Survey of Machine Learning Random Forest Algorithms

Zexing Hao [a], Gaizhi Guo

School of computer science technology, Inner Mongolia Normal University, Hohhot 010022, China

[a]4661883@qq.com

Keywords: random forest, decision tree, performance index, machine learning.

Abstract: Machine learning is an important method to realize artificial intelligence. The random forest algorithm, which is known for its simplicity and efficiency, is one of the representative algorithms of machine learning. It is a decision tree-based classifier, which selects the best classification tree as the classification algorithm of the final classifier by voting. Currently, it is in news classification, intrusion detection, content information filtering, sentiment analysis, there are a wide range of applications in the field of image processing. This paper will mainly introduce the decision tree, the construction process of random forests, and the research status of random forests in terms of performance improvement and performance indicators.

1. Introduction

Almost all data analysis problems in life can be divided into regression and classification problems. The main difference between the two types of problems is whether the predicted data is discrete [20]. If there are only two types of results predicted, such as predicting whether the lottery you bought will win, then the result is discrete and belongs to the classification problem; if the future stock market changes are predicted, then the result is continuity. It belongs to the issue of return. In real life, we often discretize the continuous values, so the regression problem becomes a classification problem. Therefore, this paper takes the classification algorithm as the main research object. The classification algorithm can be divided into a single classifier algorithm and a multi-classifier algorithm according to the number of classifiers. The traditional single classifier algorithm has decision tree, Naive Bayes [6], etc. Their computational complexity is not high and easy to use. It can process data with irrelevant features and easily construct rules that are easy to understand, but deal with them. Over-fitting is easy to occur in the process, and it is difficult to deal with missing data. The correlation between attributes in the data set is neglected in the process [12]. Later, people put forward the idea of integrated learning, which is to combine multiple individual classifiers into a new classification model, and finally form the current random forest algorithm.

2. Decision Tree Algorithm

Random forests are essentially composed of multiple decision trees, which are the basic classifiers that make up random forests [23]. Common decision tree algorithms include ID3, C4.5, CART (Classification and Regression Tree), etc. [4]. As one of the earliest decision tree algorithms, the ID3 algorithm constructs a decision tree by selecting the attribute with the largest information gain and the critical value for node splitting. It can only support discrete data processing, and the training model is prone to over-fitting phenomenon [11]. The C4.5 algorithm is an improvement of the ID3 algorithm. In order to prevent the over-fitting phenomenon, it introduces the pruning step on the basis of ID3. The implementation process is to specify a threshold, and the number of samples is smaller than the given threshold. The collection is seen as a leaf node, which does reduce the over-fitting phenomenon, but the threshold selection needs to rely on experience and lack the necessary theoretical support [34]. The CART algorithm [22] performs two-way recursive segmentation on the training samples according to the Gini impurity minimum criterion, and divides the current sample set into two sub-sample sets, so that each non-leaf node of the generated decision

tree has two branches, forming a binary tree. Formal decision tree classifier. It is worth noting that the CART tree is a binary tree, and ID3 and C4.5 can be multi-fork trees. With the training sample set S, the Gini impurity of the training sample set S is defined as:

$$\text{Gini}(S) = 1 - \sum_{i=1}^{n} P_i^2 \quad (1)$$

Where: P_i is the probability that category i appears in the training sample set S. Since the node of the binary tree divides the sample space into two subsets S_1 and S_2, the Gini impurity of the node is:

$$\text{Gini}(S_1, S_2) = \frac{|N_1|}{|N|}\text{Gini}(S_1) + \frac{|N_2|}{|N|}\text{Gini}(S_2) \quad (2)$$

Where: N is the number of samples in the training sample set S, and N_1 and N_2 are the number of samples in the subsets S_1 and S_2, respectively. CART selects the attribute with the smallest Gini as the split attribute of the node. The smaller the Gini value, the higher the purity of the sample and the better the partitioning effect. The decision tree generation process is shown in Figure 1.

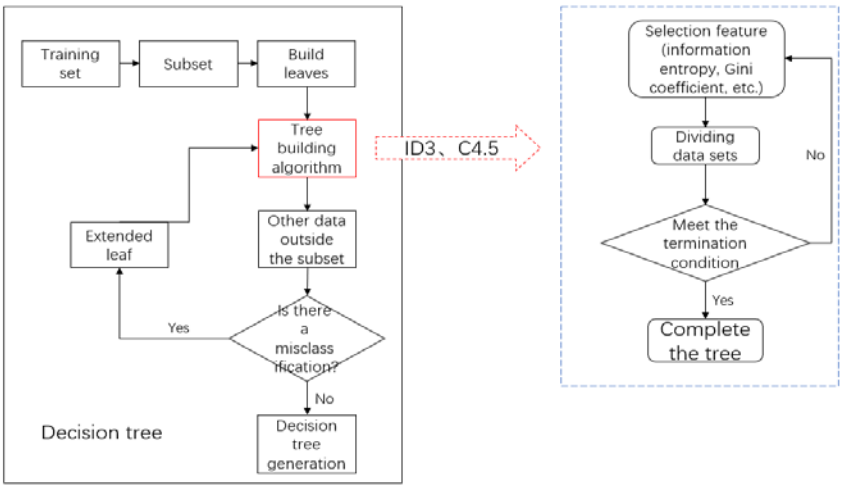

Fig. 1 Decision tree algorithm generation flow chart

3. Random Forest Algorithm

Random forest is an integrated learning model proposed by Breiman with decision tree as the basic classifier. It uses the bootstrap technique to get multiple subsets of samples, constructs a decision tree using each subset of samples, and combines multiple decision trees into a random forest. When the sample to be classified is input, the final classification result is determined by the decision tree vote [8][16]. The training process of the random forest is shown in Figure 2.

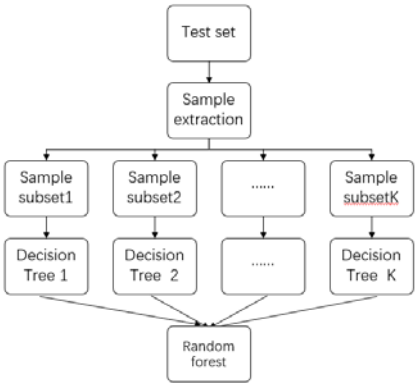

Fig. 2 Random forest training flow chart

The specific algorithm steps of the random forest are as follows:

(1) The number of samples in the given original training set is N, and the number of feature attributes is M. The bootstrap sampling technique is used to extract N samples from the original training set to form a training subset.

(2) randomly select m features from the M feature attributes as candidate features (m ≤ M), and select the optimal attributes for splitting according to certain rules (Gini index, information gain rate, etc.) in each node of the decision tree. Until all the training examples of the node belong to the same class, the process is completely split without pruning.

(3) Repeat the above two steps k times to construct a k-decision tree to generate a random forest.

(4) Using random forest for decision making, let x be the test sample, hi for a single decision tree, Y for the output variable ie classification label, I for the indicator function, H for the random forest model, and the decision formula:

$$\mathrm{H}(\mathrm{x}) = \arg\max_{\mathrm{y}} \sum_{i=1}^{k} I(h_i(x) = Y) \tag{3}$$

That is, the classification result of each test tree for the test sample is summarized, and the class with the largest number of votes is the final classification result.

In addition, some random forest promotion algorithms have appeared, and their pairs with the random forest algorithm are shown in Table 1.

Table 1. Random forest promotion algorithm

Algorithm name	Different from random forest
RandomSurvivalForest(RSF)[24]	The tree-building rule is similar to RF. Each decision tree in the RSF is a two-class survival tree to process survival data. It is superior to other survival analysis methods for high-dimensional survival data.
Extratrees[25]	A variant of RF, the difference from RF: generally does not use self-help sampling, each decision tree uses the original training set, and only randomly selects a sample feature to divide the decision tree.
IsolationFores(IForest)[26]	Using RF-like methods to test outliers, the difference with RF: the self-assisted sampling is used to sample the training set, but the number of samples is not the same as RF (equal to the number of training sets), but much smaller than the training set. For each decision tree, a partitioning feature is randomly selected, and a partitioning threshold is randomly selected for the partitioning features.

4. Performance Improvement of Random Forest Algorithm

4.1 Parameter Optimization

For the same data set, different node splitting algorithms are selected, and different decision trees are obtained because the selected attributes are different. It is concluded that the classification accuracy of random forests will be different. Therefore, it is proposed to select the optimal when generating decision trees. The attributes of the nodes are split, that is, the node splitting algorithm is linearly combined to form a new splitting rule, which is applied to the selection and partitioning of node attributes. The optimized node splitting mainly has two methods: parameter adaptive [6] and hyperparameter optimization [25]. These two methods are described in detail below.

4.1.1 Parameter Adaptation

Literature [6] proposes to combine the contents of Table 1, the node splitting criterion should aim at the higher purity of the divided data set, and update the splitting formula of the combined node as:

$$H = \min_{(\alpha,\ \beta)\in R} F\{D,a\} = \alpha Gini(D,a) - \beta Gain(D,a) \quad (4)$$

$$s.t.\begin{cases} \alpha+\beta=1 \\ 0 \le \alpha,\beta \le 1 \end{cases} \quad (5)$$

Among them, the parameters α, β represent the coefficients of the two algorithms in H(x), and the H value is the smallest, that is, both ID3 and CART are optimal as the node division criterion to improve the classification effect.

Table 2. Node splitting algorithm comparison

algorithm	Node splitting criterion	Standard indicator
ID3	Maximum information gain	Purity of the sample in the data set divided
CART	The Gini index is the smallest	Randomly pumping two samples from the data set with different probabilities

Since the characteristics of data in different data sets are different, the parameter selection in the random forest algorithm is also difficult to fix. Therefore, the adaptive parameter selection process is used to obtain the optimal combination parameters. For the parameters, the constraints in the above formula should be satisfied.

4.1.2 Hyperparametric Optimization

Hyperparametric optimization of random forests generally uses grid search and random search [25]. Grid search refers to meshing variable regions, traversing all grid points, solving the objective function values that satisfy the constraint function, and finally selecting the most advantage. Grid Search First, divide the grid with large steps in a large range, and perform rough search to select the best advantage. Then, using the small step size to divide the grid near the most advantageous, the mesh division is denser, and the search again selects the best advantage. Repeat the above steps until the grid spacing or objective function change is less than the given value.

Unlike the grid search, the random search extracts a random combination of parameters from the sampling distribution, and the calculation amount is large. Although the grid method on random forest hyperparameter selection is currently the most widely used parameter optimization method, the random search algorithm can choose a budget that is independent of the number of parameters and possible values, and adding parameters that do not affect performance does not reduce efficiency. , so the random search algorithm also has its merits.

4.2 Change Voting Weight

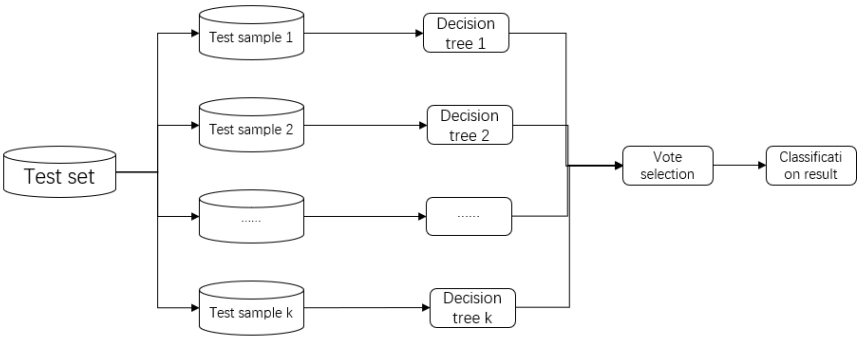

Fig. 3 Weighted random forest classification process

In [25], the decision tree of random forests is used to adjust the voting weights of different feature categories. The decision tree with strong classification ability is given high weight, and the decision tree with weak classification ability is given low weight, and finally combined by weighted voting. Use the maximum voting criteria to obtain classification results. The commonly used weight

adjustment algorithm consists of Adaboost [6], decision tree [25], weight matrix [40] and so on. Figure 4 is a technical roadmap for weighting improved random forests

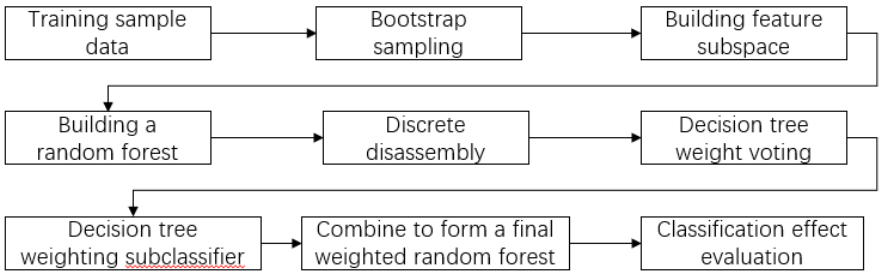

Fig. 4 Weighting algorithm flow chart

4.3 Add Penalty Factor

Based on the random forest algorithm, Xue Minglong et al. introduced a penalty factor and proposed a PRF (Penalized Random Forum) algorithm [23] to introduce a penalty factor to classify and identify user activities in a smart environment. This algorithm greatly reduces the probability that the same attribute will be repeatedly selected. In the decision tree model, the splitting attribute used by the node with lower depth has more influence on the structural change of the whole decision tree model, and the attribute used by the node with lower depth can obtain higher penalty degree (its weight wi Lower), so setting a higher penalty can reduce the probability of its occurrence in the next iteration. This method guarantees the diversity of the integrated algorithm. After the experiment, the final confirmation can improve the credibility of the classification result.

5. Random Forest Algorithm Performance Index

5.1 Classification Accuracy

The accuracy of the classifier is an important evaluation indicator for evaluating the classification performance of the model. Accuracy is often the use of classifiers to measure the classification accuracy of test data, which can estimate the ability of a given classifier to correctly classify data. The performance of the classifier is usually measured by the F1 value, the receiver operating characteristic ROC [10] (Receiver Operating Characteristic) curve or the area under the curve AUC [11] (Area Under the Curve). The ROC curve is mainly used as a performance evaluation indicator for the binary classification model to illustrate the relationship between the classifier hit rate and the false positive rate. For a second type of problem where the data is divided into positive and negative cases, one prediction may produce four different results, as shown in Table 3.

Table 3. Binary classification confusion matrix

Forecast category	Actual category	
	Positive	Negative
Positive	TruePositive	FalseNegative
Negative	FalseNegative	FalsePositive

True Positive (TP) and True Negative (TN) are correct classifications. The false positive (False Positive, FP) occurs when the negative case is predicted to be a positive case, and the false negative case (False Negative, FN) occurs when the positive case is predicted to be negative. Define the true rate (True Positive Rate, TPR) as the number of TP divided by the total number of positive cases, and the negative positive rate (False Positive Rate, FPR) is the number of FP divided by the total number of negative cases:

$$\mathrm{TPR} = \frac{TP}{TP+FN} \times 100\% \tag{6}$$

$$\mathrm{FPR} = \frac{FP}{FP+TN} \times 100\% \tag{7}$$

A typical ROC curve is a graph in which the vertical axis represents the true rate and the horizontal axis represents the false positive rate. In order to summarize the ROC curve into a single metric, AUC can be used. The value of AUC is the area under the ROC curve, and the value range is generally between 0.5 and 1. The larger the AUC, the better the model distinguishing ability. AUC metrics are usually applied to binary classification problems. If multivariate classification problems need to be extended, the literature [6] adopts the "1 vs other" method to evaluate multi-class model performance, namely:

$$AUC_{average} = \frac{1}{K} . \sum_{i=1}^{k} AUC(C_i, C_i') \tag{8}$$

Where K is the number of categories, C_i is the i-th class, and C_i' is a class consisting of other classes except C_i, so that a binary AUC can be obtained. K-class classification yields K AUCs, taking the mean as the final AUC value of the model. The AUC indicator is used to evaluate the classification accuracy of a single decision tree, and the decision tree is sorted according to the value, and a part of the tree with a lower AUC value is discarded, and a tree with a higher precision is reserved to form a sub-forest.

5.2 Diversity

Existing theoretical and experimental studies can prove that a random forest model composed of multiple decision trees is more generalized than a single decision tree, and independent, complementary and relatively accurate decision trees. The random forest obtained by integration is superior to the best performance decision tree in generalization performance. There are currently no uniform evaluation criteria for classifier diversity metrics, and most of the diversity metrics used in the study are Kappa statistic [43].

Kappa statistic can measure the classification accuracy, and can also be used to judge whether different models or analysis methods are consistent in the prediction results. The calculation of Kappa statistic is also based on the confusion matrix, and its calculation formula is:

$$\mathrm{K} = \frac{P_r(a)-P_r(e)}{1-P_r(e)} = 1 - \frac{1-P_r(a)}{1-P_r(e)} \tag{9}$$

Among them, Pr(a) represents the actual consistency of the two classifiers, and Pr(e) represents the theoretical consistency of the two classifiers, which are defined as follows: L represents the number of categories of the test set, and m represents all samples of the test set. The number, C_{ij}, represents the number of samples in the prediction result that are marked as class i by the first classifier and as class j by the second classifier.

$$P_r(a) = \frac{\sum_{i=1}^{L} C_{ii}}{m} \tag{10}$$

$$P_r(e) = \sum_{i=1}^{L} \left(\sum_{j=1}^{L} \frac{C_{ii}}{m} \times \sum_{j=1}^{L} \frac{C_{ji}}{m} \right) \tag{11}$$

The Kappa value ranges from -1 to +1, and the higher the value, the stronger the consistency. K=1 means that the two classifiers have the same prediction for the same sample, K=0 means that the consistency is the same as the accidental expectation. When K<0, the consistency is weaker than expected, but this is rarely seen. Kind of situation. Clustering is often used when data is not labeled, but has similar similarities. It divides the data according to a similarity measure. Each data sample belongs to a cluster, which is equivalent to the unsupervised form of the classification model. If the clustering algorithm is used to cluster the decision tree clusters, the similar trees are clustered into a cluster, and then the representative trees are selected from each cluster to form a sub-forest. In order to better represent the distance between classifiers, the normalized Kappa statistic as a cluster distance measure can be expressed by Equation 10 [13]. Among them, the closer K* is to 0, the more similar the classifier is.

$$K^* = 0.5 \times (1 - K) \tag{12}$$

6. Summary

The random forest algorithm is a representative algorithm in the decision-making algorithm. In the current situation of artificial intelligence big data, more and more scholars of machine learning algorithms are sought after, and random forest algorithms are widely used by people. Therefore, this paper summarizes the random forest construction process, related algorithm optimization and algorithm performance indicators. However, as a widely used algorithm, it is worthy of further study on how to improve the classification accuracy and how to deal with complex data sets.

Acknowledgements

Fund project: Inner Mongolia Autonomous Region Science and Technology Innovation Guidance Project (KCBJ2018006): Construction of postal industry supervision system and service platform based on big data.

Inner Mongolia Autonomous Region Science and Technology Plan Project (20140712): Realization of Noise Reduction Design and Algorithm Based on Embedded Underground Pipeline Leak Detection Technology.

Inner Mongolia Autonomous Region Education Department Project (NJZY17042): Research on Key Issues of Wireless Sensor Network Location Technology.

Inner Mongolia Autonomous Region Science and Technology Plan Project (201707): Design and Implementation of Leakage Monitoring System for Underground Tap Water Supply Pipeline Based on Internet of Things.

References

[1] Zhang Wei, Huang Wei, Hu Guochao. Application of Character Recognition Based on Improved Random Forest in Money Laundering[J]. Computer and Modernization, 2018(02): 101-106.

[2] Xia Xiuchen, Wang Xiuying. Improved C4.5 Decision Tree Algorithm Based on Cosine Similarity [J]. Computer Engineering and Design, 2018, 39(01): 120-125.

[3] Wen Bowen, Dong Wenbiao, Jie Wujie, Ma Jun. Optimization of random forest parameters based on improved grid search algorithm[J]. Computer Engineering and Applications, 2018, 54(10): 154-157.

[4] Liu Jiangyan, Chen Huanxin, Wang Jiangyu, Li Guannan, Shi Shuzhen. A Method for Temperature Time Series Prediction in Subway Station Based on Data Mining Algorithm[J]. Journal of Engineering Thermophysics,2018,39(06):1316-1321.

[5] Li Fangju, Lin Nan. Improved Moment Feature and Random Forest Algorithm Vehicle Classification[J]. Computer Engineering and Design, 2018, 39(06): 1664-1668+1684.

[6] Chen Weimin, Zhang Ling, Song Dongmei, Wang Bin, Ding Yaxiong, Xu Mingming, Cui Jianyong. Study on the method of improving the classification of hyperspectral imagery based on AdaBoost[J]. Remote Sensing Technology and Application,2018,33(04): 612-620.

[7] Zhang Zhiwei, Ji Yuanyuan, Man Weishi. Improved image classification application of random forest algorithm[J]. Computer Systems, 2018, 27(09): 193-198.

[8] Xu Dong, Wang Yanjun, Meng Yulong, Zhang Ziying. Improved Data Anomaly Detection Method Based on Isolation Forest[J]. Computer Science,2018,45(10):155-159.

[9] Xie Zhizhen, Wen Ruigang, Meng Anbo, Yin Hao, Liu Zhe. Research on data processing and prediction of construction people based on box shape and isolated forest[J]. Journal of Engineering Management, 2018, 32(05): 92-96.

[10] Sun Guangmin, Liu Hao, He Cunfu, Li Wei, Li Zibo, Liu Xiucheng, Zhang Ruihuan, Lu Haonan. Hardness Prediction of Ferromagnetic Materials Based on Improved Random Forest Algorithm[J]. Journal of Beijing Polytechnic University,2019,45(02) :119-125.

[11] Wu Shanfeng, Lu Xia. Design and Management System Design of Physical Education Curriculum Based on Decision Tree Algorithm[J]. Electronic,2019,42(03):131-133+138.

[12] HAN Qidi, ZHANG Xiaotong, SHEN Wei. Application of Support Vector Machine Based on Decision Tree Feature Extraction in Lithology Classification[J]. Journal of Jilin University (Earth Science Edition),2019,49(02):611-620.

[13] You Jiewen, Zou Bin, Zhao Xiuge, Xu Shan, He Rui. Estimation of NO-2 concentration in China near ground based on random forest model[J]. China Environmental Science, 2019, 39 (03):969-979.

[14] Xie Guorong, Zheng Hong, Lin Weiwei, Xu Ming, Guo Kun, Chen Jijie. Classification of Power-off Sensitive Users Based on Improved Random Forest Algorithm[J]. Computer Systems, 2019, 28(03): 104-110.

[15] Dong Na, Chang Jianfang, Wu Aiguo. A Random Forest Prediction Method Based on Bayesian Model Combination [J]. Journal of Hunan University (Natural Science), 2019, 46 (02): 123-130.

[16] Yang Xiugang. Overview of Data Mining Algorithms [J]. Science and Technology Economics Guide, 2019, 27 (05): 166.

[17] Shang Tao, Zhao Wei, Shu Wangwei, Liu Jianwei. Big Data Decision Tree Algorithm Based on Equal-Beat Privacy Budget Allocation[J]. Engineering Science and Technology, 2019, 51 (02):130-136.

[18] Fang Wei, Li Xudong, Cao Haiyan, Pan Peng. Stock Trend Prediction Based on Improved Random Forest Algorithm[J]. Journal of Hangzhou Dianzi University (Natural Science Edition), 2019,39(02):22-27.

[19] Xiao Qi, Su Kaiyu. Traffic Detection of Botnet Based on Random Forest[J]. Microelectronics & Computer, 2019, 36 (03): 43-47.

[20] Chen Wei, Ma Xiangping, Jia Chengfeng, Zhang Jie. Research on Talents' Attraction Policy Based on Decision Tree ID3 Algorithm[J]. Journal of Wuhan University of Technology (Information and Management Engineering), 2019, 41(02): 148 -153.

[21] Jiang Guoqing, Zhao Meng, Yang Tao, Peng Ruxiang, Kong Huafeng. Research on Feature Selection Method Based on Global Terrorism Database[J]. Computer Applications and Software, 2019, 36(04): 51-54.

[22] Zhang Xiaolong, Peng Yi. Audio Recognition Method Based on Residual Network and Random Forest[J]. Computer Engineering and Science, 2019, 41(04): 727-732.

[23] Ou Huajie. A review of machine learning algorithms in the context of big data [J]. China Informatization, 2019 (04): 50-51.

[24] Zhang Yuyu, Ni Rongrong, Yang Wei. Application of Improved Random Forest Classifier in RGBD Facial Expressions[J]. Journal of Nanjing Normal University (Natural Science Edition), 2019, 42 (01):82-89.

[25] Zhu Qing, Lin Jianping, Guo Jiaxin, Guo Xi. Information Extraction and Dynamic Monitoring of Rare Earth Area Based on Image Feature CART Decision Tree[J]. Metal, 2019 (05):161-169.

[26] Xue Minglong, Li Yibo. Intelligent Environment Activity Recognition Based on Improved Random Forest Algorithm[J]. Computer Engineering, 2019, 45(05): 149-154.

[27] Zhang Huining. Application of Decision Tree in Intelligent Management Platform of University Laboratory [J]. Electronic Technology and Software Engineering, 2019(10): 188-189.

[28] LIU Yong, XING Yanyun. Research and Application of Text Classification Based on Improved Random Forest Algorithm[J]. Computer Systems, 2019, 28(05): 220-225.

[29] Li Mingbo. Overview of Target Detection Algorithm Based on Machine Learning[J]. Computer Products and Circulation,2019(06):154-155.

[30] Fan Yeping, Li Yu, Yang Desheng, Wan Tao, Ma Dong, Li Wei. Face Intelligent Feedback Cognitive Method Based on Deep Integration Learning[J]. Application of Electronic Technology, 2019, 45(05): 5-8+13.

[31] Technology - Green Technology; Findings from Cardiff University Yields New Findings on Green Technology (Predictive Modelling for Solar Thermal Energy Systems: A Comparison of Support Vector Regression, Random Forest, Extra Trees and Regression Trees) [J]. Energy Weekly News, 2019.

[32] Zhao Xun, Wu Yanhong, Lee Dik Lun, Cui Weiwei. iForest: Interpreting Random Forests via Visual Analytics. [J]. IEEE transactions on visualization and computer graphics, 2018.

[33] Han Cunge, Ye Xuesun. Research and Improvement of C4.5 Algorithm in Decision Tree Classification Algorithm[J]. Computer Systems, 2019, 28(06): 198-202.

[34] Li Xiufang, Huang Zhiguo, Chen Xiaowei. Application of Bagging Integration Method in Insurance Fraud Identification[J]. Insurance Research, 2019(04): 66-84.

[35] Liu Yunxiang, Chen Bin, Zhou Ziyi. An Improved Feature Screening Algorithm Based on Random Forest[J]. Modern Electronic Technology, 2019, 42(12): 117-121.

[36] Tian Runze. Boston House Price Forecast Based on Multiple Machine Learning Algorithms[J]. China New Communications,2019,21(11):228-230.

[37] Lu Rui, Li Linwei. Crime Prediction Model Based on Random Forest[J]. Journal of China Criminal Police College, 2019(03): 108-112.

[38] Zhong Xi, Sun Xiangwei. Research on Naive Bayesian Integration Method Based on Kmeans++ Clustering[J]. Computer Science, 2019, 46(S1): 439-441+451.

[39] Lin Baiquan, Xiao Jing. Multi-criteria recommendation algorithm based on matrix decomposition and random forest[J]. Journal of South China Normal University (Natural Science), 2019, 51 (02): 117-122.

[40] Zhou Xiaoyu, Zhang Longbo, Wang Lei, Li Xinxiang. Medical image segmentation combined with spectral clustering and improved RSF model [J/OL]. Computer Engineering and Applications: 1-8[2019-07-13].

[41] Xu Guanying, Han Meng, Wang Shaofeng, Jia Tao. Overview of data stream integration classification algorithm [J/OL]. Computer Application Research: 1-11[2019-07-13].

[42] Li Xuran, Ding Xiaohong. A review of the five major categories of machine learning and their main algorithms [J/OL]. Software Guide: 1-6 [2019-07-13].

[43] Yin Ru, Men Changyi, Wang Wenjian. A Model Decision Forest Algorithm [J/OL]. Computer Science and Exploration: 1-11[2019-07-13].

[44] Yu Wei, Cao Qi, Liu Tao. Analysis of interactive characteristics of Weibo based on random forest [J/OL]. Computer Technology and Development, 2019(10):1-6[2019-07-13].

[45] Honore Yenwongfai, Nazmul Haque Mondol, Isabelle Lecomte, Jan Inge Faleide, Johan Leutscher. Integrating facies-based Bayesian inversion and supervised machine learning for

petro-facies characterization in the Snadd Formation of the Goliat Field, south-western Barents Sea [J]. Geophysical Prospecting, 2019, 67(4).

[46] Tiejun Yang, Jikun Song, Lei Li. A deep learning model integrating SK-TPCNN and random forests for brain tumor segmentation in MRI [J]. Biocybernetics and Biomedical Engineering, 2019, 39(3).