International Journal of Advanced Science and Computer Applications (2025); 4(2)
DOI: 10.47679/ijasca.v4i2.76
https://ijasca.org/index.php/ijasca

RESEARCH ARTICLE

Prediction Of Shear Wall Residential Beam Height Based On Machine Learning

Dejiang Wang , Lijun Chen

Published online: 21 September 2025

Abstract

The beam height is an important design parameter that influences structural properties such as load-bearing capacity and stability of beams. In the early stages of structural design, the existing methods for determining beam height mainly include empirical formulae. However, empirical methods are highly subjective, lack accuracy, and are poorly adapted to complex conditions. This paper establishes a beam height prediction model for shear wall residential structures. Using structural design data from projects built by a real estate company across various regions in China, a large dataset of beam heights was collected. The Permutation Feature Importance (PFI) method and six unique machine learning (ML) models were used to rank the importance of input variables. The Gradient Boosting (GB) model, consistent with the feature ranking obtained from PFI, was selected. The model evaluation method was then used to select the number of input features for the GB model, and grid search and K-fold cross-validation were employed to optimize the GB prediction model. This model was compared with a prediction model obtained from a Back Propagation Neural Network (BPNN). Finally, the SHAP method was used to interpret the "black box" machine learning model. The results show that the GB model has higher accuracy compared to the BPNN model, and the input features of the proposed GB model contribute to the beam height in accordance with mechanical laws, demonstrating the model's rationality. The research findings can provide a reference for initial beam height design.

Keyword: Feature selection, Machine learning, Model determination, Beam height, SHAP

Introduction

In urban construction, shear wall structures are a common form of structure for residential buildings. Beams, which connect the top and bottom of shear walls, are critical components. They play a vital role in bearing loads, stiffening shear walls, and providing space. The height of the beam is an important parameter in beam design, directly affecting the load-bearing capacity and structural stability of the beam. Therefore, selecting the correct beam height is crucial for ensuring the performance of residential structures.

Currently, the method primarily used for determining beam height at the early stages of structural design is based on empirical formulas. This method derives from the accumulated experience and practices of predecessors, summarizing commonly used values for beam height selection to guide actual engineering design. One commonly used empirical formula for selecting beam height based on span is $h = (1/12 \sim 1/8)\ l$, which provides a range for the beam height, h [1]. This formula is simple, easy to implement, and widely used in various engineering practices. It is especially useful in the preliminary design stage, offering a quick and effective method for estimating beam height, which provides an important reference for more detailed design. However, the empirical formula method demands high expertise from estimators and may involve substantial errors due to subjective biases. It also fails to meet the needs of complex engineering requirements. In specific engineering designs, structural analysis and calculations are still required based on the actual conditions to determine the final beam height. Therefore, there is an urgent need to establish a rapid and accurate prediction model for the preliminary design of beam heights.

In recent years, machine learning methods have increasingly gained popularity [2]. These methods can derive reliable predictive models by training and learning the deep patterns within existing empirical data, showcasing good predictive performance and generalization capabilities, which have attracted significant attention from many researchers [3, 4] and have been widely applied in the engineering field [5]. Scholars have applied machine learning algorithms to predict various structural properties of reinforced concrete (RC) beams, such as torsional bearing capacity [6, 7], shear bearing capacity [8, 9], bending capacity [10, 11], cracking performance [12], fire resistance [13, 14], stiffness [15], maximum displacement [16, 17], and seismic energy absorption [18, 19]. Abushanab et al. [20] compared an optimized GB model with four standalone machine learning models and found that the GB model could efficiently and intelligently predict the bending capacity of corroded RC beams. Cai et al. [21] discovered that BPNN could

Department of Civil Engineering, Shanghai University, Shanghai 200444, China

*) corresponding author

Email: chenlijun0326@163.com

Received 21 May 2025 | Received in revised form 07 Juli 2025 | Accepted 09 August 2025
Available online 15 September 2025

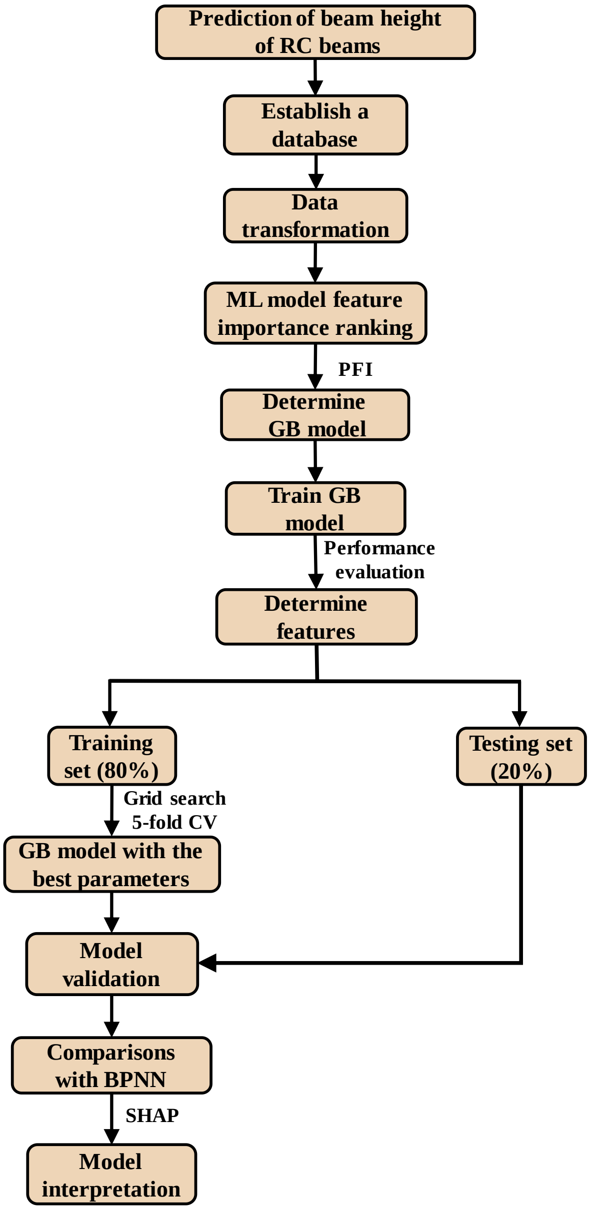
accurately predict the shear bearing capacity of RC beams after exposure to fire within the range of input parameters. Tipu et al. [22] proposed using BPNN to predict the shear performance of RC beams reinforced with a fiber-reinforced cementitious matrix. Zhao et al. [23] developed a backpropagation neural network using particle swarm and whale optimization algorithms to predict the deflection of RC beams under concentrated loads.

Although machine learning models have performed well in predicting the structural performance of beams and interpreting experimental data, the data-driven approach to predicting beam height has not yet been explored. In the early stages of structural design, predicting beam height using machine learning offers several advantages over traditional structural analysis calculations and empirical formula estimates: (1) Data-driven decision-making: Predictions are based on extensive historical data and case studies, making the decision process more reliant on data rather than solely on theory and simplified assumptions. (2) Multivariable interactions: ML models can handle complex interactions between multiple variables, which is crucial in structural design. (3) Robustness and adaptability: ML models have the ability to learn and adapt to new situations, maintaining robust predictions even when data changes or under unknown conditions. In contrast, structural analysis software typically runs on fixed algorithms and parameters and may require reprogramming or adjustments when encountering situations that do not fit preset assumptions. (4) Interpretability: While some complex machine learning models are considered "black boxes," modern interpretable machine learning techniques allow us to understand and explain the predictions of these models. Through these techniques, key factors influencing beam height predictions can be identified, understanding how different features affect the outcomes. This provides engineers with intuitive guidance when making design decisions, helps them verify the reliability of model outputs, and offers deeper insights into the design process.

In the context described above, to enhance the speed, accuracy, and adaptability of beam height in the preliminary design of structures, this paper will combine feature selection to choose regression models for predicting beam height. Optimal parameters for the regression models will be selected, and a model based on these optimal parameters will be compared with the BPNN to identify the model with the best predictive performance. Additionally, an analysis of the contribution of the features input into this model will be conducted.

Method

To construct a high-accuracy beam height prediction model that aligns with mechanical principles, this paper undertakes the following tasks, as shown in Fig. 1. The workflow begins with establishing a beam height database and performing data preprocessing. Next, the PFI method is used to rank the importance of features. Subsequently, this paper selects six regression models and ranks features based on each model's inherent feature ranking method to identify the GB model that aligns with the PFI feature ranking. The process continues by progressively reducing the number of features input into the GB model, training the model, and observing how the model evaluation metrics change with the number of input features, ultimately determining the final set of input features. To achieve a prediction model based on optimal parameters, grid search and five-fold cross-validation are used to retrain the GB model. The BPNN is introduced as a comparative model, and both models undergo performance evaluation. Additionally, using the SHAP method, an analysis of the contribution of each input feature in the GB model is conducted to examine whether the impact of these features on beam height conforms to mechanical principles. Building on this foundation, the final prediction model obtained is more accurate than traditional empirical methods and provides a reference for subsequent structural design.



Technology roadmap.

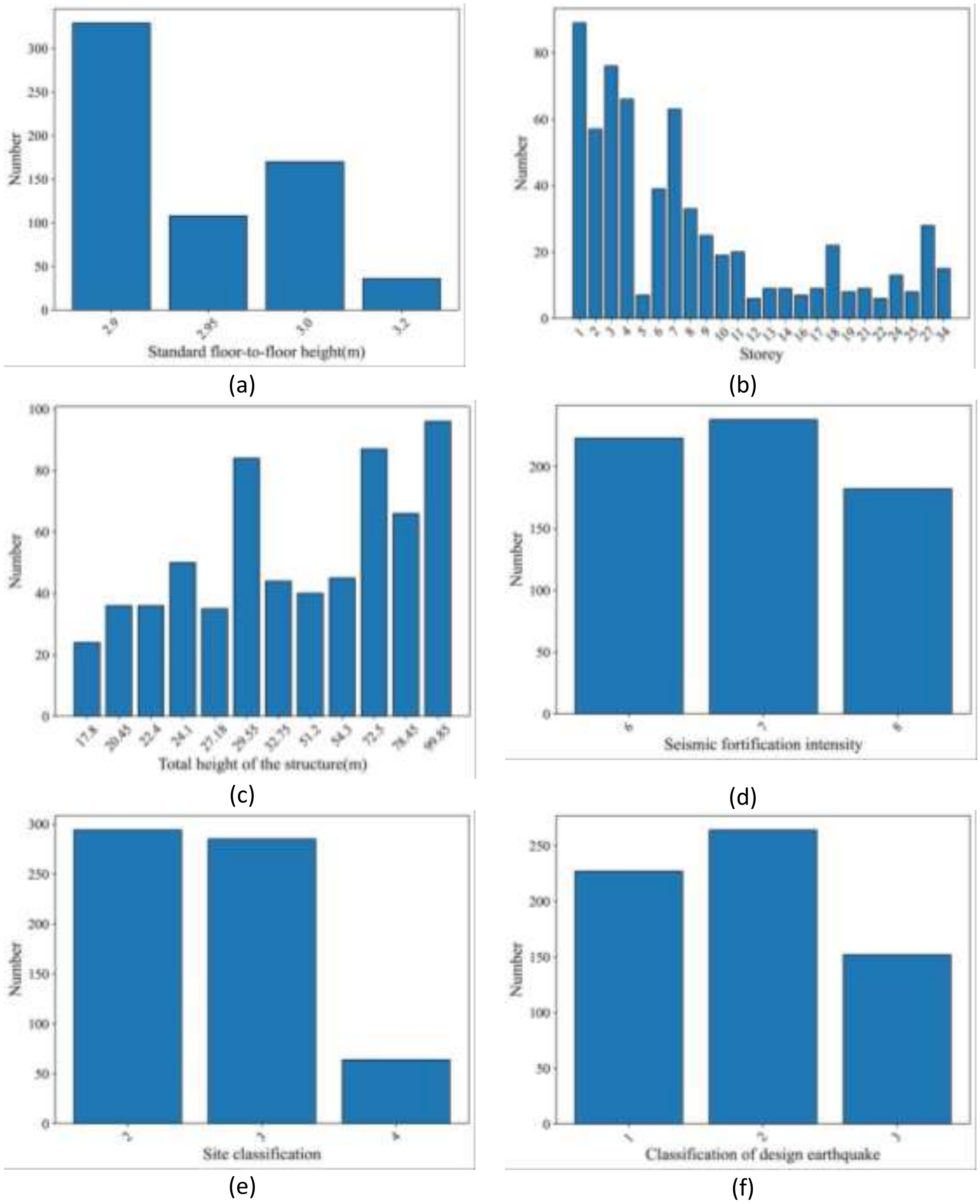
Database description

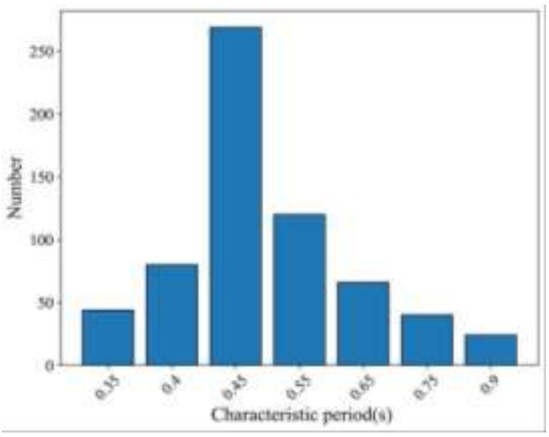
This study selected data from the project design data of existing projects built by a real estate company across various regions in China, comprising 643 data samples. The database includes input variables such as beam span, total height of the structure, beam width, building shape coefficient, storey, seismic fortification intensity, standard floor-to-floor height, site classification, basic wind pressure, terrain roughness, characteristic period, classification of design earthquake, and concrete strength grade. The output parameter is beam height. Table 1 shows the statistical characteristics of each input and output parameter in the dataset, and Fig. 2 illustrates the relationship between each input parameter and beam height in the dataset. The dataset exhibits a high degree of dispersion and a wide range of values, with a relatively uniform distribution of parameters, thus providing a reliable data foundation for establishing a data-driven model for predicting beam height.

Table 1. Statistical information of parameters included in the database.

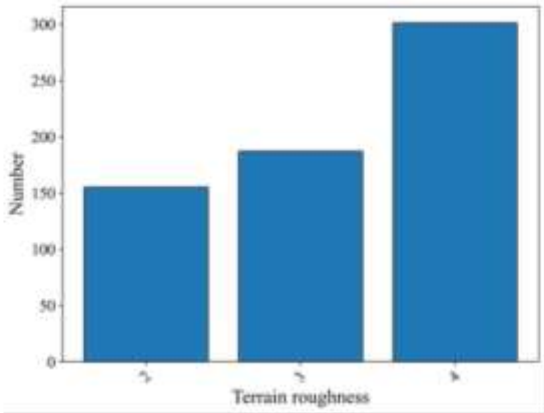
Variables	Maximum	Minimum	Median	Mean	Standard deviation
standard floor-to floor height (m)	3.2	2.9	2.9	2.95	0.074
storey	34	1	6	9	8
total height of the structure (m)	99.85	17.8	51.2	52.27	28.39
seismic fortification intensity	8	6	7	7	0.79
site classification	4	2	3	3	0.66
classification of design earthquake	3	1	2	2	0.76
characteristic period(s)	0.9	0.35	0.45	0.51	0.13

Input	terrain roughness	4	2	3	3	0.81
	basic wind pressure (kN/m²)	0.75	0.35	0.45	0.48	0.12
	building shape coefficient	0.37	0.21	0.29	0.29	0.05
	concrete strength grade (MPa)	35	30	30	31	1.94
	beam span (m)	6.4	0.6	2.7	3.03	1.43
	beam width (mm)	600	150	200	207	31.3
Output	beam height (mm)	1700	200	500	537	206

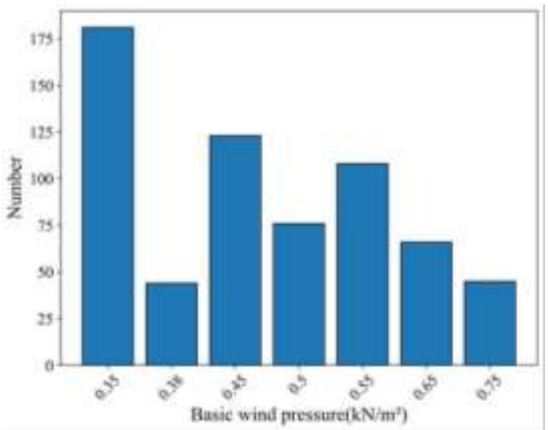




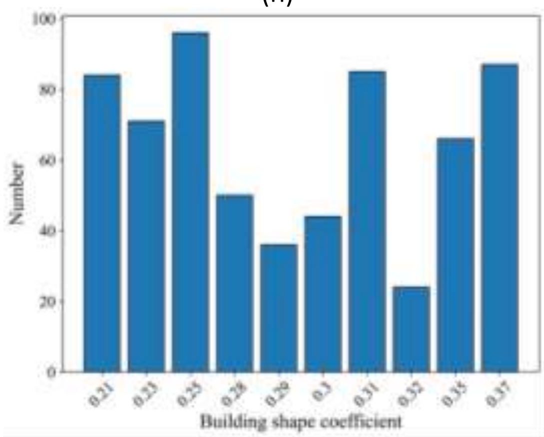
(g)



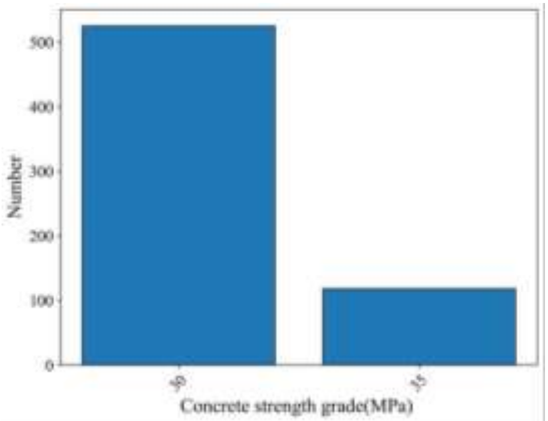
(h)



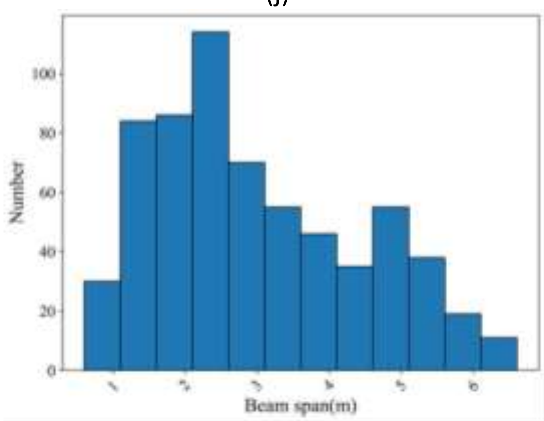
(i)



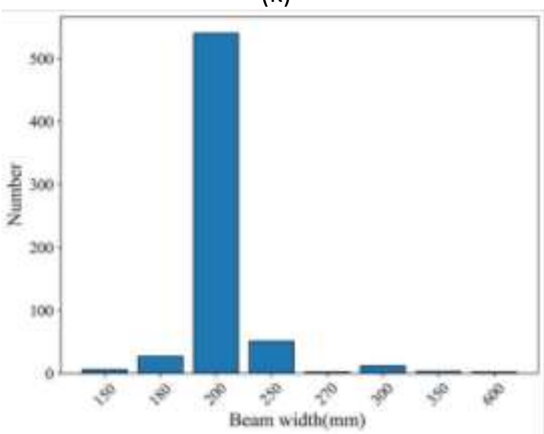
(j)



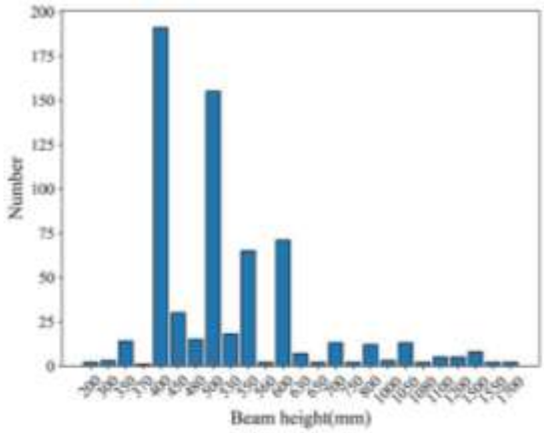
(k)



(l)



(m)



(n)

Fig 1. Statistical distributions of the feature from the different models.

Data transformation

To facilitate the model's handling and computation of sample data, it is necessary to quantify and normalize the preliminary data obtained. Quantitative and qualitative indicators are important metrics for evaluating engineering projects or other entities. Quantitative indicators can be directly quantified using numbers or mathematical methods, such as total height of the structure and beam width. The advantage of quantitative indicators is that the data are accurate, specific, and quantifiable, making them easier to process and analyze when building predictive models, resulting in more objective and reliable outcomes. Qualitative indicators, on the other hand, refer to metrics that cannot be directly quantified with numbers and require descriptive language to express. For example, seismic fortification intensity, site classification, and terrain roughness. Since qualitative indicators cannot be directly quantified, they need to be converted into discrete values when establishing predictive models so that the model can recognize and utilize this data. In this paper, qualitative indicators like site classification I, II, III, and IV are quantified as 1, 2, 3, 4, respectively; classification of design earthquake groups I, II, III, and IV are quantified as 1, 2, 3, 4; and terrain roughness categories A, B, C, D are quantified as 1, 2, 3, 4. The remaining 10 input variables and 1 output variable are represented quantitatively by their original values. Additionally, to account for the influence of storey level on different residences, this paper defines the storey feature in the database using the ratio of the storey level to the total number of storeys in the structure, referred to as the storey ratio.

After quantification, the training samples still pose some challenges for model recognition and processing. To simplify the computational complexity of the model, linear normalization is used to process the quantified data samples. This method removes the effects of dimensions and ensures that the input data is evenly distributed and converges easily. All input features' quantified values are compressed within the range of [0, 1]. The basic principle is as follows:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

In the formula, x' is the value after data normalization, x is the original data value, and $x_{\max}$, $x_{\min}$ are respectively the specified maximum and minimum values for the indicators.

Input features selection

In this section, the PFI method is used to rank the importance of input features. To further enhance the accuracy of the prediction model, once the model is finalized, the number of input features will be selected based on model evaluation metrics.

Feature importance ranking

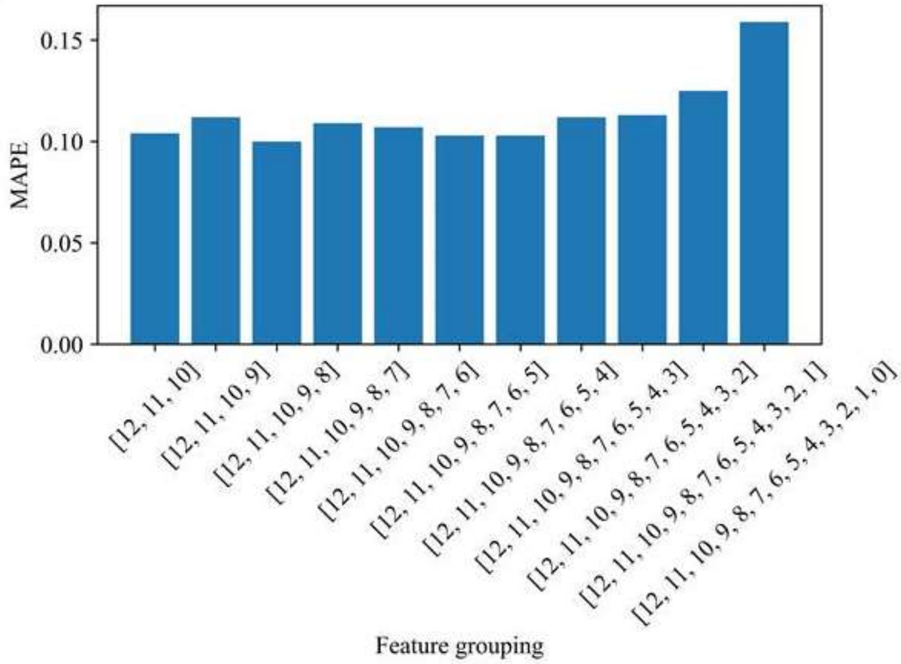
The PFI method measures the importance of a feature by calculating the increase in model prediction error when permuting that feature during training data. If scrambling the values of a feature increases the model's error, then that feature is important because the model relies on that characteristic for making predictions. Conversely, if scrambling a feature's values does not change the model's error, then that feature is considered unimportant, as the model ignores it for predictions [24]. Table 2 shows the importance ranking of 13 features for six ML models, calculated using the PFI method. Table 2 indicates that the model prediction error for each feature fluctuates to some extent across different models, but the importance rankings are similar, which in turn validates the reliability of the six ML models. Features such as beam span, total height of the structure, and beam width have a significant impact on beam height, whereas the classification of design earthquake and concrete strength grade have negative PFI values, indicating a lesser impact on beam height, thus these two features are not considered. Seismic fortification intensity and design earthquake classification both affect the seismic design of the structure, and Table 2 shows that different ML models prioritize seismic fortification intensity as an important factor affecting beam height. In typical residential buildings, the variation in concrete strength of beams is not large, and conventional strength concrete is generally sufficient to meet structural requirements, thus the collected concrete strength grade data reflects its minor impact on beam height, making it reasonable for concrete strength grade to rank last in the PFI order.

Table 2. Permutation feature importance of 13 features.

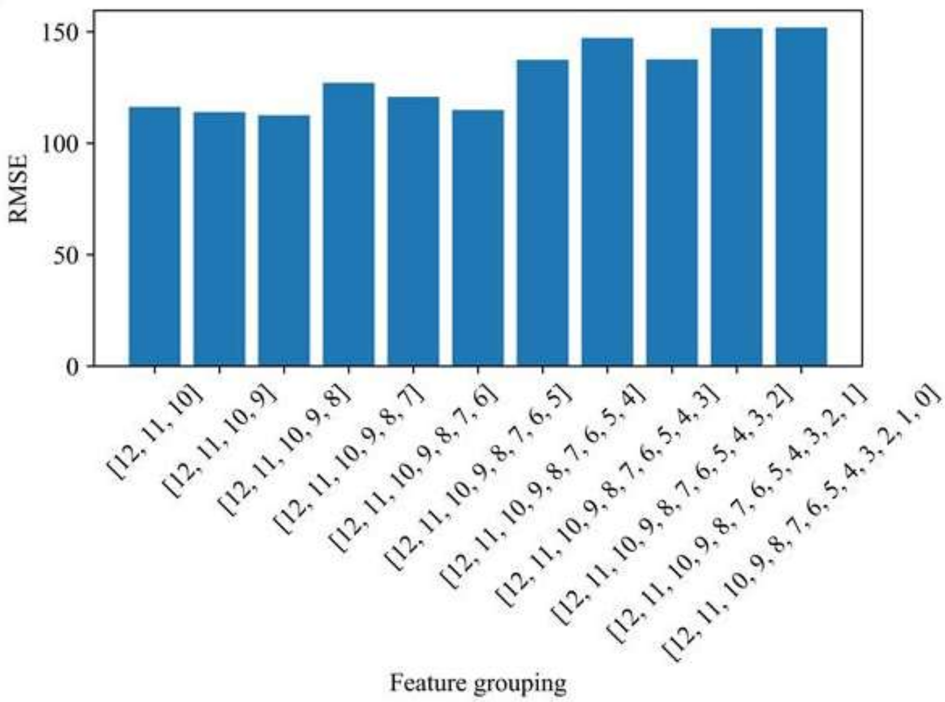
Model	Number	LASSO	SVR	KNN	DT	RF	GB	mean
Beam span (m)	0	0.137	0.006	0.193	1.602	1.061	1.016	0.669
Total height of the structure (m)	1	0.091	0.005	0.147	1.190	0.391	0.702	0.421
Beam width (mm)	2	0.073	0.004	0.126	0.658	0.324	0.139	0.221
Building shape coefficient	3	0.036	0.003	0.049	0.118	0.108	0.049	0.061
Storey ratio	4	0.029	0.003	0.044	0.117	0.084	0.046	0.054
Seismic fortification intensity	5	0.018	0.003	0.039	0.113	0.063	0.015	0.042
Standard floor-to-floor height (m)	6	0.009	0.002	0.039	0.083	0.060	0.012	0.034
Site classification	7	0.005	0.002	0.031	0.056	0.025	0.007	0.021
Basic wind pressure (kN/m²)	8	-0.002	0.001	0.030	0.020	0.015	0.006	0.012
Terrain roughness	9	-0.004	0.001	0.009	0.010	0.013	0.006	0.006
Characteristic period (s)	10	-0.005	0.001	0.001	0.008	0.008	0.005	0.003
Classification of design earthquake	11	-0.006	0.001	-0.011	0.003	0.001	0.002	-0.002
Concrete strength grade (MPa)	12	-0.059	-0.002	-0.042	0.000	-0.005	0.000	-0.018

Feature determination

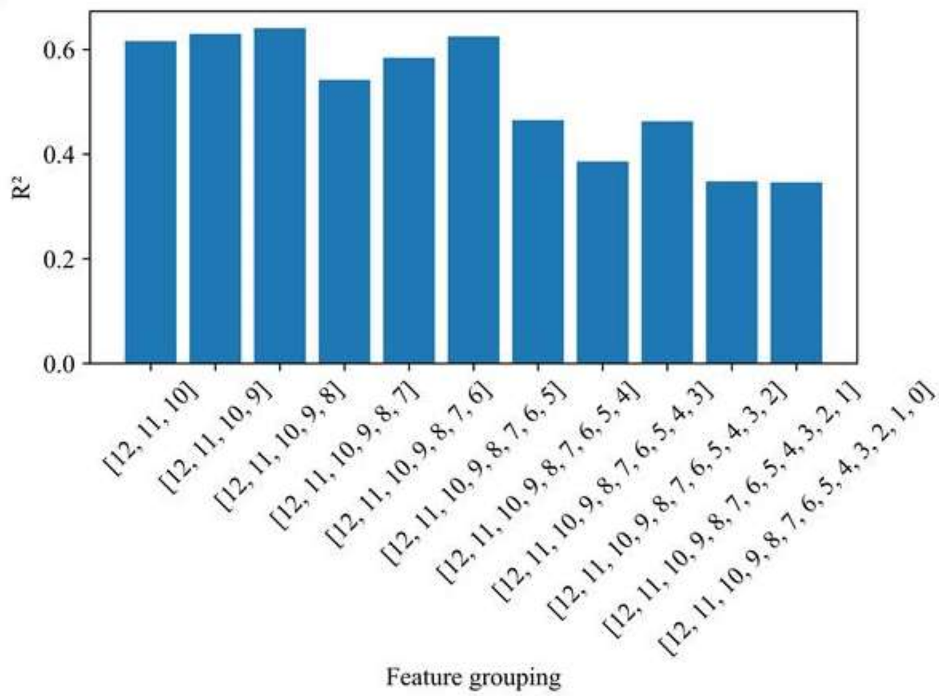
For ML models, the number of input features can significantly impact the model's predictive accuracy. Therefore, this section evaluates the accuracy of predictions made by the predictive model using different numbers of input features, based on the feature importance ranking obtained through the PFI method. The predictive accuracy is illustrated in Fig. 3. The predictive model referred to in this section is the GB model mentioned in section 5.1. The evaluation metrics used include the Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Coefficient of Determination (R^2). The input features are taken from the sequence numbers corresponding to Table 2. As shown in Fig. 3, the combination that excludes features 8, 9, 10, 11, and 12, namely, basic wind pressure, terrain roughness, characteristic period, classification of design earthquake, and concrete strength grade, results in the smallest MAPE and RMSE and the largest R^2. Therefore, this combination yields the highest model accuracy.



(a)



(b)



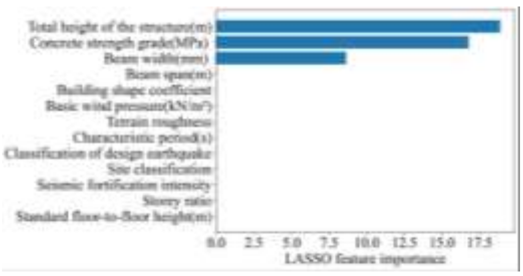
(c)

Fig 2. Model evaluation of different feature grouping.

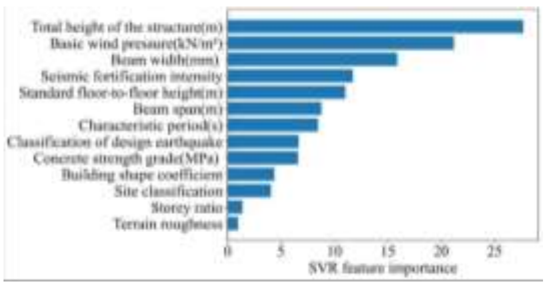
Results and discussion

Model determination

To minimize redundancy in predicting beam height, this section employs the unique feature importance ranking methods of six ML models to identify models consistent with the PFI method ranking. For instance, LASSO regression uses L1 regularization to constrain the coefficients of the regression model, causing some coefficients to become zero and thus enabling feature selection. The absolute values of the model's coefficients can be used as a measure of feature importance, with larger coefficients indicating more important features [25]. The importance of features in the SVR model is determined by observing the distribution of support vectors, which are key sample points in prediction. Features corresponding to these support vectors are considered important and can be measured using the coefficients of the support vectors [26]. The importance of features in the KNN model is usually determined based on the distance and weight of the nearest neighbors. Neighbors closer to the sample being predicted contribute more to the prediction, and their corresponding features are considered important [27]. The DT model builds trees recursively by splitting features, where each node represents a feature and a split point. Its feature importance is determined by calculating the reduction in the splitting criterion for each feature during the splitting process, measuring each feature's contribution to the predictive power of the individual decision tree. In contrast, the RF model's feature importance is global, considering the contributions of all decision trees in the entire random forest. Its feature importance is calculated through the average split gain or impurity reduction across the whole forest [28]. The GB model iteratively trains a series of weak learners (usually decision trees), adjusting the weights of features in each iteration based on the performance of the previous model to focus more on features with stronger predictive capabilities. This allows GB to progressively select the most important features [29]. In summary, the feature importance for the tree-based DT, RF, and GB models can be obtained using the feature_importances_ attribute in the scikit-learn library in Python [30]. For the LASSO and SVR models, feature coefficients are obtained by fitting each respective model [31]. The KNN model determines feature importance by considering each feature's contribution to the overall sample distance metric and ranks them accordingly. The unique feature importance rankings for the six regression models are shown in Fig. 4(a)-(f).



(a)



(b)

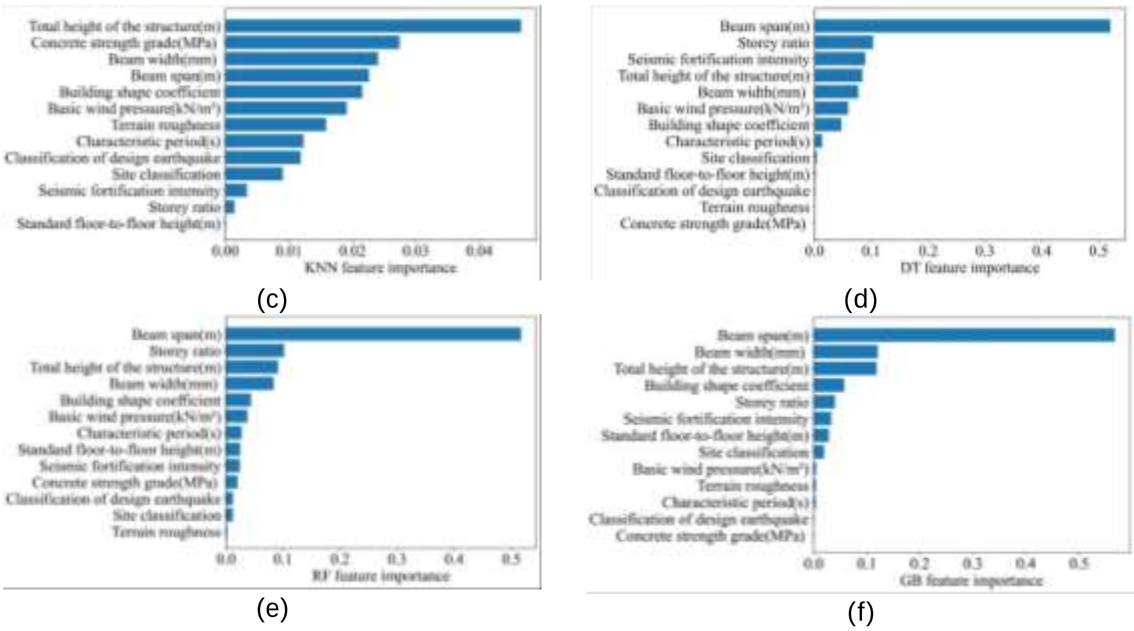


Fig 3. Feature importance of six ML models.

By comparing the model-independent PFI feature importance ranking method shown in Table 2 with the feature importance rankings derived from the individual methods of the six ML models shown in Fig. 4, it is observed that the feature ranking of the GB model in Fig. 4(f) is essentially consistent with the ranking obtained from PFI. Therefore, this paper selects the GB model as the ML model for predicting beam height.

Model training process

Following the workflow outlined in Fig. 1, a beam height prediction model based on the GB model has been established. The specific steps in model training include dataset division, model training, and performance evaluation. The dataset is first normalized and then divided into a training set and a test set in an 8:2 ratio, with the training set comprising 514 data samples and the test set comprising 129 data samples. The GB model is then trained using the training set, applying K-fold cross-validation and grid search to tune the hyperparameters. The training aims to optimize the MAE as the objective, ultimately obtaining a predictive model under the globally optimal hyperparameter combination. Cross-validation involves further dividing the training set into subsets for training and validation, cycling through multiple iterations to replace the training set, and assessing the model's predictive performance based on the average prediction accuracy from multiple results to minimize biases caused by random sample selection. Grid search employs an exhaustive method to combine different hyperparameter values, selecting the best-performing combination to avoid settling for a locally optimal set. Finally, performance evaluation tests the model based on the optimal parameters using the test set. The model's predictive accuracy is quantified using metrics such as MAE, MAPE, RMSE, and R^2.

The hyperparameter combination for the GB-based beam height prediction model is presented in Table 3.

Table 3. Hyperparameter combination of GB model.

Parameters	Adjustment range	Retrieve values	Implication
a	[0, 1]	0.4	learning rate
s	[0, ∞]	300	n_estimators
d	[0, ∞]	5	max_depth
w	[0, ∞]	1	min_samples_leaf
r	[0, ∞]	0	min_impurity_decrease

Model comparison

To compare the accuracy of the beam height prediction model based on GB with other models, a beam height prediction model using BPNN is also established using the same dataset. The design of the BPNN includes selecting the number of hidden layers, the number of nodes in each hidden layer, the choice of activation function, and the selection of performance evaluation metrics. The specific design of the BPNN is as follows:

(1) The number of layers can be set arbitrarily, but more layers make the neural network structure more complex and the computation more involved. Research shows that a three-layer BPNN can almost handle all mapping relationships in fitting problems [32]. To reduce the risk of overfitting during training, the Dropout technique is applied to the hidden layers. Therefore, this study opts for a three-layer network structure.

(2) The number of nodes in the hidden layers is determined using common empirical formulas, which include the following four formulas [33]:

$$y = \sqrt{x + l} + a \tag{2}$$

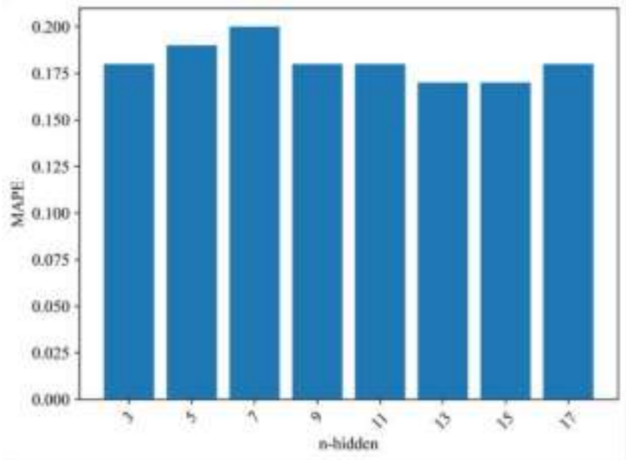
$$y = \log 2^{x} \tag{3}$$

$$y = \sqrt{xl} \tag{4}$$

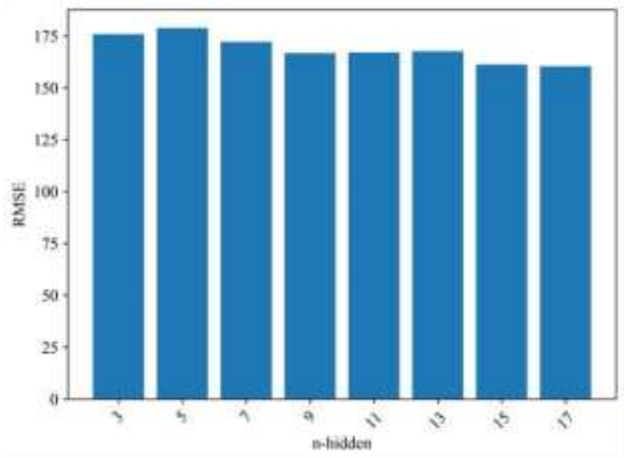
$$y = 2x + 1 \tag{5}$$

y represents the number of neurons in the middle layer, x represents the number of indicators in the first layer, l is the number of indicators in the last layer, α is any number between [1,10]. In this paper, x is 8, l is 1, applying the above four formulas, y ranges from 3 to 17. During the actual modeling process, adjustments can be made based on the model error results. The model accuracy evaluation for each number of neurons in the hidden layers is shown in Fig. 5, where n-hidden represents the number of hidden layers. It can be observed that when the number of neurons in the hidden layers is 17, the RMSE is the smallest, R^2 is the largest, and MAPE is the second smallest. Overall, the BPNN model achieves the highest accuracy when the number of neurons in the hidden layers is 17.

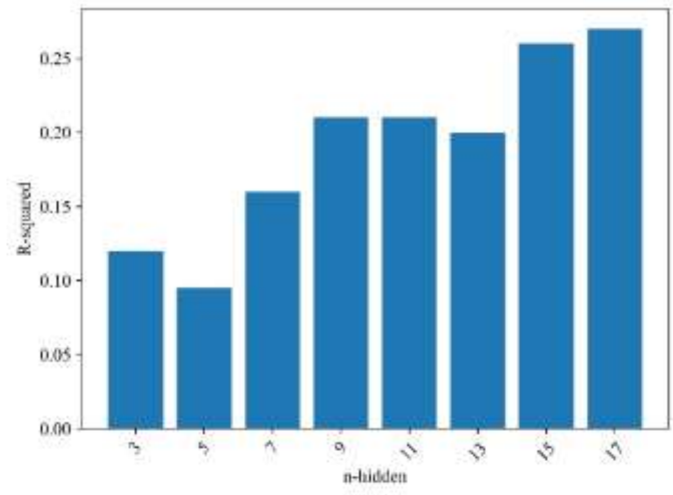
(3) The activation function selected is tanh, which is used to enhance the network's fitting capabilities, making it suitable for mapping non-linear relationships; the iteration count is set to 300 times.



(a)



(b)

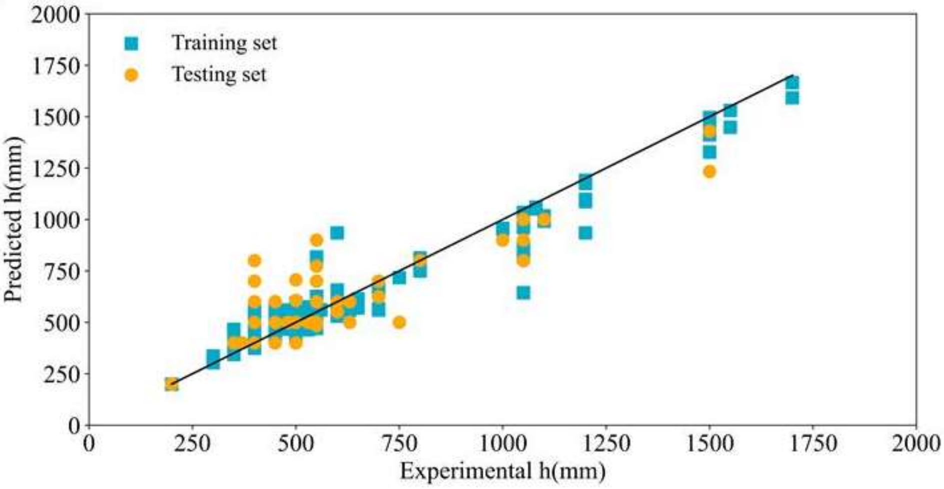


(c)

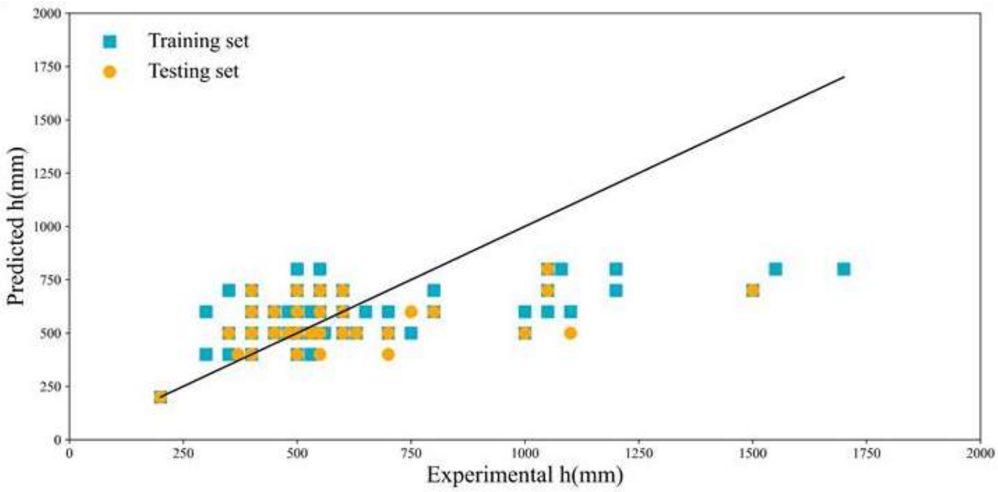
Fig 4. BPNN accuracy under different number of hidden layer neurons.

Discussion

Figure 6 shows the comparison of predicted values versus actual values for the GB model and the BPNN model. From the graph, it is visually apparent that the predictions from the GB-based model are more closely clustered along the line $y=x$ compared to those from the BPNN. This indicates that the beam height prediction model based on the GB algorithm provides better prediction accuracy for the samples.



(a)GB



(b)BPNN

Fig 5. Comparison of predicted and experimental beam height.

Table 4 lists the comparison results of the two models in terms of R^2, RMSE, MAPE, and MAE. From the table, it can be further seen that the R^2 value for the model based on the GB algorithm reached 0.67, which is significantly higher than the 0.27 achieved by the BPNN, representing an improvement of about 148% and indicating a better fit to the samples. At the same time, compared to BPNN, the RMSE decreased by about 32.5%, and MAPE and MAE decreased by about 46.1% and 53.3% respectively. This demonstrates that the GB model's predictions are closer to the actual measurements, and its predictive accuracy is clearly superior to that of BPNN.

Table 4. Evaluation of different models.

Basic algorithms	R^2	RMSE	MAPE	MAE
GB	0.67	108.82	0.097	48.25
BPNN	0.27	160.37	0.18	103.3

Model interpretations using SHapley Additive exPlanations

In model prediction, understanding the reasons behind a model's predictions and their accuracy is just as important as the predictions themselves. For simple models (such as linear models or single decision tree models), the model itself is interpretable; however, for ensemble learning algorithms or deep learning models, although these models offer better predictive performance, their complexity reduces their interpretability. To address the interpretability issue of the GB model, the results of the GB model after parameter optimization are incorporated into the SHAP attribution analysis model to analyze the impact of various variables on beam height in detail.

SHAP is an additive feature attribution method introduced by Lundberg et al [34]. It is an explanation framework designed to elucidate how models process features to arrive at their final predictions. Based on cooperative game theory, it quantifies the impact of each feature on the outcome by calculating the contribution value of each feature to the prediction result. These contribution values can be positive or negative, where positive values indicate an enhancement of the prediction result, and negative values indicate a reduction. Furthermore, the larger the role of a feature in the model, the greater the absolute value of its contribution, and consequently, the higher its importance.

Global feature interpretation

The predictive model established in this article includes eight input features. The global influence of these features in the GB model according to SHAP is displayed in Fig. 7. From top to bottom, the importance of the features gradually decreases. The color of the dots changes from blue to red, indicating that the value of the feature increases from low to high. Each dot represents the SHAP value for a sample, which signifies the contribution of that feature to an individual prediction. The aggregation of these points illustrates the overall direction and magnitude of the impact that the feature has on the prediction outcomes.

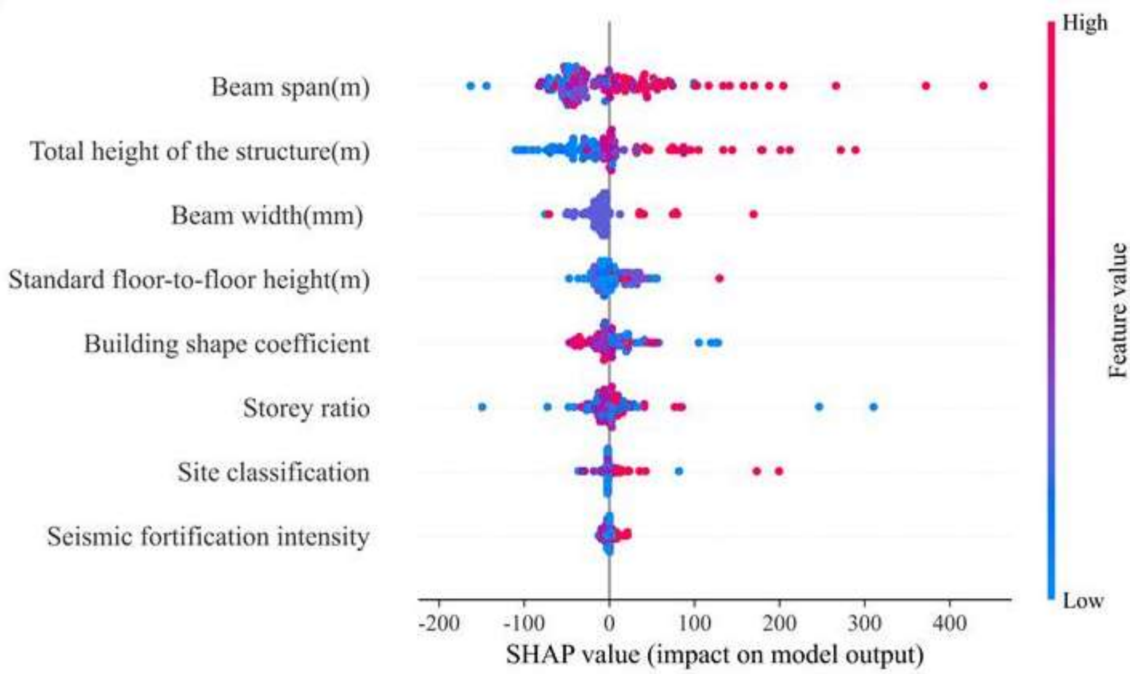


Fig 6. Global interpretation of the GB model.

As shown in Fig. 7, among the eight features, the beam span is the most important feature affecting the model's prediction outcomes. Beam span, total height of the structure, and beam width show a clear positive correlation with the predicted beam height. This correlation exists because as the beam span increases, the beam must bear more load, and its deflection increases. To enhance bending stiffness and reduce deflection, it is necessary to correspondingly increase the beam's height. When the total height of the structure is low, the change in wind load is minimal due to blockage by surrounding buildings, thus having little impact on beam height. Conversely, when the total height is considerable, there is no blockage from surrounding structures, leading to a greater wind load, which significantly influences the beam height. Regarding beam width, to ensure the beam's strength, stability, and load-bearing capacity, an increase in beam width also generally necessitates an increase in beam height. Hence, the higher the values of beam span, total height of the structure, and beam width, the higher their SHAP values and the predicted beam height. The global impact of the other five factors on beam height is not as distinct.

Local feature interpretation

In summary, the features such as beam span, total height of the structure, and beam width show a clear positive correlation with beam height, while building shape coefficient, storey, seismic fortification intensity, standard floor-to-floor height, and site classification have a relatively weaker feedback mechanism on beam height. Therefore, this section provides a more detailed judgment on the local SHAP impact of single factors. The local impacts of each feature are displayed in Fig. 8, where the horizontal axis represents the magnitude of the feature values and the vertical axis represents the size of the SHAP values for that feature. Based on the single-variable SHAP attribution analysis shown in Fig. 8 and its relationship with beam height, the following conclusions can be drawn:

There is a clear positive correlation between the seismic fortification intensity and the beam height. Higher seismic fortification intensities require structures to withstand greater earthquake forces. As a component of the structure, beams need to have sufficient load-bearing capacity. In the design process, it may be necessary to increase the beam height to increase its cross-sectional area to meet the load-bearing requirements. The SHAP values of storey ratio, standard floor-to-floor height, and building shape coefficient are near zero, indicating that these three factors do not have a significant positive or negative correlation with beam height; the site classification being in Class II and III has little change in the impact on beam height, while in Class IV, where the ground is generally loose sand or clay, prone to uneven settlement of buildings, thus the impact on beam height in Class IV is more pronounced.

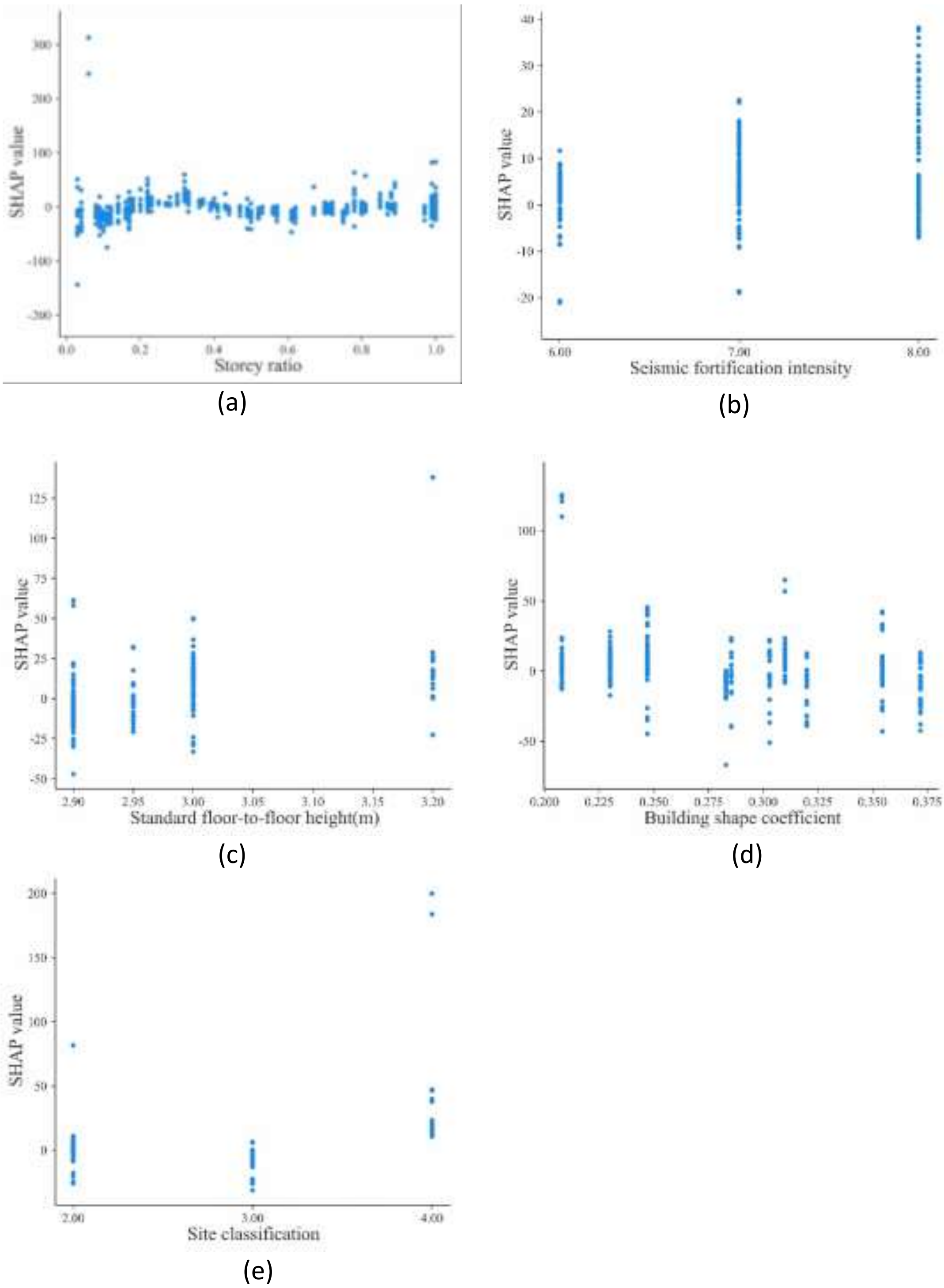


Fig 7. **Single variable SHAP distribution.**

Conclusions

This article proposes a machine learning model for predicting the beam height of shear wall residential buildings. It involves permutation feature importance and model selection across six regression models, detailing the related research ideas and methods. The following conclusions are drawn:

(1) The beam height prediction regression model was confirmed as the GB model through the permutation feature importance methods that shuffle feature order inputs to six models and the feature ranking methods specific to the six regression models. The analysis of feature importance shows that beam span, total height of the structure, and beam width have a greater impact on beam height.

(2) The beam height prediction model based on the ensemble learning algorithm GB with optimal parameters significantly outperforms the model based on the traditional machine learning algorithm BPNN. Compared to the BPNN-based model, the GB-based model has a 148.15% increase in the coefficient of determination, a 32.14% improvement in root mean square error, a 46.11% increase in mean absolute percentage error, and a 53.29% increase in mean absolute error, demonstrating a significant performance improvement.

(3) For the GB model predicting the beam height of shear wall residential buildings, by combining SHAP and mechanical laws, the contributions of various components to beam height were analyzed and discussed. The results show that beam span, beam width, total height of the structure, and seismic fortification intensity are generally positively correlated with beam height; Class IV sites have a greater impact on beam height than Class II and III sites; the ratio of storeys to total storeys, building shape coefficient, and standard floor-to floor height have no significant positive or negative impact on beam height, which is consistent with traditional mechanical laws. The machine learning model proposed in this article conforms to these laws.

Availability of data and material

The data used in this research can be found via the corresponding author.

Competing interests

The authors declare that they have no competing interests.

Funding

No funding.

Acknowledgements

NA

References

[1] sMa RJ, Chen AR, Hu XH (2002) Decision on Height of Stiffened Girder of Suspension Bridges Based on Performance of Aerodynamic Stability. J Tongji Univ, Nat Sci (05): 573-577.
[2] Amezquita-Sancheza JP, Valtierra-Rodriguez M, Adeli H (2020) Machine learning in structural engineering. Sci Iran 27(6): 2645-2656.
[3] He DQ, Wang PJ, Yang J (2021) Application of deep neural networks in EMD false component identification. Eng Mech 38(Suppl): 195-201.
[4] Deng L et al (2021) Optimization on bearing capacity and ductility of cold formed thin-walled steel flexural member. Eng Mech 38(4): 93-101.
[5] Pan ZD et al (2019) Research on sliding mode decentralized control based on adaptive learning rate RBF neural network for large-scale engineering structures. Eng Mech 36(9): 120-127.
[6] Chen SG et al (2023) Machine learning models for cracking torque and pre-cracking stiffness of RC beams. Archiv Civ Mech Eng 23(6).
[7] Cao X et al (2023) Torsional capacity of ultra-high-performance concrete beams using rectangle stirrup. J Build Eng 69: 106231.
[8] Yu Y et al (2020) Machine learning-based evaluation of shear capacity of recycled aggregate concrete beams. Materials 13(20): 4552.
[9] Olalusi OB, Awoyera PO (2021) Shear capacity prediction of slender reinforced concrete structures with steel fibers using machine learning. Eng Struct 227: 111470.
[10] Wakjira TG et al (2022) Explainable machine learning model and reliability analysis for flexural capacity prediction of RC beams strengthened in flexure with FRCM. Eng Struct 255: 113903.
[11] Gao D et al (2023) Flexural capacity prediction of steel fiber reinforced concrete beam with recycled fine and coarse aggregate. Adv Struct Eng 26(6): 1042-1058.
[12] Demir A, Caglar N, Ozturk H (2019) Parameters affecting diagonal cracking behavior of reinforced concrete deep beams. Eng Struct 184: 217-231.
[13] Nicoletti RS et al (2023) Evaluation of the TRF of reinforced concrete beams through thermal and thermostructural analysis. J Struct Fire Eng 15(1).
[14] Mei Y et al (2022) Probabilistic prediction model of steel to concrete bond failure under high temperature by machine learning. Eng Fail Anal 142: 106786.
[15] Nariman N et al (2021) Optimum design of flexural strength and stiffness for reinforced concrete beams using machine learning. Appl Sci 11(18): 8762.
[16] Lai D, Demartino C, Xiao Y (2023) Interpretable machine-learning models for maximum displacements of RC beams under impact loading predictions. Eng Struct 281: 115723.
[17] Almustafa MK, Nehdi ML (2022) Novel hybrid machine learning approach for predicting structural response of RC beams under blast loading. Structures 39: 1092-1106.
[18] Hamidia M, Kaboodkhani M, Bayesteh H (2024) Vision-oriented machine learning-assisted seismic energy dissipation estimation for damaged RC beam-column connections. Eng Struct 301: 117345.
[19] Mangalathu S et al (2019) Rapid seismic damage evaluation of bridge portfolios using machine learning techniques. Eng Struct 201: 109785.
[20] Abushanab A, Wakjira TG, Alnahhal W (2023) Machine learning-based flexural capacity prediction of corroded RC beams with an efficient and user-friendly tool. Sustainability 15(6): 4824.
[21] Cai B, Xu L-F, Fu F (2019) Shear resistance prediction of post-fire reinforced concrete beams using artificial neural network. Int J Concr Struct Mater 13(1).
[22] Kumar Tipu R et al (2023) Shear capacity prediction for FRCM-strengthened RC beams using Hybrid ReLU-Activated BPNN model. Structures 58: 105432.
[23] Zhao J et al (2021) Improved Levenberg–Marquardt backpropagation neural network by particle swarm and whale optimization algorithms to predict the deflection of RC beams. Eng Comput 38(S5): 3847–3869.

[24] Rajaratnam B et al (2016) Lasso Regression: Estimation and Shrinkage via the Limit of Gibbs Sampling. J R Stat Soc 78(1): 153-157.
[25] Xie MH, Xie LL, Zhu P (2021) An Efficient Feature Weighting Method for Support Vector Regression. Math Probl Eng 2021: 6675218.
[26] Manzali Y et al (2024) A feature weighted K-nearest neighbor algorithm based on association rules. J Ambient Intell Humaniz Comput.
[27] Norouzi H et al (2021) Assessment of groundwater vulnerability using genetic algorithm and random forest methods (case study: Miandoab plain, NW of Iran). Environ Sci Pollut Res 28: 39598–39613.
[28] Gunasekara N et al (2024) Gradient boosted trees for evolving data streams. Mach Learn 113(5): 3325-3352.
[29] Liang GQ et al (2023) Fatigue life prediction of concrete under cyclic compression based on gradient boosting regression tree. Mater Struct 56: 172.
[30] Tong B, Guo JJ, Fang S (2021) Predicting Budgetary Estimate of Highway Construction Projects in China Based on GRA-LASSO. J Manage Eng 37(3).
[31] Breiman L (2001) Random forests. Mach Learn 45: 5–32.
[32] Ba LJ, Caruana R (2014) Do deep nets really need to be deep? in Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2. MIT Press, Cambridge, MA, USA, 2654–2662.
[33] Liu Y (2019) Predicting Construction Cost Using Fuzzy Logic and Neural Networks. Dissertation, Chongqing Jiaotong University.
[34] Lundberg S, Lee SI (2017) A Unified Approach to Interpreting Model Predictions. in Proceedings of the 31st International Conference on Neural Information Processing Systems. Curran Associates Inc, Red Hook, NY, USA, 4768–4777.