present a study on loan prediction by building logistic regression with a sigmoid function model and analyzing the problem of predicting defaulters. A logistic regression model is built and different performance measures are calculated. Models were compared based on performance measures of sensitivity and specificity. The best case accuracy obtained was 81.1%. The researchers concluded that the logistic regression method effectively detects the right target customers to grant loans [7].

3 Models

3.1 Logistic Regression

The classification models all seek to determine the boundary that divides groups between data. In Logistic regression, we also look for such a division boundary to solve the binary

classification problem between two groups 0 and 1. In linear regression, we rely on a hypothetical regression function $h_w(x) = w^T x$ to predict the continuous target variable y. Because the value y can be out of range [0, 1], in Logistic Regression a function is needed that projects the predicted value on the probability space within the interval [0, 1] and at the same time creates nonlinearity for the regression equation to help it there is a better dividing line between two groups. That is the Sigmoid function or Logistic function that we will learn below [8].

Sigmoid Function
The logistic regression model is a continuation of the idea of linear regression into classification problems. From the output of the linear function, we feed the Sigmoid function to find the probability distribution of data. Note that the Sigmoid function is only used in the binary classification problem. For the classification problem of more than two labels, the Softmax function (to be explored in later chapters) is a generalized form of the Sigmoid function that will be used. The Sigmoid function is a nonlinear transform function based on the formula Eq. (1).

$$\sigma(t) = \frac{1}{1 + \exp(-t)} \tag{1}$$

The logistic—noted σ(.)—is a sigmoid function (i.e., S-shaped) that wrong form a number between 0 and 1.

3.2 Decision Tree

The decision tree in the diagram above is also called a binary decision tree because a question has only two options, True or False. We have some concepts related to decision trees:

Root node: The node at the top of the decision tree. All alternatives originate from this node.
Parent node: A node that can branch down to other nodes below. The underlying node is called a child node.
Child nodes: These are nodes where the parent node exists.
A Leaf node: The final node of a decision. Here we get the forecast result. The leaf node is in the last position so there will be no child nodes.
A Non-leaf node: Nodes other than leaf nodes.

From the above decision tree diagram, we see a decision tree composed of nodes and edges. At each node, a yes/no question is asked of an input variable. Depending on the answer, you will next turn to the True or False branch. Continue doing the same branching recursively until the answer is obtained at the last node [9].

3.3 Artificial Neural Networks (ANN)

A neural network, also known as an Artificial Neural Network, is a network that uses complex mathematical models to process information. They are based on the activity

pattern of neurons and synapses in the human brain. Similar to the human brain, an artificial neural network connects simple nodes, also known as neurons. And such a set of nodes forms a network of nodes, hence the name artificial neural network. Similar to the human brain, in an artificial neural network, a series of algorithms are used to identify and recognize relationships in data sets. Artificial neural networks are used across a variety of technologies and applications such as video games, computer vision, speech recognition, social network filtering, automatic translation, and medical diagnostics. Surprisingly, neural networks are used for traditional and creative activities, like painting and art. The three main components of a neural network include: The input layer represents the input data. The hidden layer represents the intermediate nodes that divide the input space into regions with (soft) boundaries. It takes a set of weighted inputs and produces the wrong form through an activation function. The output layer represents the output of the neural network [12].

4 Hybrid Methodology

4.1 Ensemble Learning

Voting

A voting ensemble is an ensemble machine learning model that combines the predictions from multiple other models.

It is a technique that may be used to improve model performance, ideally achieving better performance than any single model used in the ensemble.

A voting ensemble works by combining the predictions from multiple models. It can be used for classification or regression. In the case of regression, this involves calculating the average of the predictions from the models. In the case of classification, the predictions for each label are summed and the label with the majority vote is predicted.

Regression Voting Ensemble: Predictions are the average of contributing models.

Classification Voting Ensemble: Predictions are the majority vote of contributing models.

There are two approaches to the majority vote prediction for classification; they are hard voting and soft voting.

Hard Voting. Predict the class with the largest sum of votes from models.

Soft Voting. Predict the class with the largest summed probability from models.

4.2 Suggestion Model

Voting Classifier

Majority Vote-based Ensemble Classifier:

Step 1: input data from column 1 to column 13 are independent variables, and output data from column 14 is the dependent variable.
Step 2: Data Preprocessing

Data cleaning: remove data missing value.
Handling Text and Categorical Attributes
Split Data into Training, Validation, and Testing Dataset
Feature Selection
Feature Scaling

Step 3: Apply 3 classifiers: Logistic Regression, Decision Tree, and Artificial Neural Networks to the training data.
Step 4: Predict the result of testing data, and compare the performance of the 3 classifiers.
Step 5: Performing Majority Voting for every observation.
Step 6: Compare the performance of the Majority Voting with the Logistic Regression, Decision Tree, and Genetic Algorithm classifiers.

The steps involved in the methodology are graphically represented below in Fig. 1

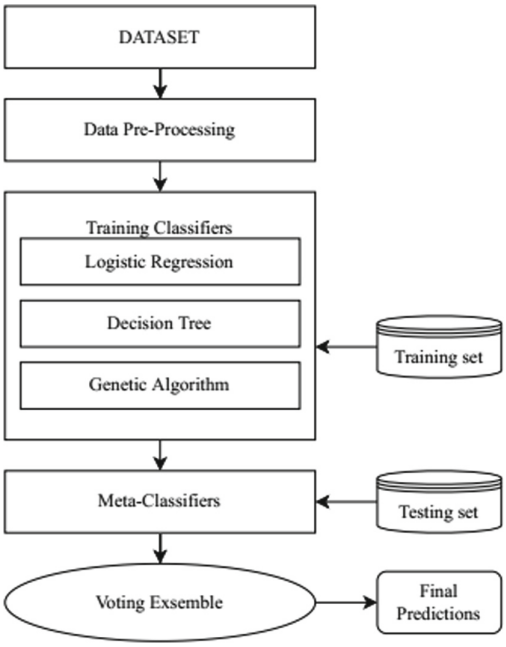


Fig. 1. Voting ensemble classifier

In this paper, we use Classification Voting Ensemble and Hard Voting. We develop voting ensembles in the Logistic Regression and Decision Tree model, Logistic Regression and Artificial Neural Networks model, in there, Logistic Regression and Decision Tree model we collect the highest Precision rate of 89.74% and ROC rate of 80.66% (Table 3).

5 Experiment

5.1 Dataset: Feature Analysis

In this paper, the dataset got from www.lendingclub.com. The dataset includes 37,066 loans between January 2018 and September 2020. Table 1 shows the 5 rows of the dataset.

Table 1. First five rows of the dataset.

credit_policy	purpose	int_rate	installment	log_annual_inc	dti	fico	days_with_cr_line	revol_bal	revol_util	inq_last_6mths	delinq_2yrs	pub_rec	not_fully_paid
1	credit_card	0.143	498.35	150000	15.7	694	37865	19748	0.674	3	5	0	0
1	home_improvement	0.11	870.29	55000	30.13	734	35947	11898	0.476	2	1	0	0
1	credit_card	0.088	785.32	165000	16.11	694	36373	20681	0.567	0	0	0	0
1	credit_card	0.17	285.03	40000	32.07	744	40269	8514	0.226	2	0	0	0
1	debt_consolidation	0.088	570.81	36000	23.73	714	38565	7555	0.256	1	0	0	1

- credit_policy: 1 if the customer meets the credit underwriting criteria of Lending-Club.com, and 0 otherwise.
- purpose: The purpose of the loan such as credit card, debt consolidation, etc.
- int_rate: The interest rate of the loan (proportion).
- installment: The monthly installments ($) owed by the borrower if the loan is funded.
- log_annual_inc: The natural log of the annual income of the borrower.
- dti: The debt-to-income ratio of the borrower.
- fico: The FICO credit score of the borrower.
- days_with_cr_line: The number of days the borrower has had a credit line.
- revol_bal: The borrower's revolving balance.
- revol_util: The borrower's revolving line utilization rate.
- inq_last_6mths: The borrower's number of inquiries by creditors in the last 6 months.
- delinq_2yrs: The number of times the borrower had been 30 + days past due on a payment in the past 2 years.
- pub_rec: The borrower's number of derogatory public records.
- not_fully_paid: indicates whether the loan was not paid back in full (the borrower either defaulted or the borrower was deemed unlikely to pay it back).

5.2 Data Preprocessing

Data Cleaning

Machine Learning algorithms cannot predict missing features, We can see that the dti, revol_util, inq_last_6mths, and delinq_2yrs attribute has missing values, so let's fix this, so we have three options to take care of missing values.

We have three options [8]:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 37066 entries, 0 to 37065
Data columns (total 14 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   credit_policy      37066 non-null  int64
 1   purpose            37066 non-null  object
 2   int_rate           37066 non-null  float64
 3   installment        37066 non-null  float64
 4   log_annual_inc     37066 non-null  float64
 5   dti                36966 non-null  float64
 6   fico               37066 non-null  int64
 7   days_with_cr_line  37066 non-null  int64
 8   revol_bal          37066 non-null  int64
 9   revol_util         37054 non-null  float64
 10  inq_last_6mths     37066 non-null  int64
 11  delinq_2yrs        37032 non-null  float64
 12  pub_rec            37066 non-null  int64
 13  not_fully_paid     37066 non-null  int64
dtypes: float64(6), int64(7), object(1)
memory usage: 4.0+ MB
```

Fig. 2. Data cleaning

– Get rid of the corresponding attribute.
– Get rid of the whole attribute.
– Set the values to some value (zero, the mean, the median, etc.).

In this paper, we choose option 1, to get rid of the corresponding attribute of the missing values in the training set. Figure 3 description of data cleaning.

Handling Text and Categorical Attributes.
In the dataset, we see the attribute 'purpose', let's fix this attribute, a common solution is to create one binary attribute per category: one attribute equal to 1 when the category is "all_other"(and 0 otherwise), another attribute equal to 1 when the category is "credit_card" (and 0 otherwise), another attribute equal to 1 when the category is "debt_consolidation" (and 0 otherwise), and so on [8] (see Fig. 4).

credit_card	debt_consolidation	educational	home_improvement	major_purchase	small_business
0	1	0	0	0	0
1	0	0	0	0	0
0	1	0	0	0	0
0	1	0	0	0	0
1	0	0	0	0	0

Fig. 3. Handling text and categorical attributes

Split Data into Training, Validation, and Testing Dataset
Loan Repayment data mentioned above will be divided into 2 parts: 70% of the time points used for training and 30% of the remaining will be used for testing in the case of all models.

Feature Selection
In this paper, we use the Genetic Algorithms (GAs) technique for feature selection, Genetic algorithms are a computer science technique for solving combinatorial optimization problems. GAs are based on evolutionary adaptations of biological populations based on Darwin's theory. It employs the principles of heredity, mutation, natural

selection, and crossover. GAs use some genetic terminology such as chromosome, population (Population), and Gene. Chromosomes are made up of Genes (represented by a linear sequence). Each gene carries some characteristics and has a certain position in the chromosome. Each chromosome represents a solution to the problem. In this article, I will explain the concepts of parallelism with programming in a specific problem. GAs are used for difficult problems, and have been successfully applied to some problems such as planning, control systems, travelling people problems, etc., Fig. 5.

The algorithm will be performed through the following steps:

Population initialization: Randomly generate a population of n individuals (where n is the solution to the problem).
Evaluation: Estimate the fitness of each individual.
Stop condition: Check the condition to end the algorithm.
Selection: Select two parents from the old population according to their fitness (the higher the fitness, the more likely it is to be selected).
Crossover: With a chosen probability, crossover two parents to create a new individual.
Mutation: With a selected mutation probability, transform the new individual.

Select result: If the stopping condition is satisfied, the algorithm terminates and chooses the best solution for the current population.
We have the overall diagram:

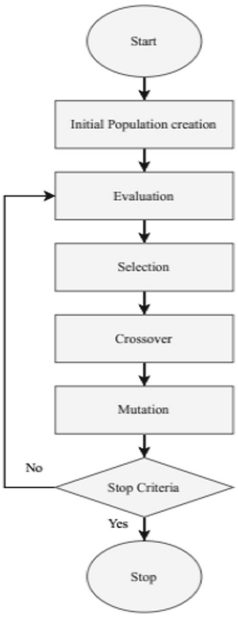


Fig. 4. How the genetic algorithm works

Feature Scaling

We apply technology feature scaling in the data processing is data normalization. Machine Learning algorithms do not work well when the input features have different values. Min-max normalization (also known as normalization) is the simplest method: the values are shifted and scaled so that they are between 0 and 1. We do this by subtracting the smallest value and dividing it by the largest and smallest values [8].

There are two common ways to get all attributes to have the same scale: min-max scaling and standardization.

In this paper, we choose min-max scaling.

$$x' = \frac{x - min(x)}{(x) - min(x)} \tag{2}$$

0.491807	1.268411	0.035292	1.071630	1.004770	0.498436	0.331567	0.162603	1.829279	0.714583	0.300395	0.238926	0.386843	1.193412	0.193019	0.260663	0.221177	0.262183
2.033318	0.158325	0.296288	0.323093	0.102024	0.288378	0.003432	0.285254	0.384146	1.087817	0.300395	0.238926	0.386843	0.837934	0.193019	3.836369	0.221177	0.262183
0.491807	1.229304	0.460964	0.118105	0.926396	0.944057	0.542356	0.072044	0.128309	0.263983	0.300395	0.238926	2.585025	0.837934	0.193019	0.260663	0.221177	0.262183
0.491807	0.479074	0.413384	0.364756	1.685458	0.629572	0.326793	0.254474	0.294258	0.263983	1.484679	0.238926	0.386843	0.837934	5.180838	0.260663	0.221177	0.262183
2.033318	0.831126	0.873532	0.601936	1.370512	1.416386	0.518787	0.252099	0.843961	0.637217	0.300395	0.238926	0.386843	1.193412	0.193019	0.260663	0.221177	0.262183
0.491807	0.547435	0.230648	0.583476	0.732509	0.367301	0.164245	0.050795	1.189687	0.263983	1.484679	0.238926	0.386843	0.837934	0.193019	0.260663	0.221177	3.814134
2.033318	1.976220	0.234153	0.060315	2.157150	1.022979	0.338752	0.216476	0.543180	1.989018	1.484679	0.238926	2.585025	0.837934	0.193019	0.260663	0.221177	0.262183

5.3 Performance Measures

Measuring Accuracy Using Cross-Validation

The K-fold cross-validation method is to randomly divide 3 separate subsets called folds, then train and evaluate the model three times, each time with a different fold to evaluate. Evaluate and train the remaining folds.

Confusion Matrix

The confusion matrix is a technique to evaluate the performance of the model in the classification problem. A confusion matrix is a matrix that represents the amount of data that belongs to a class and predicts which class the data belongs to [8].

True Positive (TP): there are 7147 records in the Positive class, the model classifies the records in the Positive class (correct prediction). True Negative (TN): there are 2230 records in the Negative class, the model classifies the records in the Negative class (correct prediction). False Positive (FP): is the record in the Negative class, but the model classifies 888 records in the Positive class (false prediction) → wrong type 1. False Negative (FN): is the record in the Positive class, but the model classifies 811 records in the Negative class (false prediction) → wrong type 2, see Fig. 6.

Precision

The precision determines that of the records classified by the model into the Positive class, how many records belong to the Positive class. The closer the Precision value

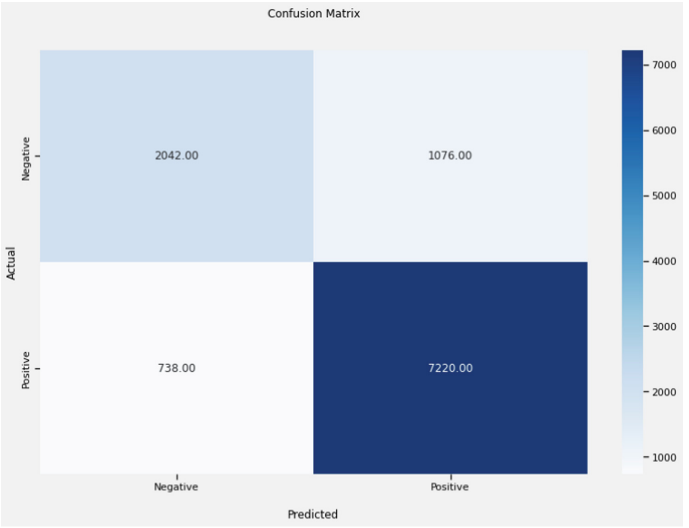


Fig. 5. Confusion matrix

is to 1, the more accurate the model is. The higher the precision, the more accurately classified records are in Eq. (3) [8].

$$precision = \frac{TP}{TP + FP} \tag{3}$$

TP is the number of true positives, and FP is the number of false positives.

Recall

Recall determines how many records actually in the Positive class are correctly classified by the model into the Positive class. The closer the Recall value is to 1, the more accurate the model is. The higher the recall, the more correct information is not missed in Eq. (4) [8].

$$recall = \frac{TP}{TP + FN} \tag{4}$$

FN is of course the number of false negatives.

F1-Score

It is often convenient to combine precision and recall into a single metric called the F1 score, in particular, if you need a simple way to compare two classifiers. The F1 score is the harmonic mean of Precision and Recall in Eq. (5). Whereas the regular mean treats all values equally, the harmonic mean gives much more weight to low values. As a result, the classifier will only get a high F1 score if both recall and precision are high [8].

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 2 \times \frac{precision \times recall}{precision + recall} = \frac{TP}{TP + \frac{FN+FP}{2}} \tag{5}$$

The ROC Curve

ROC (receiver operating characteristic) curve is a commonly used method for precise toxicity measurement used with classification problems, the ROC curve plots the true positive rate (TPR), another name for Recall, according to the false positive rate (FPR), FPR is the proportion of negative samples that are falsely classified as positive and is equal to 1-TNR (true negative rate). TNR is the proportion of negative samples that are correctly classified, also known as specificity [8].

One way to compare classifiers is to measure the area under the curve (AUC). A perfect classifier will have a ROC AUC equal to 1, whereas a purely random classifier will have a ROC AUC equal to 0.5.

5.4 Predicting Loan Repayment

Logistic Regression

Logistic regression is commonly used to estimate the probability that a sample of data belongs to a particular class (for example, the probability that an email is spam). If the estimated probability for a class is greater than 50%, then the model predicts that this sample belongs to that class (called the positive class, labelled "1"); otherwise, the prediction model does not belong to that class (i.e., it is in the negative class, labelled as "0"). So here is a binary classifier [11]. Below is the Cost Function for Logistic Regression with Ridge Penalty, see Table 2, Table 3.

$$\mathcal{L}_{\text{ridge}}(\beta;\lambda) = \|Y - X\beta\|_2^2 + \lambda\|\beta\|_2^2 = \sum_{i=1}^{n}\left(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 \quad (6)$$

This loss function is the traditional sum-of-squares augmented with a penalty. The particular form of the penalty, $\lambda\|\beta\|_2^2$ referred to as the ridge penalty and λ as the penalty parameter.

Decision Tree

Use Classification and Regression Tree (CART) to train Decision Trees. First, the algorithm will divide the training set into two subsets according to the feature k and threshold t_k. The algorithm will search for pair (k, t_k) and generate subsets, Eq. (7) is the cost function that needs to be minimized.

$$J(k, t_k) = \frac{m_{\text{left}}}{m}G_{\text{left}} + \frac{m_{\text{right}}}{m}G_{\text{right}} \quad (7)$$

$$\text{Where}\begin{cases}\frac{G_{\text{left}}}{\text{right}} & \text{measure the doping of the left/right subset}\\ \frac{m_{\text{left}}}{\text{right}} & \text{is the number of samples in the left/right subsets}\end{cases}$$

By default, Gini doping is used, but the entropy phase can also be selected by assigning the criterion hyperparameter to "entropy". The concept of entropy comes from thermodynamics. It's a measure of the chaos of molecules: entropy approaches zero when

the molecules are stationary and ordered. Entropy is often used as a measure of doping [9].

Artificial Neural Networks
Build an Artificial Neural Network (ANN) model by default using the Scikit-learning package and use the accuracy score, precision, recall, confusion matrix, and ROC the see the model's accuracy on the loan dataset, see Table 2 and, Table 3.

Model Comparison

Table 2. Genetic algorithm

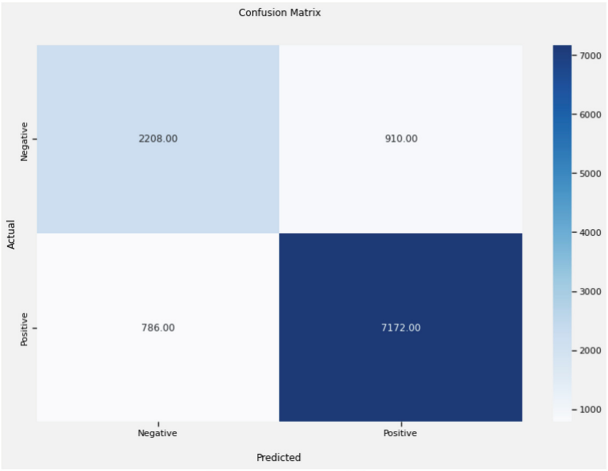
Model	Accuracy	Precision	Recall	F1-score	ROC
Genetic Algorithm + Logistic Regression	0.75686	0.75801	0.97185	0.85172	0.59
Genetic Algorithm + Decision Tree	0.84697	0.88742	0.90136	0.89433	0.80475
Genetic Algorithm + Artificial Neural Networks	0.84381	0.8818	0.90374	0.89264	0.79729

Table 3. Ensemble learning - voting

Model	Accuracy	Precision	Recall	F1-score	ROC
Logistic Regression + Decision Tree	0.84688	0.8974	0.90123	0.89426	0.80669
Logistic Regression + Artificial Neural Networks	0.83371	0.8798	0.91563	0. 90254	0.80234

The results of the selected models: the method of combining the genetic algorithm with the logistic regression model, the genetic algorithm with the decision tree model, the genetic algorithm with the artificial neural network, and classifier models. Voting types will be compared. Table 2 is the value for Precision, Recall, and ROC curves for selected models. The highest accuracy was 88.74%, Recall 90.13% and AUC score was 80.47%. Table 3 Logistic regression with Decision Trees had an accuracy score of 84.68% and the highest ROC rate of 80.66%. Therefore, in our analysis, the Decision Tree model is preferred among the selected model. In Table 3, we used the techniques of voting ensemble learning, we can see an ensemble between the Logistic Regression with Decision Tree model has the value Precision, Recall, F1-Score, AUC-Score is better (Fig. 7).

Receiver Operating Characteristic – Voting:



	precision	recall	f1-score	support
0	0.74	0.71	0.72	3118
1	0.89	0.90	0.89	7958
accuracy			0.85	11076
macro avg	0.81	0.80	0.81	11076
weighted avg	0.85	0.85	0.85	11076

Fig. 6. Ensemble learning - voting logistic regression and decision tree

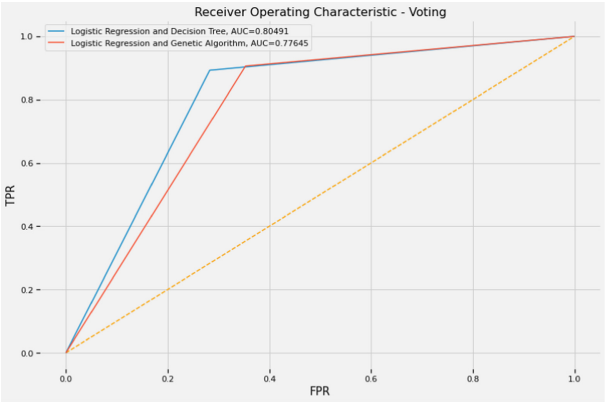


Fig. 7. Receiver operating characteristics for voting ensemble

There are Two Main Reasons to use the Ensemble Learning Model, that is:
Performance: An ensemble can make better predictions and achieve better performance than any single contributing model.

Robustness: An ensemble reduces the spread or dispersion of the predictions and model performance.

Ensembles are used to achieve better predictive performance on a predictive modelling problem than a single predictive model. The result shows that the ensemble learning model Logistic regression with Decision Trees had the highest accuracy score of 84.68% and the highest ROC rate of 80.66% [10].

6 Enhancement and Conclusion

Regarding the use of Machine Learning techniques in Finance, and Banking to achieve high profits, from which we see increasing interest. Research is conducted in the areas of credit scoring, risk management, and default prediction using Machine Learning methodology. Customers who can make loans from financial institutions, Banks and now can make loans directly on mobile devices. It is an opportunity and challenge for financial institutions, and banks to make loan decisions to determine whether customers can repay or not, avoiding the lowest possible default risks. In this study, I proposed a method of ensemble learning to improve the accuracy of predicting whether customers are likely to repay the loan or not? This helps to better understand customer behaviors to improve the prediction of customers' ability to repay loans and avoid default for financial institutions, and banks. The study also explores and understands the properties of loan data that contribute to default risk. Data analysis shows the correlation between the attributes to select the appropriate attributes to train the Machine Learning model. The testing and training data set is applied to Machine Learning algorithms to identify and find an algorithm with the best results. Indicators to evaluate the accuracy of the Machine Learning model include confusion matrix, precision, and recall, which were applied for predicting loan repayment of customers. This study explores the use of Machine Learning algorithms to improve the accuracy of loan repayment predictions. The model will be a public tool for financial institutions and banks to assess credit risk and decide whether to accept a customer's loan or not. The model that worked best in the study was the model that combined Logistic Regression with the Decision Tree with the highest Accuracy of 89.74%, Recall of 90.12%, and AUC score of 80.66%. This is a good result and can be further improved through parameter tuning and attribute selection methods that can bring improvements to the model. The model would probably be better if doing a different data set.

Acknowledgement. This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number DS2022-26-03.

References

1. Wang, B., Liu, Y., Hao, Y., Liu, S.: Defaults assessment of mortgage loan with rough set and SVM. In: 2007 International Conference on Computational Intelligence and Security (CIS 2007), pp. 981–985. IEEE (2007)
2. Reddy, M.J., Kavitha, B.: Neural networks for prediction of loan default using attribute relevance analysis. In: 2010 International Conference on Signal Acquisition and Processing, pp. 274–277. IEEE (2010)

3. Hassan, A.K.I., Abraham, A.: Modeling consumer loan default prediction using Neural Network. In: 2013 International Conference on Computing, Electrical and Electronic Engineering (ICCEEE), pp. 239–243. IEEE (2013)
4. Hamid, A.J., Ahmed, T.M.: Developing prediction model of loan risk in banks using data mining. Mach. Learn. Appl. An Int. J. **3**(1), 1–9 (2016)
5. Turkson, R.E., Baagyere, E.Y., Wenya, G.E.: A machine learning approach for predicting bank credit worthiness. In: 2016 Third International Conference on Artificial Intelligence and Pattern Recognition (AIPR), pp. 1–7. IEEE (2016)
6. Odegua, R.: Predicting bank loan default with extreme gradient boosting. arXiv preprint arXiv:2002.02011 (2020)
7. Sheikh, M.A., Goel, A.K., Kumar, T.: An approach for prediction of loan approval using machine learning algorithm. In: 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 490–494. IEEE (2020)
8. Aurélien, G.: Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow (2019)
9. Jason, B.: Machine Learning Algorithms from Scratch with Python (2016)
10. Jason, B.: Ensemble Learning Algorithms with Python Make Better Predictions with Bagging, Boosting, and Stacking (2021)
11. Natasha, A., Prastyo, D.D., Suhartono. Credit scoring to classify consumer loans using machine learning. In: AIP Conference Proceedings (2019)
12. Jason, B.: Deep Learning with Python Tap The Power of TensorFlow and Theano with Keras (2016)