

PAPER • OPEN ACCESS

Research on XGboost academic forecasting and analysis modelling

To cite this article: Ting Hu and Ting Song 2019 *J. Phys.: Conf. Ser.* **1324** 012091

View the [article online](#) for updates and enhancements.

You may also like

- [GSCV-XGBoost based information reconstruction and fusion method for SINS/DVL integrated navigation system](#)
Hongmao Qin, Ying Wang, Guangcai Wang et al.
- [Predicting Solar Proton Events of Solar Cycles 22–24 Using GOES Proton and Soft-X-Ray Flux Features](#)
Aatiya Ali, Viacheslav Sadykov, Alexander Kosovichev et al.
- [Machine learning models to predict nitrate concentration in a river basin](#)
Diana Yaritza Dorado-Guerra, Gerald Corzo-Pérez, Javier Paredes-Arquiola et al.



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

Research on XGboost academic forecasting and analysis modelling

Ting Hu¹ and Ting Song

College of Aerospace Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing, 210016, China

¹Email: ht_01@nuaa.edu.cn

Abstract. The analysis and evaluation of student achievement is an important part of teaching work and an important part of school routine management. Scientifically analysing and evaluating students' academic achievements can not only enable teachers to accurately grasp the students' learning status, but also enable students to understand their own learning situation, and also provide necessary analysis for teaching management and improving teaching. In order to evaluate students' learning situation comprehensively, objectively and reasonably, this paper uses XGBoost algorithm to classify and evaluate students' performance based on statistical analysis of basic data, and establishes a performance evaluation model. For the curriculum relevance, the student's performance data is statistically compiled according to the statistical knowledge. The subjective and objective structural entropy weight method is used to classify the characteristic importance results, and finally the relevant courses of the completed courses are obtained. For the results of the unfinished course using the completed course results, the XGBoost method is used to predict each student's grades.

1. Introduction

In the school educational administration, how to accurately evaluate the quality of college students' academic completion and further promote the application of academic achievement prediction, regression warning, comprehensive assessment has gradually attracted the attention of data analysis researchers [1]. Achievement is not only an important indicator for assessing the quality of teaching in universities, but also closely related to student management and employment guidance. It's important to make a comprehensive and objective analysis of student's achievements, avoid "partial focus" caused by subjective factors, teach students in accordance with their aptitude, and guide them scientifically. Effective prediction of students' achievements and timely intervention can provide an important basis for the dynamic guidance of students' learning thoughts and the improvement of teaching quality [2].

How to maximize the use of the available data (mainly students' scores), reasonable analysis and modelling predictions are of great significance for improving the quality of teaching [3]. Some facts need to be considered in the forecasting process:

- 1) Course completion scores often include two parts: usual grades and test scores.
- 2) Different majors should have different teaching plans and take different courses.
- 3) There is a situation in which a course has no grades at the end of the semester due to violation of discipline, lack of exams, and delays in exams.



4) There may be changes in the corresponding teaching arrangements for students of different grades, such as changes in class hours, course type, etc.

Considering these facts, we analyze the correlation between the achievements of the follow-up courses and those of the completed courses, and find out the relevant courses. On the basis of the foregoing, this correlation is further clarified and the statistical law describing the correlation is obtained. We are considering how to use the results of closed courses to predict the results of non-open courses [4].

In the field of academic forecasting, there are many studies that use machine learning to predict student academic performance. Shaobo Huang and Ning Fang focus on developing a predictive model to forecast students' academic performance in an introductory engineering course titled Engineering Dynamics. They utilize multivariate linear regression (MLR), multilayer perceptron (MLP) neural networks, radial basis function (RBF) neural networks, and support vector machines (SVMs) to realize their objective. On the problem of how to use the student academic scores of the previous semester to predict the student's grades in the current semester, Neural Network and Decision Tree were used by Halde R R, Deshpande A, Mahajan A for numeric prediction of CGPA and for classification of failures. Meanwhile Al-Saleem M, Al-Kathiry N, Al-Osimi S make use of decision tree classification algorithms to build a performance prediction model based on previous students' academic records. All the above studies provide guidance and direction for our research.

In the application of the XG boost algorithm, the researches developed by Chen T, Guestrin C and Chen T, He T, Benesty M. guide us to flexible utilize XG boost to resolve problems in curriculum classification and prediction.

This paper focuses on a comprehensive analysis of the achievements of students majored in mainframe engineering based on the results of students of Grade 15 in Nanjing University of Aeronautics and Astronautics in the previous semesters. We have obtained the correlations among advanced mathematics, engineering graphics, C++ language programming, college physics, theoretical mechanics, electrical and electronic technology, engineering materials, linear algebra, material mechanics, mechanical principles, mechanical design, interchangeability and technical measurement, hydraulic and pneumatic transmission, etc. On the basis of the correlations, we have predicted and evaluated the students' follow-up achievements, and established forecasting model.

2. Course relevance model and solution

2.1. Model assumptions

- 1) Assume that the grade of each semester is 100 points.
- 2) Assume that each student is in the same learning and examination environment.
- 3) Assume that each student's learning ability remains basically unchanged.
- 4) Assume that the difficulty of each test paper is the same.
- 5) The students' unusual and abnormal performance in the examination does not take into account.
- 6) Assume that all test scores can reflect the true level of the students, without considering the impact of cheating on the test results [5].

2.2. Symbolic description

Table 1. Symbol description table.

Symbol	Interpretative statement
m	Feature dimension
n	Sample size
$x_i (i = 1, 2, \dots, n)$	The i-th sample (Eigenvector) representing the overall score data for the i-th student
$x_i^{(j)} (j = 1, 2, \dots, m)$	The j-th feature of the i-th sample represents the results of the j-th course
$y_i (i = 1, 2, \dots, n)$	Label value of the i-th sample used to represent the classification result
$X_{n \times m}$	Training Sample Matrix (Characteristic Matrix)
$Y_{n \times 1}$	Training sample label
$T = \{(x_i, y_i), i = 1, 2, \dots, n\}$	Training Data Set

2.3. Model establishment and solution

In order to establish a performance evaluation model, it is required to analyze the correlation between the results of the follow-up courses and the completed courses, and find out the relevant courses. The given data are divided into different semesters, so we take the semester courses as the completed courses first, and then the semester courses as the unfinished courses.

Based on the data and the classification method, the results are classified as follows: 60 points is the first category, 60~70 points is the second category, 70~80 points is the third category, 80~90 points is the fourth category, 90~100 points is the fifth category and 100 points is the sixth category. Due to the large number of semesters, in order to improve the accuracy of the model, we use a circular form as shown in the following Table 2:

Table 2. Completed and unfinished course forms.

Completed Courses (Terms)	Unfinished Courses (Terms)
1, 2, 3	4
1, 2, 3, 4	5
1, 2, 3, 4, 5	6

Taking the completed courses as training data. Each student's grade is a sample in the data set, and the scores of different completed subjects are considered as features. In order to establish a classification model, the six categories are used as the labels. One of the unfinished courses is chosen at a time as a target for classification or prediction. We use the XGBoost algorithm to classify students' grades and establish a performance evaluation model.

We use the XGBoost algorithms to analyse the relevance of the course, and the XGBoost algorithm is used to evaluate the relevant courses. Based on the level of characteristic importance degree, we conduct evaluation and analysis of relevant courses. The characteristic importance degree is defined as the sum of the number of times that the characteristics split in each tree. Using the structure-entropy weight method of subjective and objective combination to classify the results of characteristic importance degree, we analyse the relevant completed courses for each target courses.

2.3.1. Establishment of hypothesis function. For the training data of the n samples (course scores of n students) and m -dimensional features (m completed courses), one-hot representation is used for each class.

$$T = \{(x_i, y_i)\} (i=1, 2, \dots, n, x_i \in \mathbb{R}^m, y_i \in (0, 1, 2, 3, 4, 5))$$

According to Table 1, $x_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(m)})^T$, the i -th student's sample including the grades of m completed course. y_i , the label of x_i .

The hypothetical function of the output value of the input sample is represented by the tree ensemble model (the process of adding multiple tree models), as follows:

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^K f_k(x_i), f_k \in F \quad (1)$$

$$F = \{f(x) = \omega_{q(x)}\} (q: \mathbb{R}^m \rightarrow T, \omega \in \mathbb{R}^T) \quad (2)$$

Among them, f_k is the k -th tree model. K is the number of tree models. F is the hypothesis space. q maps the sample instances to leaf nodes. T is the number of leaf nodes. ω is the fraction of leaf nodes. $\omega_{q(x)}$ is the predicted value of an independent tree for sample instances.

2.3.2. Establishment of regularized objective function. In order to learn the functions required by the model and prevent overfitting, we need to define the regularized objective function as follows:

$$L(\phi) = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (3)$$

Among it, $\Omega(f) = \gamma T + \frac{1}{2} \lambda \|\omega\|^2$. l is a convex loss function, which indicates the error of predicted value \hat{y}_i and label value y_i , and Ω represents the complexity of the model.

The next step is to solve this optimization problem.

2.3.3. Gradient lifting tree. Because the integration model of trees can't find the appropriate solution in the traditional Euclidean space, it should find the approximate solution through iteration.

Let $\hat{y}_i^{(t)}$ denote the approximation of the i -th sample in the t -th iteration, so that the iteration updating term f_t can be added to minimize the objective function.

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (4)$$

Expand the formula (4) by Taylor Formula at $\hat{y}_i^{(t-1)}$:

$$L^{(t)} \approx \sum_{i=1}^n \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \quad (5)$$

Among it, $g_i = \frac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}$, $h_i = \frac{\partial^2 l(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2}$. Remove the constant term:

$$\tilde{L}^{(t)} = \sum_{i=1}^n \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \quad (6)$$

$I_j = \{i | q(x_i) = j\}$ is defined as the sample set of leaf node j and the above formula is converted into the following form:

$$\begin{aligned}\tilde{L}^{(t)} &= \sum_{i=1}^n \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T \omega_j^2 \\ &= \sum_{j=1}^T \left[\left(\sum_{i \in I_j} g_i \right) \omega_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) \omega_j^2 \right] + \gamma T\end{aligned}\quad (7)$$

For the given tree structure $q(x)$, the weight of each leaf node can be calculated as follows:

$$\omega_j^* = - \frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \quad (8)$$

Then the optimal value of the objective function is

$$\tilde{L}^{(t)}(q) = - \frac{1}{2} \sum_{j=1}^T \frac{\left(\sum_{i \in I_j} g_i \right)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \quad (9)$$

The above formula can be used to measure the quality of tree structure q .

2.3.4. The solution of tree structure. (1) It is impossible to list all possible tree structures. Branches should be added iteratively through **greedy algorithm** on the initial leaf nodes.

Let I_L and I_R be the divided set of left and right nodes ($I = I_L \cup I_R$), so the loss function becomes

$$L_{\text{split}} = \frac{1}{2} \left[\frac{\left(\sum_{i \in I_L} g_i \right)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{\left(\sum_{i \in I_R} g_i \right)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{\left(\sum_{i \in I} g_i \right)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (10)$$

As an evaluation index for the evaluation, it is only necessary to maximize the above formula. We can get the greedy algorithm for segmentation points:

Input: I (Sample set of the current node)

Input: m (Feature dimension)

$gain \leftarrow 0$

$G \leftarrow \sum_{i \in I} g_i, H \leftarrow \sum_{i \in I} h_i$

for $k = 1$ to m do

$G_L \leftarrow 0, H_L \leftarrow 0$

for j in sorted($I, \text{bg } x_{jk}$) do

$G_L \leftarrow G_L + g_j, H_L \leftarrow H_L + h_j$

$G_R \leftarrow G - G_L, H_R \leftarrow H - H_L$

$score \leftarrow \max \left(score, \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right)$

end

end

Output: Breakdown value of maximum score.

(2) Boundary point finding algorithms in eigenvalue domain

The boundary point of the eigenvalue domain is $\{s_{k1}, s_{k2}, \dots, s_{kl}\}$, and the weighted quantile method is adopted. In order to approximate the optimal splitting point as much as possible, we need to ensure that the data distribution after sampling is as consistent as possible with the original data. $D_k = \{(x_{1k}, h_1), (x_{2k}, h_2), \dots, (x_{nk}, h_n)\}$ denotes the k-dimensional eigenvalues of each training sample and the corresponding second-order derivatives. Next, the sort function is defined as $r_k(\bullet): \mathbb{R} \rightarrow [0, +\infty]$, i.e.:

$$r_k(z) = \frac{1}{\sum_{(x,h) \in D_k} h} \sum_{(x,h) \in D_k, x < z} h \quad (11)$$

This function represents the proportion of sample distribution whose feature value is less than z , in which the second derivative h can be regarded as the weight. Under this sort function, we find a set of points $\{s_{k1}, s_{k2}, \dots, s_{kl}\}$ satisfies

$$|r_k(s_{k,j}) - r_k(s_{k,j+1})| < \varepsilon \quad (12)$$

Among it, $s_{k1} = \min_i x_{ik}$, $s_{kl} = \max_i x_{ik}$. ε is the sampling rate, and finally we get $\frac{1}{\varepsilon}$ dividing points due to the following formula.

$$\sum_{i=1}^n \left[\frac{1}{2} (f_i(x_i) - \frac{g_i}{h_i})^2 \right] + \Omega(f_i) + \text{constant} \quad (13)$$

2.3.5. Computing method of correlation. Based on the training data, XGBoost is adopted to evaluate the correlation between various courses. Through the algorithm, the feature importance which is defined as the sum of the number of times that the characteristics split in each tree can be calculated and compared. The higher the feature importance is, the higher its relevance to the target course.

Using the structural entropy weight method, we analyse the weight ratio and obtain the threshold of feature importance which is used to judge the correlation between courses. We consider the course of which the feature importance is above the threshold to be related to the target.

The structural entropy weight method combining subjective and objective can reduce the influence of subjective factors on objective facts. Specific steps are as follows.

For the target course, using different classification criteria, the weights of different division criteria will be generated, and the one with the highest weight will be the final classification criterion. The weight calculation uses the combination of "Delphi method" and entropy method to reduce the uncertainty of weight sorting.

(1) Determine the importance ordering matrix using the Delphi method.

A partitioning scheme **I** consists of n partitioning criteria. Suppose there are k experts involved in the consultation, each expert sorts the importance of each division standard according to experience and knowledge, corresponding to the expert sort number vector $A(i)$. Where a_{ij} denotes the sorting number of the j -th criterion by the i -th expert. After multiple evaluations by k experts, a typical sorting matrix **A** can be formed.

$$A(i) = (a_{i1}, \dots, a_{ij}, \dots, a_{in})$$

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & a_{ij} & \dots \\ a_{k1} & a_{k2} & \dots & a_{kn} \end{bmatrix}$$

(2) Calculate weight using entropy value information

Based on the ranking matrix A , the entropy theory is used to reduce the uncertainty caused by different expert cognitions, reduce the bias caused by subjective factors, and then calculate the relatively stable weight of the indicators.

The weight vector obtained by dividing scheme I is $w_I = (\alpha_1, \alpha_2, \dots, \alpha_n)$.

By analyzing the weight of the weight, we will match the threshold of feature importance which can find the corresponding course.

We use 12 completed courses as features to analyse their relevance to the course of Mechanics of Materials I. The result is shown as the following diagram.

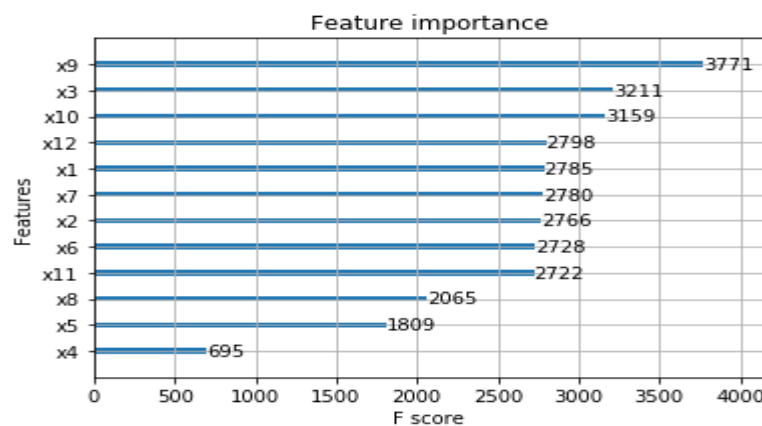


Figure 1. Feature importance of 12 features for Mechanics of Materials I.

Table 3. Standard table for classification of Mechanics of Materials I.

Classification standard	Weight
>3500	0.3472
>3000	0.4423
>2500	0.1294
>2000	0.0811

Table 3 shows that when the label is Mechanics of Materials I, the threshold of feature importance is 3000, in this case, the weight of these relevant courses can reach about 80%. If the feature is greater than 3000, it is considered to be related to Mechanics of Materials I. According to Figure 1, the feature related to Mechanics of Materials I are x_{10} , x_3 , x_9 . They represent the following courses: Theoretical Mechanics I, C++ Language Programming and Electrical and Electronic Technology II (1).

3. Model and solution of achievement prediction

3.1. Modeling

In the model of Table Problem 1, the predicted results of the open class have been divided into six categories, and the concept of classified decision tree is introduced into the process of solving the importance of sample features. Through the XG Boost algorithm, we found courses that were closely

related to each course before it was opened. We will use the algorithm of the classification decision tree to use the information of these courses to predict the result of the course.

Classification decision tree model is a tree structure used to classify instances. Decision trees are composed of nodes and directed edges. There are two types of nodes: internal nodes and leaf nodes. Internal nodes represent one feature or attribute, and leaf nodes represent one class. In this paper, the completed course scores are taken as the characteristics, and the leaf nodes represent the classification of the predicted results of the follow-up courses.

Decision trees represent conditional probability distributions of classes under given feature conditions. This conditional probability distribution is defined on a partition of feature space. The feature space is divided into disjoint units or regions, and the probability distribution of a class is defined in each unit to form a conditional probability distribution. A path of the decision tree corresponds to a unit in the partition. The conditional probability distribution represented by decision tree is composed of conditional probability distribution of classes under the given conditions of each unit. In this paper, it represents the conditional probability of the grades of follow-up courses in one of the six categories of grades under certain conditions. Assuming that X is a random variable representing the final grade and Y is a random variable representing six types of follow-up grades, the conditional probability distribution can be expressed as $P(Y | X)$. X is taken as the set of units under a given partition, and Y is taken as the set of classes. The conditional probability of each leaf node tends to lean to a certain class, that is, the probability belonging to a certain class is larger. In decision tree classification, the instances of the node are forcibly classified into those with high conditional probability.

Question 1 has used the XGBoost algorithm to get the scores of six types of follow-up courses. At this time, according to the softmax function, we can calculate the probability that the predicted results of the follow-up courses belong to the category i .

$a_1, a_2, a_3, a_4, a_5, a_6$ is set as the output category. For each student's score sample, the probability that it belongs to a category is

$$y_i = \frac{e^{a_i}}{\sum_{k=1}^6 e^{a_k}} \quad \forall i \in 1 \dots 6 \quad (14)$$

By deriving the softmax function and substituting the expression of the softmax function, we can get:

$$\frac{\partial y_i}{\partial a_i} = \frac{\partial \frac{e^{a_i}}{\sum_{k=1}^6 e^{a_k}}}{\partial a_j} \quad (15)$$

When $i = j$:

$$\frac{\partial y_i}{\partial a_i} = \frac{\partial \frac{e^{a_i}}{\sum_{k=1}^6 e^{a_k}}}{\partial a_j} = \frac{e^{a_j} \sum_{k=1}^6 -e^{a_i} e^{a_j}}{\sum^2} = \frac{e^{a_i}}{\sum} \frac{\sum -e^{a_j}}{\sum} = y_i (1 - y_j) \quad (16)$$

When $i \neq j$:

$$\frac{\partial y_i}{\partial a_i} = \frac{\partial \frac{e^{a_i}}{\sum_{k=1}^6 e^{a_k}}}{\partial a_j} = \frac{0 - e^{a_i} e^{a_j}}{\sum^2} = -\frac{e^{a_i}}{\sum} \frac{e^{a_j}}{\sum} = -y_i y_j \quad (17)$$

The above formulas satisfy $\sum = \sum_{k=1}^6 e^{a_k}$.

Through a series of derivations, the probability distribution of the predicted follow-up course scores in each category can be obtained. The class with the highest probability is taken as the result of

the follow-up prediction, and its probability is taken as the statistical probability distribution of the correlation between the completed courses and the follow-up courses.

In order to verify the accuracy of the model in predicting students' performance, the accuracy of the defined forecast is the number of students who are in the same class as the unscheduled grades predicted based on the results of the completed course, divided by the total number of students in all students. If θ is the accuracy of prediction, there are:

$$\theta = \frac{n}{N} \quad (18)$$

Among it, n denotes the number of students in the same category as the number of students whose grades are predicted based on the results of the completed courses, and N denotes the total number of students in the sample.

3.2. Model solution

The total number of validated students in this experiment is 145. The results of the completed courses in the first three semesters are used to predict the results of the fourth semester, the first four semesters are used to predict the results of the fifth semester and the first five semesters are used to predict the results of the sixth semester.

There are three courses in the fourth semester, namely, Material Mechanics I, Electrical and Electronic Technology II (2), Mechanical Principles. For Material Mechanics I, 92 people are predicted correctly, and the accuracy rate is 63.45%; for Electrical and Electronic Technology II (2), 92 people are predicted correctly, and the accuracy rate is 63.45%; for Mechanical Principle, 84 people are predicted correctly, and the accuracy rate is 57.93%.

There are two courses in the fifth semester, namely Mechanical Design and Interchangeability Measurement and Technical Measurement. For Mechanical Design, 87 people are correctly predicted, and the accuracy rate is 60%. Similarly, the accuracy rate of Interchangeability Measurement and Technical Measurement is 66.21%.

According to Figure 2, there is a course in the sixth semester, namely Hydraulic and Pneumatic Transmission. For this course, the correct number of people is 80, and the prediction accuracy is 55.17%.

With curricula as abscissa and veracity as ordinate, the histogram can be obtained as follows:

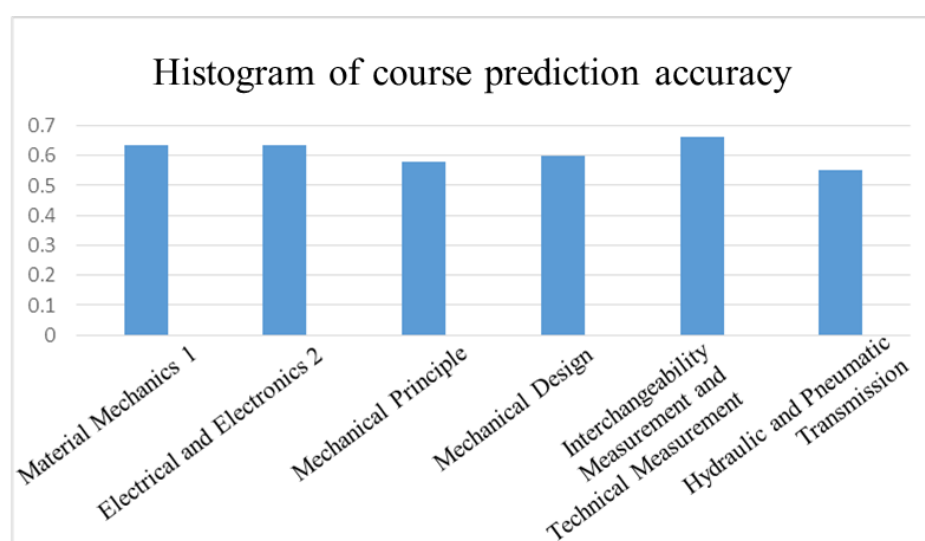


Figure 2. Histogram of course prediction accuracy.

4. Conclusions

This paper uses XGBoost algorithm to find more relevant courses, and draws the following conclusions:

The following courses including Material Mechanics I and Theoretical Mechanics I, C++ Language Programming and Electrical and Electronic Technology II (1), Electrical and Electronic Technology II (2) and Theoretical Mechanics I, Electrical and Electronic Technology II (1), Mechanical Principles and Theoretical Mechanics I, Linear Algebra, Electrical and Electronic Technology II (1) and Engineering Materials, Mechanical Design and Mechanical Principles, Engineering Materials, Materials Mechanics I, Linear Algebra and Engineering Graphics I (2), Hydraulic and Pneumatic Transmission and Interchangeability and Technical Measurements have strong correlation with each other.

The total number of sample students in this paper is 145. The results of the first three semesters are used to predict the scores of the fourth semester. The results of the first four semesters are used to predict the results of the fifth semester. The first five semester grades are used to predict the results of sixth semester. The prediction accuracy is over 55%.

The classification idea is used to divide the student's score into six large intervals, and the regression problem of the score prediction is converted into the classification problem. This allows the model to have better generalization performance. It's permitted that the predicted student grades float up and down. The XGBoost algorithm has the advantages of high speed, high precision and low resource consumption. It can be parallelized to deal with big data. It combines feature importance and expert evaluation method to calculate course relevance. It not only guarantees objective correctness, but also take subjective factors and human prior knowledge into consideration, which can significantly improve the robustness of the model.

The model classifies the scores in only one way, that is, it is divided into 6 categories. In actual situations, there may be other classification methods to better predict the scores and correlations.

Compared with other academic forecasting algorithms, the XG boost algorithm has faster computing speed, higher precision, and takes up less resources. At the same time, when the amount of calculation is large, we can use the XG boost algorithm to implement parallel computing which is difficult for traditional algorithms to achieve.

At the same time, the model uses the feature importance degree and the expert evaluation method to calculate the curriculum relevance, that is, to ensure the objective correctness and consider the subjective factors and human prior knowledge, so as to significantly improve the robustness of the model.

In general, as China's learning and early warning research is still in its infancy, next we will further strengthen the theory, technology and applied research of learning and early warning, and further improve the efficiency and effectiveness of online learning early warning.

References

- [1] Huang S B and Fang N 2011 Work in progress-Prediction of students' academic performance in an introductory engineering course[P]. *Frontiers in Education Conference (FIE)*
- [2] Chen T and Guestrin C 2016 XGBoost: A Scalable Tree Boosting System[J]
- [3] Chen T, He T and Benesty M 2016 XGBoost: Extreme Gradient Boosting[J]
- [4] Halde R R, Deshpande A, Mahajan A 2017 Psychology assisted prediction of academic performance using machine learning[C] *IEEE International Conference on Recent Trends in Electronics IEEE*
- [5] Al-Saleem M, Al-Kathiry N, Al-Osimi S, et al. 2015 Mining Educational Data to Predict Students' Academic Performance[C] *International Workshop on Machine Learning and Data Mining in Pattern Recognition Springer-Verlag New York, Inc.*