Loan Defaults Prediction Based on Stacked Models Trained by Personalized Features

Letong Zhou

Warwick Mathematics Institute, University of Warwick, Coventry, United Kingdom

letong.zhou@warwick.ac.uk

Abstract. Training one machine learning model with features that all clients have will result in a waste of features, which is likely to adversely affect the model's performance. To solve the problem, the study attempts a new method, which it to train an individual stacked model for each loan client based on personalised features. Data used contains information of about fifteen million loan applicants, their default status, and 468 features in all. 41 of the features that can be quantitively analysed are selected according to the feature importance output by a Random Forest model. Default prediction of every client is made by a stacked model trained with all selected features he/she has. The stacked model consists of two layers, in which a Light Gradient-Boosting Machine (LGBM) classifier is the base learner, and a Logistic Regression model is the meta learner. As the defaulters account for only 3.14%, which is significantly unbalanced, Area Under the Curve (AUC) and F1 scores are employed to evaluate the method, instead of accuracy. Test results show that models trained by personalised features perform better than the ones trained by shared features. Additionally, the stacked model outperforms individual Logistic Regression model, but performs nearly the same as individual LGBM Classifier. In detailed, the stacked models trained with personalised features result in AUC=0.772 and F1=0.188. Due to data unbalance, although the method's F1 score is relatively low, it's considered to be passable. In the future, stacked models combining different models will be attempted.

Keywords: Credit Assessment, Feature Engineering, Model Stacking.

1. Introduction

In finance, a loan is a sum of money borrowed from a party, with an agreement by the borrower to pay it back over a specified period, generally along with interest rate. The interest refers to the cost of borrowing and is calculated as a percentage of the borrowed amount [1]. However, borrowers may fail to meet the agreed repayment, denoted by loan default. Loan defaults can lead to several types of losses for the lender. The most direct impact is the financial loss of the principal amount that has not been repaid. Another typical effect is that frequent defaults can damage a lender's reputation, which makes it less attractive to potential clients. To avoid the risk of defaults, a lender will evaluate the creditworthiness of a client before approving the loan application. The process is known as credit assessment, credit evaluation, or credit analysis. Traditional credit assessment methods rely on manual review and analysis of an applicant's financial information. One of the classic evaluation methods is by reports from credit bureaus, which contain the applicant's credit history, including past loans, repayment history, and any instances of defaults. Lenders will give a score to the applicant based on the report, which is a numerical representation of the applicant's creditworthiness [2].

However, manual assessments require significant human labour to review and interpret the documents given, which takes a long time. This is very likely to result in long waiting time for loan approvals. Additionally, manual assessments often rely on a narrow set of features (e.g. main income, loan amount applied), but do not consider other relevant information (e.g. type of employment, living address, etc.), which may provide a more comprehensive view of the applicant's creditworthiness. This will lead to a lower accuracy in judging whether it's worth to approve the loan application or not. To deal with the disadvantages of traditional methods, Machine Learning (ML) methods have been implemented in credit assessment [3]. Unlike traditional programming, ML algorithms build a model that learns how to make decisions based on sample inputs by itself, instead of being explicitly programmed for each specific task [4]. Due to the use of computers, time needed for assessments is

much lower than manual methods, leading to higher efficiency. In addition, ML models can automatically process and analyse a wide range of data with large volumes from diverse sources (e.g. bank transactions, bill payments, etc.). Furthermore, relying on data-driven models, ML methods can help reduce the subjectivity and biases associated with human judgment [5], which has been demonstrated in many tasks [6-8].

Generally, ML models can only process sample with the same features. However, in the case of credit assessment, the available information of clients is different. To achieve a thorough evaluation, relevant data of the client should be taken into consideration in as much detail as possible. Previous research in 2010 provided a possible solution. In this research, Classification & Regression Trees (CART) and Adaptive Boosting (AdaBoost) are used to process a sample of customer transactions and credit bureau data of a major commercial bank from January 2005 to April 2009. The sample is divided into four groups based on available features. It has been found that the predictive accuracy of models improves as number of features used for training increases [9].

However, with the development of financial industry and technology, the information of loan clients has become more detailed and intricate. As a result, the difference of available features among clients is greater. Training assessment model with only the shared features or simply dividing them into groups for training will lead to a waste of features.

This study aims to address the issue mentioned above and give each loan client a more personalised credit score. In detail, two classic ML methods, LGBM Classifier, and Logistic Regression model, are used as a combination for default prediction. For each client, an individual model is trained to give a default rate based on his/her features provided. The sample used in this study contains detailed information of over twenty-five thousand loan applicants and whether they default from Jan 2019 to Oct 2010.

2. Methodology

2.1. Data Description

Data used in this study for training provided contains 32 csv files in all [10], which can be divided into 7 groups, explained in Table 1.

Table 1. Data Grouping & Description

Description	File Names
date of approval & whether default	train_base
information of the recorded application in base	train_static_0_0-1, train_static_cb_0
personal information of the applicant	train_person_1, train_person_2
cash flow of the applicant	train_deposit_1, train_debitcard_1, train_other_1
tax situation of the applicant	train_tax_registry_a_1, train_tax_registry_b_1, train_tax_registry_b_1
information of previous applications	train_applprev_1_0-1, train_applprev_2
information of the applicant from credit bureau	train_credit_bureau_a_1_0-3, train_credit_bureau_a_2_0-10, train_credit_bureau_b_1, train_credit_bureau_b_2

Each approved loan application is recorded by a unique case_id. All recorded applications have their default status, approval date in *train_base* and corresponding monthly annuity, credit amount as well as interest rate in *train_static_0*. However, whether the other types of information are available is uncertain.

All the files above are selected for model training, except for *train_credit_bureau_a_2_0-10* and *train_applprev_2*. In detail, *train_applprev_2* contains card blocking reasons, which is difficult to be analysed quantitatively, and *train_credit_bureau_a_2_0-10* records the date and amount of every repayment the applicant made before. For each applicant included, he/she may have several

loans/debts to be repaid, and for each loan/debt, several repayments are recorded, which are very difficult to be compared according to case ids.

2.2. Feature Selection Based on Random Forest

Data provided contains 468 types of information related to the applications. Not all of them are close related to the default status. For instance, type_25L denotes the contact type of a client, including mobile phone, email, etc, which doesn't have a direct relation to defaults. Using such features for model training will result in higher workload and lower accuracy. Therefore, a Random Forest model is employed for feature selection, based on the output feature importance.

The Random Forest mentioned above is a classic ML modelling method, made up of a large amount of decision trees. Each decision tree is a tree-shaped classifier that helps make decisions by splitting data into smaller groups based on certain attributes, leading to a final outcome at the leaf nodes. Multiple decision trees are integrated to reduce overfitting and improve generalization [11]. Random forest models can output feature importance, which represents the significance of each feature in predicting the target variable. The following part explains the feature selection process.

For the selected files, each of them is joined with *train_base* separately. For each joined file, features that can be quantitatively analysed are chosen as training features of a Random Forest model and the targets are whether the applicants default, represented by 0 (not default) and 1 (default).

The output importance of each feature is compared with the importance of the week number of the application within the year, denoted by *WEEK_NUM*. Features with higher importance than week number are selected for the modelling section. The reason why *WEEK_NUM* is chosen as the selection standard is that week number is a feature that every application has and relatively more stable. The selected features from each file are shown in Table 2. The meaning of feature names is explained in *feature_definitions.csv.*

In Table 2, *Relative Feature Importance to Week_NUM* = Feature Importance / Importance of WEEK_NUM, all precise to two decimal places.

Table 2. Selected Features & Relative Importance

Feature Name	Relative Feature Importance to Week_NUM
train_static_0	
price_1097A	1.37
totalsettled_863A	1.30
disbursedcredamount_1113A	1.26
credamount_770A	1.25
avgdpdtolclosure24_3658938P	1.25
maxdpdlast24m_143P	1.23
maxdpdlast12m_727P	1.22
maxannuity_159A	1.22
annuity_780A	1.20
maxdebt4_972A	1.08
pctinstlsallpaidlate1d_3546856L	1.07
numrejects9m_859L	1.05
numinstlswithdpd10_728L	1.05
pctinstlsallpaidlate6d_3546844L	1.02
maxdpdtolerance_374P	1.01
maxdpdlast9m_1059P	1.01
train_applprev_1	
annuity_853A	1.77
credamount_590A	1.56
mainoccupationinc_437A	1.21
train_credit_bureau_a_1	
debtoutstand_525A	1.51
totaloutstanddebtvalue_39A	1.47
overdueamountmax2_14A	1.46

overdueamountmax_155A	1.36
monthlyinstlamount_332A	1.36
dpdmax_139P	1.22
numberofoverdueinstlmax_1039L	1.16
train_credit_bureau_b_1	
totalamount_881A	1.12
dpdmax_851P	1.02
train_tax_registry_a_1	
amount_4527230A	11.41
train_tax_registry_b_1	
amount_4917619A	21.23
train_tax_registry_c_1	
pmtamount_36A	17.61
train_static_cb_0	
\	\
train_other_1	
amtdepositbalance_4809441A	1.18
amtdebitincoming_4809443A	1.17
amtdebitoutgoing_4809440A	1.16
train_person_1	
mainoccupationinc_384A	3.00
train_debit_card_1	
all_last180dayturnover_1134A	1.99
all_last180dayaveragebalance_704A	1.33
train_deposit_1	
total_deposit	16.61
train_credit_bureau_b_2	
all_pmts_dpdvalue_108P	3.13
all_pmts_pmtsoverdue_635A	2.29

All the selected features are joined to *train_base* as the base data for training.

2.3. Modelling

Most of the time, available information for credit assessment of loan clients is different. If the model for default prediction is trained with the information that every client has, a large number of personalised features will be wasted, resulting in a low prediction accuracy. Therefore, to utilize as many valid features as possible, for each client, an individual model is trained and used for his/her default prediction.

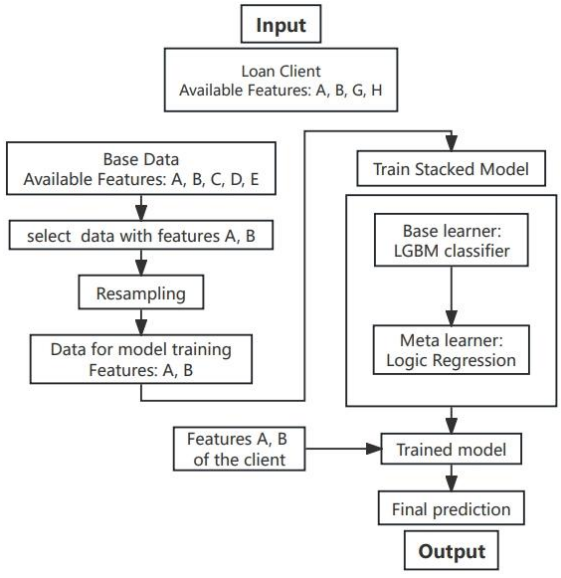

Figure 1. Workflow of The Model (Picture credit: Original)

Fig. 1 demonstrates how the whole model works with an example client. As Figure 1 shows, the client has features A, B, G, H, of which A, B lie in the selected features. Data that contain features A and B are selected from base data and resampled. Then these samples are input into the stacked model as training set, with target being default status (0 or 1). In the end, features A and B of the client will be used for his/her default prediction with the trained model.

The base data is extremely unbalanced, with only 3.14% of 1526659 samples being class 1 (default). Therefore, the training data selected from the data base for individual client is resampled. The ratio of samples of class 1 and samples of class 2 is controlled to be 1:7.

This stacked model is composed of a LGBM classifier as the base learner and a Logistic Regression model as the meta-learner. The two ML modelling methods used will be explained separately in the following text.

LGBM Classifier stands for Light Gradient Boosting Machine Classifier, which works by building a series of decision trees. Gradient boosting framework is used in LGBM Classifier to combine weak learners (typically decision trees) sequentially, where each new tree corrects errors made by previously trained trees. Unlike other boosting methods, LGBM grows trees leaf-wise (vertically) rather than level-wise (horizontally), leading to higher efficiency [12]. In this study, the parameters of LGBM Classifier are set as follows, n_estimators=250, learning_rate=0.05, max_depth=15, num_leaves=25.

The Logistic Regression model is a statistical method used for binary classification problems, which works by estimating probabilities using a logistic function. The logistic function is an S-shaped curve that maps any real number into a value between 0 and 1. In essence, logistic regression calculates the odds of a certain event occurring based on the input features. This is implemented by computing a weighted sum of the input features plus a bias term, and then applying the logistic function to the sum. This process transforms the output into a probability score that represents the likelihood of the dependent event being true [12]. In detail, logistic function is $f(x) = \frac{1}{1-e^{-x}}$, $x = b_0 + b_1x_1 + b_2x_2 + \cdots$. f(x) represents the probability of the target event. $x_1, x_2, \ldots$ represent the input features. $b_0, b_1, b_2, \ldots$ represent the coefficients of the model, which are learned from the training data [13].

LGBM Classifier is trained with the resampled data and make predictions. These predictions are known as the meta-features of the base learner, which are used as input for the meta learner, Logistic Regression. Then, Logistic Regression learns how to optimally combine these base model predictions to produce the final prediction result. Stacking model can leverage strengths of individual models and mitigate their weaknesses, often leading to higher overall prediction accuracy compared to any single model [14]. LGBM Classifier is chosen as the first layer because of its strong capability for handling large datasets. Logistic Regression is chosen as the second layer so that the whole model can capture both non-linear patterns through LGBM and linear relationships through logistic regression.

2.4. Evaluation of the Model

To achieve a comprehensive evaluation of the whole model, 32847 applications are sampled from the base data as the test set. The ratio of defaulted applications and not defaulted ones in the test set is controlled to be the same as the original data, which is 3.14% (1000:31847). The rest of the base data is used for training. Due to data unbalance, the performance of models is evaluated through F1 score and AUC score, instead of accuracy.

Evaluation focuses on two aspects: 1. Compare the performance of the stacked model and individual models. 2. Compare the performance of models trained with personalized features and those trained with shared features in the selected ones (Specifically, the shared ones are 'WEEK_NUM', 'numrejects9m_859L', 'disbursedcredamount_1113A', 'annuity_780A', 'credamount_770A', and 'mainoccupationinc_384A'). Principles of the two scoring methods are explained as follows.

The F1 score is a metric used to evaluate the accuracy of a classification model, particularly useful in situations of imbalanced datasets. F1 score is calculated by F1 = 2 × (Precision × Recall) /

(Precision + Recall). In the formula, Precision denotes the ratio of correctly predicted positive observations to the total predicted positives, while Recall (also known as sensitivity) represents the ratio of correctly predicted positive observations to all actual positives [15]. In this study, probability threshold is set to be 0.35.

The Area Under the Curve (AUC) score is a performance measurement for binary classification models. It refers to the area under the Receiver Operating Characteristic (ROC) curve, which plots the true positive rate (TPR, also known as recall) against the false positive rate (FPR) at various threshold setting [16].

3. Results & Discussion

Results of evaluation are recorded in Table 3, in which values are round to 3 significant figures. According to Table 3, all three modelling methods result in significantly higher AUC and F-1 scores when trained with personalised features. Higher AUC scores indicate higher accuracy in distinguishing two classes, which is default or not in this study [16]. And Higher F-1 scores suggests better performance in maintaining a balance between Precision and Recall, particularly in positive class (default) predictions [15]. Therefore, personalised features are more effective than shared ones, with stronger association to the default status.

Table 3. The Performance of Various Models

Feature Type	Personalised			Shared Features		
model type	stacked model	Logistic Regression	LGBM Classifier	stacked model	Logistic Regression only	LGBM Classifier
AUC score	0.772	0.581	0.772	0.639	0.490	0.639
F-1 score	0.188	0.0651	0.188	0.00987	0.00196	0.00598

Furthermore, under the condition that same features are used, individual Logistic Regression model leads to lower scores than the stacked model, while individual LGBM Classifier performs nearly the same as the stacked model. With personalised features, scores of both methods are identical. On the other hand, with shared features, although AUC scores are exactly the same, F-1 score of the stacked model is slightly higher.

In detail, stacked models trained by personalised features score 0.772 in AUC, which is generally considered to have acceptable classification ability, substantially more accurate than a random classifier (AUC=0.5). In addition, K-1 score of the method is 0.188, which is a relatively low level. This is mostly likely caused by unbalanced data. However, in practice, loan defaults are inherently rare events. Thus, low K-1 score is considered to be passable.

4. Conclusion

This study aims to solve the problem that in credit assessment, available features of loan clients are very likely to be different. Making default predictions with a model trained by features that all clients have may lead to a waste of features, resulting in lower accuracy. In this study, a set features are selected according to the feature importance output by a Random Forest model. For each client, an individual stacked model is trained with every feature the client has in the selected ones to predict his/her loan default rate. The stacked model is made up of two layers, with a LGBM Classifier being the first layer and a Logistic Regression model being the second. Experiments were conducted to evaluate this method. Experimental results showed that personalised features are indeed more effective than shared ones. In addition, the stacked model outperforms individual Logistic Regression model, but has nearly the same performance as individual LGBM Classifier. In the future, stacked models composed of different model combinations will be attempted.

References

[1] Ergungor OE. Theories of bank loan commitments. Economic Review, 2001, 37 (3): 2-19.

[2] Mester LJ. What's the point of credit scoring. Business review, Sep/Oct 1997, 3: 3-16.

[3] Qiu Y, Wang J. A Machine Learning Approach to Credit Card Customer Segmentation for Economic Stability. InProceedings of the 4th International Conference on Economic Management and Big Data Applications, ICEMBDA 2023, October 27–29, 2023, Tianjin, China 2024 Jan 19.

[4] Bi Q, et al. What is machine learning? A primer for the epidemiologist. American journal of epidemiology, 2019, 188 (12): 2222-2239.

[5] Wuest T, et al. Machine learning in manufacturing: advantages, challenges, and applications. Production & Manufacturing Research, 2016, 4 (1): 23-45.

[6] Wang H, Zhou Y, Perez E, Roemer F. Jointly Learning Selection Matrices for Transmitters, Receivers and Fourier Coefficients in Multichannel Imaging. arXiv preprint arXiv:2402.19023. 2024 Feb 29.

[7] Li M, He J, Jiang G, Wang H. DDN-SLAM: Real-time Dense Dynamic Neural Implicit SLAM with Joint Semantic Encoding. arXiv preprint arXiv:2401.01545. 2024 Jan 3.

[8] Qiu Y, Wang J, Jin Z, Chen H, Zhang M, Guo L. Pose-guided matching based on deep learning for assessing quality of action on rehabilitation training. Biomedical Signal Processing and Control. 2022 Feb 1; 72: 103323.

[9] Khandani AE, Kim AJ, Lo AW. Consumer credit-risk models via machine-learning algorithms. Journal of Banking & Finance, 2010, 34 (11): 2767-2787.

[10] Kaggle. Home Credit – Credit Risk Model Stability. https://www.kaggle.com/competitions/home-credit-credit-risk-model-stability/data, 2024.

[11] Biau G, Scornet E. A random forest guided tour. Test, 2016, 25: 197-227.

[12] Ke G, et al. Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 2017, 30.

[13] Hilbe JM. Logistic regression models. Chapman and hall/CRC, 2009.

[14] Kleinbaum DG, et al. Logistic regression. New York: Springer-Verlag, 2002.

[15] Sokolova M, Japkowicz N, Szpakowicz S. Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. Australasian joint conference on artificial intelligence. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[16] Lobo JM, Jiménez-Valverde A, Real R. AUC: a misleading measure of the performance of predictive distribution models. Global ecology and Biogeography, 2008, 17 (2): 145-151.