Research on the Prediction Method for Personal Loan Default Based on Two-Layer Stacking Ensemble Learning Model

Zhirui Ma[1] and Qinglie Wu[1,2(✉)]

[1] School of Economics and Management, Southeast University, Nanjing 211189, China
wql@seu.edu.cn

[2] Jiangsu Academy of Smart Industries and Digitalization, Nanjing 210031, China

Abstract. Accurate identification of loan risks to ensure the interests of financial institutions is the core of intelligent risk control. It has become an important research area to accommodate the requirements of Internet financial platforms for processing large amounts of high-latitude user data by using machine learning algorithms to build loan default prediction models. In this paper, we propose a two-layer model based on Stacking ensemble learning algorithm for personal loan default prediction, which uses LightGBM, Adaboost, XGBoost and Gradient boosting as the primary classifiers and random forest as the secondary classifier. The prediction effect of the model is verified on the personal loan default dataset of Alibaba Cloud Tianchi platform. Experimental results revealed that the Stacking ensemble learning model significantly outperforms four single algorithm model in five evaluation metrics: accuracy, precision, recall, F1 score, and AUC, with a prediction accuracy of 82.03% for the test set. Compared with the single algorithm model, the proposed Stacking ensemble learning model has better generalization ability and prediction performance in personal loan default prediction.

Keywords: Loan default · stacking algorithm · ensemble learning · two-layer stacking model · prediction method

1 Introduction

In recent years, major financial platforms have launched multiple types of online personal loan products under the guidance of macro policies, and their approval process is easier than that of commercial banks. With the expanding scale of personal credit business, the risks borne by the financial platforms have continued to increase. As for the default risk of online personal loan products, it becomes an urgent issue to construct effective loan default prediction and evaluation system. Traditional commercial banks' financial risk control review lenders' personal information and loan history manually, however, this approach is difficult to adapt to the large amount of high-latitude user data processing requirements of Internet financial platforms 1. By using machine learning algorithms to establish loan default prediction models can provide support for loan issuance decisions, which not only can satisfy massive user credit needs, but also can effectively reduce the

J. Yen et al. (Eds.): ICBIS 2023, AHCS 14, pp. 1099–1110, 2023.
https://doi.org/10.2991/978-94-6463-198-2_113

bad debt costs arising from loan defaults while expanding the credit business of financial institutions.

Research on loan default prediction using a single machine learning algorithm has mainly improved on high-performance classifier. Malekipirbazari et al. proposed a random forest (RF) based classification method to predict lender status 2. Ampountolas et al. tested the effectiveness of the random forest algorithm for lender classification 3. Chen et al. built a credit scoring model to assess risk using logistic regression algorithm 4. Ruiz et al. used logistic regression (LR) and support vector machine (SVM) models to identify lender credit 5. Dushimimana et al. evaluated the effectiveness of logistic regression, decision tree and random forest in classifying loan defaults using cross-validation methods 6. Yu et al. proposed a novel two-weight fuzzy approximation support vector machine for credit risk analysis 7. Munkhdalai et al. built an adaptive PIA-Soft regression model to identify lender defaults 8. Arora et al. validated the stability of feature evaluation using RF, SVM, KNN and NB classifiers 9.

Due to the drawbacks of traditional machine learning models, some scholars have tried to build prediction models by ensemble method, and numerous studies have shown that ensemble learning algorithm can better perform classification prediction. Yao et al. built a hybrid RF-SVM model using bagging ensemble learning algorithm and verified the potential of the model in the field of financial prediction 10. Luo et al. proposed a bagging ensemble learning based prediction model for financial credit evaluation and verified the effectiveness of the model 11.

The Stacking ensemble learning algorithm has also been widely studied and applied by many scholars in the recent years. Gyamerah et al. used AdaBoost and KNN algorithms to construct Stacking models and verified the stability and validity of the models on the test dataset 12. Li et al. built Stacking models based on DT, LR, NB, and SVM and evaluated the model performance 13.

In this paper, we propose a two-layer model based on Stacking ensemble learning algorithm for personal loan default prediction. To improve the generalization ability of the model, we used LightGBM, Adaboost, XGBoost and Gradient boosting as the primary classifiers and random forest as the secondary classifier. The predictive efficacy of each model was assessed by calculating five metrics: accuracy, recall, F1 score, and AUC.

2 Stacking Ensemble Learning Model

There are two general approaches to improve the classification effect and generalization ability of a single algorithm model. One is to optimize the parameters of the model, and the other is to carry out feature engineering processing of the dataset. In addition, a stacking model can be obtained by fusing single models of different algorithms to improve the prediction effect 14. Stacking is an ensemble learning method that combines different single classifiers in a certain way to form a strong classifier, and the basic idea is to combine the prediction results output by different primary single classifiers and input them into the secondary classifier, then use the secondary classifier to output the prediction results. The prediction accuracy and generalization ability of the model can be further enhanced by the Stacking ensemble learning method.

2.1 Principle of Stacking Ensemble Learning Algorithm

To avoid the problem of weak generalization ability caused by too small a test set division ratio, it is generally necessary to introduce the K-fold cross-validation method for training in the process of Stacking ensemble learning algorithm application 15. Taking the two-layer Stacking algorithm as an example, assuming that the training set is D, the test set is T, the first layer of the model has four primary classifiers M1, M2, M3, and M4, and the secondary classifier in the second layer is M5, and the five-fold cross-validation is used, the specific steps of the algorithm are as follows:

1. Slice the training set D into 5 equal parts to get the data set $D_i, i \in \{1, 2, 3, 4, 5\}$.
2. In the first layer of the prediction model, the four primary classifiers are trained using five-fold cross-validation. Taking M1 as an example, the 1-part dataset in $D_i, i \in \{1, 2, 3, 4, 5\}$ is used as the test set in turn, and the remaining 4 parts are used as the training set for training to get the prediction dataset $a_{1i}, i \in \{1, 2, 3, 4, 5\}$ on the training set and merge them vertically to $A1$, and then the prediction dataset $b_{1i}, i \in \{1, 2, 3, 4, 5\}$ on the test set T is obtained and averaged to get $B1$.
3. After all the 4 primary classifiers are trained and predicted, the matrix of predicted datasets ($A1, A2, A3, A4$) of the 4 primary classifiers on the training set is used as the training set of the secondary classifiers for training.
4. The prediction dataset matrix ($B1, B2, B3, B4$) obtained from the 4 primary classifiers on the original test set T is used as the test set of the secondary classifier for prediction, and the final prediction results are output.

2.2 Two-Layer Stacking Ensemble Learning Model Design

In this paper, we construct four single algorithm models, optimize the parameters of the models by Grid Search method, and then integrate the four models with Stacking algorithm after parameter tuning. The model follows a two-layer structure, with the first layer 's primary classifiers based on Boosting algorithm: LightGBM, Adaboost, XGBoost, and Gradient Boosting. Boosting algorithm is training iteratively through weak classifier (usually decision tree) to obtain the optimal model, which has the advantages of good training effect and less overfitting 16. In the second layer of the model, a relatively simple random forest classifier is adopted as the secondary classifier in order to prevent overfitting. The framework of the two-layer Stacking ensemble learning model is shown in Fig. 1.

3 Feature Engineering

Feature engineering is the process of extracting effective features from the original data set to improve the accuracy of the model computation results 17. In predictive models applying machine learning methods, feature engineering is considered critical to model fitting and classification 18. The feature engineering covered in this paper includes data preprocessing, feature transformation, data normalization and feature selection.

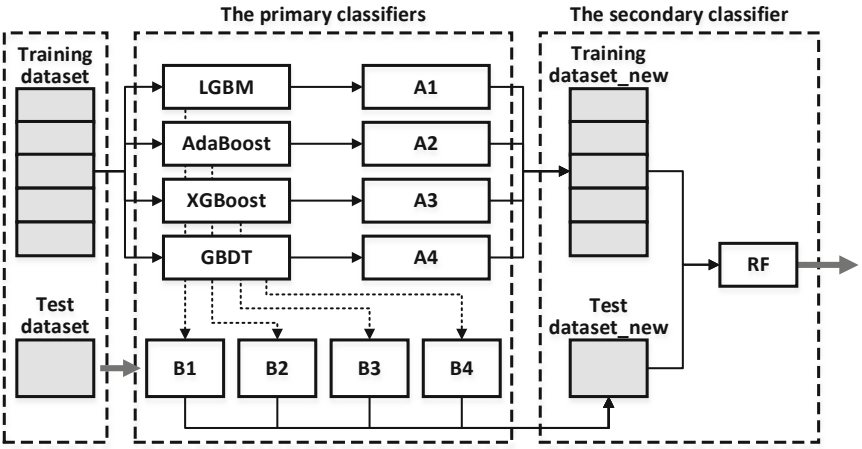

Fig. 1. The framework of the two-layer Stacking ensemble learning model

3.1 Dataset Overview

The data in this paper comes from the personal loan default dataset of Alibaba Cloud Tianchi platform. The original dataset contains 1 million pieces of data with 47 features, including 15 features of anonymous user behavior from n0 to n14. The data type in the original dataset contains int64, float64 and object.

3.2 Data Pre-processing

Due to the diversity of data type and the absence of individual feature data, which may affect the accuracy and convergence speed of the models, preprocessing of the original data is required. The data preprocessing in this paper includes the extraction and processing of default values, feature transformation of category features and normalization of the training set data.

3.2.1 Default Value Processing

Generally, there are several methods to deal with default values in dataset. One is to delete feature with default values directly. The second is to delete samples with default values. The third is to populate the default values of samples. In this paper, the missing rate and the attributes of features are combined to select the default value solution.

The features with default values in the original dataset are counted, and the missing rate of each feature is calculated and plotted in a histogram, as shown in Fig. 2. A total of 22 features were found to have default values, starting with the deletion of 6 features with a missing rate of less than 0.1%. The missing rate of the remaining 16 features ranged from 4% to 9%, and the data type of these anonymous features was int value type and dispersed, so the missing values of these features were filled in with the mode.

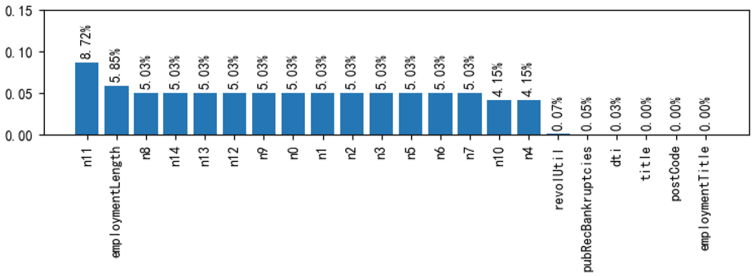

Fig. 2. The missing rate ranking of the features

3.2.2 Feature Transformation

Since the prediction model is built based on the sklearn, it is necessary to convert the data of object type in the original dataset to numeric type.

In this paper, we convert object type features with gradient relationships to int value type, split the years and months in the two features of "issueDate" and "earliesCreditLine" into four new features, and convert their features to int value type.

3.2.3 Data Normalization

Since algorithms such as Gradient Boosting and Adaboost involved in Stacking ensemble learning model need to solve the optimal solution by gradient descent, the training set data are normalized by normalization which can improve the convergence speed and accuracy of the model, thus improving the model performance 19. Normalization refers to the scaling of a column of numerical features in the training set to between 0 and 1. The scaling method is as follows:

$$x_i\prime = \frac{x_i - \min(x)}{max(x) - min(x)}, i = 1, 2, \cdots, n \tag{1}$$

x_i is the i-th feature in the feature set x. There are n features, $x_i\prime$ is the value of x_i after scaling, $max(x)$ and $min(x)$ are the maximum and minimum values of the features, respectively.

3.2.4 Sampling Method

As a dichotomous classification problem, the original data set training model is susceptible to sample imbalance. The distribution of the target variable "isDefault" shows that most of the data in the sample are non-default data, accounting for 80.05%, while the default data account for 19.95%, which shows a serious data imbalance. If this dataset is used directly, the classifiers may have difficulty in extracting the laws from the dataset due to the data imbalance, resulting in failure to meet the classification requirements, and even if a classification model is obtained, it may easily lead to overfitting problems due to over-reliance on a relatively limited sample of dichotomous data 20.

To avoid the effect of data imbalance on the training results of the models, SMOTE sampling method is used to balance the data.

3.3 Feature Selection

After feature transformation, there are 49 features in the dataset. By feature selection, irrelevant features can be eliminated, and the computational complexity of the model can be reduced. In this paper, we choose to combine the relevance matrix and the feature importance ranking of random forest model for feature selection.

Before feature selection, irrelevant feature "id" and single-valued feature "policyCode" are first eliminated, and then the feature relevance matrix is calculated, and the results are output in the form of a heat map as shown in Fig. 3.

The random forest model is introduced to fit and train the data set, and the importance values of individual features are obtained from the model, then the importance values are sorted in descending order and plotted in a histogram as shown in Fig. 4.

Combining the correlation heat map and the importance ranking of the features, the feature variables with correlation less than 0.01 with the target variable "isDefault" and the feature importance values of the random forest model are eliminated, and finally 33 features are retained for input to the models.

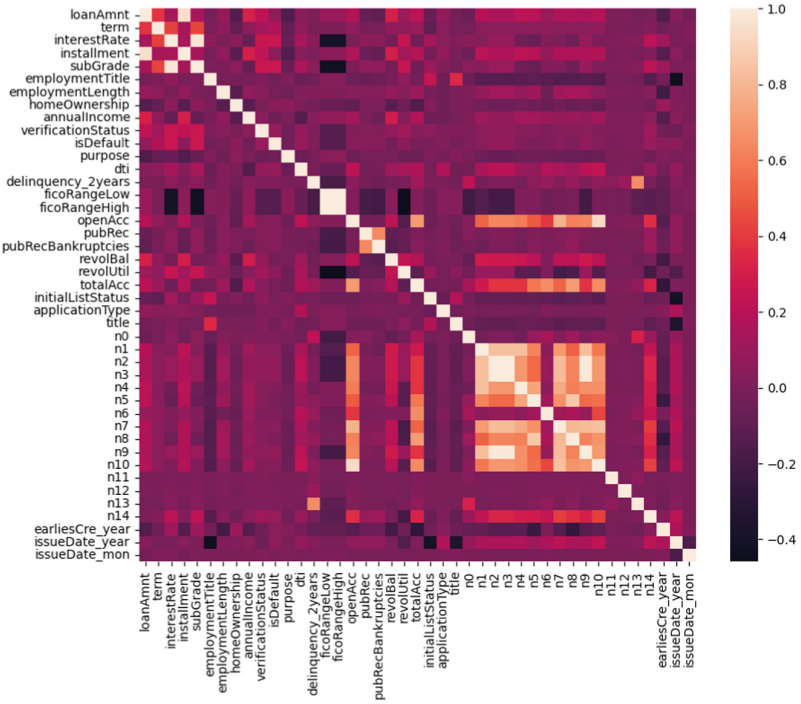

Fig. 3. The correlation heat map of the features

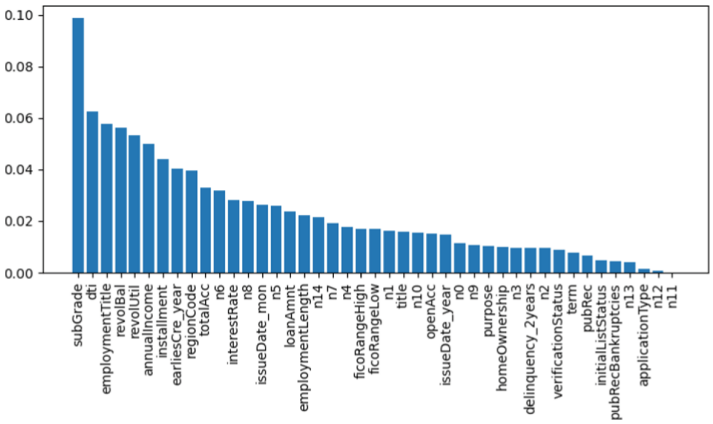

Fig. 4. The importance ranking of the features

4 Experiments and Evaluation

The experimental environment is 64-bit Windows 10 system with Intel i7–6700 HQ, 16 GB of running memory, Anaconda, and Python 3.8 programming language. Firstly, the model evaluation metrics were determined, then the single algorithm models based on LGBM, Adaboost, XGBoost, and GBDT, were established and the model parameters were adjusted by Grid Search method. Finally, a two-layer stacking model with LGBM, Adaboost, XGBoost, and GBDT as the primary classifiers and random forest as the secondary classifier was built, and the evaluation metrics of each model were calculated to evaluate and compare the prediction effects.

4.1 Model Evaluation Metrics

The loan default prediction is a dichotomous problem, and the model predicts either default or non-default. The confusion matrix of the model is shown in Table 1.

In this paper, five metrics, accuracy, precision, recall, the harmonic mean of precision and recall (F1_score), and the area under the ROC curve (AUC), are used to evaluate the prediction results of the model 21. The calculation of each metric is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Table 1. The confusion matrix of the model

Confusion Matrix		**Predicted**	
		Positive	Negative
True	Positive	TP	FN
	Negative	FP	TN

Table 2. Optimal combination of parameters for each classifier

Model	LGBM	Adaboost	XGBoost	GBDT
Optimal parameters	learning_rate: 0.1 max_depth: 5 subsample: 0.8 num_leaves: 80	learning_rate: 0.05 base_estimator: None n_estimators: 210	learning_rate: 0.1 max_depth: 7 subsample: 0.8 gamma: 0.3 reg_alpha: 0.1	learning_rate: 0.1 n_estimators: 95 max_depth: 7 subsample: 0.8

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

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

$$F1\_score = 2\frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

4.2 Model Parameter Settings

The model constructed in this paper adopts the Grid Search method in the sklearn during training, combined with the five-fold cross-validation, and the F1_score is used as the evaluation criterion for the optimization of the model parameters, and the optimal combination of parameters for each classifier is shown in Table 2.

4.3 Experimental Results

The single algorithm model parameters are set according to the tuning results and trained on the training set using five-fold cross-validation, and the fitted models are used to predict the test set to obtain the evaluation metrics.

The stacking algorithm improves the classification results by combining the classification advantages of each single classifier, and in general the stacking model classifies better than the single algorithm model. The performance of two-layer stacking model is also evaluated using five evaluation metrics: accuracy, precision, recall, F1_score, and AUC, and is compared and analyzed with the evaluation metrics of the single algorithm models. The comparison of the evaluation metrics of each model is shown in Table 3.

As can be seen from Table 3, each evaluation metric of the classification results of the Stacking model is significantly higher than the four single algorithm models of LGBM, Adaboost, XGBoost, and GBDT. In the accuracy index, the Stacking model is 8.0% higher than the LGBM model, which is the highest among the single algorithm models; in the accuracy index, the Stacking model is 8.4% higher than the GBDT model, which is the highest among the single algorithm models; in the recall index, the Stacking model is 6.2% higher than the LGBM model, which is the highest among the single algorithm models, indicating that the Stacking model is more predictive than the single model.

Table 3. Evaluation metrics of each model

Model	Accuracy	Precision	Recall	F1_score	AUC
AdaBoost	71.22%	73.13%	71.50%	0.7231	0.7191
XGBoost	71.69%	72.08%	73.38%	0.7273	0.7248
LGBM	74.08%	74.28%	75.54%	0.7491	0.7316
GBDT	73.27%	74.63%	73.20%	0.7391	0.7251
Stacking	82.03%	82.98%	81.75%	0.8236	0.8017

Among the four single algorithm models LGBM model has the best prediction effect with the AUC value of 0.7316, as shown in Fig. 5.

Figure 6 shows that the AUC value of Stacking model reaches 0.8017, which suggests that the Stacking model is more generalizable than the single algorithm model.

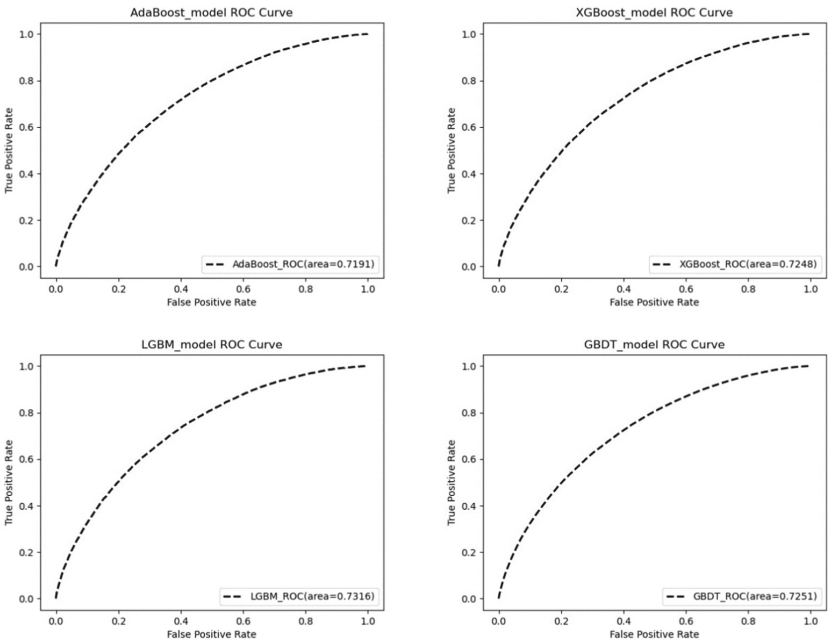

Fig. 5. ROC Curves of single algorithm models

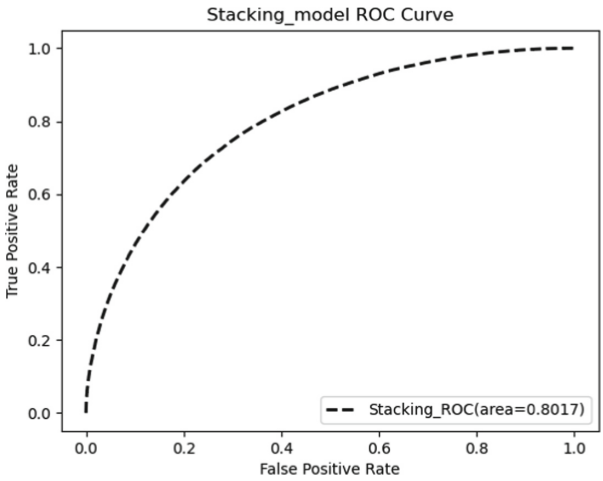

Fig. 6. ROC Curve of Stacking model

5 Conclusion

In this paper, four single algorithm models based on Boosting algorithm are established to predict loan default classification. The model parameters are adjusted by Gird Search method, then the evaluation metrics are determined, and the model prediction effect is analyzed. We propose a two-layer model based on Stacking ensemble learning algorithm to predict personal loan default, which combines LGBM (LightGBM), Adaboost, XGBoost, and GBDT (Gradient Boosting) as the primary classifiers and random forest as the secondary classifier. The dataset is selected based on feature correlation matrix and random forest feature importance ranking, and the data imbalance is solved by SMOTE sampling method. The prediction results of Stacking model and four single algorithm models are compared through experiments, which show that five evaluation metrics of Stacking model are significantly higher than the four single algorithm models in terms of accuracy, precision, recall, F1_score, AUC, and have better generalization ability and application value.

References

1. Guo, Y., Zhou, W., Luo, C., Liu, C., & Xiong, H. (2016). Instance-based credit risk assessment for investment decisions in P2P lending. EUROPEAN JOURNAL OF OPERATIONAL RESEARCH, 249(2), 417-426. doi: https://doi.org/10.1016/j.ejor.2015.05.050
2. Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. EXPERT SYSTEMS WITH APPLICATIONS, 42(10), 4621-4631. doi: https://doi.org/10.1016/j.eswa.2015.02.001
3. Ampountolas, A., Nyarko Nde, T., Date, P., & Constantinescu, C. (2021). A Machine Learning Approach for Micro-Credit Scoring. RISKS, 9(503). doi: https://doi.org/10.3390/risks9030050

4. Chen, K., Yadav, A., Khan, A., & Zhu, K. (2020). Credit Fraud Detection Based on Hybrid Credit Scoring Model. INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE AND DATA SCIENCE, 167, 2-8. doi: https://doi.org/10.1016/j.procs.2020.03.176
5. Ruiz, S., Gomes, P., Rodrigues, L., & Gama, J. (2019). Credit scoring for microfinance using behavioral data in emerging markets. INTELLIGENT DATA ANALYSIS, 23(6), 1355-1378. doi: https://doi.org/10.3233/IDA-184239
6. Dushimimana, B., Wambui, Y., Lubega, T., & McSharry, P. E. (2020). Use of Machine Learning Techniques to Create a Credit Score Model for Airtime Loans. JOURNAL OF RISK AND FINANCIAL MANAGEMENT, 13(1808). doi: https://doi.org/10.3390/jrfm13080180
7. Yu, L., Yao, X., Zhang, X., Yin, H., & Liu, J. (2020). A novel dual-weighted fuzzy proximal support vector machine with application to credit risk analysis. INTERNATIONAL REVIEW OF FINANCIAL ANALYSIS, 71(101577). doi: https://doi.org/10.1016/j.irfa.2020.101577
8. Munkhdalai, L., Ryu, K. H., Namsrai, O., & Theera-Umpon, N. (2021). A Partially Interpretable Adaptive Softmax Regression for Credit Scoring. APPLIED SCIENCES-BASEL, 11(32277). doi: https://doi.org/10.3390/app11073227
9. Arora, N., & Kaur, P. D. (2020). A Bolasso based consistent feature selection enabled random forest classification algorithm: An application to credit risk assessment. APPLIED SOFT COMPUTING, 86(105936). doi: https://doi.org/10.1016/j.asoc.2019.105936
10. Yao, J., & Chen, J. (2019). A New Hybrid Support Vector Machine Ensemble Classification Model for Credit Scoring. JOURNAL OF INFORMATION TECHNOLOGY RESEARCH, 12(1SI), 77–88. doi: https://doi.org/10.4018/JITR.2019010106
11. Luo, C. (2022). A comparison analysis for credit scoring using bagging ensembles. EXPERT SYSTEMS, 39(e122972). doi: https://doi.org/10.1111/exsy.12297
12. Gyamerah, S. A., Ngare, P., & Ikpe, D. (2019). On Stock Market Movement Prediction Via Stacking Ensemble Learning Method. 2019 IEEE CONFERENCE ON COMPUTATIONAL INTELLIGENCE FOR FINANCIAL ENGINEERING & ECONOMICS (CIFER 2019), 113–120
13. Li, Y., & Chen, W. (2020). A Comparative Performance Assessment of Ensemble Learning for Credit Scoring. MATHEMATICS, 8(175610). doi: https://doi.org/10.3390/math8101756
14. Su, Y., Wang, S., & Li, Y. (2022). Research on the improvement effect of machine learning and neural network algorithms on the prediction of learning achievement. NEURAL COMPUTING & APPLICATIONS, 34(12SI), 9369-9383. doi: https://doi.org/10.1007/s00521-021-06333-8
15. Li, M., Yan, C., & Liu, W. (2021). The network loan risk prediction model based on Convolutional neural network and Stacking fusion model. APPLIED SOFT COMPUTING, 113(107961B). doi: https://doi.org/10.1016/j.asoc.2021.107961
16. Nalic, J., & Martinovic, G. (2020). Building a Credit Scoring Model Based on Data Mining Approaches. INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING AND KNOWLEDGE ENGINEERING, 30(2), 147-169. doi: https://doi.org/10.1142/S0218194020500072
17. Trivedi, S. K. (2020). A study on credit scoring modeling with different feature selection and machine learning approaches. TECHNOLOGY IN SOCIETY, 63(101413). doi: https://doi.org/10.1016/j.techsoc.2020.101413
18. Emekter, R., Tu, Y., Jirasakuldech, B., & Lu, M. (2015). Evaluating credit risk and loan performance in online Peer-to-Peer (P2P) lending. APPLIED ECONOMICS, 47(1), 54-70. doi: https://doi.org/10.1080/00036846.2014.962222
19. Zhang, H., He, H., & Zhang, W. (2018). Classifier selection and clustering with fuzzy assignment in ensemble model for credit scoring. NEUROCOMPUTING, 316, 210-221. doi: https://doi.org/10.1016/j.neucom.2018.07.070

20. Coser, A., Maer-matei, M. M., & Albu, C. (2019). PREDICTIVE MODELS FOR LOAN DEFAULT RISK ASSESSMENT. ECONOMIC COMPUTATION AND ECONOMIC CYBERNETICS STUDIES AND RESEARCH, 53(2), 149-165. doi: https://doi.org/10.24818/18423264/53.2.19.09
21. Abelian, J., & Castellano, J. G. (2017). A comparative study on base classifiers in ensemble methods for credit scoring. EXPERT SYSTEMS WITH APPLICATIONS, 73, 1-10. doi: https://doi.org/10.1016/j.eswa.2016.12.020