

Enhancing uncollateralized loan risk assessment accuracy through feature selection and advanced machine learning techniques

Shahrul Nizam Salahudin, Yosza Dasril, Yosy Arisandy

Faculty of Technology Management and Business, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia

Article Info

Article history:

Received Jun 1, 2024

Revised Nov 6, 2024

Accepted Nov 11, 2024

Keywords:

Accuracy

Feature selection

Machine learning

Risk assessment

Stacking

Uncollateralized loan

ABSTRACT

Accuracy in evaluating the risk of credit applications is crucial for lenders, particularly when dealing with unsecured loans. Accuracy can be enhanced by selecting suitable features for a machine learning model. To better identify high-risk borrowers, this study applies an elaborate feature selection technique. This study uses the light gradient boosting machine (LGBM) Classifier model with boosting type gradient boosting decision tree (GBDT) algorithm and $n_estimator$ value 100 for feature selection process. This work uses advanced machine learning techniques namely stacking to improve accuracy model perform. The dataset consists of 307,506 applicants from European lenders who have applied for loans in Southeast Asia. Each applicant is described by 126 different features. Using GDBT algorithm GBDT, 30 best features were selected based on their maximum accuracy compared to another feature. By employing a stacking technique that combines the LGBM, gradient boosting (GB), and random forest (RF) models, and utilizing logistic regression (LR) as the final estimator, an accuracy of 0.99637 was reached. This study demonstrates an improved the accuracy compared to previous research. This discovery indicates that utilizing feature selection and stacking method can provide one of the most precise choices for modelling the binary class classification among the current models.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Yosy Arisandy

Faculty of Technology Management and Business, Universiti Tun Hussein Onn Malaysia

86400 Johor, Malaysia

Email: gp200002@student.uthm.edu.my

1. INTRODUCTION

Each year, a considerable percentage of borrowers with unsecured loan are default [1]. Emphasizing the essential requirement for precise data and dependable early detection models to accurately evaluate default risk [2], machine learning techniques are progressively employed for this assessment to ensure the quality of targets in the dataset [3], [4]. The integration of machine learning and enhancing the algorithm has provided a more nuanced approach [5] to get solutions to global optimization modelling problems than traditional risk evaluation model [6]. However, some outstanding concerns continue to exist, especially concerning feature selection. Despite the completion of feature importance analysis, there is potential for enhancement in determining which features most significantly contribute to loan default.

Feature selection in machine learning is the process of selecting the optimal features for a classification problem in order to increase the classification's accuracy [7]. Recursive feature elimination is a technique for lowering the effect of noisy data and increasing computational performance [8]. The accuracy

of the model is influenced by the number of features that are examined. When a greater number of optimally selected features are examined, the level of accuracy will increase [9]. Pathan *et al.* [10], feature selection techniques, such as those used in gradient boosting decision trees (GBDTs), is useful in identifying important features by taking into account their contribution to the model's performance and dealing with complex relationships in the data, and achieve higher predictive accuracy and avoid being influenced by irrelevant or noisy features [11]. Moreover Several previous studies have found that using feature selection techniques, particularly GBDT, can improve model performance, reduce computational complexity, and improve generalization to new, previously unseen data [12]–[14]. In contrast to previous research that focused on generic GBDT applications, this study explores the features of credit data to identify the most predictive factors of borrower defaults.

The dataset we utilize is particularly compelling, having been explored in multiple studies that demonstrate its relevance and robustness across various research contexts including Chen *et al.* [15] used 104 features in his study as the result of tree features selection and using the integrated method that combines the deep learning framework DeepGBM with CatNN handling sparse categorical data and GBDT2NN handling dense numerical data, thus obtaining the best area under the curve (AUC) value of 0.755832. Tian *et al.* [16] utilized the Pearson correlation coefficient as a method of feature selection was used in the selected feature so that around 80 features were produced which produced the best accuracy of 90.99% using the GBDT model. Feature engineering and comparing features across all models by Mahmudi *et al.* [17] extracted 40 features and found the best accuracy of 98.47% using extreme gradient boosting (Xgboost). XGBoost exhibits the superior efficacy of the XGB classifier, demonstrating a significant capacity to forecast creditworthiness with considerable precision [18]. As an added benefit, this paper utilized XGBoost for modelling but also focus on picking the most optimal features, that have a significant impact on achieving the highest accuracy value by utilized boosting feature selection technique called GBDT embedded method and purpose the stacking approach for model evaluation.

2. METHOD

Figure 1 demonstrates the dataset processing steps in this research. In exploratory data analysis, relevant data is collected and visualized. The data sources and domain knowledge elements used to enhance the dataset should be considered along with the correlation, impacts, and interactions between variables. The second phase in machine learning is data preprocessing, which cleans and organizes raw data for model creation and training. Data preprocessing improves data quality to enable useful insights. In preprocessing, unbalanced data is handled with synthetic minority over-sampling technique (SMOTE). The next step after dataset cleaning is identifying important features for target prediction. GBDTs are an effective machine learning approach for feature importance determination. This study set training sets at 80% and test sets at 20%. In training, the training and validation sets are merged to create a model. Finally, evaluation metrics are utilized to evaluate the risk assessment model.

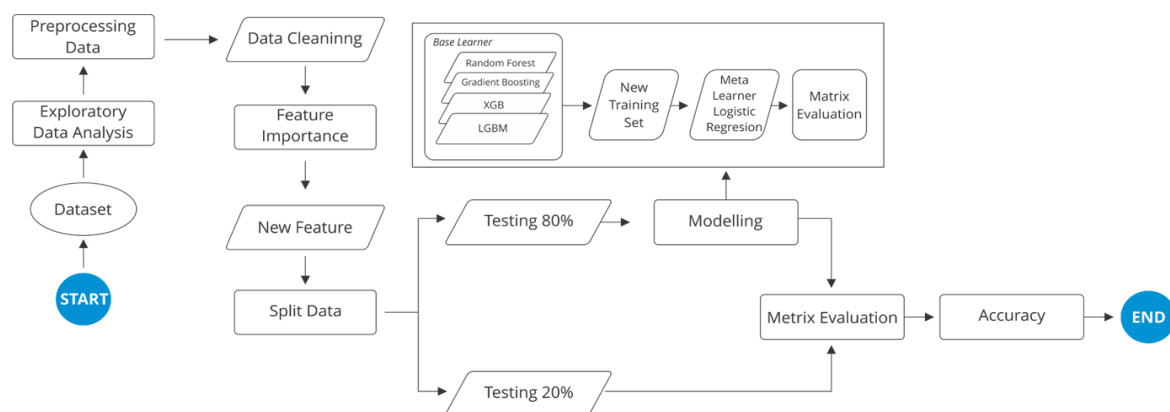


Figure 1. Research flowchart

2.1. Data collection

The dataset provides a comprehensive description of each applicant, consisting of 126 features or columns, encompassing a total of 307,506 applications. The dataset is a compilation of customer data loan

from Kazakhstan, Russia, Vietnam, China, Indonesia, and the Philippines [19]. This thorough set of features encompasses an extensive array of application information, including demographic data, financial condition, and prior loan history. A dataset that is completely lets us do comprehensive analyses and discover significantly regarding the issues that affect loan applications and results.

2.2. Data preprocessing

Data preparation involves numerous essential procedures. These encompass importing the requisite libraries, rectifying missing values, encoding categorical variables, remove outliers, splitting dataset, and executing feature scaling [20]. Furthermore, eliminating outliers and using feature scaling enhance the dataset's balance and representativeness, hence improving model performance and generalization.

2.2.1. Importing libraries and dataset

Importing data into the Python environment constitutes the initial phase of data analysis. The import format for comma separated values (CSV) files, which stands for comma-separated values. This is the format employed by pandas to import local datasets into Python for preprocessing in this research. Subsequently, the libraries utilized for pretreatment and additional data processing were imported. Machine learning projects invariably utilize the NumPy library for the management of vectors and matrices. NumPy encompasses fundamental array data types and operations, including indexing, sorting, reshaping, and elemental functions. SciPy encompasses all numerical code. The widely utilized Panda's library is renowned for its efficacy in managing time series and tabular data structures. Subsequently, there is Matplotlib. The pyplot library is a powerful data visualization and graphical charting package created for Python and NumPy, capable of operating on multiple platforms [21].

2.2.2. Finding missing value and handling

This study used two different methods to handle missing values, which are critical to maintaining data integrity. These methods include the averaging technique to impute missing numeric values and the use of substitute values to fill in missing categorical values. By implementing these techniques, this study ensured that the data set remained as complete as possible, thereby minimizing the impact of missing information on the analysis results [22].

2.2.3. Label encoding

During this stage, category data is converted or encoded into numerical values. Some types of machine learning, like Deep learning, need numerical data to work. It needs to be turned into numbers before category data can be used to fit and test a model. Dummy variables or Label Encoding can be used to solve that problem. In this study, the LabelEncoder method was used to turn categorical data into number values. Additionally, the StandardScaler method is used in the preprocessing step of this study. This improves the performance, interpretability, and resilience of machine learning models trained on the dataset [23].

2.2.4. Remove outlier

The present study addressed outliers utilizing the inter-quartile range (IQR) Score. This works similar to a box plot and z - score in the sense that a threshold IQR value is defined. IQR is the first quartile subtracted from the third quartile. Any point below the threshold IQR is removed [24]. This method identifies and removes data points that fall significantly outside the normal range, helping to improve the quality of the dataset. By managing outliers, the study ensures a more reliable analysis, minimizing the influence of extreme values on model accuracy.

2.2.5 Balancing data

The dataset used for model training contains imbalanced classes. This leads to high variation in performance results, especially in accuracy [25] and specificity rate [26]. Hence, to tackle such a situation, we applied SMOTE oversampling approach to producing reliable and accurate results by reducing the effects of a biased class. In terms of accuracy, Alamsyah et.al suggests that oversampling is preferable to undersampling [27]. Undersampling runs the risk of eliminating some portions of the dataset that include crucial information, perhaps resulting in model overfitting. Conversely, the most effective oversampling approaches are those that generate new data for the minority class instead of essentially duplicating existing data [28].

2.2.6 Splitting dataset

Following data collection and preparation, the dataset is split into two sets, including training and testing. The training set serves as the foundation for training the machine learning model, whereas the testing set is required to evaluate the model's performance. To achieve a fair assessment, a random data split is

performed. The comparison of the percentage of datasets for training and validation 80% and 20% testing. In this study, the Training sets setting by 80% and test set by 20% based on Rosebrock [29].

2.3. Feature selection

This study employs a boosting technique utilizing the embedding method to determine the optimal number of features for modeling purposes. The light gradient boosting machine (LGBM)Classifier is configured with a boosting type of GBDT and a n_estimator value of 100, as detailed in Table 1. Train a model with one feature and compare its performance against the model with all features to determine feature relevance. While Table 2 outline the GBDT feature important process accuracy comparison.

Table 1. Hyperparameters of feature selection techniques

Method	Technique	Hyperparameter
Embedded method	boosting	LGBMClassifier, objective=binary, boosting_type=gbdt, n_estimators=100

Table 2. Feature importance selection

n_features	prop_features	mean_accuracy	mean_roc_auc	mean_fit_time
55	1.000000	0.919211	0.710908	6.944838
54	0.981818	0.919224	0.710332	9.486787
53	0.963636	0.919218	0.710459	13.077004
52	0.945455	0.919218	0.710460	8.646953
51	0.927273	0.919218	0.710460	6.890522
50	0.909091	0.919231	0.710761	10.145021
49	0.890909	0.919218	0.710689	10.490020
48	0.872727	0.919228	0.710718	10.069559
47	0.854545	0.919224	0.710708	7.080661
46	0.836364	0.919221	0.710742	6.779140
45	0.818182	0.919263	0.710969	9.284734
44	0.800000	0.919224	0.709597	8.494219
43	0.781818	0.919231	0.710692	6.491265
42	0.763636	0.919244	0.709787	8.234161
41	0.745455	0.919224	0.710035	9.734142
40	0.727273	0.919250	0.710104	6.893979
39	0.709091	0.919221	0.710098	6.337167
38	0.690909	0.919263	0.710808	9.169578
37	0.672727	0.919205	0.709817	6.813418
36	0.654545	0.919263	0.709824	6.196522
35	0.636364	0.919263	0.709265	9.449198
34	0.618182	0.919267	0.709957	7.074746
33	0.600000	0.919237	0.708952	5.952127
32	0.581818	0.919231	0.710415	8.975714
31	0.563636	0.919208	0.709981	6.181975
30	0.545455	0.919289	0.709048	6.089981
29	0.527273	0.919270	0.709712	8.730765
28	0.509091	0.919276	0.708639	5.783935
27	0.409091	0.919234	0.709885	6.425910

Source: data processing

GBDT is the most popular standard approach for training DT-based models [30]. The algorithm executes training by iteratively starting with a base model, where the next model is going to utilize the mistake obtained in the previous process [31]. This is different from the random forest (RF) model, which utilizes many DT models independently [32]. One characteristic of GBDT is that in the last iterations, the model tends to over-specialize, focusing only on a few features that have a high correlation with the training outcome [33]. After utilizing LGBM with boosting type GBDT, the criteria for maximum accuracy compared to other collections, only 31 of the 126 characteristics, including the target, were picked. The features mentioned above are recognized in the Table 3.

2.4. Modelling

Supervised machine learning and stacking are modeled by several machine learning models Table 4, including RF, logistic regression (LR), gradient boosting (GB), and LGBM, CatBoost (CB), and XGBoost. Stacking approach outperforms the other techniques in terms of yields high performance, not only in terms of classification accuracy [34], outperforms traditional credit scoring models in terms of accuracy and efficiency [35], prediction accuracy but also in precision and recall [36].

Table 3. Feature importance with description and weight

Feature importance	Feature description	Weight	importance_normalized
CREDIT TERM	The length of the payment in months (since the annuity is the monthly amount due)	550	0.183333
DAYS BIRTH	Client's age in days at the time of application	174	0.058000
DAYS ID PUBLISH	How many days before the application did client change the identity document with which he applied for the loan	154	0.051333
AMT GOODS PRICE	For consumer loans it is the price of the goods for which the loan is given (consumer loan)	150	0.050000
DAYS LAST PHONE CHANGE	How many days before application did client change phone	145	0.048333
DAYS REGISTRATION	How many days before the application did client change his registration	130	0.043333
AMT ANNUITY	Loan annuity	120	0.040000
REGION POPULATION REACTIVE	Normalized population of region where client lives (higher number means the client lives in more populated region)	118	0.039333
AMT CREDIT	Credit amount of the loan	114	0.038000
DAYS EMPLOYED	How many days before the application the person started current employment	92	0.030667
DAYS EMPLOYED PERCENT	The percentage of the days employed relative to the client's age	92	0.030667
CREDIT INCOME PERCENT	The percentage of the credit amount relative to a client's income	89	0.029667
AMT REQ CREDIT BUREAU YEAR	Number of enquiries to Credit Bureau about the client one day year (excluding last 3 months before application)	70	0.023333
ANNUITY INCOME PERCENT	the percentage of the loan annuity relative to a client's income	70	0.023333
AMT INCOME TOTAL	Income of the client with standard base currency by Homecredit (Tuananhkk, 2019)	63	0.021000
OCCUPATION TYPE	What kind of occupation does the client have	62	0.020667
OWN CAR AGE	Age of client's car	56	0.018667
NAME FAMILY STATUS	Family status of the client	54	0.018000
REGION RATING	Our rating of the region where client lives with taking city into account (1,2,3)	51	0.017000
CLIENT W CITY			
ORGANIZATION TYPE	Type of organization where client works	50	0.016667
HOUR APPR PROCESS START	Type of organization where client works	48	0.016000
NAME CONTRACT TYPE	Identification if loan is cash or revolving	48	0.016000
CODE GENDER	Gender of the client	44	0.014667
AMT REQ CREDIT BUREAU QRT	Number of enquiries to Credit Bureau about the client 3 month before application (excluding one month before application)	38	0.012667
DEF 30 CNT SOCIAL CIRCLE	How many observations of client's social surroundings defaulted on 30 DPD (days past due)	35	0.011667
NAME EDUCATION TYPE	Level of highest education the client achieved	32	0.010667
FLAG WORK PHONE	Did client provide home phone (1=YES, 0=NO)	31	0.010333
OBS 30 CNT SOCIAL CIRCLE	How many observations of client's social surroundings with observable 30 DPD (days past due) default	27	0.009000
EMERGENCYSTATE MODE	Normalized information about building where the client lives	23	0.007667
FLAG OWN CAR TARGET	Means loan applicant have owns a car (1=YES, 0=NO) Target variable (1 - Client with payment difficulties: he/she had late payment more than X days on at least one of the first Y installments of the loan, 0 - all other cases)	-	-

Source: data processing

Table 4. Hyperparameters of algorithms

Type	Algorithms	Hyperparameter
Supervised machine learning	RF	Random forest classifier, n_estimators=800
	LR	Logistic regression
	GB	Gradient boosting classifier
	LGBM	lgb.LGBM classifier
	CB	Catboost classifier, iterations=5, learning_rate=0.1, loss_function = CrossEntropy
Stacking	XGBoost	XGB classifier
	GB, RF, and LGBM with LR as final estimator	Stacking classifier, estimators = stac,final_estimator = lr

This study the staking method was used by combining GB, RF, and LGBM with LR as final estimator. The research employed LR as the final estimator due to its demonstrated reliability in selecting manageable subsets of indicators [37] and Offers probabilities for potential results, which can be valuable for making informed decisions [38].

2.5. Performance evaluation metrics

In order to evaluate the efficacy of the proposed methodology, many performance metrics have been utilized, including accuracy, precision, recall, and F1-score and it consists of a matrix of four different combinations true negative (TN), false negative (FN), false positive (FP), and true positive (TP) of predicted and actual values on Figure 2 Confusion matrix.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \times 100 \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 - Score = 2 \frac{Recall \times Precision}{Recall + Precision} \quad (4)$$

		Predicted value	
		Negatif (0)	Positive (1)
Actual Value	Negatif (0)	TN	FP Type I Error
	Positive (1)	FN Type II Error	TP

Figure 2. The components of 2×2 confusion matrix

3. RESULTS AND DISCUSSION

In this section, the supervised learning algorithms implemented include LGBM, CB, LR, RF, GB, and XGBoost. Additionally, an ensemble learning algorithm, stacking, is utilized to combine the strengths of multiple models. Each of these algorithms is selected for its effectiveness in handling different data patterns and improving model performance.

3.1. Light gradient boosting machine

According to the LGBM algorithm confusion matrix in Figure 3, there were 111279 instances where the model accurately predicted the positive class, also known as TP. A total of 0 occurrences were recorded in which the model accurately predicted the negative class, also known as TN. The number of occurrences in which the model produced false predictions by categorizing them as positive (Type I error), generally known as FP, is 0. The captured number of instances in which the model generated inaccurate predictions for the negative class, known as Type II error or FN, is 1794. The accuracy value is 0.98413, precision value is reported as 1.0, the recall value as 0.98413, and the F1-score value as 0.99200.

3.2. CatBoost

The number of instances when the model correctly predicted the positive class, known as TP, is 100789, as shown in the confusion matrix CB Figure 4. There were no instances in which the model successfully predicted the negative class, also referred to as TN. The quantity of instances in which the model made an inaccurate prediction of the positive class (Type I error) referred to as FP, is 0. The number of instances in which the model produced inaccurate predictions for the negative class, classified as Type II errors or FN, is 12284. The accuracy number is 0.89136, while the precision 1.0, recall 0.89136, and F1-score values are 0.94256.

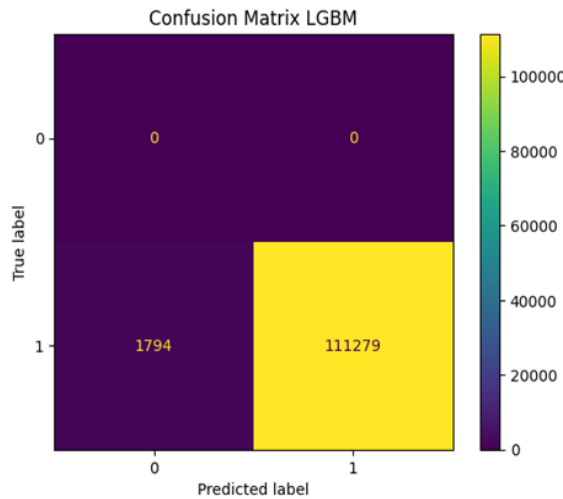


Figure 3. Confusion matrix of LGBM

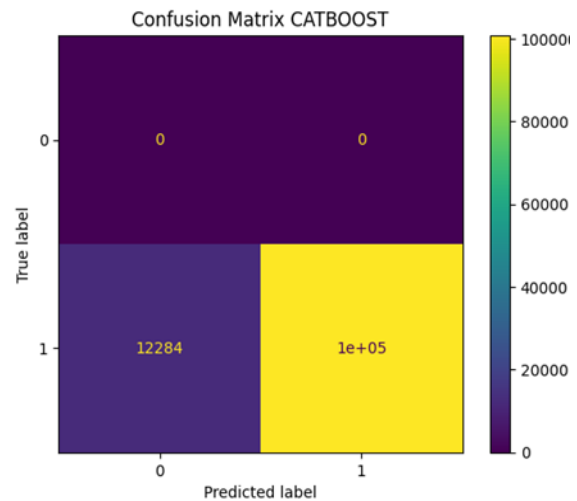


Figure 4. Confusion matrix of CB

3.3. Logistic regression

Figure 5 demonstrates that the confusion metric for LR model’s accuracy in predicting outcomes, categorized by class, and includes the counts of both correct and incorrect predictions. According to the confusion matrix LR, the count of occurrences in which the model accurately predicted the positive class, also known as TP, is 61601. There was a total of 0 cases in which the model accurately predicted the negative class, often known as TN. The number of cases in which the model wrongly predicted the positive class (Type I error), also known as FP, is 0. The count of occurrences in which the model made inaccurate predictions for the negative class, often known as Type II error or FN, is 51472. The accuracy value is 0.54479, the precision value is 1.0, the recall value is 0.54479, and the F1-score value is 0.70533.

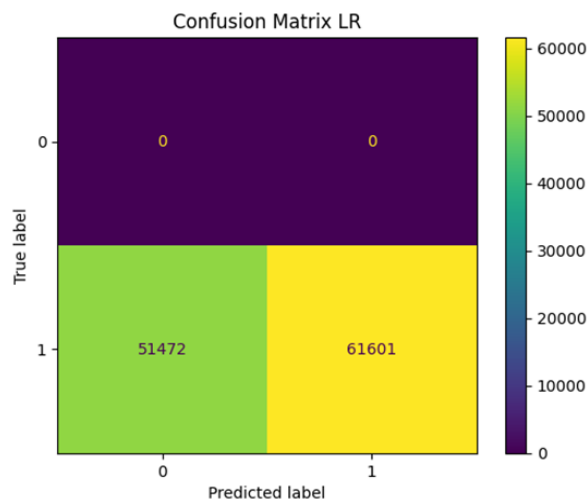


Figure 5. Confusion matrix of LR

3.4. Random forest

Based on the confusion matrix presented in Figure 6, the RF algorithm properly predicted the positive class, referred to as TP, in a total of 110354 occurrences. A cumulative count of 0 instances was observed in which the model successfully predicted the negative class, commonly referred to as TN. The quantity of instances in which the model made an inaccurate prediction of the positive class (Type I error), widely known as FP, is 0. The frequency of instances in which the model produced inaccurate predictions for the negative class, referred to as Type II error or FN, amounts to 2719.

The Accuracy metric is reported as 0.97595, indicating the proportion of correctly classified instances in the dataset. The Precision metric is reported as 1.0, representing the proportion of correctly predicted positive instances out of all instances predicted as positive. The Recall metric is reported as 0.97595, indicating the proportion of correctly predicted positive instances out of all actual positive instances. Lastly, the F1-score metric is reported as 0.98783, which is the harmonic mean of Precision and Recall, providing an overall measure of the model's performance.

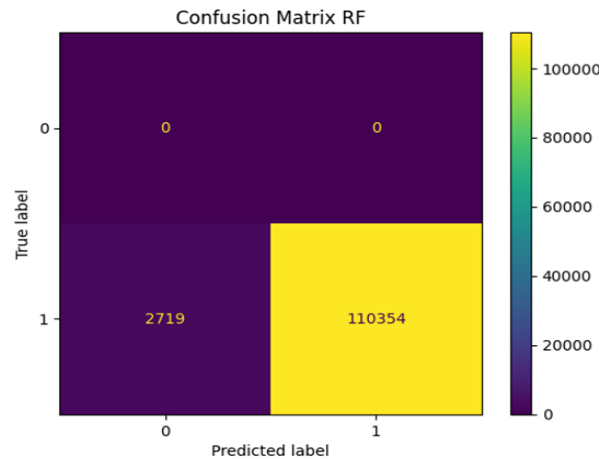


Figure 6. Confusion matrix of RF

3.5. Gradient boosting

According to the confusion matrix depicted in Figure 7, the GB method accurately identified instances belonging to the positive class, denoted as TP, in 106138 instances. A total of 0 occurrences were recorded in which the model accurately predicted the negative class, also known as TN. The number of occurrences in which the model produced an incorrect forecast of the positive class (Type I error), often known as FP, is zero. The occurrence rate of erroneous predictions for the negative class, often known as Type II mistake or FN, is 6935.

The accuracy measure is provided as 0.93867, which signifies the proportion of instances in the dataset that have been properly categorized. The precision measure is provided as 1.0, indicating the ratio of accurately predicted positive instances to the total number of instances projected as positive. The recall measure is reported as 0.93867, denoting the ratio of accurately anticipated positive instances to the total number of actual positive instances. Finally, the F1-score metric is shown as 0.96836, representing the harmonic mean of precision and recall. This metric serves as a comprehensive assessment of the model's performance.

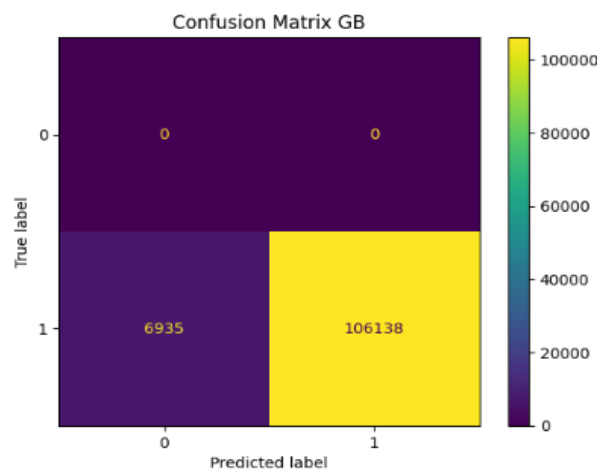


Figure 7. Confusion matrix of GB

3.6. XGBoost

The number of instances where the model correctly predicted the positive class, commonly known as TP, was 111398, as shown in the original XGBoost algorithm confusion matrix Figure 8. A cumulative count of 0 instances was observed in which the model successfully predicted the negative class, commonly referred to as TN. The quantity of instances in which the model made inaccurate predictions by classifying them as positive (Type I error), commonly referred to as FP, is 0. The quantity of instances in which the model produced erroneous forecasts for negative class, commonly referred to as Type II error or FN, is recorded as 1675. The precision value is reported as 1.0, the Recall value as 0.98519, and the F1-score value as 0.99254.

The highest count of accuracy compared to the LR, XGBoost, RF, GB and CB algorithms, XGBoost has a significant impact on the achieved accuracy 0.98519, which is the best accuracy when compared to the other five algorithms. The calculation results for accuracy, precision, recall, and F1-score are presented in Table 5.

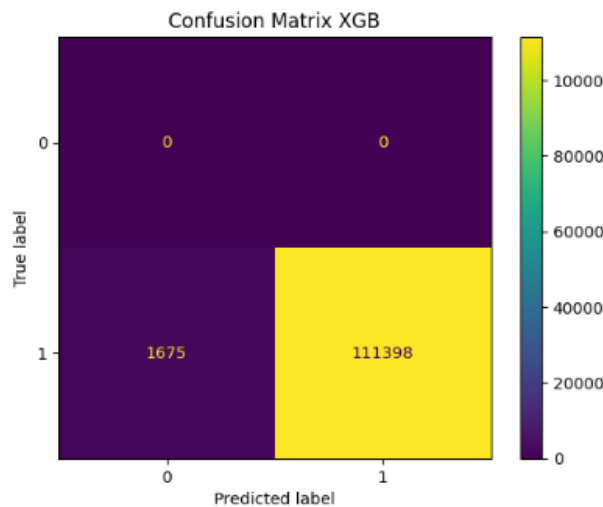


Figure 8. Confusion matrix of XGBoost

Table 5. Summary of confusion metrics

Algorithm	Accuracy	Precision	Recall	F1-score
LGBM	0.98413	1.0	0.98413	0.99200
CB	0.89136	1.0	0.89136	0.94256
LR	0.54479	1.0	0.54479	0.70533
RF	0.97595	1.0	0.97595	0.98783
GB	0.93707	1.0	0.93707	0.96751
XGBoost	0.98519	1.0	0.98519	0.99254

According to the data shown in Table 1. The XGBoost algorithm demonstrates the highest level of accuracy, with a notable accuracy rate of 98.52%. XGBoost is a GB algorithm that uses DT as weak learners to create a strong learner, which can handle both numerical and categorical features effectively, making it suitable for a wide range of datasets. This is particularly useful for the home credit dataset, which contains a mix of numerical and categorical features with large dataset.

One of the notable strengths of XGBoost is its efficacy in managing imbalanced datasets. The home credit dataset exhibits an imbalance, characterized by a relatively low proportion of defaulters. This study employed strategies such as SMOTE to address class imbalance in the dataset and enhance the efficacy of GB algorithms, specifically XGBoost. The present study exclusively employs the SMOTE technique as a means to address the issue of imbalanced datasets, without using the ADASYN (adaptive synthetic) approach utilized by Mahmudi *et al.* [17]. Nevertheless, the research model demonstrates improved accuracy in its outcomes than Mahmudi *et al.* [17] finding using XGBoost.

Moreover, XGBoost is renowned for its notable efficiency and scalability, enabling it to effectively manage extensive datasets including a substantial number of features. The home credit dataset, with 30 features and 307,506 captures, offers significant advantages. In conclusion, the notable accuracy of XGBoost in binary class classification utilizing the dataset may be ascribed to its capacity to manage mixed feature

types, its efficacy in addressing imbalanced datasets, and its efficiency and scalability in managing extensive datasets with numerous features. So, XGBoost is a highly recommended method for binary class classification in the context of assessing default risk, particularly when applied to home credit datasets.

3.7. Stacking

LR was used as the final estimator in this research because its reliability is most evident in its capacity to select manageable subsets of indicators [37] and Provides probabilities for outcomes, which can be useful for decision-making [38]. LR is a statistical modeling technique used for predictive analysis. This methodology is employed to elucidate the association between discrete binary variables. LR exhibits sensitivity to the presence of multivariate collinearity among the independent variables within the model, wherein the influence of a single variable can significantly impact the other variables. When confronted with a multitude of factors, the resulting performance may not meet expectations. The remark elucidates the rationale for the very modest accuracy outcome of LR, which recorded a value of 0.54479, in the context of binary classification inside this research. This performance was comparative in comparison to the results achieved by alternative methodologies employed in this investigation.

The outcomes of employing the Stacking technique, which integrates LGBM, RF, and GB models, with LR serving as the ultimate estimator, encompass the following metrics: accuracy: 0.99637, precision: 1.0, recall: 0.99637, and F1-score: 0.99818 display in Table 6. The outputs of the confusion matrix resulting from the integration of LGBM, RF, and GB models in the stacking model are depicted in Figure 9.

Table 6. Confusion matrix stacking

Method	Accuracy	Precision	Recall	F1-score
Stacking	0.99637	1.0	0.99637	0.99818

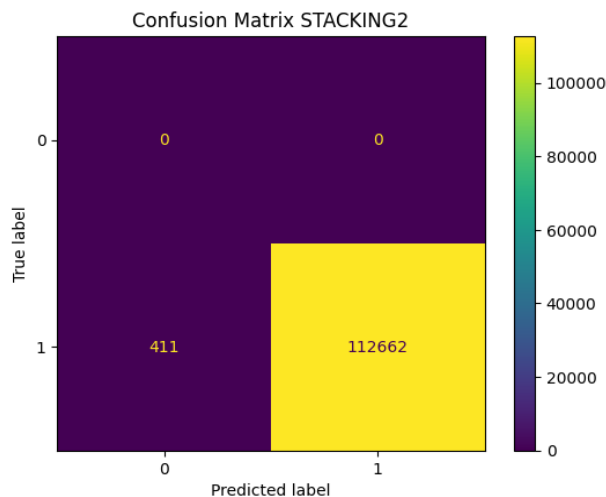


Figure 9. Confusion matrix for stacking combines GB, RF, and LGBM

The stacking method's outstanding accuracy demonstrates its promise as reliable instruments for assessing uncollateralized loan risk. This has important ramifications for financial institutions looking to improve their model accuracy. Furthermore, the use of GBDT with $n_{estimator}=100$ to select feature importance in classification affects accuracy.

Table 7 presents a comparison of the accuracy achieved using GBDT with $n_{estimator}=100$ against previous studies. As demonstrated in Table 7, our model has the highest accuracy of 99.64%, which is much higher than prior research' accuracies ranging from 75% to 98%. This improvement can be due to GBDT's stringent feature selection method, which ensures that the model contains only the most useful features.

This study employed strategies such as SMOTE to address class imbalance in the dataset and enhance the efficacy of GB algorithms, specifically XGBoost. The present study exclusively employs the SMOTE technique without using the ADASYN (Adaptive Synthetic) approach utilized by Mahmudi [17]. Nevertheless, the research model demonstrates improved accuracy in its outcomes. LR was used as the final estimator in stacking research methode because its reliability is most evident in its capacity to select

manageable subsets of indicators [37] and provides probabilities for outcomes, which can be useful for decision-making [38]. The results demonstrate that applying the LGBM classifier model with the GBDT boosting type and `n_estimator` set to 100, together with ensemble learning approaches like XGBoost and stacking, achieved considerable gains in accuracy for assessing uncollateralized loan risk. The model performed optimally with a selection of 30 features, verifying the hypothesis that effective feature selection might improve predicted accuracy.

Table 7. Comparing the accuracy of this study in relation to prior research

Research	Number of feature selection	Feature metode	Accuracy
[15]	104	Tree features selection and integrated method	DeepGBM (AUC value 0.755832)
[16]	80	Pearson correlation coefficient	GBDT 90.99%
[17]	40	Feature engineering and comparing features	XGboost 98.47%
This paper	30	Boosting_type = gbdt, n_estimators = 100	XGboost 98.52%, Stacking (GB, RF, LGBM with LR final estimator) 99.64%

4. CONCLUSION

The purpose of this study is to improve the accuracy by finding optimum number of features of unsecured loan risk assessment. This research confirmed that employing the LGBM with GBDT boosting algorithm and a `n_estimator` value of 100, along with XBoost and stacking model, showcases remarkable performance that outshines models in previous research. These models are extremely accurate, making them important tools for financial firms seeking improved forecasting capabilities and providing a benchmark for future studies aiming to optimize feature selection in classification tasks. The stacking with LR as the final estimator outperforms individual artificial intelligence and statistical methodologies when attempting to identify the best machine learning techniques for predicting default risk. The incorporation of the stacking approach is consistent with contemporary advances in machine learning and data science, acting as a reference for a more precise credit risk calculation model.

The present study, however, makes a valuable contribution to the field of predicting default risk in uncollateralized loan services. The proposed model, by effectively identifying borrowers with a high repayment rate, lenders can minimize the risk of default and improve the quality of their loan portfolios. An additional assessment tool can be used by the credit worthiness assessor to evaluate the quality of registering borrowers. Additionally, investors can utilize it as a reference to choosing investment projects with lower risk. The research lays a solid groundwork for further investigation, with the dataset having the potential to be widely used in future studies on predicting default risk. This research has important practical implications for the lending industry, especially in the areas of risk management and investment decision making. The study also emphasizes the significance of feature selection in enhancing the performance of default risk prediction models, providing valuable guidance to lenders and uncollateralized lending platforms in credit decision-making.

This study raised several questions that warrant further investigation. A key area for further research is to determine whether the modelling and feature selection strategies identified in this study can be adapted effectively for multiclass classification problems instead of being limited to binary classification. Exploring this adaptation could enhance the generalisability of the models and broaden their application across a wider range of classification challenges.

REFERENCES




- [1] Z. Li, K. Li, X. Yao, and Q. Wen, "Predicting prepayment and default risks of unsecured consumer loans in online lending," *Emerging Markets Finance and Trade*, vol. 55, no. 1, pp. 118–132, 2019, doi: 10.1080/1540496X.2018.1479251.
- [2] J. P. Noriega, L. A. Rivera, and J. A. Herrera, "Machine learning for credit risk prediction: a systematic literature review," *Data*, vol. 8, no. 11, pp. 111–138, 2023, doi: 10.3390/data8110169.
- [3] L. Barbaglia, S. Manzan, and E. Tosetti, "Forecasting loan default in europe with machine learning," *Journal of Financial Econometrics*, vol. 21, no. 2, pp. 569–596, Mar. 2023, doi: 10.1093/jfinec/nbab010.
- [4] B. Ligar, S. Madenda, S. Mardjan, and T. Kusuma, "Design of a traceability system for a coffee supply chain based on blockchain and machine learning," *Journal of Industrial Engineering and Management*, vol. 17, no. 1, pp. 151–167, Feb. 2024, doi: 10.3926/jiem.6256.
- [5] Y. Arisandy, Y. Bin Dasril, S. N. Bin Salahudin, M. A. Muslim, A. Adnan, and G. K. Wen, "Buy now pay later services on generation z: exploratory data analysis using machine learning," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 11, pp. 4194–4204, 2023.
- [6] Y. Dasril, G. K. Wen, N. Bin Bujang, and S. N. Salahudin, "New approach on global optimization problems based on meta-heuristic algorithm and quasi-Newton method," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 5, pp. 5182–5190, 2022, doi: 10.11591/ijece.v12i5.pp5182-5190.

- [7] A. A. Alhussan *et al.*, “A binary waterwheel plant optimization algorithm for feature selection,” *IEEE Access*, vol. 11, pp. 94227–94251, 2023, doi: 10.1109/ACCESS.2023.3312022.
- [8] M. Awad and S. Fraihat, “Recursive Feature elimination with cross-validation with decision tree: feature selection method for machine learning-based intrusion detection systems,” *Journal of Sensor and Actuator Networks*, vol. 12, no. 5, p. 67, 2023, doi: 10.3390/jsan12050067.
- [9] Y. Dasril, Y. Arisandy, and S. N. Salahudin, “Home credit default risk assessment using embedded feature selection and stacking ensemble technique,” *Journal of Numerical Optimization and Technology Management*, vol. 1, no. 2, pp. 59–68, 2023, doi: 10.0000/jnotm.0000.00.00.000.
- [10] M. S. Pathan, A. Nag, M. M. Pathan, and S. Dev, “Analyzing the impact of feature selection on the accuracy of heart disease prediction,” *Healthcare Analytics*, vol. 2, Nov. 2022, doi: 10.1016/j.health.2022.100060.
- [11] L. Ghoulmi and M. E. A. Benkechache, “Feature selection based on machine learning algorithms: a weighted score feature importance approach for facial authentication,” in *2022 3rd International Informatics and Software Engineering Conference (IISEC)*, Dec. 2022, pp. 1–5. doi: 10.1109/IISEC56263.2022.9998240.
- [12] P. V. Balachandran, D. Xue, J. Theiler, J. Hogden, J. E. Gubernatis, and T. Lookman, “Importance of feature selection in machine learning and adaptive design for materials,” in *Springer Series in Materials Science*, 2018, vol. 280, pp. 59–79, doi: 10.1007/978-3-319-99465-9_3.
- [13] M. Yousef, M. D. Saçar Demirci, W. Khalifa, and J. Allmer, “Feature selection has a large impact on one-class classification accuracy for micrnas in plants,” *Advances in Bioinformatics*, vol. 2016, pp. 1–6, Apr. 2016, doi: 10.1155/2016/5670851.
- [14] M. R. Islam, A. A. Lima, S. C. Das, M. F. Mridha, A. R. Prodeep, and Y. Watanobe, “A comprehensive survey on the process, methods, evaluation, and challenges of feature selection,” *IEEE Access*, vol. 10, pp. 99595–99632, 2022, doi: 10.1109/ACCESS.2022.3205618.
- [15] X. Chen, Z. Liu, M. Zhong, X. Liu, and P. Song, “A deep learning approach using deepgbm for credit assessment,” in *Proceedings of the 2019 International Conference on Robotics, Intelligent Control and Artificial Intelligence*, Sep. 2019, pp. 774–779. doi: 10.1145/3366194.3366333.
- [16] Z. Tian, J. Xiao, H. Feng, and Y. Wei, “Credit risk assessment based on gradient boosting decision tree,” *Procedia Computer Science*, vol. 174, pp. 150–160, 2020, doi: 10.1016/j.procs.2020.06.070.
- [17] H. Mahmudi, R. Bhargava, and R. Das, “Evaluation of gradient boosting algorithms on balanced home credit default risk,” in *2022 International Conference on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT)*, Oct. 2022, pp. 1–6. doi: 10.1109/TQCEBT54229.2022.10041584.
- [18] A. Mukhanova *et al.*, “Forecasting creditworthiness in credit scoring using machine learning methods,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 5, pp. 5534–5542, Oct. 2024, doi: 10.11591/ijece.v14i5.pp5534-5542.
- [19] Tuananhkk, “More domain knowledge from former home credit analyst,” *Kaggle*. 2019. [Online]. Available: <https://www.kaggle.com/competitions/home-credit-default-risk/discussion/63032>. Accessed: 15-Jan-2023]
- [20] S. Bouhissin, N. Sael, F. Benabbou, and A. Soultana, “Enhancing machine learning algorithm performance through feature selection for driver behavior classification,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 35, no. 1, pp. 354–365, Jul. 2024, doi: 10.11591/ijeecs.v35.i1.pp354-365.
- [21] K. Dheyaa Ismael and S. Irina, “Face recognition using viola-jones depending on python,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 3, pp. 1513–1521, Dec. 2020, doi: 10.11591/ijeecs.v20.i3.pp1513-1521.
- [22] K. Seu, M.-S. Kang, and H. Lee, “An intelligent missing data imputation techniques: A review,” *JOIV: International Journal on Informatics Visualization*, vol. 6, no. 1–2, pp. 278–283, 2022.
- [23] M. M. Ahsan, M. A. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, “Effect of data scaling methods on machine learning algorithms and model performance,” *Technologies*, vol. 9, no. 3, p. 52, 2021, doi: 10.3390/technologies9030052.
- [24] H. P. Vinutha, B. Poornima, and B. M. Sagar, “Detection of outliers using interquartile range technique from intrusion dataset,” in *Information and decision sciences: Proceedings of the 6th international conference on ficta*, 2018, pp. 511–518.
- [25] A. J. Barid, Hadiyanto, and A. Wibowo, “Optimization of the algorithms use ensemble and synthetic minority oversampling technique for air quality classification,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 33, no. 3, pp. 1632–1640, 2024, doi: 10.11591/ijeecs.v33.i3.pp1632-1640.
- [26] N. Santoso, W. Wibowo, and H. Himawati, “Integration of synthetic minority oversampling technique for imbalanced class,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, no. 1, pp. 102–108, 2019, doi: 10.11591/ijeecs.v13.i1.pp102-108.
- [27] N. Alamsyah, B. Budiman, T. Parama Yoga, and R. Y. Rakhman Alamsyah, “A stacking ensemble model with SMOTE for improved imbalanced classification on credit data,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 22, no. 3, pp. 657–664, Feb. 2024, doi: 10.12928/telkomnika.v22i3.25921.
- [28] A. Khalaf Hamoud *et al.*, “A prediction model based machine learning algorithms with feature selection approaches over imbalanced dataset,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 28, no. 2, pp. 1105–1116, Nov. 2022, doi: 10.11591/ijeecs.v28.i2.pp1105-1116.
- [29] A. Rosebrock, “Deep Learning for Computer Vision with Python(ImageNet),” *PyImageSearch*, 2017.
- [30] F. Fu, J. Jiang, Y. Shao, and B. Cui, “An experimental evaluation of large scale GBDT systems,” *arXiv preprint arXiv:1907.01882*, 2019.
- [31] W. Xu, L. Ning, and Y. Luo, “Wind speed forecast based on post-processing of numerical weather predictions using a gradient boosting decision tree algorithm,” *Atmosphere*, vol. 11, no. 7, p. 738, Jul. 2020, doi: 10.3390/atmos11070738.
- [32] X. Zhou, P. Lu, Z. Zheng, D. Tolliver, and A. Keramati, “Accident prediction accuracy assessment for highway-rail grade crossings using random forest algorithm compared with decision tree,” *Reliability Engineering & System Safety*, vol. 200, 2020, doi: 10.1016/j.res.2020.106931.
- [33] A. Ghods and D. J. Cook, “A survey of techniques all classifiers can learn from deep networks: Models, optimizations, and regularization,” *arXiv preprint arXiv:1909.04791*, 2019.
- [34] M. Munsarif, M. Sam’an, and S. Safuan, “Peer to peer lending risk analysis based on embedded technique and stacking ensemble learning,” *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 6, pp. 3483–3489, Dec. 2022, doi: 10.11591/eei.v11i6.3927.
- [35] Y. Xia, C. Liu, B. Da, and F. Xie, “A novel heterogeneous ensemble credit scoring model based on bstacking approach,” *Expert Systems with Applications*, vol. 93, pp. 182–199, 2018, doi: 10.1016/j.eswa.2017.10.022.
- [36] W. Yin, B. Kirkulak-Uludag, D. Zhu, and Z. Zhou, “Stacking ensemble method for personal credit risk assessment in Peer-to-Peer lending,” *Applied Soft Computing*, vol. 142, p. 110302, Jul. 2023, doi: 10.1016/j.asoc.2023.110302.




- [37] C. J. Greenwood *et al.*, "A comparison of penalised regression methods for informing the selection of predictive markers," *PLOS ONE*, vol. 15, no. 11, p. e0242730, Nov. 2020, doi: 10.1371/journal.pone.0242730.
- [38] W. Liu, S. Zeng, G. Wu, H. Li, and F. Chen, "Rice seed purity identification technology using hyperspectral image with LASSO logistic regression model," *Sensors*, vol. 21, no. 13, p. 4384, Jun. 2021, doi: 10.3390/s21134384.

BIOGRAPHIES OF AUTHORS






Shahrul Nizam Salahudin    is a senior lecturer at the Tun Hussein Onn University in Malaysia with a field category in Economic, business and management. Areas of research interest in organisational studies, finance, management and economic. Author contribution: conceptualization, methodology and review. He can be contacted at email: shahrulns@uthm.edu.my.



Yosza Dasril    is a senior lecturer at the Tun Hussein Onn University in Malaysia with a field of specialization in Mathematical Programming and Optimization. Areas of research interest in Fuzzy decision-making computational mathematics and optimization. Author contribution: supervision and validation. He can be contacted at email: yosza@uthm.edu.my.



Yosy Arisandy    is a researcher in management technology at the Faculty of Management and Business Technology, Universiti Tun Hussein Onn Malaysia, research subject of interest are risk management, computer science, and big data analysis. Author contribution: data processing, writing original draft and editing. She can be contacted at email: gp200002@student.uthm.edu.my.