# The Application of Machine Learning Algorithms in Credit Card Default Prediction

Yue Yu
Institute of Computer and
Information Technology
Fujian Agriculture and Forestry
University
Fuzhou, P.R.China
lucckyuyue@163.com

***Abstract*—With the rapid development of the credit cards industry, there is an increasing number of delinquency rates on credit card loans, which imposes a financial risk for commercial banks. Therefore, successful resolutions of the risks are important for the healthy development of the industry in the long term. The existed methods, such as FICO model [1] (developed by Fair Isaac Company) can estimate the probabilities of credit card defaults, but it based totally on the subjective judgement of people. This means that FICO model comes with several undesirable problems, including low efficiency, low accuracy, time-consuming and high labor costs. In addition, the data of credit card defaults is always unbalanced, since few clients default in real world, which brings challenges to default model construction. In current big-data era, machine learning methods [2] are popular for its high efficiency and high accuracy. In this paper, we employed several classical machine learning algorithms, including logistic regression[3],decision tree[4] and ensemble learning[5] (adaboosting [6], random forest[7]), to build credit default prediction models. To solve the problem of unbalanced data [8], we further build corresponding weighted models so that it can improve the prediction accuracy of default class with slightly higher prediction error of non-default class. The results show that random forest models with weight is the best, which has achieved an accuracy of 82.12%. It achieves the goal of fast learning speed, high parallelism efficiency and high-volume data. Overall, machine learning algorithms has practical application value which can evaluate the delinquency precisely.**

***Keywords*—*machine learning, credit card default prediction, logistics regression, decision tree, random forest, adaboosting***

## I. Introduction

Many financial banks and institutes become more and more attentive in the issue of credit card default because it brings about a high probability of commercial risks. However, evaluated data in current society is so complex that it is both seriously lopsided and high-volume. Though credit card defaults are not so common, it would take its toll on banking systems once it happens. For instance, as the large scale of home-loan defaults and credit card fraud in America, the subprime mortgage crisis [9] has been evoked in 2008, which has spread to global credit markets and banking systems and hit the integral economics seriously in the last. This shows that prediction of credit card default plays a critical role in the maintenance of the banking function and global economics.

There have been several methods proposed to address the problem. The traditional judgement systems have been applied for a long time. It assesses the default rate according to several consumer factors, such as income, property and credit. These traditional systems are relative objective as they rely on inspector's opinions. Besides, it is a waste of time to collecting information and scanning data one by one. Another method is FICO model built by Fair Isaac Company. It has been used widely in American banks, which uses scores to represent the credit safety of clients. The major problem with FICO model is that it depends on the subjective opinion of professional people so that it is difficult to assure the precision and consistency in the practical application. Overall, traditional predict systems are lack of accuracy and they cannot ensure the explanation power of the data. Moreover, it would pose a great threat to the flourish of credit card industry, even the whole economics. Such approaches who use raw data directly are not suitable for current big data era.

Recently, machine learning algorithms [10] have been widely used in credit card default prediction. By constructing several models, it can predict clients' payment precisely and effectively without time spending and intensive labor. Besides, compared with traditional models, machine learning algorithms can address the problem of large-scale data and operate promptly. Due to the unbalanced data, we raise the amounts of default class in the process of establishing models. To conclude, machine learning algorithms are logical, objective and exact, which has a significant practical application in the real world. They are ideal to make the intellectual forecast.

In this paper, we tend to estimate the probabilities of client's default in next month through a few machine learning algorithms. To begin with, as a processing phase, we transformed strings [11] into numerical variables [12] and set up training set [13] and testing set [14] in an attempt to handle the data better. Then, we built a conventional logistics model. Because less consumers tend to be delinquent, we constructed relevant weighted logistic models in order to raise the sample capacity of delinquent type.

However, we cannot obtain expected level of accuracy owing to the linear [15] logistics models. Hence, we built a nonlinear [16] decision tree model and corresponding weighted model to address the linear limitation. Since decision tree model exists a problem of instability, the algorithms of tree-based ensemble learning (including adaboosting, random forest and relevant weighted model) are adopted to avoid the issue, both of which could improve the predicted outcomes and output an optimal accuracy. Finally, the testing set in random forest model came out the accuracy of 82.12%,which was the best in the work.

978-1-7281-7106-7/20/$31.00 ©2020 IEEE
DOI 10.1109/CDS49703.2020.00050

The essay has been organized in the following way. In Section II, we briefly described data information and background. In Section III, we built a series of machine learning models to addressing problems, including logit, decision tree, adaboosting and random forest. We also demonstrated and analyzed the results and comparison with a different view. Finally, we concluded the outcomes and observations.

TABLE I. THE 23 ATTRIBUTES IN CREDIT DEFAULT DATA

| **Attribute** | **Description** | **Type** |
|---|---|---|
| LIMIT_BAL | Amount of the given credit | numerical |
| SEX | Gender | categorical |
| EDUCATION | Education | categorical |
| MARRIAGE | Marital status | categorical |
| AGE | Age | categorical |
| PAY_0 | the repayment status in September, 2005 | numerical |
| PAY_2 | the repayment status in August, 2005 | numerical |
| PAY_3 | the repayment status in July, 2005 | numerical |
| PAY_4 | the repayment status in June, 2005 | numerical |
| PAY_5 | the repayment status in May, 2005 | numerical |
| PAY_6 | the repayment status in April, 2005 | numerical |
| BILL_AMT1 | amount of bill statement in September, 2005 | numerical |
| BILL_AMT2 | amount of bill statement in August, 2005 | numerical |
| BILL_AMT3 | amount of bill statement in July, 2005 | numerical |
| BILL_AMT4 | amount of bill statement in June, 2005 | numerical |
| BILL_AMT5 | amount of bill statement in May, 2005 | numerical |
| BILL_AMT6 | amount of bill statement in April, 2005 | numerical |
| PAY_AMT1 | amount paid in September, 2005 | numerical |
| PAY_AMT2 | amount paid in August, 2005 | numerical |
| PAY_AMT3 | amount paid in July, 2005 | numerical |
| PAY_AMT4 | amount paid in June, 2005 | numerical |
| PAY_AMT5 | amount paid in May, 2005 | numerical |
| PAY_AMT6 | amount paid in April, 2005 | numerical |

## II. DATA DESCRIPTION

The Taiwan Credit Data collected the consumers' attributes and the data of default payments, which is public available at the UCI (University of California Irvine). (http://archive.ics.uci.edu/ml/datasets/default+of+credit+card +clients).

There are 30000 samples and 23 attributes. 23 attributes and their description are displayed in Table I respectively.

To begin with, as a processing phase, we transformed categorical variables [17] into dummy variables [18] and set up training and testing data in an attempt to handle the data better. In order to construct models, we merged several small-sample types together. Specifically, we merged samples whose attributes without description into others for marriage attribute, and attributes without description into others for education attribute. We extracted 70% samples randomly as the training set(2100 samples), and the rest samples as the testing set (900 samples).

## III. MODELS

### *A. Logistics Regression*

Logistics is a classification method whose outcome is categorical values. It was developed by statistician David Cox in 1958[19] and has been widely used in various fields, such as medical statistics, financial area and machine learning. Logistic model is a generalized linear model, which is applied for prediction of the possibility of a binary response.

We first built a logistic model to predict the consumer's default in next month in the training set. Based on the logistics model which has been trained already, we set a threshold value which is 0.5, and then we predicted default for samples with default possibility over 0.5. Then we did the same thing in the testing set. The regression result has been presented in the Table II.

According to the Table II, we found several variables, including LIMIT_BAL, EDUCATION4, MARRIAGE2, AGE, PAY_0, PAY_3 BILL_AMT1, PAY_AMT1, PAY_AMT2 ,PAY_AMT4 are significant in credit default prediction. Table III have shown the confusion table. The error in the training set and testing set was 7.48% and 18.61% respectively.

Since the unbalanced data, the default accuracy is low while the nondefault accuracy is high, up to almost 100%. However, in the reality, it is more crucial to pay attention to the default prediction as default may cause enormous economic loss. Therefore, this model is not suitable for the real world. In order to handle this problem, we created a weighted model to improve the prediction of default class, which sets the weights of default and nondefault as 1:3.5. After this, we conducted the logistics model again, the result has shown in Table IV, and the regression outcome of weighted logistics is in Table II. Compared with the previous results, the training accuracy has reduced, but the default accuracy has been raised. We sacrificed the prediction accuracy of nondefault class to ensure the predication accuracy of default class.

### *B. Decession Tree*

Since logistics is a linear model while the relationship between dependent variables and independent variables is nonlinear, we used the decision tree model which is a method can address nonlinear problems. Decision tree is a flowchart-like structure which aims at handing the problem of decision analysis. It has three types of nodes, including root node, internal node and leaf node [20]. Each branch shows the result of the test and each leaf node shows the tag after

TABLE II. THE RESULT OF LOGISTIC REGRESSION AND WEIGHTED LOGISTIC REGRESSION

| **Attribute** | **Logistics** | | **Weighted Logistics** | |
|---|---|---|---|---|
| | **Estimate** | **Pr(>\|z\|)** | **Estimate** | **Pr(>\|z\|)** |
| **INTERCEPT** | -1.0207 | 0.0000 | 0.0646 | 0.3718 |
| **LIMIT_BAL** | 0.0000 | **0.0000** | 0.0000 | **0.0000** |
| **SEX2** | -0.0820 | 0.0273 | -0.0808 | 0.0018 |
| **EDUCATION2** | -0.0939 | 0.0281 | -0.0806 | 0.0063 |
| **EDUCATION3** | -0.1075 | 0.0610 | -0.0875 | 0.0284 |
| **EDUCATION4** | -1.3861 | **0.0000** | -1.3243 | **0.0000** |
| **MARRIAGE2** | -0.1959 | **0.0000** | -0.1856 | **0.0000** |
| **MARRIAGE3** | -0.2667 | 0.0882 | -0.2240 | 0.0397 |
| **AGE** | 0.0064 | **0.0044** | 0.0071 | **0.0000** |
| **PAY_0** | 0.5882 | **0.0000** | 0.5296 | **0.0000** |
| **PAY_2** | 0.0627 | 0.0101 | 0.0662 | 0.0001 |
| **PAY_3** | 0.0876 | **0.0013** | 0.0853 | **0.0000** |
| **PAY_4** | 0.0002 | 0.9948 | 0.0032 | 0.8786 |
| **PAY_5** | 0.0407 | 0.2091 | 0.0323 | 0.1489 |
| **PAY_6** | 0.0026 | 0.9218 | -0.0145 | 0.4293 |
| **BILL_AMT1** | 0.0000 | **0.0001** | 0.0000 | **0.0000** |
| **BILL_AMT2** | 0.0000 | 0.1304 | 0.0000 | 0.0432 |
| **BILL_AMT3** | 0.0000 | 0.9491 | 0.0000 | 0.9048 |
| **BILL_AMT4** | 0.0000 | 0.7726 | 0.0000 | 0.7556 |
| **BILL_AMT5** | 0.0000 | 0.3728 | 0.0000 | 0.0763 |
| **BILL_AMT6** | 0.0000 | 0.9336 | 0.0000 | 0.4478 |
| **PAY_AMT1** | 0.0000 | **0.0000** | 0.0000 | **0.0000** |
| **PAY_AMT2** | 0.0000 | **0.0066** | 0.0000 | **0.0000** |
| **PAY_AMT3** | 0.0000 | 0.1490 | 0.0000 | 0.0299 |
| **PAY_AMT4** | 0.0000 | **0.0220** | 0.0000 | **0.0002** |
| **PAY_AMT5** | 0.0000 | 0.8995 | 0.0000 | 0.8063 |
| **PAY_AMT6** | 0.0000 | 0.3350 | 0.0000 | 0.2509 |

TABLE III. LOGISTICS MODEL CONFUSION TABLE

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 15730 | 3480 | **Non-default** | 6769 | 1490 |
| **Default** | 447 | 1106 | **Default** | 166 | 474 |
| **Class accuracy** | 97.24% | 24.12% | **Class accuracy** | 97.61% | 24.13% |

TABLE IV. WEIGHTED LOGISTICS MODEL CONFUSION TABLE

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 12800 | 1945 | **Non-default** | 5477 | 817 |
| **Default** | 3377 | 2641 | **Default** | 1458 | 1147 |
| **Class accuracy** | 79.12% | 57.59% | **Class accuracy** | 78.98% | 58.40% |

classification. Decision tree has been commonly applied in operations research and management for a long time.

In the process of constructing decision tree, min-samples split is the parameter that controls minimum samples if continue to split. The model would be simpler with bigger min-samples split parameter. We needed to adjust the model complexity through controlling min-samples split in order to avoiding the problem of overfitting [21]. Hence, we tried a variety of parameter values to build the model and calculate the train error in addition to the test error. The result has been plotted in Fig. 1. According to Fig 1, we selected 1200 as the optimal parameter. We listed the confusion table of corresponding decision tree in Table V.

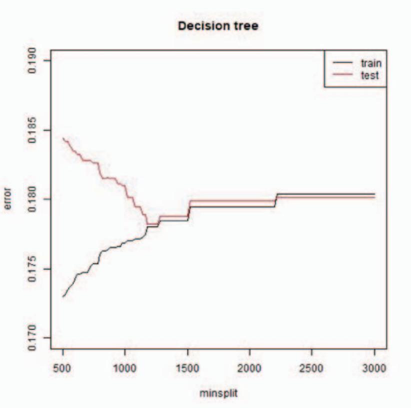

Fig. 1: The train and test error of decision tree with different complexity parameters

It is clear that the whole accuracy of decision is better than that of logistics model. However, the default class accuracy is low similarly. Similar to weighted logistic model, we raise default class accuracy by setting the samples weight. Then we used the weighted data to build the weighted decision tree based on the above-mentioned flow. The error changed with min-samples split has been presented in Fig. 2. The optimal parameter was 1200. Corresponding confusion table has been shown in the Table VI. By comparison, we found that the weighted decision tree accuracy is higher than weighted logistics and prediction accuracy about default class is higher than unweighted decision tree.

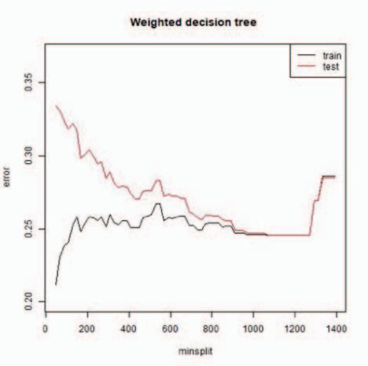

Fig. 2: The train and test error of weighted decision tree with different complexity parameters

## C. Adaboosting

Adaboosting, in definition, is the abbreviation of 'Adaptive Boosting'. It is a machine learning algorithm put forward by Schapire in 1990. Aiming at a training set, Adaboosting trains a number of different classifiers (weak classifiers), then gather them together into a stronger final classifier (strong classifier) [22], which could achieve a superior outcome. The application of adaboosting mostly concentrates on classified problems as well as regression issues.

Similar to decision tree, we set mfinal to control the number of iterations in boosting, which is a parameter of complexity. The curve of error rate fluctuated with mfinal is in the Fig. 3. According to Fig. 3, we selected parameter when mfinal is 400 to be our optimal value. Corresponding confusion table has been listed in Table VII.

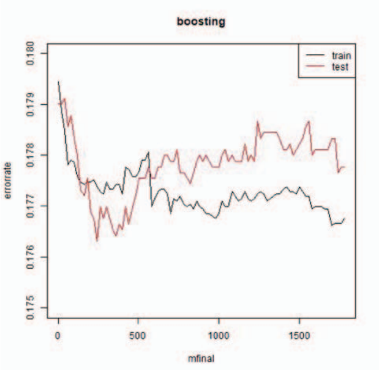

Fig. 3: The train and test error of boosting with different iterations number

Because of the low predicted accuracy of default class, we operated repeated sampling in the default class with a view to making the number of default samples the same as nondefault samples. The weighted boosting has been built on the basis of new data. The outcome is in the Fig. 4, and corresponding confusion table has been shown in the Table VIII. The weighted boosting is better than random forest and prediction accuracy of default is higher than unweighted boosting.

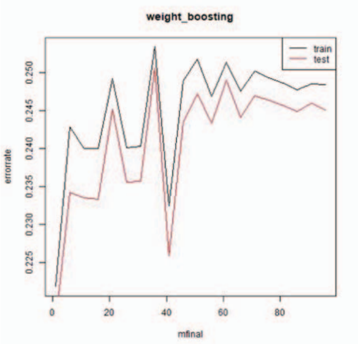

Fig. 4: The train and test error of weighted boosting with different iterations number

## D. Random Forest

Owing to the drawbacks of adaboosting, such as low stability and accuracy, we used ensemble learning to predict default possibility. Random forest is a traditional ensemble learning method, which is for regression, classification and other researches. The earliest random forest was proposed by TinKam Ho in 1995 who used the method to execute the "stochastic discrimination" approach [23] to classification proposed by Eugene Kleinberg. Random forest is a classifier that contains several decision trees and the exporting classes depends on the mode of individual decision tree's output.

TABLE V. THE CONFUSION TABLE OF DECISION TREE

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 15341 | 2860 | **Non-default** | 6584 | 1235 |
| **Default** | 836 | 1726 | **Default** | 351 | 729 |
| **Class accuracy** | 94.83% | 37.64% | **Class accuracy** | 94.94% | 37.12% |

TABLE VI. THE CONFUSION TABLE OF WEIGHTED DECISION TREE

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 12788 | 1708 | **Non-default** | 5490 | 741 |
| **Default** | 3389 | 2878 | **Default** | 1445 | 1223 |
| **Class accuracy** | 79.05% | 62.76% | **Class accuracy** | 79.16% | 62.27% |

TABLE VII. THE CONFUSION TABLE OF BOOSTING

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 15375 | 2877 | **Non-default** | 6598 | 1234 |
| **Default** | 802 | 1709 | **Default** | 337 | 730 |
| **Class accuracy** | 95.04% | 37.27% | **Class accuracy** | 95.14% | 37.17% |

TABLE VIII. THE CONFUSION TABLE OF WEIGHTED BOOSTING

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 12906 | 1696 | **Non-default** | 5528 | 717 |
| **Default** | 3271 | 2890 | **Default** | 1407 | 1247 |
| **Class accuracy** | 79.78% | 63.02% | **Class accuracy** | 79.71% | 63.49% |

TABLE IX. THE CONFUSION TABLE OF RANDOM FOREST

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 16164 | 138 | **Non-default** | 6582 | 1238 |
| **Default** | 13 | 4448 | **Default** | 353 | 726 |
| **Class accuracy** | 99.92% | 96.99% | **Class accuracy** | 94.91% | 36.97% |

TABLE X. THE CONFUSION TABLE OF WEIGHTED RANDOM FOREST

| **Training data** | | | **Testing data** | | |
|---|---|---|---|---|---|
| **True / False** | **Non-default** | **Default** | **True / False** | **Non-default** | **Default** |
| **Non-default** | 14026 | 0 | **Non-default** | 5490 | 700 |
| **Default** | 2151 | 4586 | **Default** | 1445 | 1264 |
| **Class accuracy** | 86.70% | 100.00% | **Class accuracy** | 79.16% | 64.36% |

TABLE XI. THE ACCURACY OF LOGIT, DECISION TREE, RANDOM FOREST AND BOOSTING

| | **Train** | **Test** | **Train(weight)** | **Test(weight)** |
|---|---|---|---|---|
| **Logit** | 81.09% | 81.39% | 74.37% | 74.44% |
| **Decision Tree** | 82.20% | 82.18% | 75.96% | 76.04% |
| **Boosting** | 82.28% | 82.35% | 76.11% | 76.10% |
| **Random Forest** | 99.27% | 82.12% | 89.64% | 75.90% |

TABLE XII. THE CONFUSION TABLE OF LOGIT, DECISION TREE, RANDOM FOREST AND BOOSTING

| | **Logit** | **Decision Tree** | **Boosting** | **Random Forest** |
|---|---|---|---|---|
| **Non- default(train)** | 97.24% | 94.83% | 95.04% | 99.92% |
| **Default(train)** | 24.12% | 37.64% | 37.27% | 96.99% |
| **Non-default(test)** | 97.61% | 94.94% | 95.14% | 94.91% |
| **Default(test)** | 24.13% | 37.12% | 37.17% | 36.97% |
| **Non-default(train,weight)** | 79.12% | 79.99% | 79.61% | 86.70% |
| **Default(train,weight)** | 57.59% | 61.75% | 63.76% | 100.00% |
| **Non-default(test,weight)** | 78.98% | 80.06% | 79.50% | 79.16% |
| **Default(test,weight)** | 58.40% | 61.86% | 64.10% | 64.36% |

Similarly, we controlled the complexity by number of trees to prevent overfitting [24]. The corresponding test error and oob (out of bag) error are plotted in Fig. 5. As shown in Fig. 5, we choose the tree number 220 to be our optimal tree number parameter when the test error was smallest. The corresponding random forest confusion table has been represented in the Table IX.

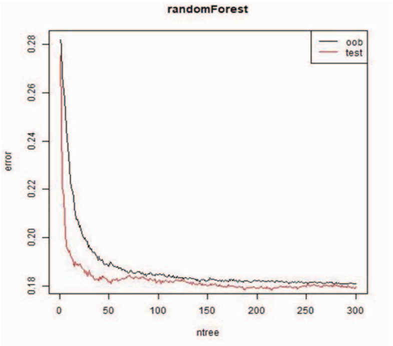

Fig. 5: The oob and test error of random forest with different complexity parameters

Because of the insufficient samples of default class, the predicted accuracy of default clients was inclined to be particular low compared with the nondefault clients. In order to handle the problem, we accomplished the weighted random forest through setting the sampling ratio to be 3000:4000. In line with the above flow, we singled out the excellent parameter to build weighted random forest.

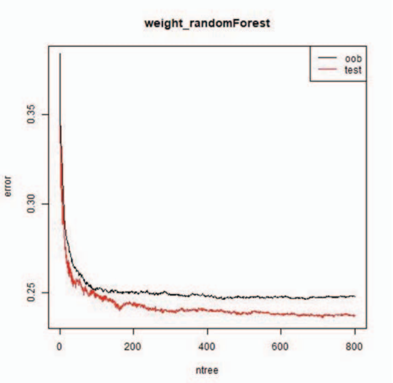

Fig. 6: The train and test error of weighted random forest with different complexity parameters

Fig. 6 provides the information about error fluctuated with tree number, and the optimal tree number is 300. The corresponding confusion has been shown in Table X. The accuracy of weighted random forest is better than weighted decision tree. In addition, the prediction accuracy of default is higher than the unweighted random forest.

## IV. Summary

The application of machine learning algorithms is more significant beneficial to the prediction economic default, which has a high accuracy and saving time. We can reach almost 100% accuracy in the random forest with just a few seconds. Hence, machine learning algorithms will play a more vital role in banking rise management and have a more necessary function in business prediction moving forward.

In this paper, we constructed a series of machine learning algorithms, including logistics, decision tree, random forest and adaboost. Due to the unbalanced samples, we built corresponding weighted models in every model. The accuracy and confusion table showed in Table XI and Table XII. Generally speaking, random forest has the highest accuracy in both unweighted and weighted models and run fast. 99.27% has been achieved by the random forest in the training data. In rest models, both in unweighted and weighted models, boosting was slightly better than decision tree while operated much more time than the other models. The accuracy of logit was the lowest among all models. Compared with unweighted models and weighted models, it is clear that weight will sacrifice the whole accuracy, but according to confusion table, we observed that weighted model can increase accuracy of default class. Overall, we can operate precise as well as efficient prediction through weighted machine learning algorithms

## References

[1] Cieslak D A , Chawla N V . Learning Decision Trees for Unbalanced Data[C]// Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, Heidelberg, 2008.

[2] Jorge Galindo, Pablo Tamayo. Credit Risk Assessment using Statistical and Machine Learning Methods as an Ingredient for Risk Modeling of Financial Intermediaries[J]. computing in economics & finance, 2001.

[3] Sohn S Y , Kim H S . Random effects logistic regression model for default prediction of technology credit guarantee fund[J]. European Journal of Operational Research, 2007, 183(1):472-478.

[4] Pang S , Yuan J . WT Model & Applications in Loan Platform Customer Default Prediction Based on Decision Tree Algorithms[M]// Intelligent Computing Theories and Application. Springer, Cham, 2018.

[5] Larry Shoemaker. Ensemble Learning With Imbalanced Data[J]. 2010.

[6] Ramakrishnan S , Mirzaei M , Bekri M . Adaboost Ensemble Classifiers for Corporate Default Prediction[J]. Research Journal of Applied Sciences Engineering & Technology, 2015, 9(3):224-230.

[7] Lakshmi Devasena C. Comparative Analysis of Random Forest, REP Tree and J48 Classifiers for Credit Risk Prediction[C]// 2014.

[8] Cieslak D A , Chawla N V . Learning Decision Trees for Unbalanced Data[C]// Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Berlin, Heidelberg, 2008.

[9] Wilton, Brandon. Subprime mortgage crisis[J]. alphascript publishing, 2008, 549(6):87.

[10] Tudor L . Machine Learning Algorithms[M]. 2017.

[11] Manber U , Myers G . Suffix Arrays: a New Method for On-Line String Searches[C]// First Acm-siam Symposium on Discrete Algorithms. ACM, 1990.

[12] Kuchemann, D. Children's understanding of numerical variables[J]. Mathematics in School, 1978, 7(4):23-26.

[13] Sheridan R P , Feuston B P , Maiorov V N , et al. Similarity to Molecules in the Training Set Is a Good Discriminator for Prediction Accuracy in QSAR[J]. Journal of Chemical Information and Computer ences, 2004, 44(6):1912-1928.

[14] Tsai J , Bonneau R , Morozov A V , et al. An improved protein decoy set for testing energy functions for protein structure prediction[J]. Proteins, 2003, 53(1):76-87.

[15] Ulrike Graßhoff, Holling H , Schwabe R . Optimal Designs for Linear Logistic Test Models[J].

[16] Chang H H , Ying Z . Nonlinear sequential designs for logistic item response theory models with applications to computerized adaptive tests[J]. Annals of Statistics, 2009, 37(3):1466-1488.

[17] Banker R D , Morey R C . The Use of Categorical Variables in Data Envelopment Analysis[J]. Management Science, 1986, 32(12):1613-1627.

[18] Melissa A. Hardy. Regression with Dummy Variables[J]. bms bulletin of sociological methodology, 1993(40):96-96.

[19] Varin, Cristiano, Cattelan, Manuela, Firth, David. David Cox (statistician)[J]. crism research reports, 2013.

[20] Brocklebank J C , Weir B S , Czika W . Method for selecting node variables in a binary decision tree structure: US 2010.

[21] Reunanen J , Guyon I , Elisseeff A . Overfitting in Making Comparisons Between Variable Selection Methods[J]. Journal of Machine Learning Research, 2003, 3:1371–1382.

[22] Zhang, Yu, Du, Zhijun, Wang, Minjie. Method and apparatus for generating strong classifier for face detection[J]. 2017.

[23] Breton M L , Michelangeli A , Peluso E . A stochastic dominance approach to the measurement of discrimination[J]. journal of economic theory, 2012, 147(4).

[24] Reunanen J , Guyon I , Elisseeff A . Overfitting in Making Comparisons Between Variable Selection Methods[J]. Journal of Machine Learning Research, 2003, 3:1371–1382.