

Feature Selection and Multi-Model Performance Comparison in E-Commerce Customer Churn Prediction

Chengchuan Xu

*Shenyang Ligong University-Tomsk Polytechnic University Institute of International Engineering,
Shenyang Ligong University, Shenyang, China
xcl8859313928@outlook.com*

Abstract. With the rapid development of the e-commerce sector, customer churn has become an increasingly important issue. Controlling customer churn can ensure the development of e-commerce platforms at a relatively low cost, and predicting customer churn is one of the best methods for controlling it. This study faces the absence in this field that simultaneously considers the impact of model performance and parameter variations on the final prediction results. Multiple experimental analyses were conducted on the Kaggle, where publish Ecommerce Customer Behavior Dataset. First, Random Forest was used to perform feature importance analysis on the preprocessed dataset, ranking the features by importance. Then, different models were trained on four datasets retaining different numbers of important features, comparing both the performance differences between models and the differences in the contribution of features to predicting customer churn. Finally, the impact of parameter variations on the prediction results of the best-performing model was investigated, resulting in the highest-performing prediction model. Experiments show that for this dataset, Random Forest performs best. The closer the model performance is to the optimal value, the higher the cost becomes, while the benefits decrease.

Keywords: Machine learning, data analysis, e-commerce

1. Introduction

The rapidly developing e-commerce sector has attracted much attention from the media and academia. Since 2010, the global e-commerce market transaction volume has increased from US\$571 billion to US\$4.75 trillion [1]. As a country with a huge share of global e-commerce, China's share grew rapidly from 2012 to 2020 to 48.3%, and then gradually declined [1]. This demonstrates that the expansion of the e-commerce market leads to faster customer churn. The high cost of acquiring new customers also becomes apparent that retaining existing customers costs only 1/5 to 1/6 of acquiring new ones, making it crucial to addressing the industry's declining scale [2]. Therefore, customer churn prediction has become a key research area in the development of e-commerce. In order to predict customers' future behavior through various behavioral characteristics, the inefficient manual identification has been gradually phased out, and researchers are studying how to use Machine Learning (ML) to better predict data [2, 3].

After learning from a database through algorithms, machines can behave more like humans or improve their abilities in specific professional fields, which enables them to efficiently and automatically handle problems [4]. Their advantages in efficiency and error rate are particularly significant, and this is known as ML [4]. To address the limitations that traditional methods struggle to learn and analyze customer behavior characteristics, Wei proposed a prediction framework using Bi-LSTM with Synthetic Minority Oversampling Technique (SMOTE), which has proven to have high prediction accuracy on real datasets [2]. The experiment only compared the performance of the algorithm in different hyperparameters, and lacked experimental analysis on whether changes in dataset features would affect the prediction results [2]. Kumar et al. conducted experiments comparing six algorithms, including XGBoost, Random Forest and Decision Tree to address issues such as data imbalance and data suggestion in customer churn prediction and concluded that XGBoost performed best [3]. The experiment only compared the differences in prediction performance between algorithms on the dataset, and did not determine whether changes in the data characteristics of the dataset and the hyperparameters of the algorithms affected the prediction results [3].

To investigate the model's performance in e-commerce data processing and the impact of hyperparameter changes on prediction results, this experiment uses the publicly available Ecommerce Customer Behavior Dataset obtained from Kaggle. At first, the feature importance ranking is obtained through the random forest algorithm after preprocessing. Then, retaining different numbers of the top-ranked important features, the results are predicted using four algorithms: Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), and K-Nearest Neighbors (KNN), and the accuracy of the four algorithms is compared in different numbers of important features. After obtaining the optimal algorithm, experiment was conducted to compare the prediction results of the algorithm on the original dataset in different hyperparameters, and to compare the impact of hyperparameter changes on the algorithm's prediction accuracy. Through multi-dimensional ablation and comparative experiments, the best-performing model and hyperparameters were determined. The experiment and analyses provided in this study may be used as a reference for future applications of ML in analyzing and predicting e-commerce data.

2. Method

2.1. Dataset preparation

The Ecommerce Customer Behavior Dataset comes from Kaggle and collects basic information, online behavior and transaction information of 50,000 customers from global e-commerce platforms [5]. Preprocessing involves calculating the number of missing features for each user firstly. If the number exceeds 25% of the total number of features, delete the user's information to avoid experimental bias caused by a large number of missing feature values. If the number is less than 25% of the total number of features, fill in the missing feature values. For numerical features, fill in the missing values with the mean of the column; for categorical features, fill in the missing values with the mode of the column.

2.2. Machine learning models

This study utilizes classes and functions provided by scikit-learn (Version: 1.6.1) to construct, train, and evaluate ML models. Because the Ecommerce Customer Behavior Dataset contains input

feature data and target values, this study uses four common supervised learning ML models including DT, LR, RF, and KNN for comparative experiments.

2.2.1. Decision Tree (DT)

Decision trees (DT), composed of nodes and branches, are among the earliest and most effective machine learning algorithms [4, 6]. Nodes are divided into root nodes (decision nodes), internal nodes (chance nodes), and leaf nodes (end nodes) [6]. The root node is at the top level, dividing the dataset into two or more mutually exclusive subsets, representing the first choice; leaf nodes are at the end of each branch, representing the final result of a series of choices, and there are no more branches below; internal nodes are at a certain step of a branch in the structure, representing the possible branches of that branch at this step [4, 6]. The constraints for splitting downwards from the root node and internal nodes are called branches, which connect the nodes in the structure [6]. Each path from the root node to the leaf node is composed of hierarchical branches [6].

The principle of training a decision tree is to start from the root node, select an optimal feature for partitioning, divide the dataset into multiple subsets, and then recursively repeat the above process for the child nodes until the stopping condition is met, at which point leaf nodes are generated, resulting in a complete tree structure that can analyze and predict data [7]. Due to DT's effectiveness and structural characteristics, it has a wide range of applications, including patient subgrouping, stock price prediction, housing price prediction, book usage prediction, and other scenarios in medicine, finance, daily life and so on [4, 6, 7].

2.2.2. Random Forest (RF)

A model composed of multiple different DTs constructed by randomly sampling training set samples with replacement is called RF, which predicts by voting among all trees [8]. As the number of trees increases, because of their convergence and randomness, the final prediction error will be greatly reduced, which can significantly improve the accuracy and avoid overfitting [8]. By proving a consistency result of Breiman's original algorithm, it can be seen that the better the performance of a single tree, or the smaller the correlation between them in terms of the raw margin functions, the more accurate the model's prediction effect [8, 9]. RF is extremely good at processing structured data, such as user behavior, air quality prediction, and bioinformatics data, which have many features but weak information [8, 9].

2.2.3. Logistic Regression (LR)

Logistic Regression is a classic machine learning and statistical analysis algorithm for binary classification problems [4, 10]. The model relies on log-dominant transformation to construct a linear expression, uses maximum likelihood estimation (MLE) to solve for parameters, and outputs the probability of events between 0 and 1 [4, 10]. It often uses 0.5 as the threshold to complete the classification and can include various types of independent variables and correct for confounding interference [4, 10]. The algorithm has a simple structure, high computational efficiency, and fast training speed [4, 10]. The output probability values can intuitively assist decision-making and are widely used in disease diagnosis, risk control screening and other fields [4, 10]. However, due to the linear nature of the algorithm, it is difficult to capture the nonlinear relationship between variables [10]. When the independent variables are not screened or there are extreme abnormal samples in the data, the model performance is prone to decline [10].

2.2.4. K-Nearest Neighbors (KNN)

KNN is a lazy learning algorithm, which has high-performance and great speed of implementation [11]. When dealing with test samples of unknown class, KNN only stores training samples and does not train an explicit model [4, 11]. It then relies entirely on the local information around the test sample to select the most frequent of the K nearest neighbors as the result for the classification stage, without using a global distribution [4, 11]. The value of K is crucial to the performance of KNN, that a larger K value is prone to underfitting and a smaller K value is prone to overfitting [11]. Due to the assumption-based design of KNN based on the training set, it tends to favor the class with the greatest number of tests when selecting the K nearest training examples, which can easily lead to performance degradation due to irrelevant features [4, 11]. Because of its susceptibility to irrelevant features, KNN performs well in detection applications such as medical diagnosis, intrusion detection, and credit rating [4, 11].

3. Results and discussion

3.1. Feature importance

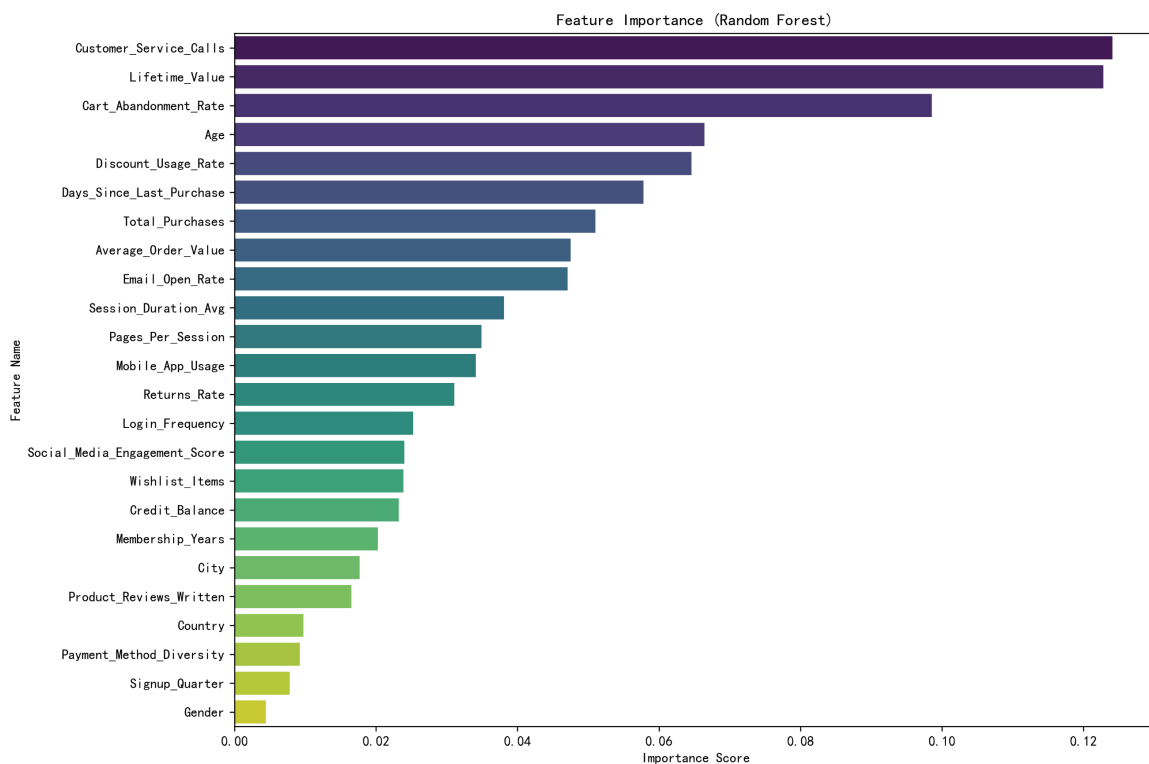


Figure 1. Feature importance by RF (picture credit: original)

Figure 1 shows the ranking of feature importance in the preprocessed Ecommerce Customer Behavior Dataset using RF. Customer_Service_Calls and Lifetime_Value are the most important features, followed by Cart_Abandonment_Rate, changing in them has the greatest impact on the results. It is clear that features in the Customer Service and Purchase Behavior categories have a significant impact on the final result. The quality of Customer Service is the most critical factor in determining customer retention and farewell, and it is something that e-commerce platforms need to

focus on and strengthen today. Furthermore, Purchase Behavior also greatly influences customer retention or farewell, and maintaining customer purchasing habits through advertising and promotion may be a better way to retain existing users.

3.2. Performance comparison

After obtaining the importance of all features, this experiment used four datasets retaining different numbers of the most important features and predicted the final results using ML models including DT, LR, RF, and KNN, as shown in Figure 2 and Table 1. The experimental results show that the more features a model has, the better its training performance. Based on this dataset, RF performed best in the four experiments using different numbers of features, significantly outperforming the other models. Comparing the experiments with 6 and 8 features, DT's accuracy was higher after training with 6 features than after training with 8 features, indicating slight overfitting. Compared with the six-feature setting, the four-feature setting leads to lower accuracy for all models, indicating that using only four features may cause information loss and insufficient model fitting.

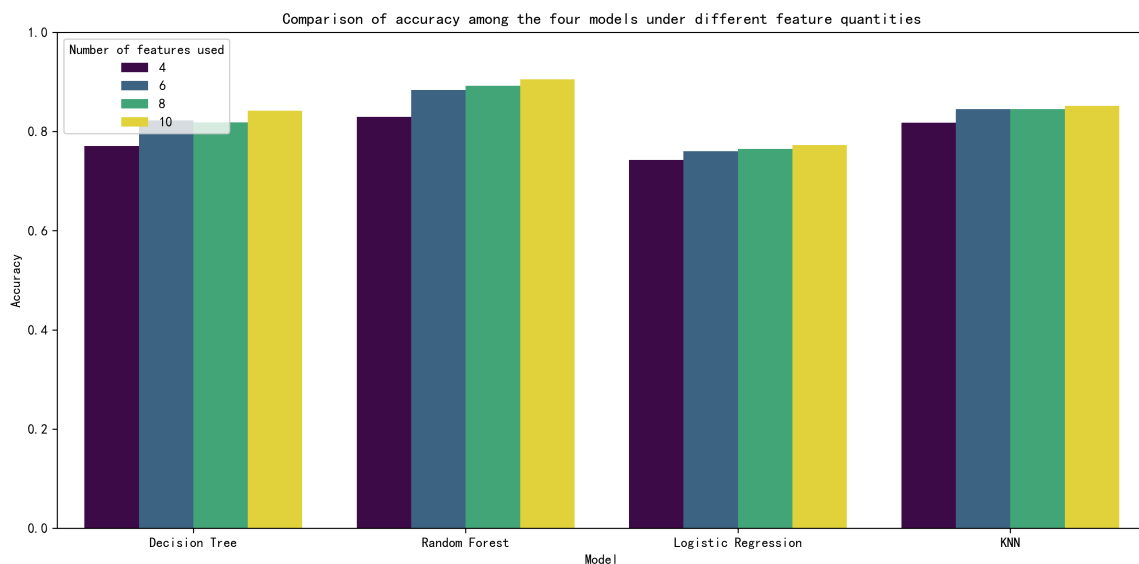


Figure 2. Visualization of model accuracy comparison under different feature quantities (picture credit: original)

Table 1. Numerical comparison of accuracy among the four models under different feature quantities

Models	Number of feature used			
	4	6	8	10
DT	0.7705	0.8222	0.8187	0.8418
RF	0.8296↑	0.8836↑	0.8922↑	0.9055↑
LR	0.7428	0.7601	0.7647	0.7728
KNN	0.8176	0.8450	0.8454	0.8518

As shown in Figure 2 and Table 1, the model can still achieve an accuracy of around 0.8 when trained using only four the most important features. This indicates that the four features including Customer_Service_Calls, Lifetime_Value, Cart_Abandonment_Rate, and Age have a significant

impact on the prediction results. They are key factors for e-commerce companies to accurately predict customer churn. Comparing the underfitting observed in the two sets of experiments with 4 and 6 features shows that the 5th and 6th ranked features, Discount_Usage_Rate and Days_Since_Last_Purchase, also make significant contributions to predicting customer churn. DT exhibits slight overfitting, indicating that redundant or weakly correlated features have a significant impact on the model, making it unsuitable as a prediction model.

3.3. Sensitivity analysis

When the number of decision trees increases from 30 to 300 shown in Figure 3, the model accuracy fluctuates slightly between 0.909 and 0.920, showing an overall upward trend. This indicates that once the number of trees exceeds 30, the model performance improvement is minimal. 200 trees already achieve peak performance, and further increasing the number of trees only increases computational cost without bringing any gain in accuracy.

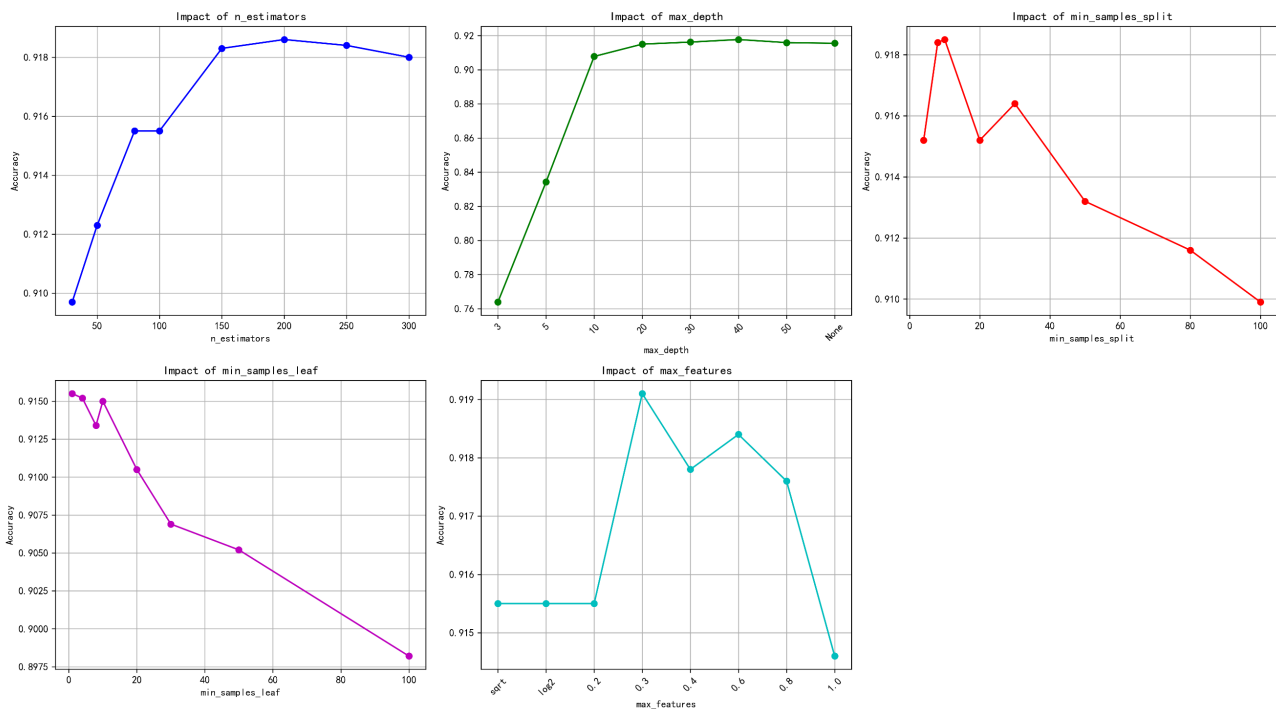


Figure 3. Comparison of parameters change (picture credit: original)

As the maximum depth gradually increases from 3, the accuracy rises from approximately 0.764 to 0.910 and remains stable. The lowest accuracy (0.764) is achieved at a depth of 3, and the accuracy stabilizes at around 0.910 after reaching a depth of 10. Further increasing the depth (20, 30, 40, None) does not significantly change the accuracy. Too small a depth limit leads to severe underfitting, failing to learn the complex interactions between features. However, the model achieves a good fit after a depth exceeding 10, and further increases in depth do not cause overfitting, indicating that the ensemble mechanism of random forests is not sensitive to the depth of individual trees.

As the minimum number of samples required for node splitting increases from 2 to 100, the overall accuracy shows a downward trend, gradually decreasing from 0.910 to 0.895, with the highest value reached at a node split count of 10. Similarly, as the minimum number of samples

increases from 1 to 100, the accuracy of leaf nodes also continuously decreases, from 0.910 to approximately 0.895. These two parameters control the growth complexity of the tree. Smaller number of samples required allow the tree to grow deeper and finer, better fitting the training data. Larger number of samples required force the tree to stop splitting prematurely, leading to underfitting, indicating that maintaining a certain level of model complexity is necessary in this experiment.

As the minimum number of samples required for node splitting increases from 4 to 100, the overall accuracy shows a downward trend, gradually decreasing from 0.915 to 0.910, with the highest reached at a node split count of 10. Similarly, as the minimum number of samples increases from 1 to 100, the accuracy of leaf nodes also continuously decreases, from 0.915 to approximately 0.898. These two parameters control the growth complexity of the tree. Smaller number of samples required allow the tree to grow deeper and finer, better fitting the training data. Larger number of samples required force the tree to stop splitting prematurely, leading to underfitting, indicating that maintaining a certain level of model complexity is necessary in this experiment.

When the feature selection strategy is 0.3, the accuracy reaches its highest level of 0.92, while using more extreme proportions slightly reduces the accuracy. This indicates that splitting approximately one-third of the features each time yields the best results, balancing the diversity and predictive power of a single tree. By contrast, considering too many or few features will reduce the accuracy.

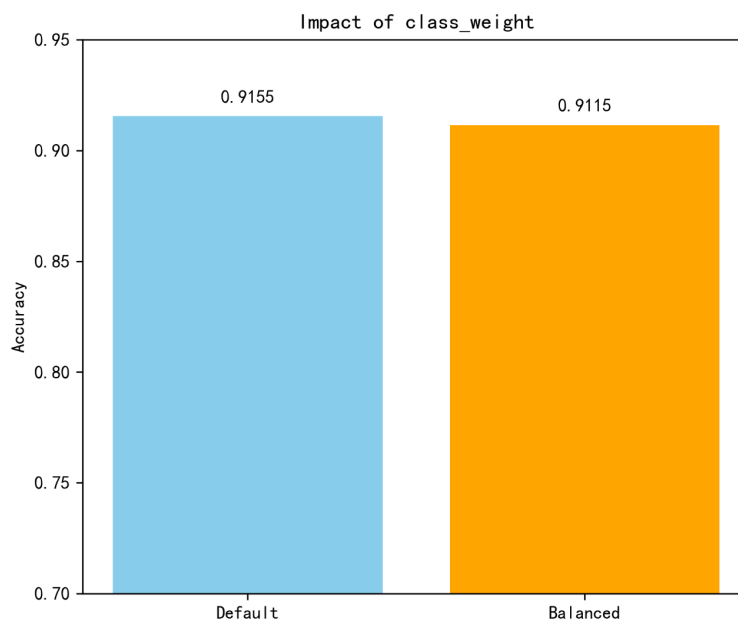


Figure 4. Comparison of class_weight changes (picture credit: original)

In this experiment, the accuracy of the default unweighted method was only slightly better than that of the balanced method shown in Figure 4, indicating that the class imbalance ratio in this dataset may be relatively balanced. Therefore, automatic weight balancing did not improve accuracy and even slightly reduced it.

4. Conclusion

In today's e-commerce platforms, customer retention is increasingly crucial. It makes customer churn prediction more and more paramount. This study uses the Ecommerce Customer Behavior Dataset and conducts multiple comparative experiments to explore the performance differences of different models on this dataset, the performance differences when training models with different numbers of features, and the impact of parameter changes on prediction results. The experiments show that RF performs best among the four ML models and is most suitable for predicting customer churn. With this dataset, when the number of decision trees is 200, the maximum depth is 40, the minimum number of samples required for node splitting is 10, the minimum number of samples for leaf nodes is 1, and the feature selection strategy is 0.3, RF achieves the optimal performance of 0.9195. Experiments show that achieving the highest prediction accuracy requires considering not only the model's fit to the dataset but also whether the parameters are optimal. However, once performance is close to optimal, the cost of further performance enhancements far outweighs the benefits.

The data source for this study is relatively limited, lacking temporal dimension information, and the number of tested parameters is also limited, resulting in certain limitations. Future experiments will continue to investigate data richness and the diversity of measurement methods to achieve more effective and accurate conclusions.

References

- [1] China Data Portal (2026). E-commerce Market: China vs World. Retrieved from <https://chinadata.live/data/ecommerce-market-china-vs-world/>
- [2] Wei, Q. (2024). Personalized and contextualized data analysis for e-commerce customer retention improvement with bi-lstm churn prediction. *IEEE Transactions on Consumer Electronics*, 71(2), 4406-4414.
- [3] Kumar, S., Deep, S., & Kalra, P. (2024). A comprehensive analysis of machine learning techniques for churn prediction in e commerce: A comparative study. *International Journal of Computer Trends and Technology (IJCTT)*, 72(5), 163-170.
- [4] Rahaman, M. J. (2024). A comprehensive review to understand the definitions, advantages, disadvantages and applications of machine learning algorithms. *Int J Comput Appl*, 186(31), 43-47.
- [5] Singh, D. (2025). Ecommerce customer behavior dataset [Data set]. Kaggle. <https://www.kaggle.com/datasets/dhairyajeetsingh/ecommerce-customer-behavior-dataset>
- [6] Ying, L. U. (2015). Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, 27(2), 130.
- [7] Timofeev, R. (2004). Classification and regression trees (CART) theory and applications. Humboldt University, Berlin, 54(2004), 48.
- [8] Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.
- [9] Scornet, E., Biau, G., & Vert, J. P. (2015). Consistency of random forests.
- [10] LaValley, M. P. (2008). Logistic regression. *Circulation*, 117(18), 2395-2399.
- [11] Sun, B., & Chen, H. (2021). A survey of k nearest neighbor algorithms for solving the class imbalanced problem. *Wireless communications and mobile computing*, 2021(1), 5520990.