Membrane Technology
ISSN (online): 1873-4049

A Comprehensive Score Prediction Method for College Students Based on Multi Model Fusion

Ance Zhao[1], Weixuan Liang[2,3,*], Zhen Wu[2]

[1]*Department of Students Affairs (Baoding), North China Electric Power University, Baoding, P.R.China*
[2]*School of Control and Computer Engineering, North China Electric Power University, Baoding, P.R.China*
[3]*Baoding key Laboratory of State Detection and Optimization Regulation for Integrated Energy System, Baoding, P.R.China*
**Corresponding Author.*

Abstract:
The rapid development of information and artificial intelligence technologies has brought new opportunities for innovation and transformation to higher education. The integration of these technologies with higher education teaching has led to innovative applications of data-driven decision support in teaching. This paper analyzes the characteristics of various machine learning models, comprehensively utilizes the strengths of each model, constructs a stacking model, and optimizes the model through Bayesian hyperparameter optimization to achieve accurate prediction of comprehensive grades for college students. The academic performance of students in a specific major in Central China is used as experimental data for model validation, achieving ideal predictive results. This method is characterized by a minimal feature set, high predictive accuracy, and strong stability. The comprehensive score prediction model, constructed through analysis of a selected specialty collection, holds significant guidance value in assisting teaching and research departments with curriculum and subject development.

Keywords: data-driven modeling, comprehensive score prediction, stacking model, Bayesian hyperparameter optimization, machine learning.

INTRODUCTION

With the rapid development of information and artificial intelligence technology, the global education system is undergoing significant changes[1]. As a key to social development and talent cultivation, higher education is also experiencing innovation and transformation brought by big data technology. Predictive modeling in artificial intelligence is an exciting field that the education community is exploring. By using artificial intelligence-based results, proactive measures are taken to improve the quality of services and teaching experience in the learning process. In 2023, the Chinese higher education department listed "deeply implementing the digital strategy action, and shaping new advantages in the reform and development of higher education" as a key point of work. It emphasized the role of informatization and data technology in promoting the development and modernization of education in the new era[2].

Traditionally, comprehensive student achievement has been used as a means to assess and predict the probability of students' success or failure in specific courses. Through this process, potential dropouts or candidates for further studies and/or career choices are identified[3]. Building a comprehensive academic achievement prediction model can provide important decision-making basis for the higher education department in terms of curriculum setting, teacher allocation, and subject construction, which is an important demand in the context of intelligent education. However, there are challenges such as numerous courses, strong relevance, and difficulty in identifying key courses.[4]

Recently, the education community has been using data mining and machine learning techniques to provide increasingly accurate predictions related to students. This paper constructs a comprehensive academic achievement prediction method for college students based on the idea of multi-model fusion, and completes the following work: 1) Based on the Stacking algorithm framework, it integrates multiple machine learning models to achieve data-driven precise prediction of college students' comprehensive academic level through the integration of model prediction results; 2) It constructs a key feature extraction method, extracting key features that affect model performance from a large amount of data features, that is, key courses that affect comprehensive academic level, thereby providing decision-makers with data and model-based decision support.

RECENT WORK

Academic performance analysis and prediction is an innovative application of information and artificial intelligence technology in the field of higher education, achieving the transformation from teaching experience-based decision-making to data-driven decision-making at the teaching level[5,6]. It is of guiding significance for college teachers to get rid of "empiricism" and to correct the arrangement of teaching courses; it is also of important reference significance for college students to reasonably plan their learning tasks. Previously, scholars have conducted relevant research on data-driven academic performance analysis and

prediction. Mehdi and Nachouki[7] used a neuro-fuzzy model to improve students' learning experience. He et al.[8] used a course correlation classification model based on frequent pattern spectral clustering to predict students' grades in a timely and accurate manner, providing targeted academic guidance for students. Lin and Li[9] proposed a neural network model for student grade level prediction based on an adaptive differential evolution algorithm, providing a basis for teaching decision-making. Nithiyanandam et al.[10] constructed a learning management system to evaluate students' performance in a digital form. Wu[11] constructed a weighted prediction model based on the random forest algorithm, starting from four types of student behavior characteristics, to analyze students' learning situations and assist teachers in mastering students' learning trends. Shen et al.[12] constructed a learning achievement prediction model based on the Adaboost algorithm to analyze and evaluate students' learning progress, and to take various teaching intervention measures to improve the quality of teaching. Ren[13] predicted student grades based on LightGBM and genetic algorithms. Cao and Xie[14] studied and realized the modeling of students' knowledge systems using LSTM networks, and predicted learning achievements by integrating multiple features. The above research approaches academic analysis and prediction from different perspectives, but there is a lack of research on predicting the comprehensive grades of college students in existing literature. Constructing a comprehensive student performance prediction model using student performance data can provide important decision-making basis for higher education departments in course design, faculty allocation, and discipline construction. It is an important demand in the context of intelligent education, but there are challenges such as a wide range of course categories, strong correlation, and difficulty in identifying key courses. The construction of a comprehensive score prediction method for college students based on the idea of multi model fusion in this article has strong innovation.

DATA-DRIVEN MODELS

Linear regression, decision trees, Support Vector Regression (SVR), Extreme Gradient Boosting (XGBoost), and Light Gradient Boosting Machine (LightGBM) are common data-driven modeling methods. Taking the academic achievements of students in a major in the central region of China as an example, apply the above methods to construct a comprehensive academic achievement prediction model, and achieve multi-model fusion through ensemble learning methods to form the final high-performance prediction model. This major, as a comprehensive engineering discipline, covers both strong and weak electricity, spans software and hardware, and is deeply coupled with multiple disciplines such as computer technology, mechanical engineering, and electronic engineering. It is of great importance for serving the development of national industrial applications. The comprehensive academic achievement prediction model integrates and analyzes the data of basic courses, scientifically detects the distribution of students' comprehensive academic achievements, accurately predicts the courses that significantly affect students' achievements, and helps colleges and universities reasonably formulate and reform curriculum teaching arrangements, achieving the optimization of automated talent training.

Linear Regression

Linear regression is one of the most widely used regression analysis methods in the field of artificial intelligence [15,16]. Its basic principle is to mine the linear association between feature variables and target variables through learning the regression coefficients, that is:

$$f(x) = \omega_0 + \omega_1 x_1 + \omega_2 x_2 + \ldots + \omega_n x_n \quad (1)$$

The goal of the linear regression model is to find a set of parameters ω_0 that make:

$$\omega = \arg\min L(\omega) = \arg\min \sum_{i=1}^{M} (x_i \omega - y_i)^2 \quad (2)$$

Where $L(\omega)$ is the loss function, used to evaluate the difference between the model's predicted value and the actual value. The least squares method can minimize the loss function to obtain the best parameter matrix:

$$W = (X^T X)^{-1} X^T Y \quad (3)$$

Decision Tree

A decision tree is a tree-like structure used for classification and regression problems, generated through training[17]. Suppose the training dataset $\Omega = \{(x_1, y_1), (x_2, y_2), \ldots (x_M, y_M)\}$, where x and y are input and output variables, respectively, and y is a continuous variable. For spatial division, a heuristic method is used, randomly selecting the k-th feature variable in the input space as the division variable, and its corresponding feature value s as the division feature, dividing the input space into two regions:

$$\begin{cases}\gamma_1(k,s) = \{x|x^{(k)} \le s\} \\ \gamma_2(k,s) = \{x|x^{(k)} > s\}\end{cases} \quad (4)$$

Assuming that the division has been completed, consider using the mean squared error as the loss function to evaluate the effect of spatial division. Based on the criterion of minimizing the loss function, the optimal division feature and predicted value $\widehat{a_1}, \widehat{a_2}$are obtained:

$$\begin{cases}\widehat{a_1} = ave(y_k|x_k \in \gamma_1(k,s)) \\ \widehat{a_2} = ave(y_k|x_k \in \gamma_2(k,s))\end{cases} \quad (5)$$

In the above process, based on the criterion of minimizing the mean squared error, the combination of division variable k and division feature s is traversed, and the optimal combination is selected:

$$\min_{k,s}\begin{bmatrix}\min_{a_1}\sum_{x_k \in \gamma_1(k,s)}(y_k - a_1)^2 \\ \min_{a_2}\sum_{x_k \in \gamma_2(k,s)}(y_k - a_2)^2\end{bmatrix} \quad (6)$$

The optimal division combination(k,s) is used for spatial division. For the divided regions, the above steps are repeated for optimal division, obtaining regions $I(\gamma_1,\gamma_2,...\gamma_N)$ until no further division is possible, forming a complete regression tree.

$$f(x) = \sum_{n=1}^{N} \widehat{a_n}\, I\,(x \in \gamma_n) \quad (7)$$

Support Vector Regression

SVR is a regression algorithm based on Support Vector Machines (SVM)[18]. Its basic idea is to map the dataset from the original input space to a high-dimensional space to find the best fitting hyperplane. Let ω be the weight vector, φ be the mapping function, x be the feature vector in the original sample space, and b be the bias. The regression model can be expressed as:

$$f(x) = \omega\, \varphi(x) + b \quad (8)$$

A prediction band centered on $f(x)$ with a width of 2ε is constructed, and when the training samples fall into this prediction band, the prediction is considered correct, as shown in Figure 1.

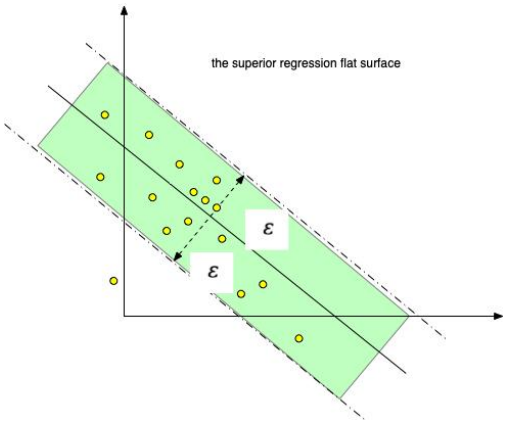


Figure 1. SVR diagram.

The loss function is defined as:

$$l_\varepsilon(z) = \begin{cases} 0, & if\ |z| \le \varepsilon \\ |z| - \varepsilon, & otherwise\end{cases} \quad (9)$$

For parameters ω and b, based on the criterion of minimizing the objective function, we get:

$$\min_{\omega,b}\frac{1}{2}||\omega||^2 + C\sum_{j=1}^{M} l_\varepsilon(f(x_j) - y_j) \quad (10)$$

In the formula, C is the penalty coefficient,M is the number of samples, $f(x_j)$ and y_j are the predicted and actual values of the j-th sample, respectively.

For the mapping function φ, SVR provides the kernel function as a tool. For feature vectors x_m and x_n in the sample space, if there exists $K(x_m, x_n) = \varphi(x_m)^T\varphi(x_n)$, it is called a kernel function. A linear kernel is used as the kernel function, expressed as:

$$K(x_m, x_n) = x_m^T x_n \tag{11}$$

XGBoost

XGBoost is an optimized Boosting algorithm that improves model accuracy by continuously splitting features to fit the residuals of the previous tree[19]. As a gradient boosting decision tree algorithm, the XGBoost algorithm model can be seen as an additive strategy model of decision trees, that is:

$$\widehat{y_p} = \sum_{t=1}^{M} f_t(x_p), f_t \in D \tag{12}$$

In the formula: x_p is the p-th input data, y_p is the corresponding predicted value, M is the number of decision trees, D is the set of decision trees, and $f_t(x_p)$ is the t-th decision tree in the space D. When the model is iteratively learning, it first initializes a predicted value, and each iteration adds a new function f, and the iterative learning process can be expressed as:

$$\begin{gathered} \hat{y}_p^{(0)}=0 \\ \hat{y}_p^{(1)}=f_1(x_p)=\hat{y}_p^{(0)}+f_1(x_p) \\ \cdots \\ \hat{y}_p^{(t)}=\hat{y}_p^{(t-1)}+f_t(x_p) \end{gathered} \tag{13}$$

The objective function (B_{Obj}) of the XGBoost algorithm is not only used to measure the model's fitting error but also includes a regularization term to limit the complexity of each tree, specifically represented as:

$$B_{Obj} = \sum_{p=1}^{n} l(y, \hat{y}) + \sum_{t=1}^{M} \Omega(f_t) \tag{14}$$

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda ||w||^2 \tag{15}$$

In the formula 14, $\sum_{p=1}^{n} l(y, \hat{y})$ is used to measure the model's prediction error, and $\sum_{t=1}^{M} \Omega(f_t)$ is the regularization term used to penalize the complexity of the tree to prevent overfitting.In the formula 15, T is the number of leaf nodes, used to limit the number of nodes and tends towards simple models; w is the score of leaf nodes, and normal form constraints are used to limit the estimated score of each leaf node.

The regularization part of the XGBoost algorithm is also one of the advantages of the algorithm. After the t-th iteration, the objective function can be written as:

$$\Gamma^{(t)} = \sum_{t=1}^{M} l(y_p, \hat{y}_p^{(t-1)} + f_t(x_p)) + \Omega(f_t) \tag{16}$$

Based on the criterion of minimizing the objective function, XGBoost uses the second-order Taylor expansion term to approximate the objective function, treating the objective function as a quadratic function w_j^* of the variables, and then finding the optimal variables and the optimal value of the objective function B_{Obj}^*.

LightGBM

LightGBM shares similarities with XGBoost, as both use the negative gradient of the loss function as an approximate residual value for the current decision tree, fitting a new decision tree[20,21]. At the same time, LightGBM has also made optimizations on this basis. It selects the histogram-based decision tree algorithm, the basic idea of which is: continuously valued feature values are discretized into integers, forming bins, and a histogram with a width of k is constructed. After traversing the data, the discrete values are used as indexes to accumulate statistical information in the histogram, and the optimal split point is found according to the discrete values obtained from the histogram shown in Figure 2.

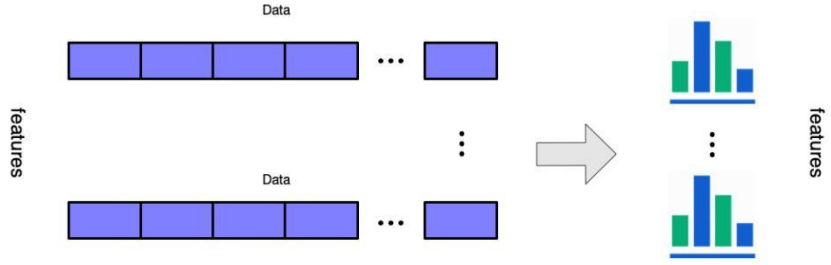


Figure 2. Histogram algorithm diagram.

Since the histogram algorithm does not consume additional storage resources to save the pre-sorting results and only needs the discretized values, LightGBM can effectively reduce memory usage. In addition, since the original feature dataset does not need to be traversed during the running process, only the histogram constructed is calculated, reducing computational difficulty and improving computational efficiency. On the basis of the histogram algorithm, the leaf-wise growth strategy shown in Figure 3 is adopted, where nodes with the largest split gain are selected for splitting first, and the same layer of leaf nodes are selectively treated to further reduce computational overhead.

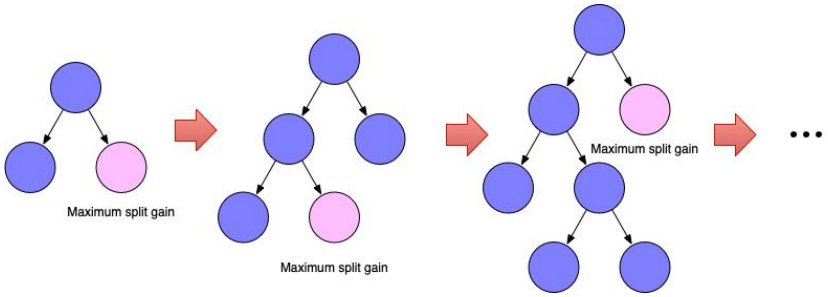


Figure 3. Leaf-wise diagram.

Data-Driven Model Comparative Analysis

The aforementioned data-driven models each have their advantages and disadvantages. The linear regression model is simple and easy to implement, with strong interpretability, but is sensitive to outliers and not well suited to handling highly complex data. The decision tree model is easy to implement with a wide range of applications and is the foundation of tree models, yet it is prone to overfitting and may overlook the intercorrelation of features. SVR can handle high-dimensional data and complex nonlinear issues, with strong generalization capabilities, but the training time for this algorithm model is long, efficiency is low, and interpretability is not strong. XGBoost has high accuracy, and the introduction of regularization reduces the risk of overfitting, but the algorithm's pre-sorting process has a complex space that consumes too much storage space. LightGBM reduces computational complexity and storage consumption with the histogram algorithm, but the algorithm is prone to creating deep decision trees, potentially leading to overfitting and incomplete consideration of features. The comparison and analysis of the strengths and weaknesses of each model are shown in Table 1.

Table 1. Analysis of advantages and disadvantages of data-driven models

Model	Advantages	Disadvantages
Linear Regression	Simple, easy to implement, strong interpretability	Sensitive to outliers, not good at handling high-complexity data
Decision Tree	Easy to implement, wide applicability, foundational tree model	Prone to overfitting, likely to ignore feature correlations
SVR	Can handle high-dimensional data, nonlinear problems	Long training time, low efficiency
XGBoost	High algorithm precision, lower overfitting risk	Consumes storage space
LightGBM	Reduces computational complexity and storage consumption	Prone to produce deeper decision trees

CONSTRUCTION OF COMPREHENSIVE GRADE PREDICTION MODEL BASED ON STACKING

To integrate the advantages of basic data-driven models, a multi-model fusion mechanism based on Stacking is introduced for student grade prediction model construction.

Introduction to Stacking Model

The Stacking model originated from Wolpert's "Stacked Generalizations"[22]. Its basic idea is to construct a two-layer model, using the output results of the first layer models as the input features of the second layer model to achieve the effect of model fusion. That is, the first layer uses various machine learning algorithms, and the predicted values obtained are used as the input features of the meta-model of the second layer, and the final predicted values are output after learning by the meta-model of the second layer. This structure is beneficial for the second layer model to correct the errors of the first layer models. The basic structure of the Stacking model is shown in Figure 4.

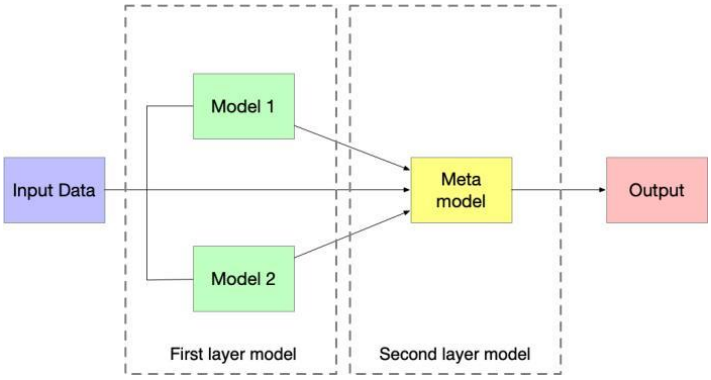


Figure 4. Basic structure of a Stacking model.

Stacking Model Construction

Based on the analysis in the previous text, linear regression models, decision tree models, SVR models, XGBoost models, and LightGBM models are selected as base learning models. They fully utilize the advantages of each model to obtain the best results, and SVR is used as the meta-model to correct the bias of the base learning models. During training, for the given dataset $\Omega = \{(x_m, y_m), m = 1, \ldots, M\}$, where x_m, y_m are the features and predicted quantities of the samples, respectively, after training the base learning models in the first layer, the predicted values of each model in the first layer are obtained. The results are stacked in column form to obtain the comprehensive output results of the first layer models, i.e., the dataset (m, l) with dimension, which is used as the input data for the second layer meta-model for fitting. The model construction process is shown in Figure 5.

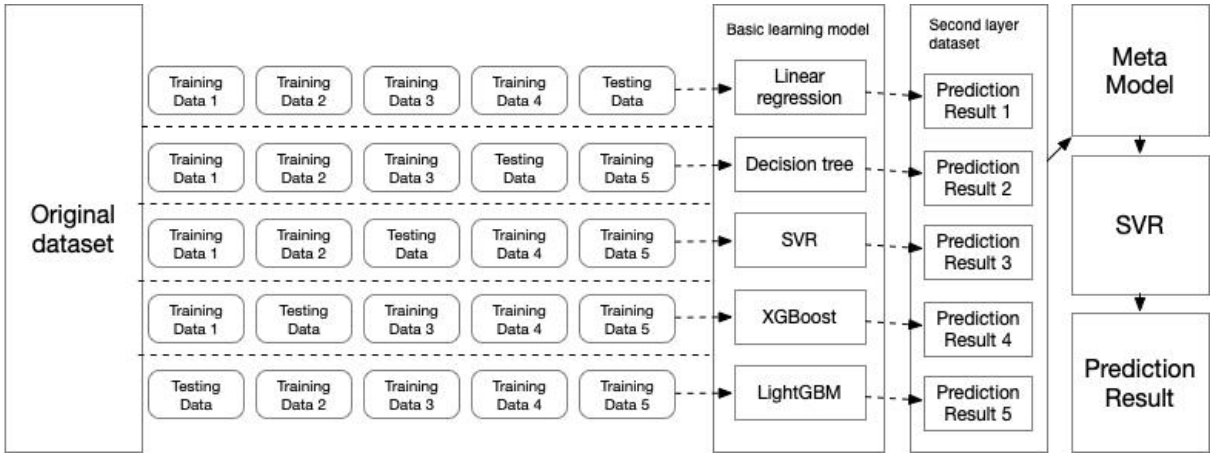


Figure 5. Multi-model fusion method based on Stacking.

To avoid overfitting due to the use of multiple model fusions, cross-validation is introduced to reduce the overfitting risk of the model. A 5-fold cross-validation is used, and the specific process is: the given dataset is divided into training and testing sets, and the dataset $S = \{(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)\}$ is randomly divided into 5 blocks with similar capacities $K_1, K_2, ..K_5$, which are mutually exclusive. Let K_i be the training set for the i-th fold, and K_i' be the testing set. Then 5 iterations are performed, for each base learning model Λ_j in each iteration, there is a corresponding trainer on the training set K_i, and the trainer $\lambda_j^{(i)}$ is used to predict the samples in the testing set, denoted as $\lambda_j^{(i)}(x_n)$. After 5 iterations of each base learning model, there is a layer result $Z_n = \{\lambda_1^{(i)}(x_n), \lambda_2^{(i)}(x_n), \ldots, \lambda_j^{(i)}(x_n)\}$ of the Stacking model, which is also imported as the input dataset into the second layer meta-model, and finally, a stacked model is constructed.

Hyperparameter Optimization

Since the stacked model uses complex structure deep learning models such as decision tree models, the choice of hyperparameters like maximum depth, number of iterations, and subsampling ratio is closely related to the model's precision and quality. Manual hyperparameter selection based on experience is relatively rough and labor-intensive, therefore, Bayesian optimization algorithm is used for hyperparameter tuning. The basic idea is: for a given general optimization objective function, establish a prior distribution, add sample points to update the posterior probability distribution of the objective function, and continuously iterate

to improve and refine the objective function. Compared with conventional grid search tuning, Bayesian optimization algorithm uses Gaussian processes, taking into account prior information, more comprehensive; fewer iterations, faster, and more efficient; not easy to fall into local optima.

Prediction Process

Using Stacking to integrate base learning models, and applying Bayesian optimization to tune the hyperparameters of each model, on the basis that the base learning models already have certain predictive accuracy, as few features as possible are used, and the stacking model selects features to achieve better prediction effects with fewer features[23]. According to the preset idea, import the dataset, use the grades of each course and the corresponding rankings in the dataset to predict the comprehensive grade separately, and sort the output results from large to small, and gradually add features in the order of feature arrangement, using the elbow method to assist in judgment, and select the best feature set, and input it into the SVR model for the final prediction. The complete prediction process is shown in Figure 6.

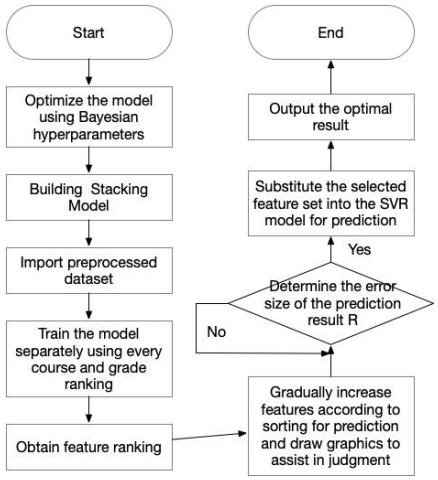


Figure 6. Predicting process of the proposed Stacking model.

EXPERIMENTAL

Data Processing

The experimental dataset consists of academic performance data of students from a certain major in a central region of China. The dataset contains 275 data entries, each representing a student's grades in various courses and a total of 50 features, all from the same academic year. During the data preprocessing phase, feature data were normalized; features with more than 75% missing values were removed to avoid significant impact on the final results; the remaining features with missing values were filled with the mean value of the respective feature column. In addition, to maintain the richness of the dataset and improve the prediction effect, the grades of each course were sorted. Random state settings were used to randomize the data, and the dataset was divided into training and testing sets with an 8:2 ratio, creating the new dataset required for the experiment.

The mean-square error (MSE), mean absolute percentage error (MAPE), and coefficient of determination were used as indicators to evaluate the prediction effect[24,25]. The calculation methods for the indicators are as follows:

$$e_{MSE} = \frac{1}{n}\sum_{q=1}^{1}(y_q - \widehat{y_q})^2 \tag{19}$$

$$e_{MAPE} = \frac{1}{n}\sum_{q=1}^{n}(\frac{y_q - \widehat{y_q}}{y_q}) * 100\% \tag{20}$$

$$R^2 = 1 - \frac{\sum_{q=1}^{n}(y_q - \widehat{y_q})^2}{\sum_{q=1}^{n}(y_q - \bar{y})^2} \tag{21}$$

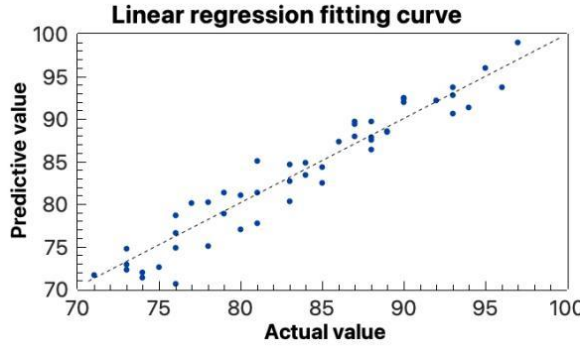
Model Prediction Effect Comparison

For this experimental dataset, based on the idea of assisting teaching arrangements through academic analysis, the experimental direction is to analyze the impact of different courses on the comprehensive academic level. Since there may be linear or nonlinear factors between feature vectors, to fully explore the attributes of input data, basic data-driven models as described

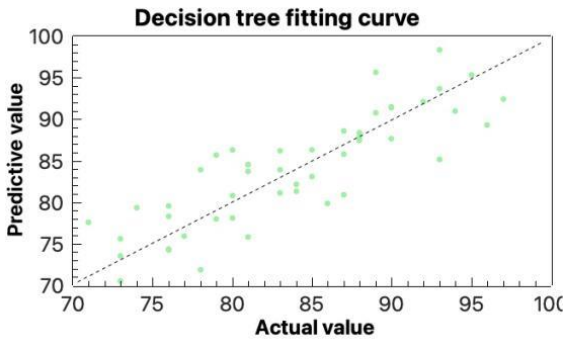
earlier are first used to perform preliminary analysis on the dataset. All 53 course grade features are used to predict the comprehensive grade, and the results are shown in Table 2. The fitting curves of each model are shown in Figure 7.

Table 2. Individual prediction effect of data-driven models

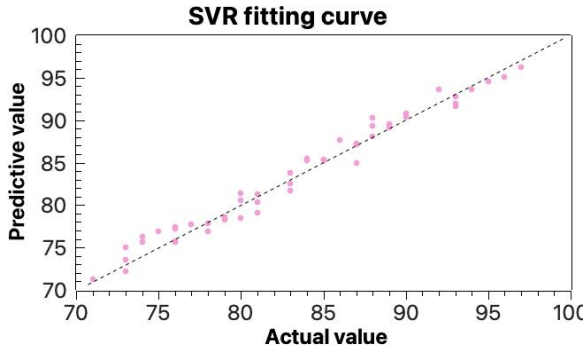
Model Name	e_{MSE}	e_{MAPE}	R^2
Linear Regression	1.6319	0.8762%	0.8729
Decision Tree	4.0013	2.1245%	0.6741
SVR	2.1136	1.1578%	0.8152
XGBoost	2.8462	1.3684%	0.7698
LightGBM	1.5631	1.0882%	0.8435



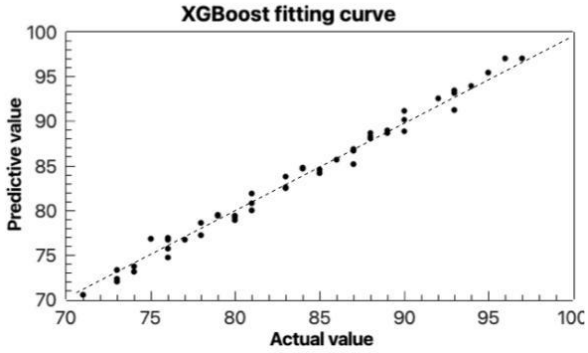
(a) Linear regression fitting curve



(b) Decision tree fitting curve



(c) SVR fitting curve



(d) XGBoost fitting curve

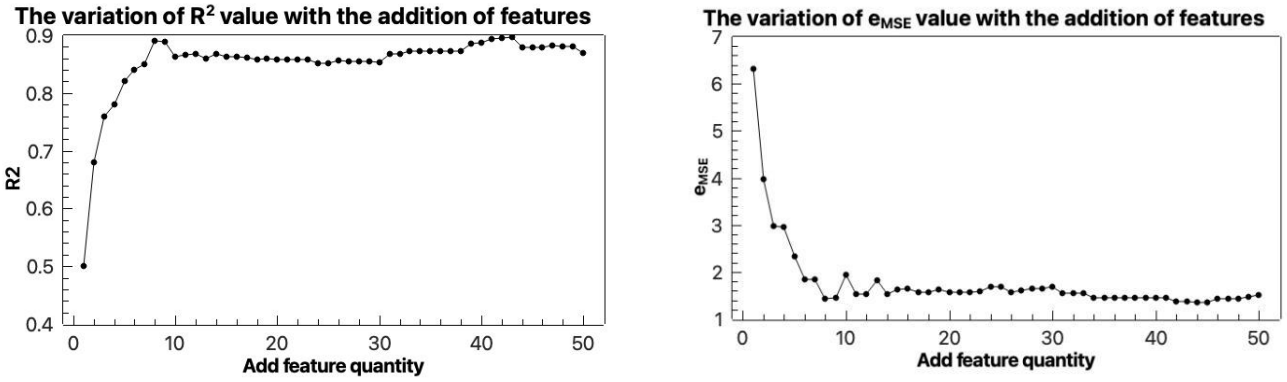
Figure 7. Model fitting curve.

From the results in Table 2, it can be seen that the basic machine learning models have a certain applicability to this dataset. However, since the training model used 50 course grade features from the dataset, the results are not highly refined, and the degree of influence of each course grade on the final grade is not clear, which limits the reference value for decision-makers. Therefore, by proposing a stacked model to integrate various basic machine learning models, each basic course grade and ranking are used to predict the comprehensive grade separately, and the results are arranged in descending order of value. The elbow method is used to assist in feature selection to obtain the optimal feature set, which is then input into the SVR model for the final prediction, achieving the goal of obtaining better prediction effects with fewer input features. Bayesian hyperparameter tuning is used to optimize the parameters of the stacked model, and the specific hyperparameter settings for each base learning model are shown in Table 3.

Table 3. Hyperparameter setting table

Model Name	Hyperparameter Settings
Decision Tree	Maximum depth 28, minimum samples per leaf 8, minimum samples per split 2
SVR	Kernel function is linear, penalty coefficient 128
XGBoost	Learning rate 0.1, maximum depth 3.823, subsample ratio 0.7
LightGBM	Learning rate 0.00783, maximum depth 6, number of iterations 500

To construct a Stacking model with fewer features, the elbow method is used to determine the number of features. The experimental results of the elbow curve are shown in Figure 8. According to the feature selection rule of the elbow method, 8 course features are finally selected to construct the Stacking model.



(a) elbow diagram of value (b) elbow diagram of value

Figure 8. Elbow diagram assisted feature selection.

The comparison of the prediction performance between the Stacking model and various basic data-driven models is shown in Figure 9. From the comprehensive comparison of various evaluation indicators in Figure 9, it can be seen that the Stacking model only needs to use 8 data features to achieve ideal prediction results. The specific features are shown in Table 4, and the number of features used is far less than the 53 features used by each model for independent prediction. The model predicted 1.333, 1.041%, and 0.834, demonstrating good predictive performance. The comparison with various independent models shows that the Stacking model has the characteristics of small error, high accuracy, and fewer features used, and its prediction performance is significantly better than that of a single prediction model.

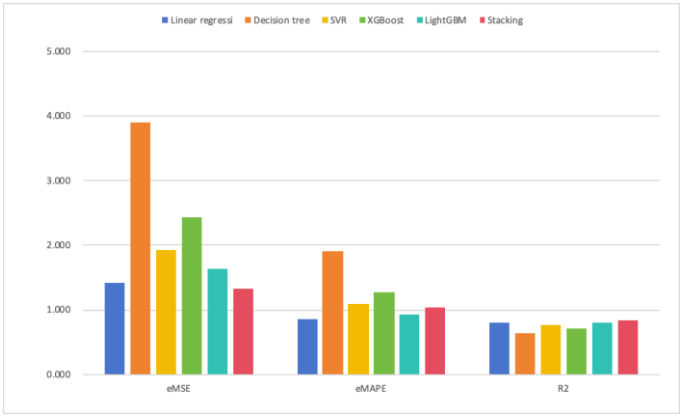


Figure 9. Comprehensive comparison of the prediction effects of the compared models.

Table 4. Features selected by the Stacking model

Course	Course
Advanced Mathematics	Microcomputer Principles and Programming
University Physics	Complex Function and Integral Transform
Probability Theory	Process Control Systems
Linear Algebra	Circuit System Analysis

CONCLUSION

This paper proposes a data-driven, ensemble-based multi-model fusion grade prediction method, which employs Bayesian optimization for hyperparameter tuning and utilizes the elbow method to identify a critical set of features. Ultimately, the desired predictive performance is achieved using only eight heading features. The method boasts advantages such as minimal feature requirements, high prediction accuracy, and strong stability. Building upon this foundation, an analysis of the selected feature set elucidates the value of prioritizing key courses for university teachers, students, and educational research departments. The constructed stacked model provides a valuable reference for university teachers and students: the chosen features include

foundational mathematics courses and specialized courses, highlighting the importance of these courses as the foundation for professional learning. Moreover, the research findings align with practical teaching experience, emphasizing the need for students to take these courses seriously, allocate study time wisely, and maximize efficiency and effectiveness.

For university educational research departments, it is crucial to focus on the teaching construction of these courses, appropriately tilt the allocation of teaching resources, enrich teaching content, and improve teaching facilities. Given the need to cultivate practical talents, it is important to increase practical teaching components such as social practice, course design, and graduation projects, thereby comprehensively enhancing students' technical innovation abilities, professional application skills, and development design capabilities. By employing participatory, discussion-based, heuristic, and inquiry-based teaching methods, the learning enthusiasm of students can be fully mobilized. Since the dataset used in this study includes only grade data from students in a specific major from the central region of China, and the courses corresponding to students from different colleges, majors, and grades also vary, the generalizability of this model still requires further validation in future research.

FUNDING

Project Supported by 2023 The Great Wall Plan of North China Electric Power University.

Project Supported by the "Rainbow Life Career Education Studio" a Leading Instructor Studio in Hebei Province.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

REFERENCES

[1] M. Chitti, P. Chitti and M. Jayabalan, "Need for Interpretable Student Performance Prediction," 2020 13th International Conference on Developments in eSystems Engineering (DeSE), Liverpool, United Kingdom, 2020, 269-272.

[2] Wu D, Li H, Yu X R. Digital transformation of education international perspective, development needs and implementation strategy. Chinese Journal of Distance Education, 2022, (7): 21-27+58+79.

[3] S. Gupta and J. Agarwal, "Machine Learning Approaches for Student Performance Prediction," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, 1-6.

[4] H. Q. Alatawi and S. Hechmi, "A Survey of Data Mining Methods for Early Prediction of Students' Performance," 2022 2nd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 2022, 171-174.

[5] Yan H B, Wang W. Strengthening the practical rationality of digital education: Model construction and application verification of data-driven decision making. Research in Educational Development, 2023, 43(04): 21-29+47.

[6] Guan C, Mou J, Jiang Z. Artificial intelligence innovation in education: A twenty-year data-driven historical analysis. International Journal of Innovation Studies, 2020, 4(4): 134-147.

[7] Mehdi R., Nachouki M. A neuro-fuzzy model for predicting and analyzing student graduation performance in computing programs. Educ Inf Technol, 2023, 28(3): 2455–2484.

[8] He C, Song J, Zhuo T. Curriculum association model and student performance prediction based on spectral clustering of frequent pattern. Application Research of Computers, 2015, 32(10): 2930-2933.

[9] Lin M N, Li J H. Student achievement grade prediction model based on neural network optimized by adaptive differential evolution. Modern Electronics Technique, 2022, 45(03): 130-134.

[10] Nithiyanandam N, Dhanasekaran S, Kumar A S. Artificial intelligence assisted student learning and performance analysis using instructor evaluation model. 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC). Coimbatore: IEEE, 2022: 1555-1561.

[11] Wu C H. Research and application of performance prediction based on improved random forest. Zhejiang: Zhejiang A&F University, 2023.

[12] Shen J, Chen L L, Pan T, et al. Analysis of Learning Trajectory Based on Data Mining under the Background of Emerging. Engineering Education. Journal of Tianjin University (Social Sciences) ,2023,25(06):499-506.

[13] Ren Y M. Study of students' performance prediction and the strategy based on the machine learning. Yunnan: Yunnan Normal University, 2023.

[14] Cao H J, Xie J. LSTM-based learning achievement prediction and its influencing factors. Journal of Beijing University of Posts and Telecommunications (Social Sciences Edition), 2020, 22(06): 90-100.

[15] Steinberg D, Colla P. CART: classification and regression trees. The Top Ten Algorithms in Data Mining, 2009, 9: 179.

[16] Gomes C M A, Lemos G C, Jelihovschi E G. Comparing the predictive power of the CART and CTREE algorithms. Avaliação Psicológica, 2020, 19(1): 87-96.

[17] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016: 785-794.

[18] Asselman A, Khaldi M, Aammou S. Enhancing the prediction of student performance based on the machine learning XGBoost algorithm. Interactive Learning Environments, 2023, 31(6): 3360-3379.

[19] Niyogisubizo J, Liao L, Nziyumva E, et al. Predicting student's dropout in university classes using two-layer ensemble machine learning approach: A novel stacked generalization. Computers and Education: Artificial Intelligence, 2022, 3: 100066.

[20] Z. Lin and S. Zhu, "DFS-Enhanced Lightgbm: An Extended Lightgbm Model Applied to ICU Heart Failure Mortality Prediction," 2023 International Conference on High Performance Big Data and Intelligent Systems (HDIS), Macau, China, 2023, 113-117.

[21] Z. Cai, H. Huang, G. Sun, Z. Li and C. Ouyang, "Advancing Predictive Models: Unveiling LightGBM Machine Learning for Data Analysis," 2023 4th International Conference on Computer, Big Data and Artificial Intelligence (ICCBD+AI), Guiyang, China, 2023, 109-112.

[22] Z. Hongwei, K. Zhipeng, Z. Yuchen, G. Ruichao, L. Shengsheng and Y. Jinhui, "RS-Stack: Defense against Stack Buffer Overflow Attack with Random Shadow Stack," 2022 IEEE 2nd International Conference on Data Science and Computer Application (ICDSCA), Dalian, China, 2022, 1432-1436.

[23] J. Faria, S. M. Azmat Ullah and M. R. Hasan, "Stroke Detection Through Ensemble Learning: A Stacking Approach," 2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS), Dhaka, Bangladesh, 2024, 01-06.

[24] Hong Qu, Ze Ye, Liang Weixuan. Research on Medium and Long Term Generation Side Deviation Predictionof New Power Market Based on Multi-Layer LSTM, Recent Advances in Electrical & Electronic Engineering; 2023, 16(6): e170423215840.

[25] A. Quilliot, J. -L. Figueroa, H. Toussaint and A. Wagler, "Algorithmic Handling of Time Expanded Networks," 2023 18th Conference on Computer Science and Intelligence Systems (FedCSIS), Warsaw, Poland, 2023.