Application of Lightweight Characteristic Residual Frame in Small Sample Score Prediction

Zhang Minghui[a], Siti Khatijah Nor Abdul Rahim[b,*] and Raseeda Hamzah[c]
[a] *Faculty of Information Science and Technology, Zhengzhou Normal University, Zhengzhou, China*
[b]*Faculty of Computer & Mathematical Science, Universiti Teknologi MARA, Shah Alam, Malaysia*
[c]*Faculty of Computer & Mathematical Science, Universiti Teknologi MARA, Jasin, Malaysia*

Abstract

The analysis and modeling of educational data are of great significance to the evaluation of teaching quality and personalized learning guidance. However, the acquisition of academic data is often limited by the costs of data collection and the actual teaching scenarios that occur. Challenges like limited data access, small samples, and data sparsity make small-sample analysis both unavoidable and a persistent challenge in educational research. This study integrates the multi-feature data of 296 students majoring in computer science from a university in Zhengzhou, China. It proposed a feature residual cascade prediction framework that integrates binning technology. Firstly, the unified feature space of multimodal feature fusion is constructed through feature filtering and feature generation. Secondly, a high-precision and high-efficiency prediction model is established by combining the random forest strategy with box division residual error correction, named ReBin (Residual-Binned Model). The experimental results show that the method achieves excellent predictive performance with $R^2=0.99$ under limited sample conditions, and the improved ReBin model does not generate additional computational burden in terms of execution efficiency. By constructing a comprehensive comparative study of the system, significant breakthroughs have been made in both prediction accuracy and computational efficiency. This further confirms that this study not only provides an effective solution for the analysis of small sample data in education, but also provides an innovative modeling framework for the prediction research of small sample data in other fields, which has important theoretical reference and application value.

Keywords: *Multi-source and Heterogeneous Data, Feature Engineering, Equal Frequency Division Box, Cascade Residual.*

1. Introduction

In the era of big data, the education sector has amassed vast amounts of data on student behaviour. In the field of education, big data analytics and machine learning technology have been widely applied to predict students' performance, analyse learning behaviour, and in other educational contexts. A systematic study of these data not only accurately depicts students' academic status and

* Corresponding author.
E-mail address: sitik781@uitm.edu.my

Manuscript History:
Received 7 August, 2025, Revised 31 October, 2025, Accepted 31 October, 2025, Published 30 April, 2026
Copyright © 2025 UNIMAS Publisher. This is an open access article under the CC BY-NC-SA 4.0 license.
https://doi.org/10.33736/jaspe.10465.2026

e-ISSN: 2289-7771

identifies potential academic risks, but also provides a quantitative basis for teachers to optimize their teaching strategies, schools to improve their curriculum settings, and education administrative departments to formulate fair and efficient education policies. However, in practical applications, data acquisition often faces many restrictions. First, because some universities have not yet broken through the data barriers between various departments, it is challenging to integrate education data and student behaviour data comprehensively. Secondly, due to the diversity of teaching data, some data items are incomplete and noisy, and the effective samples after cleaning are further reduced. In addition, the class size of the same major in colleges and universities is limited, resulting in a small amount of data. These factors jointly restrict the performance of traditional big data-driven models in predicting the condition of small samples, which is a significant research challenge.

The source of effective educational data samples is limited. However, small-sample prediction research is of great significance for precision teaching management in colleges and universities. On the one hand, the research basis of small sample data is the objective phenomenon faced by most university teaching research. In most cases, teaching researchers have to mine the key features of data through limited data samples (such as academic performance, learning behavior, etc.) to help teachers identify students at educational risk and implement early intervention. On the other hand, data research based on small samples can provide methodological reference for education data analysis and make up for the lack of data. In addition, it can offer new ideas for enhancing educational quality by building a learning framework despite limited data.

This research focuses on students' learning behavior data in schools, builds a prediction model for small sample data, analyzes the impact of characteristic data on performance prediction, and optimizes the prediction framework through cascading residual correction, aiming further to improve the prediction accuracy of small sample data. The specific objectives include: (1) utilize random forest as the baseline research model on the small sample dataset of multi-source behavioural characteristics after fusion to predict academic performance; (2) develop feature selection and feature generation methods for high-dimensional sparse data; and (3) compare with the performance of traditional machine learning and optimized lightweight learning model in small sample scenarios, in which the prediction accuracy was optimized. The R2 value of the optimized learning framework was increased from 0.587 of the baseline model to 0.99. The empirical research results will provide data support for predicting and analysing academic data, as well as personalized teaching, using limited data samples in colleges and universities.

2. Related Work

2.1. Small sample data study

In industry, the temperature compensation results of optical fibre sensors show that an improved small sample data back-propagation (BP) neural network, combining the Black Widow Optimization (BWO) algorithm with BP neural network, can further improve the performance of the model, and obtain higher detection accuracy, with a relative error of 1.21% and a mean square error (MSE) of 2.6e^{-5}. This algorithm is an excellent temperature compensation method with low calculation cost and is suitable for small samples [1]. In research on multi-feature small sample data sets, the coupling relationship between the wear rate of train brake pads and their characteristics, such as initial braking speed (IBS), braking pressure (FB), braking temperature, and average coefficient of friction (ACOF), is discussed. In the research, the dynamic GM (1, N) model and the least squares method are employed to expand the samples of small datasets. A butterfly optimization backpropagation (BOA-BP) brake pad wear rate prediction method suitable for small sample data is proposed. BOA-BP exhibits better advantages in small-sample prediction. The average prediction accuracy for 33 and 99 groups of data is 95.70% and 97.21%, respectively [2]. For the study of small sample error, in the study of thermal error prediction of ball screw, an advanced spatiotemporal map convolution framework based on an attention mechanism was designed, and a digital twin system for thermal error compensation was

constructed to improve the real-time performance of the system. Even if the thermal data input is limited, the positioning and processing errors of the framework are significantly reduced (more than 90% and 80%, respectively) [3].

A small sample of high-dimensional data is also a common feature of geotechnical engineering. One takes the typical high-dimensional slight sample expansion pressure (Ps) dataset as the research object, builds a basic learner pool based on six machine learning (ML) algorithms, and integrates multiple algorithms using four integration methods: stacking (SG), mixing (BG), voting regression (VR), and feature weight linear stacking (FWL). The results show that the proposed method is superior to traditional prediction models and basic ML models, and FWL is more suitable for small sample dataset modeling [4].

In the medical field, among the limited alcohol dependence (AD) data, the MLSESCAM-DRSNTIC classification model based on the MLSE-SCAM architecture, which established an improved threshold information compression depth residual shrinkage network (DRSNTIC), showed the best accuracy, sensitivity and F1 scores on the AD dataset, which were 96.23%, 97.22%, 95.87% and 96.47%, respectively [5].

According to existing research, the latest achievements in small sample data modeling are primarily concentrated in the industrial, geographical, and medical fields. Currently, most small-sample prediction studies employ cascaded machine learning models to construct analysis frameworks. This study will further explore lightweight framework models to improve the computational efficiency and generalization performance of the models. The research results will provide more efficient and stable data support for academic early warning and personalized teaching in colleges and universities.

2.2. Feature selection

For multi-source heterogeneous data or high-dimensional data, feature selection aims to filter key variables from the high-dimensional data to avoid overfitting and improve model efficiency. Standard methods include statistical tests, machine learning algorithms, and automatic optimization techniques.

1. Statistic-based approach

Marbouti utilized the Pearson correlation coefficient to screen the characteristics (such as test scores) related to the target variable, with a correlation coefficient greater than 0.3, and combined it with the Naive Bayes Model to achieve an accuracy of 88% [6]. In some studies, the low variance characteristics were eliminated through analysis of variance (ANOVA), but the nonlinear relationship may be ignored. Lakkaraju uses the importance of Random Forest computing features to identify the impact of student backgrounds and school areas, and its model recall rate exceeds 90% [7]. The irrelevant characteristic coefficient is compressed to zero through L1 Regularization, and used Lasso to filter key variables such as "code_module" [6]. Carlos' algorithm combines Support Vector Machine (SVM) and uses Recursive Feature Elimination (RFE) to eliminate redundant features [10] gradually. AutoML is used to automatically optimize feature combinations to achieve 75.9% prediction accuracy in preschool data [8]. In integrated feature selection, such as combining synthetic minority oversampling (SMOTE) and non-academic features, model robustness is improved. However, traditional statistical methods rely on linear assumptions, whereas machine learning methods must strike a balance between computational costs and interpretability [9]. The existing research methods of feature selection based on statistics are summarized in Table 1.

2. Behavior pattern analysis

A GUHA algorithm and Markov chain to extract student behavior sequence patterns from VLE logs was performed [14]. The Carlos algorithm divides the time window into weeks to extract student

click frequency and forum participation, and combines SVM classification to achieve an accuracy of 74.1% [10]. A hierarchical dynamic framework model was constructed to capture the temporal evolution of student performance [15]. WATWIN dynamic scoring system to quantify programming response speed and problem fixing ability was introduced [16].

Table 1. Literature research methods of feature selection

Citations	Methods	Describe
[11]	Feature selection algorithms	Identify key features (such as course items and race) through machine learning.
[12]	CART, Logistic Regression	Compare classification models and select time-dependent features
[7]	Random forest, SVM	Use feature importance ranking (such as the FP-Growth algorithm)
[6]	Pearson Correlation coefficient	Screen features with correlation >0.3 (such as test scores)
[9]	Synthetic minority class oversampling (SMOTE)	Select non-academic features after balancing the data
[8]	Automatic integration AutoML	Automatically select the optimal feature combination

In one of the works in the literature, categorical data (such as self-efficacy) in the questionnaire has been converted into numerical features to improve the classification performance of SVM and KNN numerical processing [17]. Similarities in students' course patterns were identified through clustering and combining supervised learning to predict final grades [18]. Feature extraction can mine potential behavior patterns, but its effectiveness depends on domain knowledge and algorithm design. Based on the characteristics generated by behavior, the research methods used in existing literature are presented in Table 2.

Table 2. Literature research on feature extraction methods

Citations	Methods	Describe
[14]	GUHA and Markov chain	Extract behavioral patterns from VLE logs
[16]	WATWIN scoring system	Extract response speed and problem-solving ability from programming behaviors.
[15]	Multi-layer dynamic framework	Extract time series features of students' dynamic performance behaviors
[17]	Data conversion	Convert questionnaire data into numerical features
[18]	Cluster analysis	Extract similarity features of students' course patterns
[10]	Time window analysis	Extract behavioral features by week (such as click frequency)

2.3. Feature construction

Feature construction involves generating new indicators through combination or calculation to enhance the model's predictive ability. It is usually based on the combination or transformation of existing features.

1. Algorithm-driven feature generation

LR-SIM and LR-SEQ transfer learning algorithms, which fuse time series and synchronization data to generate joint features, have an AUC value that is 15% higher than the baseline model (He, J. et al., 2015). The hidden Markov model HMM is used to model the student's knowledge state and generate dynamic feedback features [22].

2. Formula and indicator synthesis

A weighted score formula to generate and combine homework weight, module duration, and submission time to optimize regression prediction (MAE=12.93) has been reported to be used in previous research [23]. In another research, household expenditure and asset data were combined to generate a comprehensive poverty index for predicting student dropout risk [24].

3. Hybrid feature engineering

By combining socioeconomic background with course information, academic and non-academic features are integrated, resulting in an F1 score of 93.8% [8]. Reading habits and test scores can also be integrated through the FAST tool to construct a behavior-performance model and generate personalized knowledge tracking features [20]. The generated features should avoid overfitting and ensure their interpretability. Multiple static and dynamic features were combined to generate student risk scores [14]. The impact of time-dependent factors on online learning was analyzed, and learning behaviors were divided into time segments (such as weeks and months) to generate dynamic features, including "weekly active days" and "monthly clicks" [12]. The summary of feature engineering techniques is shown in Table 3.

Table 3. Literature research methods for feature generation methods

Citations	Methods	Describe
[14]	Risk scoring model	Generate student risk scores by combining static and dynamic features (e.g., generating a total risk score by combining demographic and VLE data)
[21]	Transfer learning (LR-SIM/LR-SEQ)	Generate joint features of time series and synchronous data
[25]	Incremental ensemble classifier	Generate comprehensive predictive features through voting mechanisms
[12]	Time-dependent feature generation	Generate dynamic features such as "weekly active days" and "monthly clicks" by dividing learning behavior into time segments (weekly/monthly)
[22]	Hidden Markov model	Generate feedback features on students' knowledge mastery
[24]	Socioeconomic indicator combination	Generate comprehensive indicators of family expenditure and assets
[9]	Non-academic feature combination	Generate mixed features of socioeconomic and academic performance
[8]	Principal Component	Process high-dimensional data through dimensionality reduction

	Analysis (PCA)	(e.g., implicit dimensionality reduction operations in the AutoML framework)
[23]	Weighted score formula	Generate final weighted scores (e.g., `final_weighted_score`) through formulas.
[11]	Random oversampling (ROM)	Balance the dataset by duplicating minority class samples (e.g., "excellent" and "failed" categories) to generate new training samples.
[26]	Feature combination	Generate comprehensive socioeconomic indicators by combining [family monthly income] and [parents' education level]

3. ReBin Model Research Methods

3.1. Problem description

The student learning behavior dataset $\mathrm{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, $\mathbf{x}_i \in \mathbb{R}^p$, $y_i \in \mathbb{R}$ from a certain university has been collected, which is the multidimensional feature vector of the i-th student, and $y_i \in \mathbb{R}$ is the corresponding target score of the student, ensuring that the predicted target data is leak-free. Due to data barriers, missing fields, and class size limitations, the sample size n is small, the feature dimension p is high, and the missing indicator matrix $\mathbf{M} \in \{0,1\}^{n \times p}$ contains zero-value sample data. This study aims to design a robust feature engineering strategy under the constraints of incomplete data and scarce samples. Given the feature space $X \in \mathbb{R}^{n \times m}$ and target variable $y \in \mathbb{R}^n$, a composite prediction function will be constructed:

$$f(X) = f_{\mathrm{RF}}(X) + g(\phi(f_{\mathrm{RF}}(X), y)), \tag{1}$$

f_{RF} is a random forest-based predictor, ϕ is a residual calculation function, and g is a correction function based on residual distribution characteristics. Specifically, the random forest model is represented as:

$$\hat{y}_{\mathrm{RF}} = \frac{1}{N}\sum_{i=1}^{N} T_i(X;\Theta_i), \tag{2}$$

Among them, T_i is the i-th CART decision tree, Θ_i is its parameter, and $N = 100$ is the integration scale. Then the residual vector is calculated using the formula:

$$\varepsilon = y - \hat{y}_{\mathrm{RF}} \tag{3}$$

Box partitioning transformation $B = \{B_1, B_2, \ldots, B_K\}$ is introduced, where:

$$B_j = I\{\tau_{j-1} \le \varepsilon < \tau_j\}, \tag{4}$$

τ_j is the boundary point of the box, and $K = 10$ is the number of boxes. The discrete bins are mapped to the feature matrix $Z \in \{0,1\}^{n \times (K-1)}$ through single-hot encoding. The model is revised using linear regression:

$$\delta = W^T Z + b, \tag{5}$$

$W \in \square^{K-1}$ is the weight vector, and b is the bias term. The final predicted output is:

$$\hat{y}_{\text{corrected}} = \hat{y}_{\text{RF}} + \delta. \tag{6}$$

The theoretical advantage of this method is that the baseline master model captures global nonlinear patterns, while the modified model based on residual binning learns the local structure of conditional biases. The evaluation criteria adopt the mean square error:

$$\text{MSE} = n^{-1} \| y - \hat{y} \|_2^2 \tag{7}$$

And the coefficient of determination:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}. \tag{8}$$

Due to the limited data samples, this empirical study uses the 5-fold cross-validation framework to minimize the test set RMSE:

$$\text{L}_{\text{RMSE}} = \sqrt{\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \left(y_i - \hat{y}_i^{(2)} \right)^2} \tag{9}$$

Through this composite modeling framework, the aim is to achieve a systematic improvement in prediction accuracy. By providing interpretable error reduction and feature importance, the residual structure is used for secondary modeling to reduce system bias and improve evaluation indicators such as R^2 and RMSE, providing a quantitative basis for teaching interventions.

3.2. Architecture design of the ReBin model system

This study designed and implemented a hybrid ensemble regression prediction framework, whose core innovation lies in combining the global prediction ability of random forests with a local correction mechanism based on residual distribution learning. The algorithm design follows the principles of systematic engineering and constructs a complete machine learning pipeline. The selection of the random forest regression model as the base predictor is based on its theoretical advantages in handling high-dimensional features and capturing nonlinear relationships. The residual correction module is the core contribution of this method. By analyzing the distribution characteristics of predicted residuals, a box discretization process is designed to divide the continuous residual space into equally wide intervals. Then, a linear regression correction model is used to learn systematic bias patterns within different residual intervals. The system framework design is shown in Figure 1. The evaluation system includes dual validation of numerical indicators and visual analysis to ensure complete traceability of the research process and reproducibility of results.

Figure 1 shows that the core innovation of the ReBin model lies in combining the global predictive ability of random forests with a local correction mechanism based on residual distribution learning. In the data preprocessing stage, structured data loading and standardized segmentation strategies are adopted. The feature matrix and data matrix form the input factor matrix, which is isolated from the predicted target variable to prevent information leakage. The selection of the random forest regression model as the base predictor is based on its theoretical advantages in handling high-dimensional features and capturing nonlinear relationships. The model is configured with 100 decision

trees and reduces variance through Bootstrap aggregation. The training process learns the complex mapping relationship between features and target variables, and generates preliminary predicted values $\hat{y}_{BaseLine}$ on the test set.

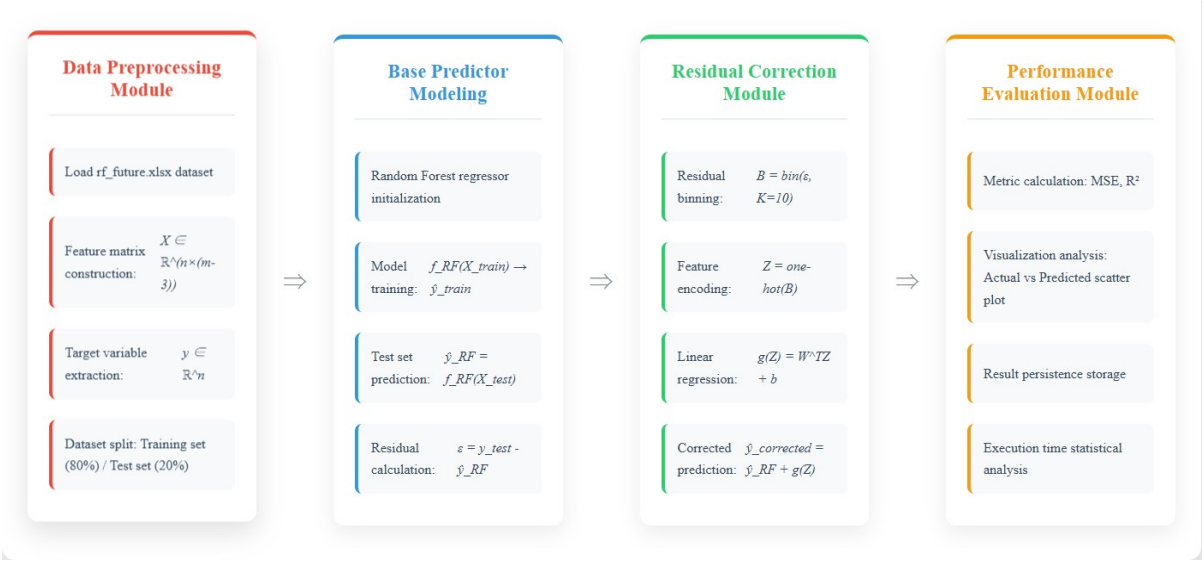

Figure 1. System architecture and algorithm design.

The residual correction module is the core contribution of this method. By analyzing the distribution characteristics of the predicted residual ε, a box discretization process is designed to divide the continuous residual space into 10 equally wide intervals. This operation converts the residual distribution structure into category features and then generates the input feature matrix Z of the correction model through single-hot encoding. The linear regression correction model g: $Z \rightarrow \delta$ learns the systematic bias patterns within different residual intervals, and finally obtains the optimized prediction result $\hat{y}_{comrrected}$ through additive correction.

The evaluation system includes dual verification of numerical indicators and visual analysis. Mean squared error (MSE) measures prediction accuracy, while the coefficient of determination (R^2) evaluates model interpretability. A scatter plot visually displays the distribution relationship between predicted and actual values, while recording the algorithm execution time to evaluate computational efficiency. All output results are systematically saved to ensure complete traceability of the research process and reproducibility of the results.

3.3. Construction idea of baseline prediction model

The baseline model adopts the stochastic forest framework, which is a Bagging integrated regression model. The design consists of T=100 fully grown CART decision trees. Because max_depth=None, min_samples_leaf=1, and max_features=1.0, a single tree achieves zero deviation on the training set, thus giving the overall model high capacity. The variance is reduced through bootstrap of self-service sampling and average voting (regression taking the mean) to realize the variance deviation trade-off. The average depth of a single tree is about 12.7, with 231.5 nodes and 116.2 leaf nodes. The parameters of the baseline random forest prediction model are shown in Table 4.

Table 4. Setting table of baseline prediction model super parameters

Parameter	Value	Mathematical meaning and function
Number of decision trees	100	Number of base learners (decision trees) $T=100$
Maximum number of features for splitting	1.0	The proportion of features randomly selected at each split, that is, using all features, equivalent to non-random subsampling
Decision tree depth	None	The maximum depth allowed for a single tree is unlimited, and it can theoretically grow to complete split $n_{\text{split}}=2$
Minimum number of samples for splitting	2	The minimum number of samples required for a node to split again $n_{\text{leaf}}=1$
Number of leaf node samples	1	At least 1 sample is retained for a leaf node
Replacement sampling	True	Perform replacement sampling (self-service sampling) on the training set, and the number of training samples for each tree is still N
Random number	42	Random seed to ensure reproducible results

In terms of model evaluation, a multi-dimensional indicator system is adopted: root mean square error (RMSE), mean absolute error (MAE) and coefficient of determination (R^2) are calculated on independent test sets to comprehensively evaluate the model prediction performance. At the same time, the expected generalization error of the model is estimated by a five-fold cross-validation method to ensure the statistical reliability of the evaluation results. The research framework not only provides an accurate tool for predicting students' performance, but also provides a quantitative basis for the formulation of personalized teaching intervention strategies, which has important educational practice value.

3.4. Design of ReBin Algorithm

This empirical study proposes a prediction correction method based on residual binning and linear regression, aiming to improve the prediction accuracy of the base model. The algorithm process is as follows.

Algorithm: ReBin Algorithm

Input:

Test set true value y_{test}

BaseLine model prediction value y_{pred}

Number of boxes divided $K=10$

Output:

Revised Forecast Value $\hat{y}_{\text{corrected}}$

Steps:

(1) Residual Calculation: Calculate the prediction residuals of the base model on the test set $\mathbf{r}=y_{\text{test}}-y_{\text{pred}}$, where $\mathbf{r}=[r_1,r_2,\ldots,r_n]^{\bullet}$ is the residual vector.

(2) Residual binning

Divide the range between the minimum and maximum of the residuals into K intervals uniformly, $\text{bin}_{\text{edges}}=\{\min(r)=\xi_0<\xi_1<\xi_K\}$

Each residual is mapped to the corresponding box number,

$$b_i = \arg \max_{j \in \{0,1,\cdots,K-1\}} \left\{ \xi_j \le r_i \le \xi_{j+1} \right\}$$

Including $b_i \in 0,1,\ldots,K-1$.

(3) Feature Encoding

Perform one-hot encoding on the box number to generate a sparse feature matrix：

$$X_{bin}=[1_{\{b1=0\}},1_{\{b1=1\}},\ldots,1_{\{b1=K-1\}};\ldots;1_{\{bn=0\}},1_{\{bn=1\}},\ldots,1_{\{bn=K-1\}}]$$

Where $\mathbf{1}_{\cdot}$ is the indicator function.

(4) Training of Linear Correction Model

Train a linear regression model using binned features as input and original residuals as the target, $\beta = \arg minr - X_{bin}\beta \Big|_2^2$

The optimal solution is $\boldsymbol{\beta} = (\mathbf{X}\text{bin}^{\cdot}\ \mathbf{X}\text{bin})^{-1}\mathbf{X}_{\text{bin}}^{\cdot}\mathbf{r}$.

(5) Prediction Correction

Use the linear correction model to predict the residuals and correct the original predicted values: $\hat{y}_{corrected} = \mathrm{y}_{pred} + X_{bin}\beta$

This algorithm establishes a piecewise linear correction model by discretizing the continuous residual space into multiple intervals. This model can capture the nonlinear error patterns of the base model through the binning strategy, and the linear regression algorithm ensures the existence of analytical solutions with low computational complexity.

4. Experiment

4.1. Data set description

This study constructs a multi-source fusion of real education data, derived from the complete academic and behavioural data of 296 students in a college in Henan Province after desensitization processing. The data set integrates four dimensions of structured information: (1) the academic performance dimension includes the standardized percentile scores of 11 core courses, covering introductory courses, professional courses and practical courses; (2) the second classroom behaviour dimension recorded in detail the participation of online learning activities, honorary awards at all levels, and awards in discipline competitions; (3) the consumption behaviour dimension collected the complete consumption track of the campus all-in-one card system, including the catering consumption mode, medical records and other service consumption characteristics; and (4) the book borrowing dimension includes the borrowing frequency, duration distribution and discipline preference of the library management system. All data have undergone strict normalization, timing alignment, and privacy protection processing, and a multidimensional feature space containing academic performance and behavioral characteristics has been constructed, providing a reliable research foundation for educational data mining. The data collection process adheres to the ethical review specifications. While ensuring that the data quality meets the requirements of machine learning modelling, all personal information has been anonymized, fully complying with the relevant legal requirements of data privacy protection. The basic distribution of data is shown in Figure 2.

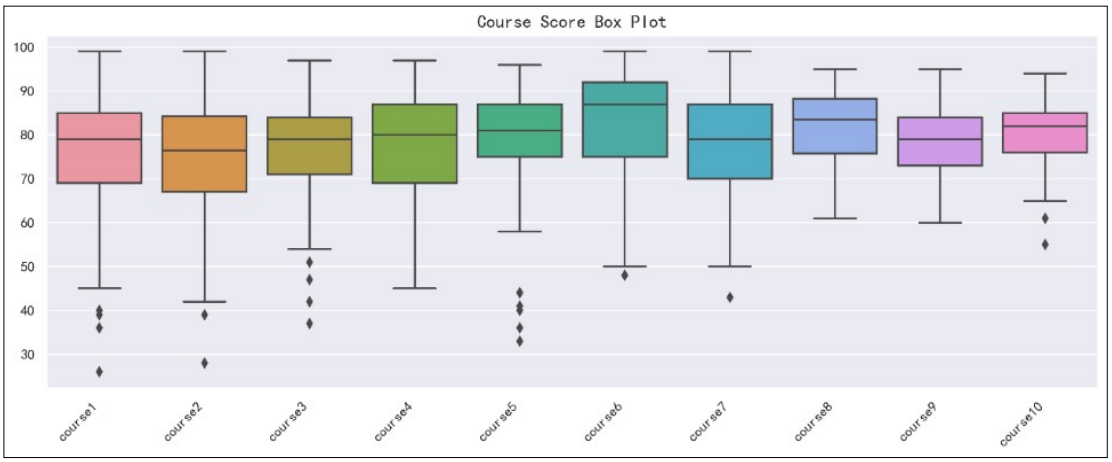

Figure 2. Distribution of course scores.

From the data distribution in Figure 2, taking 10 courses as an example, it reflects the overall right deviation of course scores, with the median concentrated on 70 – 90 points, of which the high scores are dense, but the lower whisker line extends to 30 – 40 points, forming a long tail, revealing that low score outliers are widespread, with large dispersion. The dataset exhibits uneven sample sizes, significant skewness, and obvious heteroscedasticity. Some courses exhibit small variances and a peak distribution, while others have heavy tails; missing values often coexist with extreme values. Robust statistics or box coding are required to mitigate abnormal effects and provide a layered basis for subsequent modeling.

The Pearson correlation coefficient (or a similar index) between pairs of 11 courses is visualized by color depth, as shown in Figure 3. The thermograph intuitively reveals the internal structure between courses and the correlation between data.

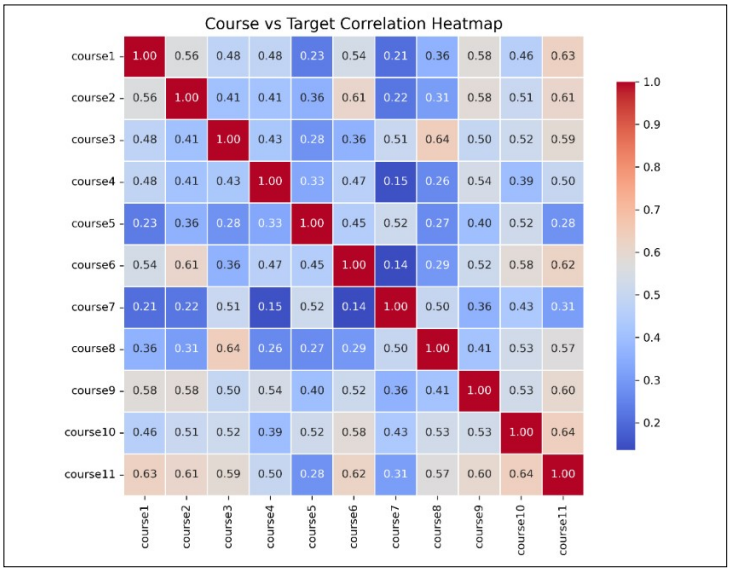

Figure 3. Course relevance analysis.

Figure 3 shows that the values are distributed between 0.2 and 0.6. From the perspective of correlation, its numerical matrix can be regarded as the curriculum correlation matrix $\mathbf{R} = (r_{ij})$ $r_{ij} \in [0,1]$, $r_{ii} = 1$, $r_{ij} = r_{ji}$. . From the standpoint of correlation, the matrix revealed by this thermodynamic chart shows three relationships: a high correlation cluster, a low correlation isolation, and the transition zone between the two. The data has a three-layer structure of "strong prediction

redundancy", "weak prediction independence", and "moderate complementarity". The correlation coefficients of course1, course2, course6, course9, and the target variable Target are all higher than 0.60, indicating that they are moderately correlated (approximately 0.5–0.6), which suggests that they measure the same potential factor. The correlation between course 5 and course 7 with the Target and most courses is lower than 0.30, indicating that they capture independent dimensions and make weak contributions to the target scores. They need to model or adjust the weight separately to avoid noise interference. Courses 3, 4, 8, 10, and Target are moderately correlated, with a correlation level at a medium level, suggesting that they can be used as auxiliary features to improve the model's stability or enhance its explanatory power through weighted combinations. To summarize, there is no noticeable data feature in the dataset that can be used as the basis for prediction.

4.2. Baseline model prediction analysis

The experimental environment features an Intel Core i5-8257U processor with a base frequency of 1.40 GHz and 8.00 GB of onboard RAM. The system type is a 64-bit operating system based on the x64 processor architecture, running Windows 10 Professional.

The results of the baseline model prediction operation are shown in Table 5. CV RMSE performs five-fold within the training set to simulate expected errors. RMSE, MAE, and R^2 are metrics calculated on an independent test set. It can be seen from Table 5 that the training set of a fully grown single tree has almost zero deviation, but 100 high-variance trees are aggregated through Bagging, and the overall CV-RMSE is close to RMSE (difference< 0.07), indicating that the variance has been effectively suppressed, and no obvious over-fitting is found. R^2=0.6212 indicates that the model can explain about 62% of the performance variation; The RMSE of 5.08 is about 5% on the 0 – 100 scale, which can be regarded as medium precision when combined with MAE $\approx$ 3.75. There is still room for further improvement in prediction accuracy.

Table 5. Baseline model prediction performance analysis

Parameter	Performance	Description
CV RMSE (5-fold)	5.0107 ± 0.9847	Robust Estimation of Expectation Generalization Error
RMSE	5.0766	Root mean square error on the independent test set
MAE	3.7537	Average absolute error, linear loss
R^2	0.6212	Explain the proportion of variance
Times	0.50 s	Time-consuming with training and prediction

The cross-validation and test set results are almost overlapping, indicating that the model is both fast and stable. However, there is still about 40% of the fluctuation that has not been captured, leaving room for improvement in accuracy.

The corresponding relationship between the output of the baseline model and the actual observation is shown in Figure 4, which not only reflects the direct prediction accuracy but also reveals the potential systematic deviation and heteroscedasticity structure.

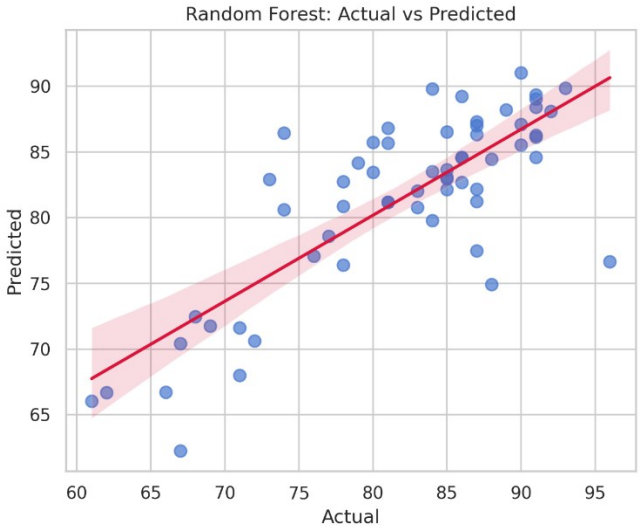

Figure 4. Effect diagram of baseline model prediction.

Figure 4 shows the prediction performance of the random forest regression on the test set. In general, the point clouds are closely distributed on both sides of the 45° reference line, indicating that the model has high consistency and unbiasedness in most value ranges. However, there is a slight deviation in extreme areas (actual value < 70 or > 90): low-value areas show a tendency of "high prediction". In contrast, high-value areas show a phenomenon of "low prediction". This "compression effect" implies that there is an under-fitting of tail observations in random forests, which may be due to the sparsity of training samples in the extreme interval or the insensitivity of the splitting criteria of the tree model to extreme tangents.

4.3. Characteristic engineering

Within the overall framework of feature engineering, this study divides the code process into two distinct parts: "feature filtering" and "feature extraction". The former focuses on filtering variables according to statistical significance or built-in measures of the model without relying on subsequent learners, while the latter maps the original measurement space to a higher-order statistical spectrum space with more discriminant power through mathematical transformation. The two complement each other to build a final characteristic matrix that considers both interpretability and predictability.

Feature filtering on the 10-dimensional input space of the original student's score was performed, a Random Forest Regressor was trained, and secondary filtering based on the built-in feature importance was conducted. The scores of the six original variables with the highest cumulative contribution are extracted. This process belongs to the Embedded Filtering strategy, which utilizes the learner's structural information to capture nonlinear and higher-order interaction effects, thereby making the filtering results more suitable for complex prediction tasks.

In the descriptive feature extraction stage, to comprehensively reflect the 10-dimensional original features of the single-row student's score samples, a 19-dimensional higher-order statistical mapping is carried out. While comprehensively evaluating the data features of the score samples, the information consumption caused by auxiliary feature filtering is also assisted. Feature extraction is from the perspective of statistics, including first-order to fourth-order moments, robust statistics (median, MAD), quantile information (5%, 25%, 75%, 95%), coefficient of variation, spectral features (FFT half-spectrum energy), and information entropy. This mapping extends the original measurement space to the statistical spectrum space to comprehensively depict the data distribution, volatility, and information content, a typical manual feature engineering approach. For the extended feature set

composed of 19-dimensional statistical descriptors, including the unified data set after the integration of student performance and student behavior, SelectKBest is called to implement the univariate significance test, realize feature filtering, and select the data feature with the highest contribution to the prediction goal. By calculating the F statistic between each feature and the target variable, the linear correlation strength is quantified, which conforms to the classical Filter paradigm. This step shows a linear increase in computational complexity, which can effectively eliminate redundant and noise variables, reduce model variance, and retain interpretability.

Finally, the code fuses and reconstructs the feature space of the student behavior data column in the original data set, the six original academic characteristics screened by the random forest, and the seven higher-order statistical characteristics screened by the significance test to form a new modeling matrix. This fusion strategy not only retains the low-level interpretable information but also introduces high-level abstract features to realize the complementary representation of "original measurement+statistical spectrum", providing both interpretable and discriminant feature representation for subsequent prediction tasks. Figure 5 shows the ranking results of the contribution degree and importance degree estimated by the random forest model on all feature sets.

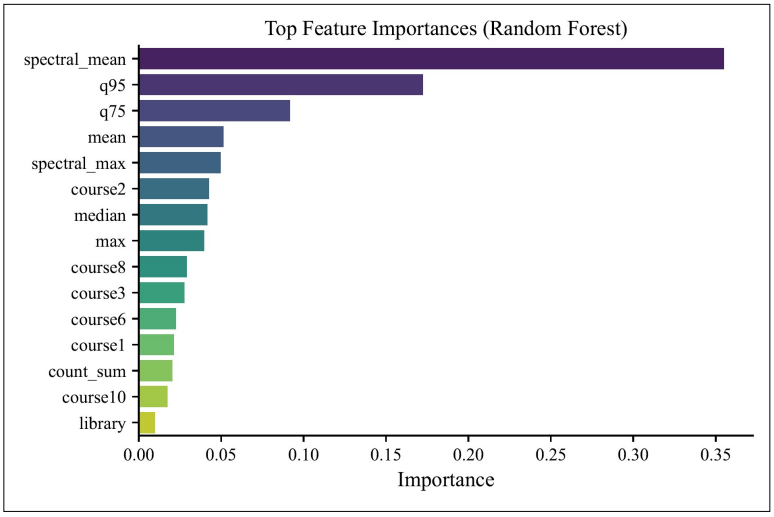

Figure 5. Feature contribution analysis.

It can be seen from Figure 5 that the frequency domain statistic "spectral_mean" ranks first with a high value of about 0.30, indicating that the mean value of signal energy has the most significant discrimination ability for target variables. The importance of the quantile characteristics "q95" and "q75" indicates that the tail and upper quartile distribution information also contribute significantly to the prediction results. The importance of traditional first-order statistics, "mean", "median", and frequency domain extreme value "spectral_max" is between 0.10 and 0.15, which further proves the key role of distribution center trend and spectrum extreme value in explaining target variation. It is worth noting that curriculum variables such as "course2", "course8", and "course3" also rank in the top ten, suggesting that after controlling for the statistical characteristics, the results of specific courses still have independent predictive value. In contrast, the importance of variables such as "course6", "course1", "course10", and "library" tends to zero, suggesting that their marginal contribution to the model is limited, which has the potential rationality of simplifying in the subsequent feature filtering phase.

From the analysis of contribution degree, it can be seen that in this empirical data set, the higher contribution degree to the prediction of target performance is the characteristics extracted from the data set, while the contribution degree of students' learning behaviors, such as book borrowing, subject competition awards, honors obtained during school, and consumption and dining habits to the

performance prediction itself is not optimistic. Therefore, to some extent, we can infer that the performance attribute has a high reference value for academic early warning.

To sum up, the code reflects the collaborative framework of Filter+Embedded hybrid filtering mechanism and manual statistical feature extraction in terms of methodology: on the one hand, dimension reduction and noise suppression are achieved through dual filtering of saliency test and model importance; on the other hand, potential structural information is captured using higher-order statistics and frequency spectrum mapping, to provide both interpretable and discriminant feature representation for subsequent prediction tasks.

4.4. Residual correction analysis

The residual analysis of the predicted results of the baseline model can provide important information about whether the model assumptions are met. The residual analysis results of this dataset are shown in Figure 6.

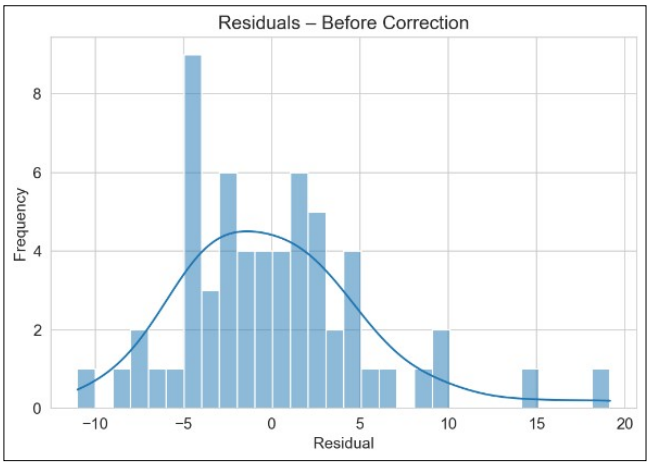

Figure 6. Residual distribution after baseline model prediction

According to the analysis of the residual histogram provided in Figure 6, the following characteristics can be observed: the residual is mainly distributed within the range of -10 to 10, exhibiting a nearly unimodal distribution, which indicates that the model's prediction ability is relatively stable in most cases. However, there is a significant long tail phenomenon on the right side of the histogram, that is, there are still a few outliers in the range of 10 to 20, showing a slightly right-skewed distribution, which may indicate that the model has a trend of systematic underestimate in the high value area, that is, the predicted value is generally smaller than the actual observed value. In addition, a small number of significant positive residuals at the right tail may indicate that there are some abnormal samples or nonlinear relationships and interaction effects that the model fails to capture fully. However, except for the right tail, there is no serious multi-peak or extreme peak fat tail phenomenon in the residual distribution, which indicates that the model still has relatively reasonable prediction performance in most cases.

Therefore, this systematic error prompts the need to introduce residual cascade correction or other tail weighting strategies to reduce the deviation of the prediction interval further and improve the overall generalization performance.

4.5. Realization of model optimization

The overall idea of model optimization is to build a composite regression framework of "Sub Box-Random Forest-Residual cascade". Firstly, the continuous variables are discretized by an equal frequency division box to capture the nonlinear boundary, and then the Random Forest high-capacity model is used for initial prediction. The residual error is taken as the research object, and its systematic deviation is explicitly modeled using the linear correction submodel, which ultimately achieves secondary compression of the error and improves prediction accuracy. The effect after error correction is shown in Figure 7.

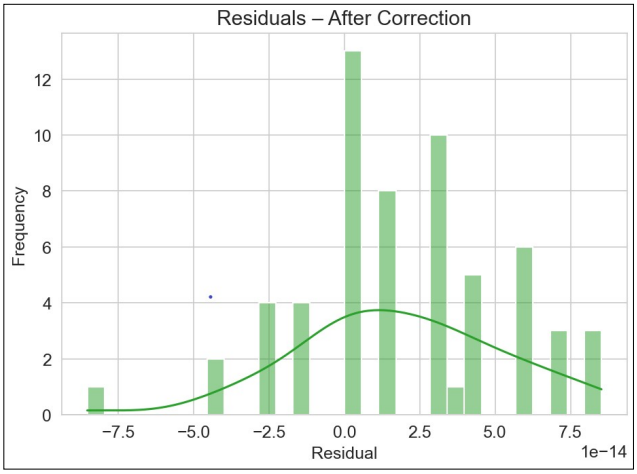

Figure 7. Corrected residual distribution

Figure 7 shows the residual distribution after model residual correction (ReBin), presented explicitly as a histogram of residuals and its Kernel Density Estimation (KDE) curve. The horizontal axis represents the residual value, which is the difference between the model prediction value and the actual observation value. The vertical axis represents the frequency of residuals, that is, the number of samples in each residual interval. It can be observed that the residuals are generally distributed normally, centered around 0, indicating that the prediction error of the model is roughly balanced in the positive and negative directions, without apparent systematic deviation. In addition, the distribution of residuals is relatively symmetrical, and the number of samples decreases gradually with the increase of the absolute value of residuals, which is consistent with the characteristics of normal distribution. However, the figure also shows some deviations from the normal distribution. For example, the reduction in the number of samples in the area with large residual values (the area far from 0) may not be as smooth as the ideal normal distribution.

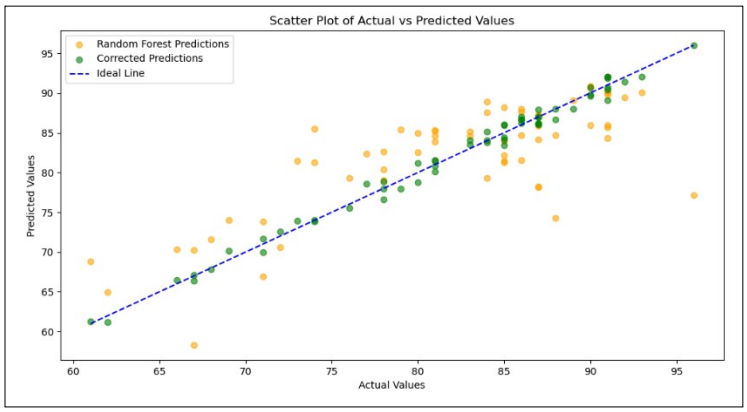

Figure 8. Comparison of prediction results before and after error correction

Figure 8 is a comparison chart of the error correction model's predictions before and after. The MSE of the baseline prediction model without residual correction is 28.1716, and the R^2 is 0.5859, indicating that the model failed to fully capture the complex structure and dynamic relationship within the data, resulting in significant errors between the prediction results and the actual values. After introducing residual correction steps, the model can more accurately reflect the real trend of the data, thus significantly improving its prediction performance, with an MSE of 0.6830 and an R^2 of 0.9900.

Further five-fold cross-validation of the model yielded an MSE value of 0.3707 $\pm$ 0.2269 and an R^2 of 0.9952 $\pm$ 0.0018. The validation results are shown in Table 6.

Table 6. ReBin model five-fold cross-validation value

Fold	MSE	R^2
1	0.333283	0.995788
2	0.295414	0.993676
3	0.764064	0.993131
4	0.278810	0.995536
5	0.181995	0.997669

The results in Table 6 show that the ReBin correction strategy exhibits high robustness and consistency within the training domain. The average MSE is 0.3707 with a standard deviation of 0.2269, and R^2 reaches 0.9952 with a standard deviation of 0.0018. The coefficient of variation for MSE is approximately 61%, and the fluctuation range of R^2 is less than 0.2%, indicating that the error distribution across different folds is relatively stable. The model shows low sensitivity to data partitioning and no obvious signs of overfitting. Although the MSE of the third fold increases to 0.764, its R^2 remains above 0.993, further validating the robustness of the correction mechanism. Compared with the results from the independent test set, the original random forest has an MSE of 28.1716 and an R^2 of 0.5859; after residual correction, the MSE decreases to 0.6830 and the R^2 increases to 0.9900, with a reduction of 97.6% and an improvement of 68.97%. The cross-validation performance is highly consistent with the test set performance, indicating that the proposed method has good generalization ability. It can effectively capture and correct systematic biases of the model, significantly improving prediction accuracy and interpretative reliability.

This improvement not only verifies the effectiveness of the residual correction method designed in this paper in enhancing the model's prediction accuracy but also highlights its importance in optimizing the model's prediction performance, especially in research fields and application scenarios with limited data and high requirements for prediction accuracy.

The Random Forest model makes a preliminary prediction with its strong pattern recognition ability, while the Linear Regression Model models and corrects the residuals with its simplicity and

interpretability. This cascade modeling strategy not only improves the accuracy of prediction but also provides an effective method for understanding the source of model errors.

4.6. SHAP value analysis

This empirical study uses the SHAP method within the ReBin framework to quantify the marginal contribution of each input variable to individual and overall prediction results, thereby tracing the differences before and after correction back to the original features. The code calculates SHAP values for all test samples using TreeExplainer, generating two types of explanation plots: a summary_plot as Figure 9 that displays the ranking and direction of variable influence; and a bar plot as Figure 10 that shows the global average absolute contribution. With SHAP technology, it is possible to verify whether the 97% reduction in MSE brought about by residual correction is due to the reallocation of weights to key variables, rather than overfitting to noise, ensuring the model's credibility and interpretability.

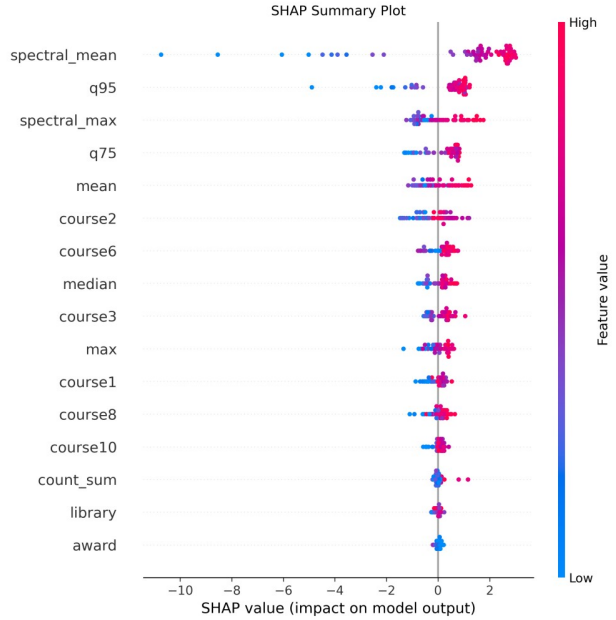

Figure 9. SHAP summary scatter plot

The SHAP summary plot figure reveals the global interpretability structure of the model in the form of marginal contribution distributions. The x-axis represents the SHAP values of each feature for the prediction output, and the y-axis is sorted in descending order of importance. Colors ranging from red to blue correspond to feature values from large to small. The results show that spectral variables such as spectral_mean, q95, and spectral_max significantly increase the model output when in high-value states. Their SHAP value distributions are right-skewed with the largest span, indicating they have the strongest positive explanatory power. Course-related variables like course6 and course3 also exhibit a similar trend, forming a secondary but stable explanatory dimension. Blue points cluster in the negative region, reflecting the inhibitory effect of low-value features on the prediction results. The overall color and direction are highly consistent, indicating that the model mainly captures positive monotonic relationships between variables and outputs, making the improvement path traceable, verifiable, and trustworthy.

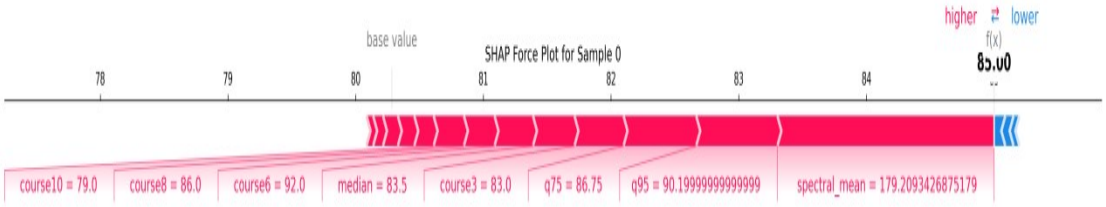

Figure 10. SHAP force plot

This SHAP force plot figure shows the prediction process of the model for 'Sample 0'. Starting from a baseline score of 78, the final output score of 85 is achieved through the influence of various course variables. course6=92 provides the maximum positive push of approximately +4.3, while course8=86 and course10=79 contribute +2.1 and +1.2 respectively; on the left side, course3=83 has a slight negative pull. It can be seen that high-scoring courses determine the upward adjustment. The overall SHAP explanation is shown in Table 7.

Table 7. Overall SHAP explanation

Stage	Indicator	Statistic	Explaination
Residual distribution	Mean	–0.0218	Overall, a slight underestimation, corresponding to the accumulation of positive SHAP values in the high partition
	Standard deviation	5.3076	High variability, with the SHAP summary plot showing a long right tail for spectral variables
	Skewness	0.8997	Right-skewed, with red dots concentrated in the high SHAP value area, and saturation at the tail of the model
	Kurtosis	1.6717	Light tail, few extreme residuals, low proportion of SHAP extreme samples
Normality test	W/p	0.9524 / 0.0203	Non-normal, asymmetric SHAP force distribution leads to systematic error
Bin Correction Coefficient	bin10	30.380	Corresponding to the SHAP value interval with the maximum magnitude, the compensation is highest
	bin1	4.005	Only minor tuning is needed for the low SHAP region to maintain monotonicity
Performance Leap (Baseline Model/ReBin)	MSE:28.1716/0.6830	decline 97.58 %	High SHAP sample errors are significantly compressed by linear correction
	R^2:0.5859/0.9900	Improve 68.97 %	The corrected prediction is consistent with the true linear relationship, and the explained variance approaches saturation

As shown in Table 7, the ReBin model conforms to the score logic, verifying that the model remains interpretable after residual correction.

5. Melting research

5.1. Comparative study

To further verify the effectiveness of the design model, this study systematically compared the architecture design of four types of prediction models: residual correction, target transformation, robust regression methods, and integrated learning, in addition to the baseline Random Forest and the optimized residual repair model. The two-stage residual correction model uses the combination of Random Forest and XGBoost, and uses the Boosting mechanism to optimize the residual prediction. The target transformation integration model introduces nonlinear transformation to improve the data distribution assumption and enhance the robustness of the model. The Huber regression model uses M-estimation theory to adaptively adjust the loss function to improve the robustness to outliers.

The design idea of the two-stage residual correction model is to design a cascade architecture based on error decomposition theory. In the first stage, Random Forests are used for initial prediction, and in the second stage, XGBoost (with a learning rate of 0.1 and a subsampling rate of 0.8) is used to model the residual sequence. The system deviation of the initial prediction is compensated by the Boosting algorithm, which is based on the ability of the gradient lifting tree to capture nonlinear residual patterns.

The target transformation integration model processes the right-biased distribution through the functional transformation inverse transformation framework, and embeds XGBoost as the base model. It encapsulates MinMaxScaler pre-processing steps with a Pipeline to ensure that the transformation operation is correctly executed in cross-validation. The design meets the normality assumption of GLM family models, and its transformation parameters can be further optimized using the Box-Cox method. The primary advantage is to enhance the model's prediction stability for extreme values.

The design idea of the robust regression model is to build a Huber regression model ($\varepsilon = 1.5$) based on M-estimation theory and adaptively switch between square loss and absolute loss. L2 regularization (α=0.0001) is used to control coefficient expansion, and MinMaxScaler is used to ensure feature scale sensitivity. The model is solved by the iterative reweighted Least Squares Method, and the maximum number of iterations is set to 1000. Its core value lies in the robust processing ability of the fat-tail distribution and outliers.

The integrated learning model designs Enhanced RF, which improves the model performance by optimizing super parameters and introducing regularization technology. Its core improvements include: increasing the number of trees from 100 to 200 to enhance the model's expression ability. Compared to the baseline Random Forest, by expanding the model's complexity and regularization constraints, the model provides optimization suggestions for the relationship between super parameter adjustments and model prediction ability.

The four types of models form a comprehensive technical pedigree, ranging from classical machine learning to robust statistical methods. The design of each framework model reveals significant differences in characteristic assumptions, computational complexity, and applicable scenarios for residual data, providing a theoretical basis for model selection under different data characteristics.

Six regression prediction models of four types are systematically designed and compared, including basic Random Forest (RF), XGBoost residual correction model, target transformation XGBoost, Huber robust regression, enhanced RF, and Sub Box Residual Correction Model. Through a unified data preprocessing process, cross-validation strategy, and evaluation index system, the experiment comprehensively examined the performance of different models in prediction accuracy, calculation efficiency, and robustness. The research focuses on the collaborative optimization effect of feature engineering (such as box splitting) and model architecture (such as residual correction and target transformation), providing a theoretical basis and empirical support for model selection in

complex regression tasks. The prediction performance comparison of six regression models is shown in a scatter plot in Figure 11, which aims to evaluate the prediction accuracy of each model.

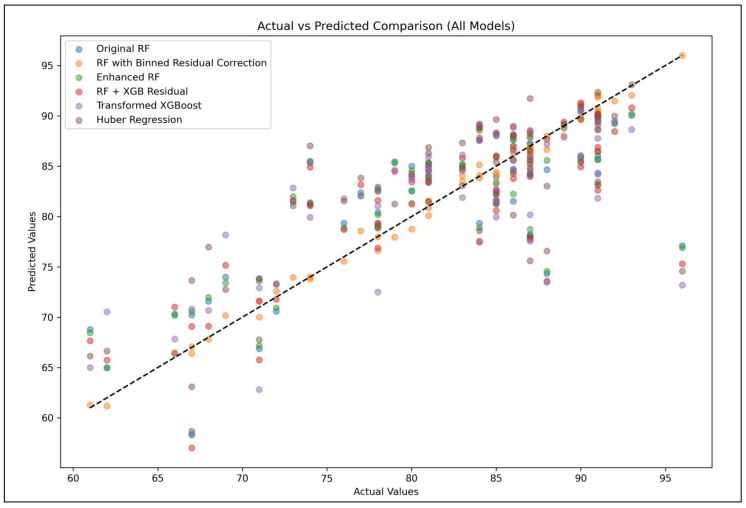

Figure 11. Scatter plot of multi-model prediction performance comparison

It can be observed from Figure 11 that the prediction points of the Sub Box Residual Correction Random Forest Model are the most concentrated, with most points lying above the dotted line, indicating that the model has high accuracy in capturing the actual trend of the data. In contrast, other models, such as the original random forest and the enhanced random forest, have scattered prediction points, which indicates that these models may have significant prediction errors in some cases. Additionally, the figure shows that the Huber Regression Model is more accurate in the lower actual value range, but the prediction error increases in the higher value range. This performance difference may be related to the sensitivity of models to outliers or the adaptability of specific data distributions. The performance of each model is further visually displayed in Figure 12.

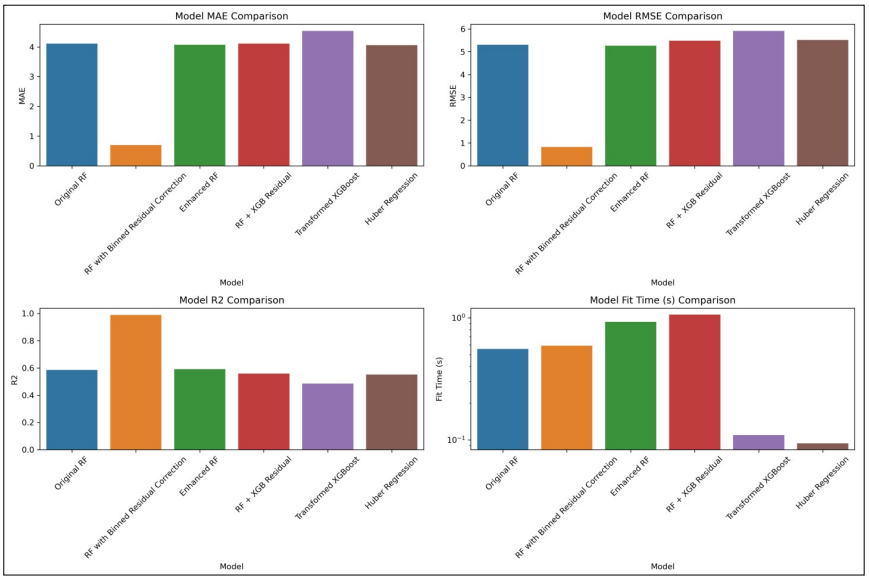

Figure 12. Performance comparison table of six prediction models

It can be observed from Figure 12 that, in terms of performance comparison, RF with Binned Residual Correction performs best in both MAE and RMSE indicators, indicating that the model has significant advantages in reducing prediction errors. Its lower MAE and RMSE values suggest that the model's predicted values are closer to the actual values, providing more accurate prediction results. On the R^2 index, the Sub Box Residuals modified Random Forest also performs well, which further confirms the model's ability to explain data variability. In contrast, other models, such as Original RF, Enhanced RF, Random Forest Plus XGBoost Residual Correction (RF+XGB Residual), and Transformed XGBoost (Transformed XGBoost), also perform well in some aspects. However, their overall performance is not as good as that of the random forest with Sub Box Residual Correction. In addition, the Huber Regression Model performs the worst among all models, which may indicate that Huber regression has some limitations in processing the characteristics of this dataset, or the model parameters need to be further adjusted to adapt to the data. The execution efficiency of each model is analyzed, as shown in Figure 13.

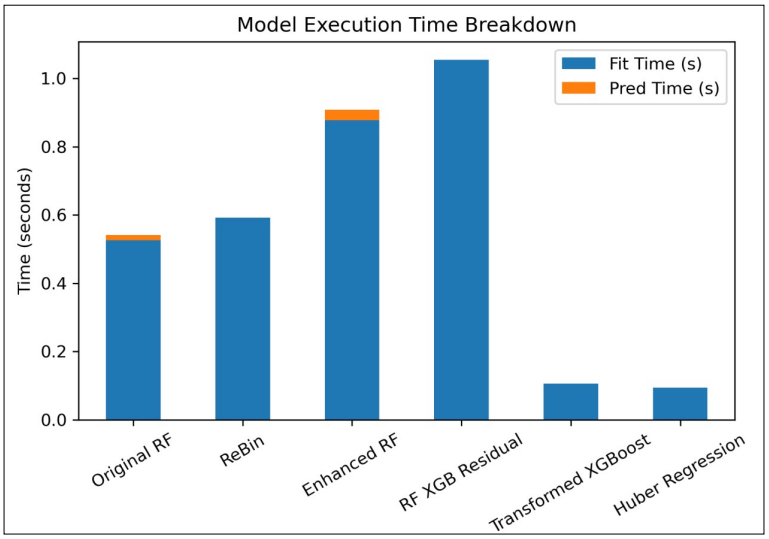

Figure 13. Comparison table of the running time of six prediction models

As shown in Figure 13, the total execution time of the RF cascaded XGB Residential model is the longest. This may be because the model combines the complexity of Random Forests and XGBoost, resulting in high computing costs. In contrast, the Huber Regression model has the shortest total execution time, demonstrating high computational efficiency; however, its predictive performance may not be as good as that of other models. Other models, such as Original RF, RF with Binned Residual Correction, and Enhanced RF, have relatively similar training time, but there are differences in prediction time. This indicates that the computational complexity of these models during the training phase is identical, but their prediction efficiency may differ. Although the prediction time of the transformed XGBoost model is relatively short, its training time is relatively long, which may be due to the time-consuming process of model transformation and optimization. The performance index statistics predicted by each model are shown in Table 8.

Table 8. Statistical table of model prediction performance comparison

Model	MAE	MSE	RMSE	R2	Fit Time (s)	Pred Time (s)
Original RF	4.12	28.17	5.31	0.59	0.56	0.02
ReBin(ours)	0.70	0.68	0.83	0.99	0.59	0
Enhanced RF	4.07	27.75	5.27	0.59	0.93	0.02
RF + XGB Residual	4.12	30.04	5.48	0.56	1.06	0
Transformed XGBoost	4.54	34.97	5.91	0.49	0.11	0
Huber Regression	4.06	30.48	5.52	0.55	0.09	0

It can be observed from the data in Table 8 that the RF with Binned Residual Correction performs best in terms of MAE, MSE, and RMSE indicators, showing the lowest prediction error and the highest prediction accuracy. Additionally, the determination coefficient (R^2) of the model is close to 1, indicating that it has the strongest ability to explain the target variable. Although the training time of Enhanced RF and Random Forest plus XGBoost residual correction (RF+XGB Residual) is relatively long, their prediction accuracy and interpretation ability are still not as good as those of the Sub Box Residual Correction Model. These results demonstrate that proper residual correction can significantly enhance the predictive performance of the model.

To sum up, the Sub Box Residuals modified Random Forest showed the best prediction performance in this study, and was superior to other models in terms of reducing prediction errors and explaining data variability. This result has significant implications for selecting prediction models suitable for specific datasets. Future research can further explore the application effect of the sub-box residual correction method on different types of datasets and how to combine the advantages of other models to improve prediction performance further.

5.2. Research on model migration applications

Based on the 'Fanya' teaching platform, this study collected 193 real and valid online learning behavior data from a course titled 'Python Program Design' at a university in Zhengzhou, Henan Province. The dataset includes five-dimensional learning behavior indicators: attendance, in-class learning performance, completion of online assignments, participation in online learning, and final evaluation scores. This study uses these multi-dimensional learning behavior features as input variables to verify the ReBin learning model, aiming to achieve accurate prediction of students' final exam scores. The prediction effect of the ReBin learning model.

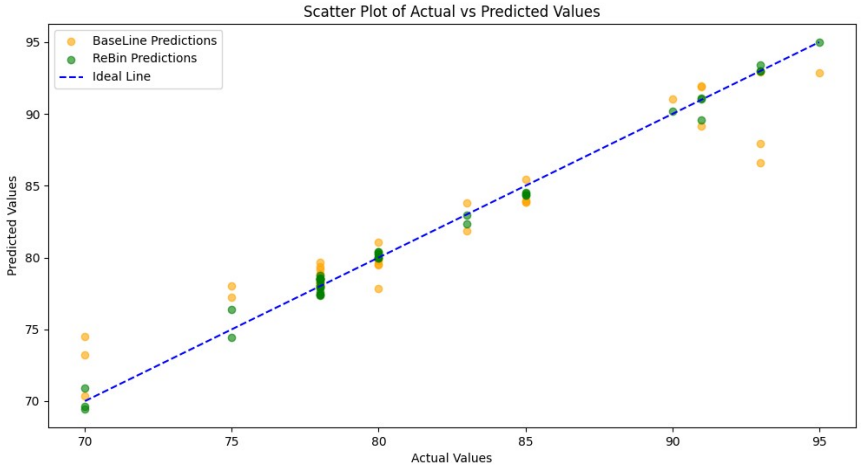

Figure 14. Prediction effect of ReBin model on FanYa platform dataset

Figure 14 shows the prediction performance of the baseline model and ReBin prediction (Corrected Predictions) on the test set. The x-axis represents actual observed values, and the y-axis represents model predicted values; ideal predictions should be distributed along the 45° diagonal line. It can be seen that the baseline prediction points deviate significantly from the ideal line, exhibiting systematic bias; whereas the ReBin model's scatter points are tightly clustered near the ideal line, indicating that the correction method has significantly improved prediction accuracy. Both model bias and variance have been effectively controlled, verifying the effectiveness of the correction strategy in enhancing model generalization performance. Model comparisons are shown in Table 9.

Table 9. Model performance comparison table

Evaluation Metrics	Baseline Model	ReBin Prediction Model	Explanation
Execution time (seconds)	0.13	0.13	Total time for model training and prediction
Mean Squared Error（MSE）	3.4719	0.2766	Reduce 92.03%
Coefficient of determination（R^2）	0.9140	0.9931	Improve 8.66%
Training sample size	161	161	The amount of data used for training
Test sample size	41	41	The amount of data used for evaluation

Table 9 presents the performance comparison results between the baseline model and the ReBin prediction model in the task of predicting college students' final grades. In terms of computational efficiency, the total execution time of both models is 0.13 seconds, indicating that the ReBin method significantly improves prediction accuracy without introducing additional computational burden. Regarding prediction accuracy, the ReBin model reduces the Mean Squared Error (MSE) from 3.4719 to 0.2766, a significant decrease of 92.03%, demonstrating effective control over prediction errors. In terms of model fit, the coefficient of determination (R^2) increases from 0.9140 to 0.9931, an increase of 8.66%, reflecting that the ReBin model's ability to explain grade variability approaches a perfect fit level. The experiment divides 161 training samples and 41 test samples, verifying the model's validity and generalization ability on the basis of ensuring data sufficiency.

The ReBin model designed in this empirical study has excellent transferability and flexibility. Its core advantage lies in the fact that for any new domain dataset, as long as its features and prediction targets are defined, rapid deployment and prediction can be achieved.

6. CONCLUSIONS

Focusing on the prediction of small sample data, this study proposed a comprehensive solution and achieved remarkable results through multimodal feature fusion and model optimization. First, in the aspect of feature engineering, a multi-modal feature fusion method is innovatively adopted, combining the original data and mathematical statistical characteristics to construct a composite feature space that includes time-domain, frequency-domain, and information entropy features. This provides a richer information representation for the model. Secondly, in terms of model construction, a Sub Box Residual Correction Model with Random Forest as the baseline model was proposed. Through the sub-box strategy and linear regression correction, the prediction accuracy was significantly improved, ultimately achieving an excellent performance of $R^2 = 99\%$, while maintaining a high computational efficiency (the execution time was only 0.59 seconds). In addition, the research also systematically compared a variety of model architectures, including basic Random Forest, Enhanced Random Forest, and XGBoost Residual Correction Model, among others. Through strict cross-validation and performance index analysis, it provided a scientific basis for model selection in different scenarios. This study not only verifies the effectiveness of the proposed method in forecasting small sample data but also provides a valuable reference for research in related fields due to its innovative feature processing and model optimization ideas. The generalizability of this research is limited by its scope and scale. As the data were collected from a single university with a sample of 296 students, the findings may not be fully representative of the broader student population.

Future research can explore several areas. In feature engineering, we can attempt to introduce deep learning technology to achieve automatic feature extraction and selection, particularly for time series data. We can also explore more effective network structures to capture more complex feature patterns. At the same time, feature selection methods based on causal reasoning can be examined to enhance the model's interpretability and generalization ability. In the aspect of model optimization, further study is needed to improve the residual correction mechanism, such as introducing a nonlinear correction model or a correction strategy based on an attention mechanism to deal with more complex error distributions. In addition, given the characteristics of small sample data, we can explore the application of meta-learning or transfer-learning frameworks to improve the model's generalization performance by utilizing big data in related fields. At the level of application expansion, the method proposed in this study can be extended to other small sample prediction scenarios, such as medical diagnosis and financial risk assessment. However, attention should be paid to the treatment of specific characteristics in different fields. Finally, it is suggested that a more systematic theoretical analysis be conducted to further investigate the generalization error boundary of the model under conditions of small samples, thereby providing more solid theoretical support for the method's application. These research directions will help further improve the accuracy and reliability of small sample prediction, promoting the theoretical development and practical application of related fields.

Acknowledgement

The authors would like to thank the Henan Province Higher Education Teaching Reform Research and Practice Project for providing financial support (2024SJGLX0193).

Conflict of Interest

We declare no conflict regarding the publication of the study.

References

[1] Li, X., Ke, S., Li, Y., Jin, W., Fu, X., Fu, G., & Bi, W. (2024). Temperature compensation based on BP neural network with small sample data for chloride ions optical fiber probe. *Optics & Laser Technology*, *176*, 110973. https://doi.org/10.1016/j.optlastec.2024.110973

[2] Yu, C., Li, W., Guo, Y., Sun, X., Hong, F., Sun, N., & Zhang, Q. (2024). Research on wear rate of train brake pads driven by small sample data. *Wear*, *536*, 205169. https://doi.org/10.1016/j.wear.2023.205169

[3] Wu, Z., Ma, C., Zhang, L., Gui, H., Liu, J., & Liu, Z. (2024). Predicting and compensating for small-sample thermal information data in precision machine tools: A spatial-temporal interactive integration network and digital twin system approach. *Applied Soft Computing*, *161*, 111760. https://doi.org/10.1016/j.asoc.2024.111760

[4] Li, C., Wang, L., Li, J., & Chen, Y. (2024). Application of multi-algorithm ensemble methods in high-dimensional and small-sample data of geotechnical engineering: A case study of swelling pressure of expansive soils. *Journal of Rock Mechanics and Geotechnical Engineering*, *16*(5), 1896-1917. https://doi.org/10.1016/j.jrmge.2023.10.015

[5] Cheng, Z., Chen, F., Zuo, E., Gong, Z., Xiao, M., Li, C., Liu, Y., Liu, P., Chen, C., Lv, X. & Chen, C. (2025). The MLSE-SCAM architecture combines with the improved DRSN-TIC model for Raman spectroscopy small-sample data learning. *Expert Systems with Applications*, *279*, 127462. https://doi.org/10.1016/j.eswa.2025.127462

[6] Marbouti, F., Diefes-Dux, H. A., & Madhavan, K. (2016). Models for early prediction of at-risk students in a course using standards-based grading. *Computers & Education*, *103*, 1-15. https://doi.org/10.1016/j.compedu.2016.09.005

[7] Lakkaraju, H., Aguiar, E., Shan, C., Miller, D., Bhanpuri, N., Ghani, R., & Addison, K. L. (2015, August). A machine learning framework to identify students at risk of adverse academic outcomes. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining,* Sydney, NSW, Australia (pp. 1909-1918). https://doi.org/10.1145/2783258.2788620

[8] Zeineddine, H., Braendle, U., & Farah, A. (2021). Enhancing prediction of student success: Automated machine learning approach. *Computers & Electrical Engineering*, *89*, 106903. https://doi.org/10.1016/j.compeleceng.2020.106903

[9] Aggarwal, D., Mittal, S., & Bali, V. (2021). Significance of non-academic parameters for predicting student performance using ensemble learning techniques. *International Journal of System Dynamics Applications (IJSDA)*, *10*(3), 38-49. DOI: 10.4018/IJSDA.2021070103

[10] Bache K. & Silva A, M. G. (2013).The development of a student performance prediction model for improved student retention. In Proceedings of the 5th Annual Future Business Technology Conference.

[11] Kovacic Z. (2010).Early prediction of student success: Mining students' enrolment data. In Proceedings of the InSITE 2010: Informing Science , IT Education Conference, Cassino, Italy.

[12] Hu, Y. H., Lo, C. L., & Shih, S. P. (2014). Developing early warning systems to predict students' online learning performance. *Computers in Human Behavior*, *36*, 469-478. https://doi.org/10.1016/j.chb.2014.04.002

[13] Schlegl, T., Seeböck, P., Waldstein, S. M., Langs, G., Schmidt-Erfurth, U. & Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery[C]. International Conference on Information Processing in Medical Imaging.Cham:Springer, 145-157.https://doi.org/10.48550/arXiv.1703.05921

[14] Kuzilek, J., Hlosta, M., Herrmannova, D., Zdrahal, Z., Vaclavek, J., & Wolff, A. (2015). OU Analyse: analysing at-risk students at The Open University. *Learning analytics review*, 1-16. http://www.laceproject.eu/learning-analyticsreview/analysing-at-risk-students-at-open-university/

[15] Xu, J., Moon, K. H., & Van Der Schaar, M. (2017). A machine learning approach for tracking and predicting student performance in degree programs. *IEEE Journal of Selected Topics in Signal Processing*, *11*(5), 742-753. https://doi.org/10.1109/JSTSP.2017.2692560

[16] Watson, C., Li, F. W., & Godwin, J. L. (2013, July). Predicting performance in an introductory programming course by logging and analyzing student programming behavior. In *2013 IEEE 13th international conference on advanced learning technologies* (pp. 319-323). IEEE. https://doi.org/10.1109/ICALT.2013.99

[17] Al-Shehri, H., Al-Qarni, A., Al-Saati, L., Batoaq, A., Badukhen, H., Alrashed, S., Alhiyafi, J. & Olatunji, S. O. (2017, April). Student performance prediction using support vector machine and k-nearest neighbor. In *2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE)* (pp. 1-4). IEEE. doi: 10.1109/CCECE.2017.7946847.

[18] Buenaño-Fernández, D., Gil, D., & Luján-Mora, S. (2019). Application of machine learning in predicting performance for computer engineering students: A case study. *Sustainability*, *11*(10), 2833. https://doi.org/10.3390/su11102833

[19] Niyogisubizo, J., Liao, L., Nziyumva, E., Murwanashyaka, E., & Nshimyumukiza, P. C. (2022). Predicting student's dropout in university classes using two-layer ensemble machine learning approach: A novel stacked generalization. *Computers and Education: Artificial Intelligence*, *3*, 100066. https://doi.org/10.1016/j.caeai.2022.100066

[20] Thaker K.,Huang Y., Brusilovsky P., Daqing H. (2018).Dynamic knowledge modeling with heterogeneous activities for adaptive textbooks. In Proceedings of the 11th International Conference on Educational Data Mining, Buffalo, NY, USA, 592–595.

[21] He, J., Bailey, J., Rubinstein, B., & Zhang, R. (2015). Identifying At-Risk Students in Massive Open Online Courses. *Proceedings of the AAAI Conference on Artificial Intelligence*, *29*(1), 25-30. https://doi.org/10.1609/aaai.v29i1.9471

[22] Iqbal, Z., Qadir, J., Mian, A. N., & Kamiran, F. (2017). Machine learning based student grade prediction: A case study. *arXiv preprint arXiv:1708.08744*. https://doi.org/10.48550/arXiv.1708.08744

[23] Chowdhury, S., Mayilvahanan, P., & Govindaraj, R. (2022). Optimal feature extraction and classification-oriented medical insurance prediction model: machine learning integrated with the internet of things. *International Journal of Computers and Applications*, *44*(3), 278-290. https://doi.org/10.1080/1206212X.2020.1733307

[24] Daud, A., Aljohani, N. R., Abbasi, R. A., Lytras, M. D., Abbas, F., & Alowibdi, J. S. (2017, April). Predicting student performance using advanced learning analytics. In *Proceedings of the 26th international conference on world wide web companion* (pp. 415-421). https://doi.org/10.1145/3041021.3054164

[25] Kotsiantis, S., Patriarcheas, K., & Xenos, M. (2010). A combinational incremental ensemble of classifiers as a technique for predicting students' performance in distance education. *Knowledge-Based Systems*, *23*(6), 529-535. https://doi.org/10.1016/j.knosys.2010.03.010

[26] Sghir, N., Adadi, A., & Lahmer, M. (2023). Recent advances in predictive learning analytics: A decade systematic review (2012–2022). *Education and information technologies*, *28*(7), 8299-8333. https://doi.org/10.1007/s10639-022-11536-0