Hindawi
Advances in Civil Engineering
Volume 2022, Article ID 2923069, 15 pages
https://doi.org/10.1155/2022/2923069

Research Article

Prediction of Transverse Reinforcement of RC Columns Using Machine Learning Techniques

Congzhen Xiao, Baojuan Qiao, Jianhui Li, Zhiyong Yang, and Jiannan Ding

China Academy of Building Research, Beijing 100013, China

Correspondence should be addressed to Baojuan Qiao; qiaobaojuan@cabrtech.com

Received 16 September 2022; Revised 25 October 2022; Accepted 7 November 2022; Published 22 November 2022

Academic Editor: Huseyin Bilgin

Transverse reinforcement of reinforced concrete (RC) columns contributes greatly to the ductility deformation capacity of RC structures. The existing models to predict the amount of transverse reinforcement required are all empirical models with low accuracy and large dispersion and have not considered the real ductility demand of individual components. This paper proposes a ductility design method of RC structure based on component drift ratio demand obtained from nonlinear structural dynamic analysis. To establish the best transverse reinforcement ratio prediction model for RC columns, based on an experimental database consisting of 498 columns, 12 machine learning (ML) models are trained. To solve the over-fitting problem caused by the current situation of "few samples and big errors" of the experimental database, feature engineering aiming at dimension reduction is systematically carried out through an iterative process. Through comprehensive performance evaluation on the testing set, an XGBoost model is selected. To interpret the "black box" ML model, the SHAP method and partial dependence plots are used to analyse the correlation between the input parameters and the transverse reinforcement ratio. The interpretation results are consistent with mechanical laws and engineering experience, which prove the reliability of the selected ML model. Compared with two existing empirical models, the proposed XGBoost model shows higher accuracy and smaller deviation. After safety probability analysis, the trained XGBoost model is transformed into C code and integrated into seismic design software for productive practice. An open-source data-driven model to predict the transverse reinforcement ratio required for RC columns is provided worldwide, with the flexibility to account for additional experimental results.

1. Introduction

Ductility refers to the ability of a structural member to bear large deformation without obvious reduction of bearing capacity in inelastic stage [1], which contributes greatly to the collapse resistance of reinforced concrete (RC) structures. The ductility of columns, which plays an important role in the ductility deformation capacity of RC structures, is mainly ensured by a sufficient number and arrangement of transverse steel bars in the potential plastic hinge zone. Generally, transverse reinforcement has three main functions [2], namely, (1) preventing the longitudinal bars from buckling, (2) avoiding shear failure, and (3) confining the concrete core to provide sufficient deformability ductility. This study mainly focuses on confinement requirements. Design codes such as the Chinese code (GB 50010-2010), American code (ACI 318-11), European code (Euro Code 8), New Zealand code (NZS 3101), and Canadian code (CSA 2004) have made detailed provisions on transverse reinforcement, such as the minimum transverse reinforcement ratio (or transverse reinforcement characteristic value), maximum transverse reinforcement spacing, and length of potential plastic hinge zone. Some codes further provide empirical formulas for calculating the minimum transverse reinforcement ratio, but few codes consider the real ductility demand of individual component. The amount of code-required transverse reinforcement can be reduced in many cases, while insufficient in other cases.

To quantify the amount of transverse reinforcement required for RC columns, many empirical models [2–10] have been proposed based on the basic principles and mechanics of RC components. For example, based on

a numerical study using cyclic analyses performed on a large set of columns, Watson et al. [3] derived refined design equations to determine the quantities of transverse reinforcement required for specified ductility levels, which were adopted by New Zealand Standard. Considering the limitations of ACI 318M-11 that the transverse requirements do not account for the axial load level and confinement demand, which has been proven to significantly affect the confinement effectiveness and the column behaviour, Sheikh et al. [4] proposed a design procedure to determine the amount of lateral steel required considering the column ductility performance, the level of axial load, and the steel configuration. Generally speaking, an empirical method usually starts with the assumed form of an equation and then carries out regression analysis, in which the assumed variables are linearly related and the unknown coefficients are determined by using experimental data so that the equation will fit the data. However, due to the complicated constitutive material relationships and the coupling of external seismic loads, the real relationship between the input variables and the transverse reinforcement required is highly nonlinear. The chosen equation may not be able to adequately represent complex nonlinear relationships. Besides, empirical formulas are always developed based on a narrow data range, and the diversity of sample results is limited. All these factors lead to poor accuracy and large dispersion of empirical models [2, 7, 8, 10]. For example, the coefficient of variation of the ratio of the calculated amount to the experimental amount of transverse reinforcement is as high as 0.616 [10]. Therefore, a new method to predict the amount of transverse reinforcement needed covering a wide range of parameters; for example, normal strength and high strength of concrete and reinforcement, with high accuracy and low dispersion, should be developed.

Recently, the merits of alternative approaches, e.g., nonparametric modelling in engineering research have been widely recognized [11, 12]. Artificial intelligence (AI) techniques have attracted great scientific interest in fields with sufficient experimental data and complicated phenomena. Machine learning (ML) has been successfully used to classify the failure mode of RC columns [13], RC beam-column joints [14], RC shear walls [15], and RC frames with infills [16], predict the shear strength of RC deep beams [17], squat RC Walls [18], RC beam-column joints [14, 19], precast concrete joints [20], steel fiber-reinforced concrete beams [21], and slender RC structures with steel fibers [22], predict the drift capacity of RC columns [23, 24], forecast the backbone curve and hysteresis loop of RC columns [25–27], predict the compressive strength of concrete [28], predict the compressive and flexural strengths of steel fiber-reinforced concrete [29], estimate the flexural capacity of ultrahigh-performance concrete beams [30], predict the punching shear capacity for fiber-reinforced concrete slabs [31], and predict the lateral strain in transverse reinforcements [32]. Although progress has been made in applying the ML technique to interpret the experimental data and predict the component-level structural properties, the data-driven method for predicting the transverse reinforcement needed for columns has not yet been studied. It is recognized that the ML methods can (1) capture the complex nonlinear relationships between the input and output variables, (2) deal with a large number of input variables without neglecting potentially important variables, and (3) gain insights from big data and take into account the diversity of massive specimens. In view of these advantages, ML methods are adopted to predict the amount of transverse reinforcement required for RC columns in this paper.

The general objectives of this research are as follows: (1) to propose a ductility design method of RC structure based on real drift ratio demand of individual components; (2) to establish ML models to predict the amount of transverse reinforcement required for columns and choose the best one; (3) to interpret the prediction of the ML model and ensure the credibility of the proposed model; (4) to create an open-source data-driven ML model that can be used in seismic ductility design worldwide, with flexibility to account for additional experimental results. The paper begins by presenting a new ductility design method of RC structure based on real component drift ratio demand in Section 2. Then, ML model training, performance evaluation, and model interpretation are presented in Section 3. Comparisons of the proposed ML model with empirical models, safety probability analysis, and ML model deployment are presented in Section 4. The conclusions are given in Section 5.

2. Methods

To predict the transverse reinforcement of RC columns, firstly, nonlinear structural dynamic analysis under specified earthquake ground motion is carried out to get the component deformation in the response history. Secondly, the component drift ratio demand is calculated based on component deformation. There are three following reasons for choosing drift ratio instead of curvature ductility as the input ductility demand: (1) drift ratio includes bending deformation and shear deformation and is suitable for flexure critical, flexure-shear critical, and shear critical columns; (2) drift ratio does not depend on the definition of yield displacement or yield curvature; (3) drift ratio is a routine record of all specimens and can be directly related to drift limits specified in building codes. Finally, component drift ratio demands along with other component features (for example, geometric dimensions, longitudinal reinforcement arrangements, and material properties, which will be elaborated in Section 3.2) are input into the trained ML model to predict the transverse reinforcement ratios required for individual columns. The work flow is shown in Figure 1.

The model establishing steps of the transverse reinforcement ratio prediction model are as follows: (1) we collect sufficient experimental data covering a wide range of parameters; (2) we carry out feature engineering to select the right features for ML models; (3) we split the data set into training set (80%) and testing set (20%) randomly; (4) we select appropriate ML methods and train the models on the training set; (5) we adopt grid search and the 10-fold cross-validation method to optimize the hyper-parameters; (6) we train ML models using the optimal hyper-parameters on training set and evaluate the performance on testing set through four typical quantitative

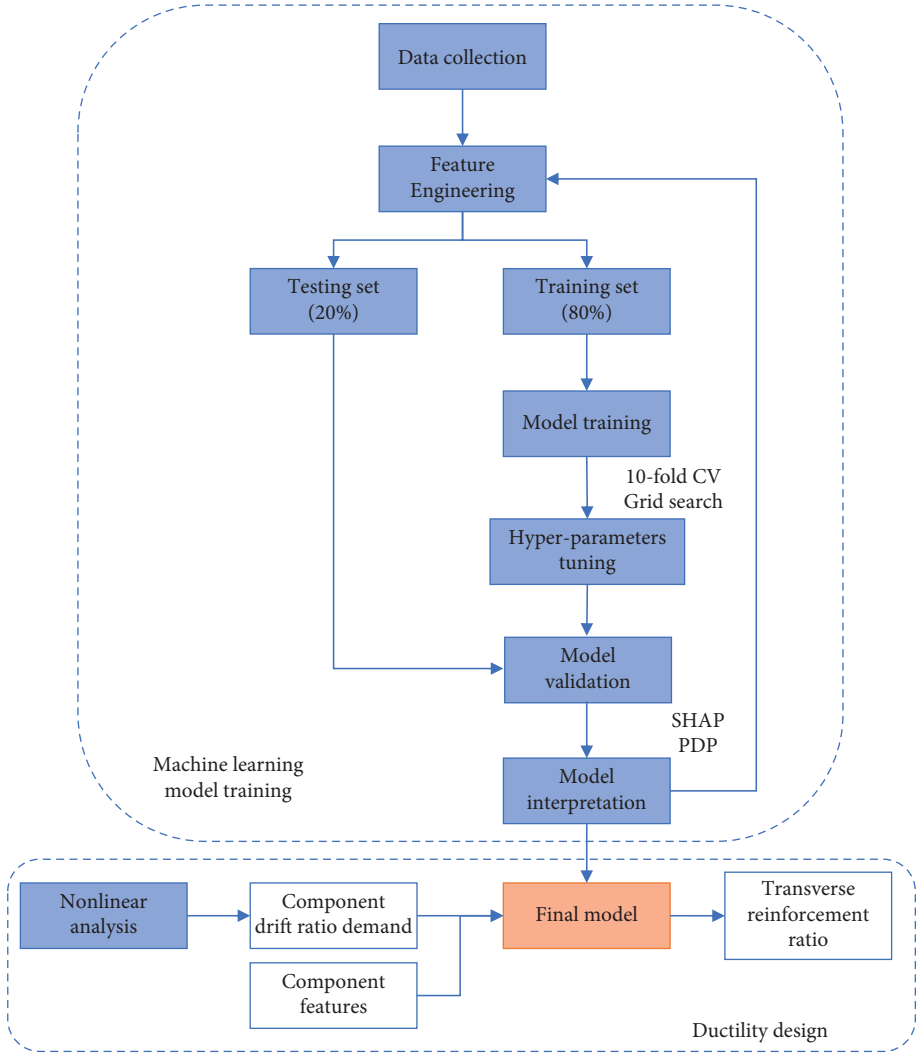

Figure 1: Flow chart of ductility design method based on ML techniques.

metrics; (7) we interpret the established model results through the SHAP method and partial dependence plot (PDP) to verify the reliability of the trained ML model; (8) we choose the best ML model as the final model to predict the transverse reinforcement ratio of RC columns.

In this study, 12 ML models are used to establish the best prediction algorithm of transverse reinforcement ratio as follows: (1) Ordinary Least Squares (OLS), (2) Lasso regression, (3) Ridge regression, (4) K-Nearest Neighbors (KNN), (5) Support Vector Regression (SVR), (6) Multilayer Perceptron (MLP), (7) Decision Trees (DT), (8) Random Forests (RF) [33], (9) AdaBoost [34], (10) XGBoost [35], (11) LightGBM [36], and (12) CatBoost [37]. They can be classified into two categories, namely, single models and ensemble models. Models (1)–(7) belong to single models, and models (8)–(12) belong to ensemble models. Ensemble techniques can be divided into two categories, namely, parallel ensemble techniques (bagging methods) and sequential ensemble techniques (boosting methods). Combining the predictions of several single models, ensemble models increase the accuracy of the results significantly. Linear regression models, including Ordinary Least Squares, Lasso regression, and Ridge regression, are the simplest and most commonly applied form of regression techniques used for the prediction of continuous variables and are used as the basic model for comparison. The ML models are developed using Scikit-learn [38], a machine learning package in the Python programming language.

3. ML Model Training

3.1. Experimental Database. The experimental database consists of 326 rectangular column tests and 172 circular column tests for a total of 498 tests, involving cyclic and monotonic lateral loading, with or without axial load [39–41]. The test configuration and experimental data of RC columns were reduced to the cast of an equivalent cantilever to consistently compare the column behaviour for a wide range of testing configurations [39].

There are many variables in the database which can be classified into five categories, namely, geometric dimensions, reinforcement arrangements, material properties, applied

loads, and extracted displacement data. The ultimate drift ratio $\theta_u = \Delta_u/L \times 100$ was extracted from the force-displacement hysteresis curves for each column [39], where L is the distance between point of maximum moment and point of zero moment, also called shear span, and Δ_u is the lateral displacement during loading at which the column experienced a 20% reduction in maximum applied lateral load V_{Max}. If a 20% drop in shear capacity occurred because of cycling at constant deformation rather than through increasing deformation demands, then the displacement at 0.8 V_{Max} was obtained through interpolation by drawing a line connecting tips of hysteresis loops before and after the horizontal line at 0.8 V_{Max}, as can be seen in Figure 2. If there was no drop in lateral load to 0.8 V_{Max}, then Δ_u was taken as the maximum lateral displacement the column test achieved. A lower bound estimate on the lateral displacement at shear failure was obtained with the procedure.

Based on the background knowledge of civil engineering, all the features related to ductility and transverse reinforcement of RC columns are extracted. Since the transverse reinforcement ratio A_{sh}/sb_c is the target variable to be predicted, features directly related to the transverse reinforcement ratio, such as bar diameter and spacing of transverse bars, cannot be used as input features.

3.2. Feature Engineering. Different from databases in other industries, civil engineering test databases are characterized by limited number of samples and large errors, especially for concrete component test databases, owing to that the test is expensive and the dispersion of material properties is large, which may lead to overfitting of ML models. In view of the difficulty of increasing the number of samples, feature engineering, aiming at dimensionality reduction, is particularly important.

Feature engineering is an essential phase to improve performance by selecting the right features for the model, ensuring that the maximum relationship with the target variable is captured. It is worth noting that feature engineering is an iterative process, which takes a lot of effort. While there is no formula for effective feature engineering, 4 steps are used in this study:

3.2.1. Data Transformation. To handle data with different units avoiding scale effect, data standardization is used to convert the data into a uniform format (zero mean and unit standard deviation), while the tree-based model does not need data standardization. For features with distribution skewed to the right, logarithmic transformation is also tried, while the performance improvement is not obvious.

3.2.2. Feature Extraction. To reduce the number of input features, new features are extracted from the existing attributes by grouping multiple variables into a feature that measures the average of these variables, such as section depth-width ratio h/b, gross area to core area ratio A_g/A_c, shear span to effective depth ratio L/d, longitudinal reinforcement ratio ρ_l, and axial load ratio $P/A_g f'_c$.

3.2.3. Feature Selection. Features with high correlations can lead to collinearity problem, which will reduce the accuracy of the ML model by preventing it from learning the interactions between independent features. Therefore, feature selection is conducted to achieve dimensionality reduction based on a correlation matrix composed of the Pearson correlation coefficient of each pair of features in the database. Among the features with strong correlation, the feature with lower correlation with the target variable will be eliminated. For example, the clear cover thickness and the gross area to core area ratio A_g/A_c are a pair of highly correlated features. Having a lower correlation with the target variable transverse reinforcement ratio A_{sh}/sb_c, the clear cover thickness is removed.

Selected features are the section shape (S for short, 0 for rectangle and 1 for circle), section depth h, longitudinal bar diameter d_l, yield strength of longitudinal bars f_{yl}, yield strength of transverse bars f_{yt}, and concrete compressive strength at 28 days f'_c. The descriptions and statistical attributes of the input and output variables, such as the mean, standard deviation, minimum, 25%, 50%, 75%, and maximum value, are given in Table 1, and their statistical distributions are displayed in Figure 3. As can be seen, the database covers a wide range of RC column parameters, including normal-strength and high-strength concrete and reinforcement, which will increase the adaptability of the trained ML model.

3.2.4. Feature Iteration. Feature iteration, also known as the wrapper method of feature selection, is an iterative process involving four steps as follows: (1) we select a subset of features; (2) we train the ML model with the selected features (the training process is introduced later); (3) we measure the model performance; (4) we make a decision to retain or remove the selected features.

Permutation feature importance [33] is used to rank the features and identify the most important features. The permutation feature importance is defined to be the decrease in a model score when a single feature value is randomly shuffled, which breaks the relationship between the feature and the target. Thus, the drop in the model score indicates how much the model depends on the feature. Table 2 lists the permutation feature importance of the 12 input features for the 12 ML models, in descending order. Although the numerical algorithms of different ML models are different, the ranking of feature importance is similar, which also reciprocally proves the reliability of 12 ML models. Ranking sixth, the section shape is not very important, which proves the rationality of training rectangular column and circular column samples together. According to the average permutation feature importance of 12 ML model, four features of least importance, such as the section depth-width ratio h/b, section depth h, longitudinal bar diameter d_l, and yield strength of longitudinal bars f_{yl}, are eliminated by trial and error through an iterative process, and the best performance on testing set is obtained.

3.3. Model Training. The database is randomly split into a training set (80%, 398 samples) and a testing set (20%, 100 samples). The training set is used to establish the prediction

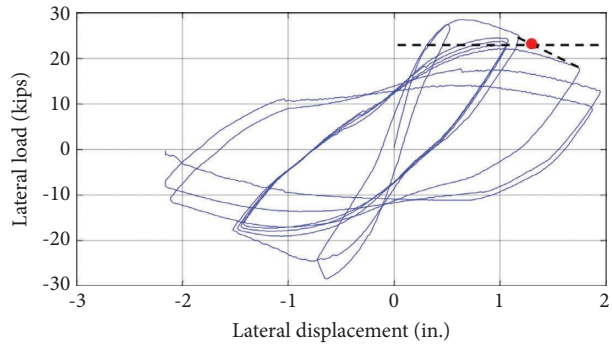

Figure 2: Estimation of drift ratio θ_u at 0.8 V_{Max}.

Table 1: Statistics summary of input and output variables.

Variable	Feature comments	Mean	St.D.	Min.	25%	50%	75%	Max.
h (m)	Section depth	0.35	0.15	0.08	0.25	0.31	0.41	1.52
h/b	Section depth-width ratio	1.07	0.27	0.50	1.00	1.00	1.00	2.01
A_g/A_c	Gross area to core area ratio	1.42	0.30	1.07	1.19	1.35	1.52	2.71
L/d	Shear span to effective depth ratio	3.93	1.94	1.15	2.50	3.49	5.00	12.50
d_l (mm)	Longitudinal bar diameter	16.68	5.05	6.00	12.70	16.00	19.40	43.00
f_{yl} (MPa)	Yield strength of longitudinal bars	424	62	240	374	436	455	587
ρ_l (%)	Longitudinal reinforcement ratio	2.42	0.99	0.46	1.82	2.22	2.68	6.94
f_{yt} (MPa)	Yield strength of transverse bars	451	186	200	352	414	476	1424
f'_c (MPa)	Concrete compressive strength at 28 days	43	24	13	28	34	41	118
$P/A_g f'_c$	Axial load ratio	0.20	0.17	0.00	0.09	0.16	0.30	0.90
θ_u (%)	Drift ratio at 0.8 V_{Max}	3.82	2.42	0.43	1.98	3.22	5.27	15.63
A_{sh}/sb_c (%)	Transverse reinforcement ratio	0.71	0.57	0.04	0.27	0.57	0.97	3.47

b is the section width; b_c is the cross-sectional core width measured with outside edges of transverse reinforcement; A_g is the gross area of column; A_c is the section area measured out-to-out with transverse reinforcement; d is the effective depth in primary direction (dimension from compression face to centroid of outermost layer of tension steel); P is the axial compressive force on column; A_{sh} is the total cross-sectional area of transverse reinforcement (including crossties) within spacing; and s is the center-to-center spacing of spirals or circular hoops.

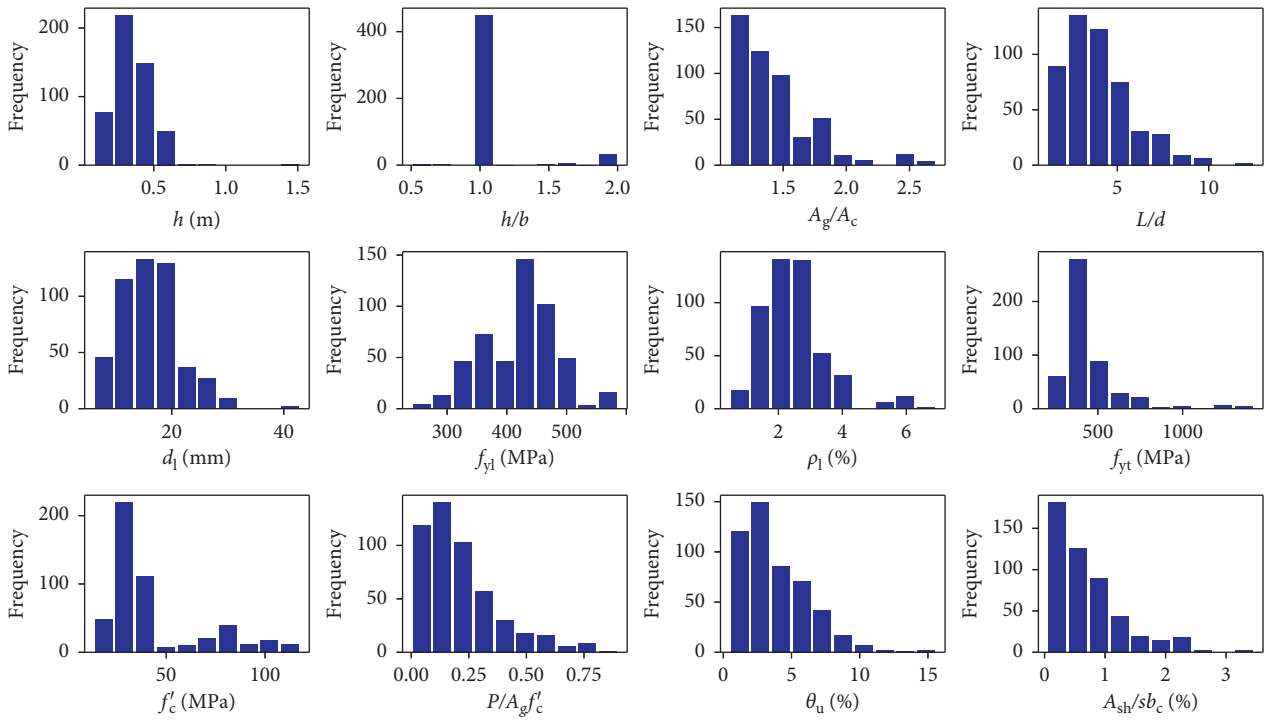

Figure 3: Distribution of input and output variables.

Table 2: Permutation feature importance of 12 features.

Model	$\mathbf{f_c'}$	θ_u	$\mathbf{P/A_g f_c'}$	$\mathbf{A_g/A_c}$	$\mathbf{f_{yt}}$	**S**	**L/d**	ρ_l	$\mathbf{f_{yl}}$	$\mathbf{d_l}$	**h**	**h/b**
OLS	0.64	0.21	0.24	0.01	0.11	0.05	0.02	<0.01	<0.01	0.01	0.04	0.03
Lasso	0.64	0.21	0.24	0.01	0.11	0.05	0.02	<0.01	<0.01	0.01	0.04	0.03
Ridge	0.59	0.18	0.23	0.01	0.09	0.05	0.01	<0.01	<0.01	<0.01	0.02	0.03
KNN	0.43	0.36	0.28	0.16	0.12	0.19	0.18	0.19	0.12	0.11	0.11	0.20
SVR	0.93	1.19	0.63	0.84	0.25	0.39	0.41	0.32	0.18	0.32	0.14	0.18
MLP	0.66	0.47	0.40	0.19	0.16	0.23	0.11	0.13	0.08	0.09	0.10	0.14
DT	1.23	0.40	0.37	0.62	0.25	<0.01	0.09	0.23	0.11	0.09	0.05	<0.01
RF	0.82	0.22	0.19	0.21	0.06	0.01	0.05	0.08	0.03	0.02	0.02	0.01
AdaBoost	0.47	0.08	0.12	0.05	0.04	0.02	0.02	0.01	0.01	0.01	0.02	0.01
XGBoost	0.73	0.34	0.29	0.32	0.14	0.10	0.05	0.05	0.04	0.03	0.02	<0.01
LightGBM	0.38	0.38	0.23	0.12	0.13	0.04	0.07	0.04	0.02	0.04	0.04	<0.01
CatBoost	0.45	0.36	0.20	0.13	0.11	0.07	0.06	0.05	0.04	0.03	0.03	0.01
Mean	0.66	0.37	0.29	0.22	0.13	0.10	0.09	0.09	0.06	0.06	0.05	0.05

ML model, and the testing set is used to evaluate the accuracy of the training model.

After feature engineering and dimensionality reduction, eight features, namely, concrete compressive strength f_c', drift ratio demand θ_u, axial load ratio $P/A_g f_c'$, gross area to core area ratio A_g/A_c, yield strength of transverse bars f_{yt}, section shape S, shear span to effective depth ratio L/d, and longitudinal reinforcement ratio ρ_l are selected as input parameters. The output parameter is the transverse reinforcement ratio.

To establish the best transverse reinforcement ratio prediction model, 12 ML models, including Ordinary Least Squares, Lasso regression, Ridge regression, K-Nearest Neighbors, Support Vector Regression, Multilayer Perceptron, Decision Trees, Random Forests, AdaBoost, XGBoost, LightGBM, and CatBoost are trained on the training set.

For the best output of each model, grid search is applied for tuning hyper-parameters. ML Models of every combination of various hyper-parameters are built and evaluated, and the model with the highest accuracy wins. To alleviate the inherent randomness in selecting training and testing samples, a 10-fold cross-validation process is employed. The training set is randomly divided into 10 folds, each fold is used as a testing set in turn and the remaining 9 folds are used as the training set. Several rounds of 10-fold cross-validation are performed and the results from all the rounds are averaged to estimate the accuracy of the ML model. We can see the Jupyter Notebook python code on GitHub for the grid search hyper-parameters ranges and the final hyper-parameters of each ML model. After effective feature engineering, most ML models even using default hyper-parameters can achieve good performance.

3.4. Training and Testing Results. The model performance on training set and testing set is evaluated through four typical quantitative metrics, namely, coefficient of determination ($R^2 = 1 - \sum_{i=1}^{m} (P_i - T_i)^2 / \sum_{i=1}^{m} (P_i - \overline{T})^2$), root mean square error ($RMSE = \sqrt{\sum_{i=1}^{m} (P_i - T_i)^2 / m}$), mean absolute error ($MAE = \sum_{i=1}^{m} |P_i - T_i| / m$), and weighted average percentage error ($WAPE = \sum_{i=1}^{m} |P_i - T_i| / \sum_{i=1}^{m} |T_i|$), where T_i is the actual value of the transverse reinforcement ratio and P_i is the predicted value, i is sample index, m is the number of samples, and $\overline{T}$ is the mean value of all the samples in the database. The reason for using weighted average percentage error (WAPE) instead of mean absolute percentage error (MAPE) is that the MAPE exaggerates the importance of the percentage error at low transverse reinforcement ratio, which is not important in engineering practice. Generally, when the predicted transverse reinforcement ratio is low, the confinement transverse reinforcement does not play a controlling role in the final amount of transverse reinforcement.

The performances of these 12 models are evaluated based on the testing set by comparing the predicted results with the experimental data, as shown in Table 3 and Figure 4. The diagonal line ($y = x$) represents that the prediction is identical to the experimental data. In general, ensemble models show higher performance than single models. Statistically, a model with high value of R^2 and corresponding low values of error measures is considered to have a high performance. Among the 12 ML models, the XGBoost model shows the best performance ($R^2 = 0.873$, $RMSE = 0.239$, $MAE = 0.161$ and $WAPE = 0.212$) on testing set and is chosen as the final prediction model of transverse reinforcement ratio.

3.5. Model Interpretation. The established ML model may have good prediction performance; however, it is still a "black box" model which cannot give an explicit explanation of the underlying physical or mechanical mechanism. An ML model whose explanation violates the mechanical law cannot be used in production practice even if it has good performance. To obtain a better understanding of the predictions and verify the reliability of the proposed XGBoost model, the SHAP method [42] and partial dependence plot (PDP) [43] are used to interpret the results.

3.5.1. SHAP Method. The SHAP (SHapley Additive exPlanations) method [42] originates from game theory and it is an additive feature attribution method, that is, the output of the model is a linear addition of input variables. The contribution of each feature is represented by the so-called Shapley value. SHAP not only offers an understanding of which features are important but also of how each feature affects the prediction, whether at the level of the whole database or at the level of single samples.

Table 3: Performance measure of the developed models.

Model	Training set				Testing set			
	R^2	*RMSE*	*MAE*	*WAPE*	R^2	*RMSE*	*MAE*	*WAPE*
OLS	0.534	0.365	0.268	0.384	0.603	0.423	0.297	0.391
Lasso	0.533	0.366	0.267	0.384	0.595	0.427	0.300	0.395
Ridge	0.534	0.365	0.268	0.384	0.603	0.423	0.297	0.391
KNN	1.000	<0.001	<0.001	<0.001	0.775	0.318	0.194	0.256
SVR	0.941	0.129	0.075	0.108	0.733	0.347	0.233	0.307
MLP	0.738	0.274	0.191	0.274	0.719	0.356	0.233	0.308
DT	1.000	5.736	7.671	1.100	0.772	0.32	0.182	0.241
RF	0.951	0.118	0.079	0.114	0.838	0.27	0.185	0.244
AdaBoost	0.638	0.322	0.280	0.402	0.717	0.356	0.299	0.395
XGBoost	0.999	0.006	0.004	0.006	0.873	0.239	0.161	0.212
LightGBM	0.977	0.08	0.053	0.076	0.817	0.286	0.192	0.254
CatBoost	0.978	0.078	0.057	0.082	0.842	0.266	0.179	0.237

Figure 5 is a SHAP summary plot of the features, which demonstrates the distribution of the SHAP values for each feature and indicates the corresponding influence trends. The horizontal axis represents the specific SHAP value and the vertical axis represents the input features, ordered by importance. The dots are the samples in the database. The colour of the dot indicates the value of the specific feature, and the colour from blue to red indicates a value from small to large. The horizontal position of the dot indicates whether the feature value leads to a higher or lower prediction. For example, the upper right dot in red indicates that a high concrete compressive strength f'_c leads to a prediction increase. It is observed that the quantity of transverse reinforcement required increases with increasing concrete compressive strength f'_c, increasing drift ratio demand θ_u, increasing axial load ratio $P/A_g f'_c$, increasing gross area to core area ratio A_g/A_c, decreasing yield strength of transverse bars f_{yt}, and decreasing shear span to effective depth ratio L/d. In addition, more transverse reinforcement is required for rectangular columns than for circular columns. While the influence trend of longitudinal reinforcement ratio ρ_l is not obvious. The law obtained by the SHAP method is consistent with existing mechanical models and experimental results of RC columns, so the proposed XGBoost model is convincible.

In addition to the global interpretations of the entire data set, SHAP also provides individual (local) interpretations of single samples. Figure 6 illustrates explanations for a typical circular sample. The base value is the average of the predictions of the whole training set, which is 0.619%. The features determine the deviation of the prediction from the base value. The red bars pointing to the right represent the contribution to increasing the transverse reinforcement ratio from the base value, while the blue bars pointing to the left represent oppositely. For this circular sample, drift ratio demand θ_u is the most critical feature and has a positive effect on transverse reinforcement ratio, whose SHAP value is 0.23%.

3.5.2. Partial Dependence Plot. To visualize the relationships between transverse reinforcement ratio and the input parameters and to provide design suggestions for practical engineering, partial dependence plot (PDP) [43] is adopted in this study. Partial dependence of a feature corresponds to the average response of an estimator for each possible value of the feature. PDP shows the marginal effect of one or two features on the predicted outcome of an ML model and whether the relationship between the target and a feature is linear, monotonic, or more complex. A flat PDP indicates that the feature is not important, and the more the PDP varies, the more important the feature is.

One-way PDPs of transverse reinforcement ratio A_{sh}/sb_c on concrete compressive strength f'_c, drift ratio demand θ_u, axial load ratio $P/A_g f'_c$, gross area to core area ratio A_g/A_c, yield strength of transverse bars f_{yt}, section shape S, shear span to effective depth ratio L/d, and longitudinal reinforcement ratio ρ_l, ordered by importance obtained from permutation feature importance, are visualized in Figure 7. The thinner lines represent individual specimens (only 50 specimens are randomly selected for clear and typical display), while the thicker line represents the average value of all the 498 samples in the database. Marks on the horizontal axis indicate the data distribution. The largest influences can be seen in concrete compressive strength, and there is an obvious step around 70 MPa. The higher the concrete compressive strength is, the more transverse reinforcement is needed. The second and third important features are drift ratio demand and axial load ratio. With the increase of drift ratio and axial load ratio, the transverse reinforcement ratio required also increases. With the increase of gross area to core area ratio A_g/A_c, the transverse reinforcement ratio mainly increases, ignoring the decreases caused by some large-sized specimens when the gross area to core area ratio A_g/A_c is small. With the increase of yield strength of transverse bars f_{yt} and shear span to effective depth ratio L/d, the transverse reinforcement ratio decreases. Rectangular columns require more transverse reinforcement than circular columns. The influence of longitudinal reinforcement ratio ρ_l on the transverse reinforcement ratio is small. The law obtained from the partial dependence plots is consistent with the SHAP method, which further proves the reliability of the proposed XGBoost model.

Two-way PDP of transverse reinforcement ratio A_{sh}/sb_c on drift ratio demand θ_u and axial load ratio $P/A_g f'_c$ is visualized in Figure 8(a). The maximum and minimum

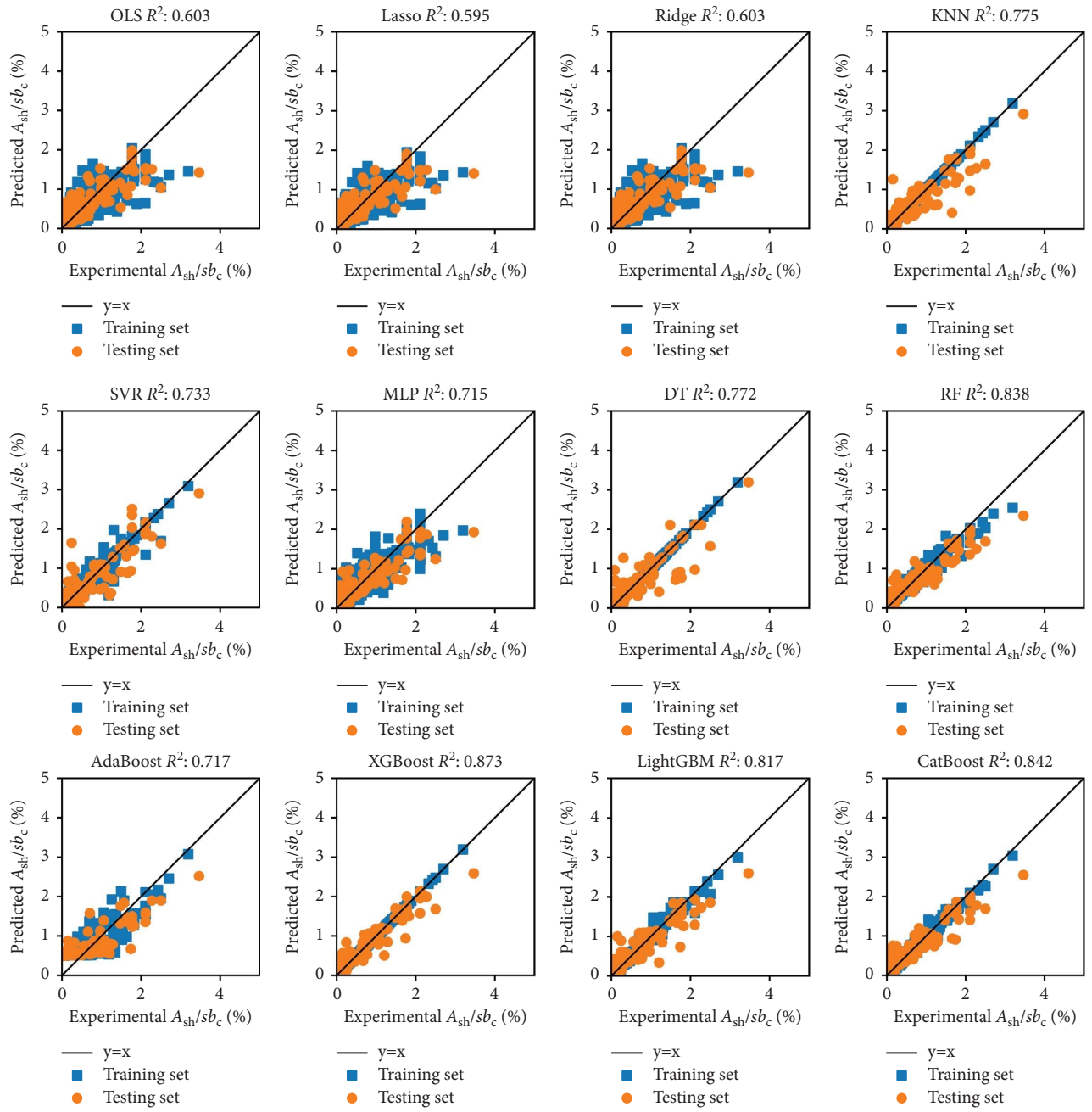

Figure 4: Comparison of predicted and experimental transverse reinforcement ratio.

average transverse reinforcement ratios are 1.29% and 0.35%, respectively. Since there is an obvious step near 70 MPa in the PDP of transverse reinforcement ratio A_{sh}/sb_c on concrete compressive strength f'_c, the samples of normal-strength concrete (less than 60 MPa, the strength definition is not important here, just for illustration) are studied separately. For normal-strength concrete, two-way PDP of transverse reinforcement ratio A_{sh}/sb_c on drift ratio demand θ_u and axial load ratio $P/A_g f'_c$ is visualized in Figure 8(b). The maximum and minimum average transverse reinforcement ratios are 1.23% and 0.25%, respectively.

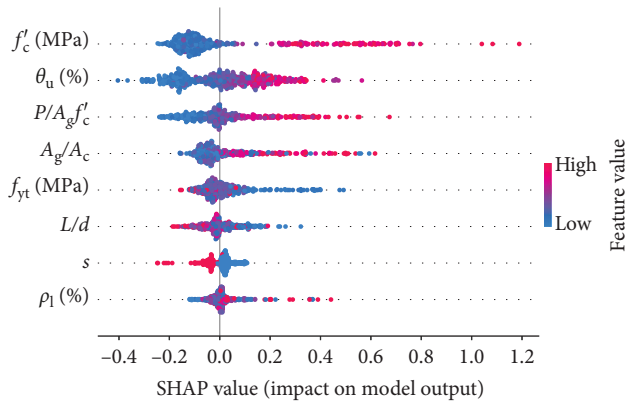

Figure 5: Global interpretation of the XGBoost model.

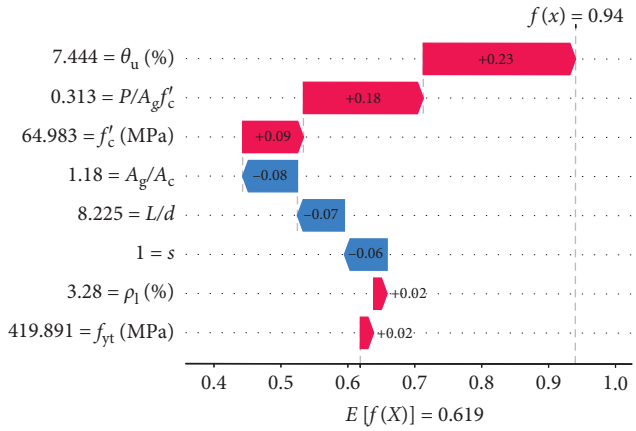

Figure 6: Individual interpretation for a typical circular sample.

4. Discussion

4.1. Comparisons with Empirical Models. To validate the superiority of the proposed XGBoost model, two traditional empirical models proposed by Watson et al. [3] and Sheikh et al. [4] are employed to predict the transverse reinforcement ratio needed for RC rectangular columns.

The Watson et al. [3] model is given as follows:

$$\frac{A_{sh}}{sb_c} = \frac{A_g}{A_c}\frac{\left(\mu_\phi - 33\rho_l m + 22\right)}{111}\frac{f'_c}{f_{yt}}\frac{P}{\phi f'_c A_g} - 0.006. \quad (1)$$

The Sheikh et al. [4] model is given as follows:

$$\frac{A_{sh,ACI}}{sb_c} = 0.3\left(\frac{A_g}{A_c} - 1\right)\frac{f'_c}{f_{yt}} \geq 0.09\frac{f'_c}{f_{yt}},$$
$$\frac{A_{sh}}{sb_c} = \alpha\left\{1 + 13\left(\frac{P}{P_0}\right)^5\right\}\frac{\left(\mu_\phi\right)^{1.15}}{29}\frac{A_{sh,ACI}}{sb_c}. \quad (2)$$

Here, P is the Axial compressive load; P_0 is the nominal axial load strength at zero eccentricity ($P_0 = 0.85f'_c A_g\ (1 - \rho_l) + A_g\rho_l f_{yl}$); μ_ϕ is the curvature ductility factor, ϕ is the strength reduction factor; $m = f_{yl}/0.85f'_c$; α is a parameter that accounts for the confinement efficiency including configuration and the lateral restraint provided to the longitudinal bars; other variables are defined previously.

Figure 9 illustrates the comparison between the transverse reinforcement ratio of the rectangular columns obtained from experiments and those from the prediction formulas. Ideally, all points are distributed on the diagonal line. A point distributed below the diagonal line means the formulation under-predicts, whereas above the diagonal line indicates the formulation over-predicts. All the ML models trained in the previous section show higher accuracy than the empirical formulas, especially the XGBoost model. Figure 9(c) illustrates the results of rectangular columns in the database predicted by the proposed XGBoost model (just for visual contrast, noting that the performance of the ML model should be evaluated on the testing set, not the whole database).

The mean absolute error (MAE) of Watson et al. [3] and Sheikh et al. [4] empirical formulas are 0.727 and 0.652, respectively, while that of the proposed XGBoost model on testing set is 0.161. The standard deviation of the error between experimental and predicted transverse reinforcement ratio of Watson et al. [3] and Sheikh et al. [4] empirical formulas are 0.990 and 1.291, respectively, while that of the proposed XGBoost model on testing set is 0.239. Both empirical formulas show less accuracy and larger uncertainties than the proposed XGBoost model. The reasons can be attributed to the fact that (1) the empirical equations were developed based on a narrow range of data having a limited diversification in specimen results; (2) the empirical equations might neglect important parameters that contribute to the transverse reinforcement ratio; (3) the ability of ML methods to capture complex and nonlinear relationships between the output and inputs is stronger than the traditional linear regression formula.

4.2. Safety Analysis. When applied in engineering practice, considering safety and structural stability, the transverse reinforcement ratio prediction model should be conservative enough for the safe design of RC structures. For primary members critical to structural stability, it is suggested that the probability that the predicted value is lower than the experimental value is less than 20%, referring to the confidence of acceptance criteria for Life Safety (LS) of primary members suggested by Ghannoum et al. [44].

The histogram and empirical cumulative distribution of the error between experimental and predicted transverse reinforcement ratio (the error is defined as experimental value minus predicted value) of the XGBoost model on testing set are plotted in Figure 10. The vertical line at error = 0 indicates that the predicted value is equal to the experimental value. Errors less than 0 indicate safety; errors bigger than 0 indicate insecurity. Most of the transverse reinforcement ratios predicted by the XGBoost model are very close to the experimental values, with a safety probability of 55%. To provide a conservative estimate of the transverse reinforcement ratio, it is suggested that an

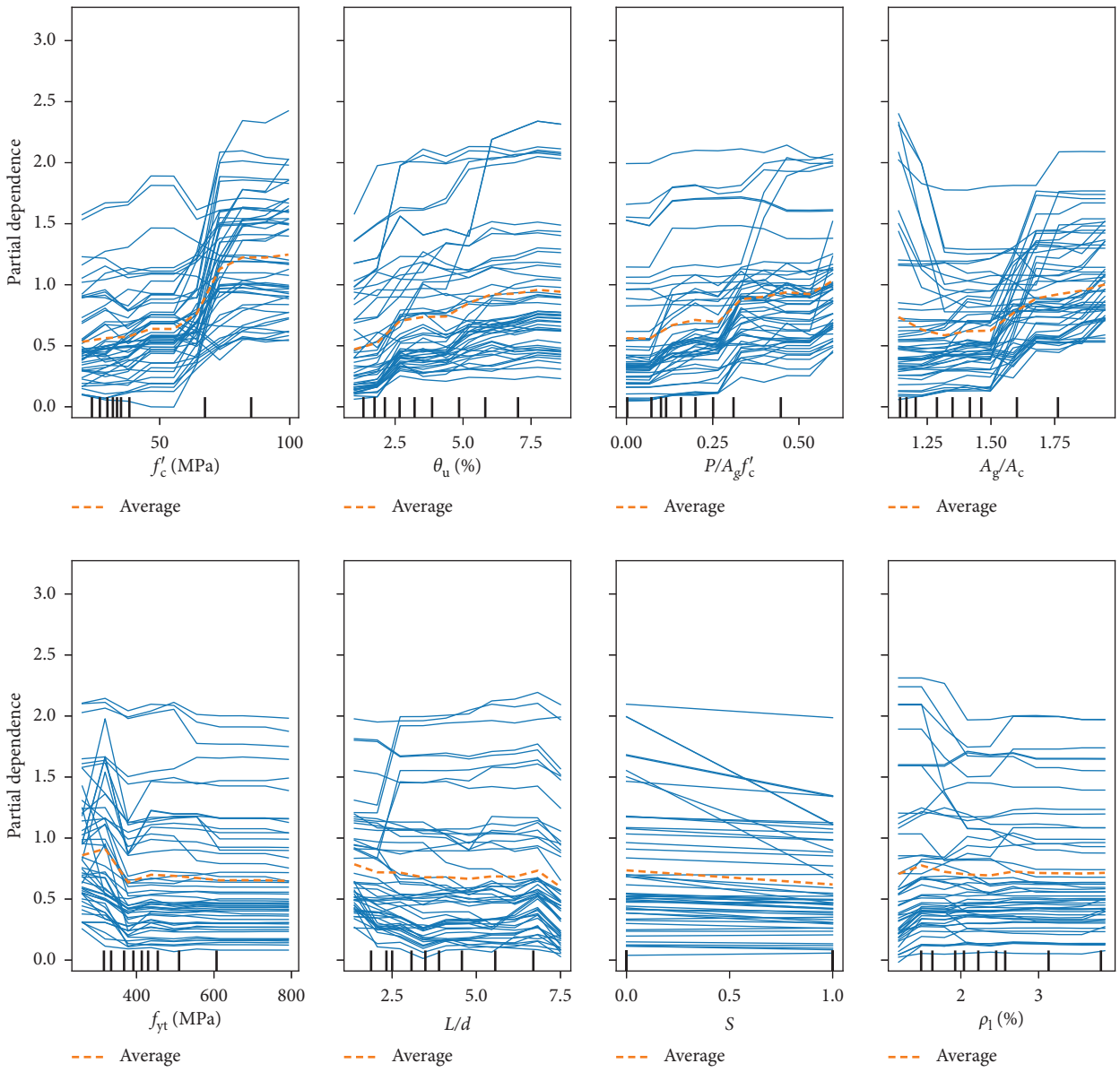

Figure 7: One-way PDPs of transverse reinforcement ratio.

additional value of 0.12% should be added to the transverse reinforcement ratio predicted by the proposed XGBoost model with a guarantee rate of 80%. In other words, the probability that the amount of transverse reinforcement predicted by the proposed XGBoost model is lower than that required for RC columns is 20%.

4.3. ML Model Deployment. After training an ML model, it is desirable to have a way to persist the model for future use without having to retrain. The pickle module in the Scikit-learn package implements a fundamental but powerful algorithm for serializing and de-serializing a Python object structure. Although this method is simple and convenient, it has poor compatibility. Pickle format can only be used in python. If the production environment needs other languages, it cannot be called directly. In order to integrate the trained ML model into a seismic design software incompatible with the ML model-building language, m2cgen (model 2 code generator) is used to transform the trained XGBoost model into C code. The m2cgen is a lightweight library which provides an easy way to transform trained statistical models into native code (Python, C, Java, et al.). The trained XGBoost model can be easily integrated into seismic design software for reliable prediction of the transverse reinforcement ratio of RC columns through source code or API (Application Programming Interface). For example, the XGBoost model is integrated into SAUSG software developed by China Academy of Building Research using C code, to predict the transverse reinforcement ratio of RC columns, as shown in Figure 11.

Through nonlinear structural dynamic analysis of the overall structural system under the specified earthquake ground motion, the maximum component deformation in

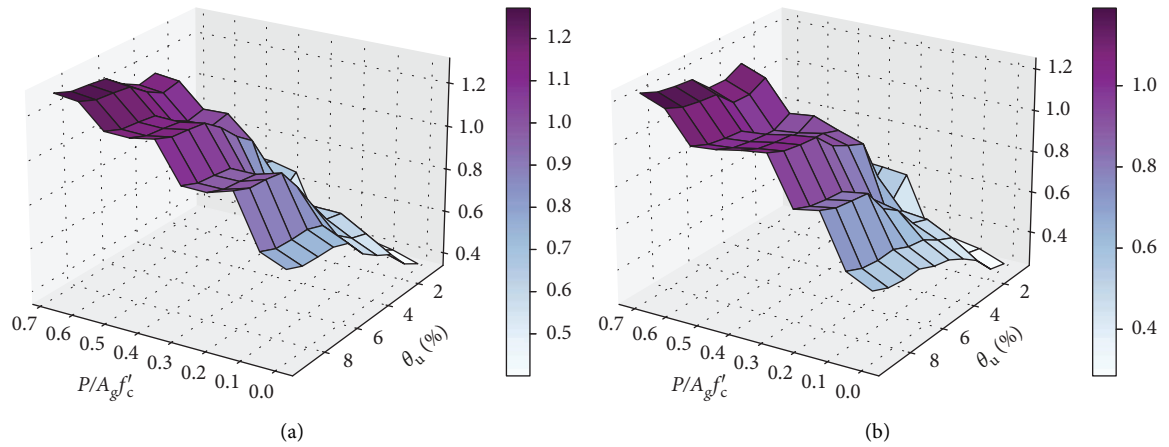

FIGURE 8: Two-way PDPs of transverse reinforcement ratio. (a) Normal-strength and high-strength concrete. (b) Normal-strength concrete.

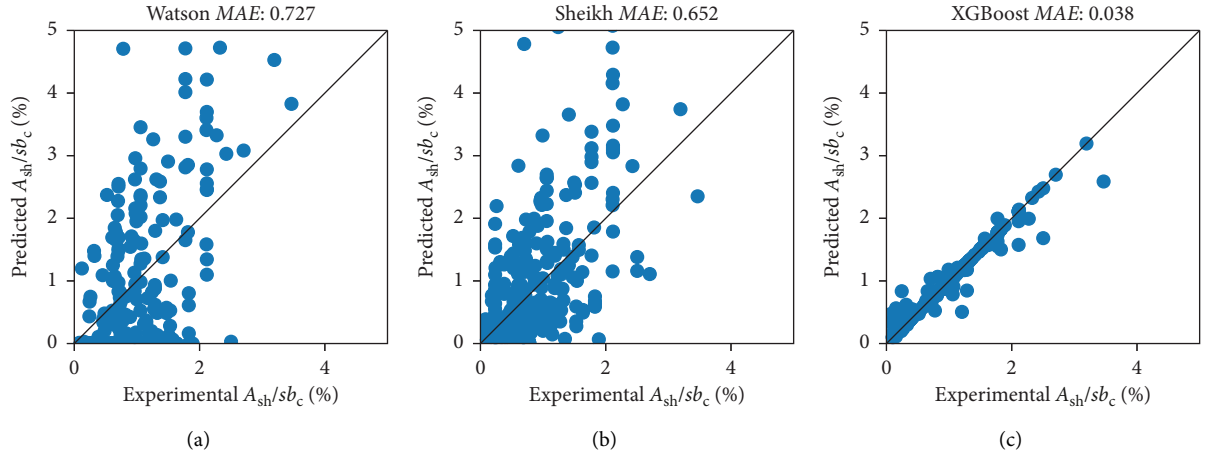

FIGURE 9: Comparison of predicted and experimental transverse reinforcement ratio. (a) Empirical model proposed by Watson et al. [3]. (b) Empirical model proposed by Sheikh et al. [4]. (c) Our XGBoost model.

the response history can be obtained. The concept of obtaining component drift ratio demand from nonlinear structural dynamic analysis is suitable for all types of structural components. This paper mainly focuses on RC columns. Because the component drift ratio demand is defined according to the experimental configuration of cantilever, deformation transformation is required. In the nonlinear structural dynamic analysis, there is usually a zero moment point near the middle point of the column. The segments below and above the zero moment point can be equivalent to two cantilever columns, and the drift ratio demands can be calculated, respectively, as shown in Figure 12.

If the moments of the top and bottom ends of the column are of different signs, firstly, we find the location of the zero moment point and then calculate the displacement of the zero moment point according to the shape function of the finite element model. For each equivalent cantilever, the displacement of the zero moment point and column end are transformed into the local coordinate system defined based on the tangent of the column end, that is, the implicated angle of the column end, which is considered as the harmless drift ratio, is automatically deducted, as shown in Figure 12(b).The drift ratios at the top and bottom ends are calculated by Equation (3).

$$\theta_t = \frac{\Delta_t}{H_t} = \frac{d_0 - d_t}{H_t},$$
$$\theta_b = \frac{\Delta_b}{H_b} = \frac{d_0 - d_b}{H_b}, \tag{3}$$

where θ_t is the drift ratio at the top end; θ_b is the drift ratio at the bottom end; d_0 is the displacement at zero moment point; d_t is the displacement of the top node of the column; d_b is the displacement of the bottom node of the column; H_t is the length of the upper equivalent cantilever; H_b is the length of the lower equivalent cantilever.

If the moments of the top and bottom ends of the column are of the same signs, the zero moment point is outside the column, and its position does not need to be

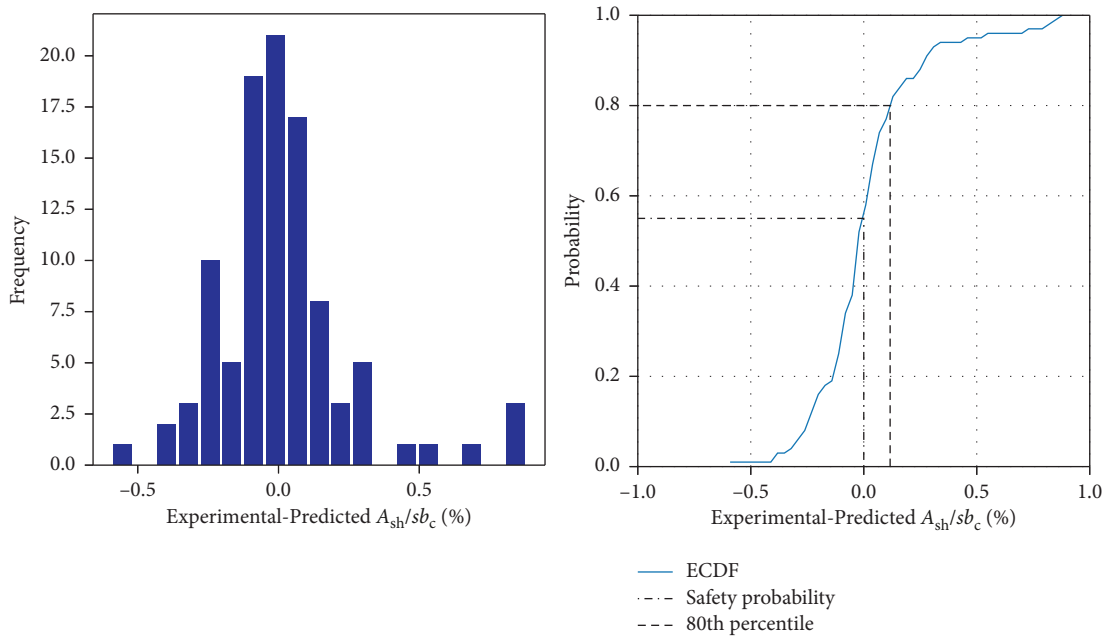

Figure 10: Distribution of the error between experimental and predicted transverse reinforcement ratio on testing set. (a) Histogram. (b) Empirical cumulative distribution function.

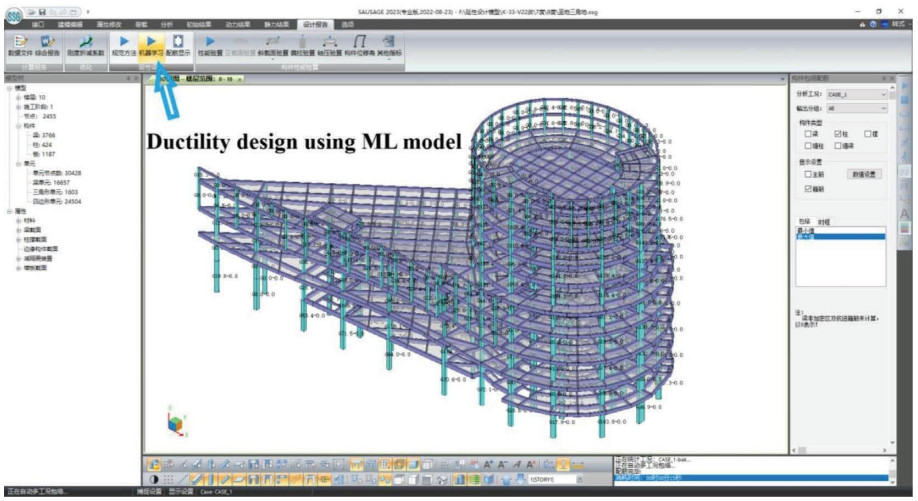

Figure 11: Ductility design using ML model in SAUSG software. Schematic diagram displayed for transverse reinforcement area.

calculated. The drift ratio of the column can be approximately calculated by Equation (4).

$$\theta_t = \theta_b = \frac{d_t - d_b}{H}, \tag{4}$$

where H is the length of the column.

The column drift ratios at all time steps can be calculated according to the abovementioned formulas, and the envelope of the absolute value of the drift ratios at all time steps can be obtained as the drift ratio demand.

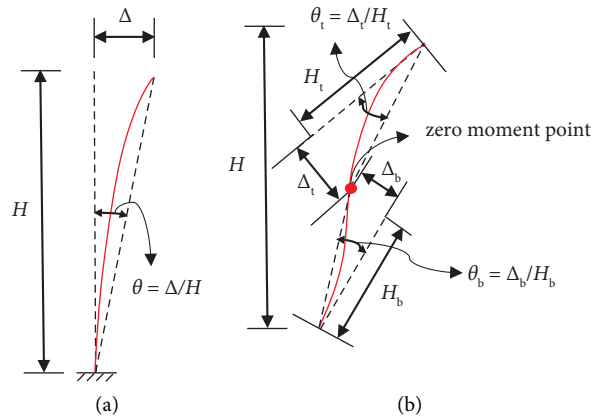

Figure 12: Schematic diagram of component drift ratio. (a) Cantilever. (b) Structural component.

Finally, the drift ratio demands θ_u, along with concrete compressive strength f'_c, axial load ratio $P/A_g f'_c$, gross area to core area ratio A_g/A_c, yield strength of transverse bars f_{yt}, section shape S, shear span to effective depth ratio L/d, and longitudinal reinforcement ratio ρ_l are input into the trained ML model to predict the transverse reinforcement ratios required at the top or bottom ends of the RC columns. Noting that the sectional dimensions and drift ratio demands in the sectional height and width direction may be quite different, the transverse reinforcement ratios in height and width direction should be calculated separately.

5. Conclusion

The existing models to predict the amount of transverse reinforcement required for RC columns are all empirical models with low accuracy and large dispersion and have not considered the real ductility demand of individual components. This paper proposes a ductility design method of RC structure based on the real component drift ratio demand and develops a method for determining the component drift ratio demand. To establish a transverse reinforcement ratio prediction model, a database consisting of 326 rectangular columns and 172 circular columns for a total of 498 tests is used. The database is randomly split into a training set (80%) and a testing set (20%). Twelve ML models including Ordinary Least Squares, Lasso regression, Ridge regression, K-Nearest Neighbors, Support Vector Regression, Multilayer Perceptron, Decision Trees, Random Forests, AdaBoost, XGBoost, LightGBM, and CatBoost are trained. Feature engineering, including data transformation, feature extraction, feature selection, and feature iteration, is systematically carried out through an iterative process. For the best output, grid search and the 10-fold cross-validation method are applied to tune the hyper-parameters. Through a comprehensive performance validation on the testing set, an XGBoost model is suggested for the best accuracy. The SHAP method and partial dependence plot are used to interpret the "black box" ML model. The following conclusions are drawn:

(1) The amount of transverse reinforcement required for columns increases with increasing concrete compressive strength, increasing drift ratio demand, increasing axial load ratio, increasing gross area to core area ratio, decreasing yield strength of transverse bars, and decreasing shear span to effective depth ratio, ordered by importance. In addition, more transverse reinforcement is required for rectangular columns than for circular columns. While the influence trend of longitudinal reinforcement ratio is not obvious, other features are less important.

(2) Compared with two empirical models, all the 12 trained ML models show higher accuracy and lower dispersion than empirical models, especially the XGBoost model. The mean absolute error of the XGBoost model on testing set is 0.161, while the two empirical models are 0.727 and 0.652, respectively. The standard deviation of the error of the XGBoost model on testing set is 0.239, while the two empirical models are 0.990 and 1.291, respectively.

(3) Simple design charts are given by partial dependence plot for reference by engineers. Partial dependence plots of transverse reinforcement ratio on drift ratio demand and axial load ratio show that the average maximum and minimum transverse reinforcement ratio are 1.29% and 0.35% for normal-strength and high-strength concrete, while 1.23% and 0.25% for normal-strength concrete (less than 60 MPa).

(4) The safety probability of the proposed XGBoost model on testing set is 55%. To provide a conservative estimate of the transverse reinforcement ratio, an additional value of 0.12% is suggested to be added to the predicted value for a guarantee rate of 80%.

The trained XGBoost model is transformed into C code and integrated into seismic design software for productive practice. An open-source data-driven model is created for continuous improvement, with the flexibility to incorporate more experimental data when available.

Data Availability

The experimental database, Jupyter Notebook python code for 12 ML models, C code of the trained XGBoost model, and other materials used to support the findings of this study

have been deposited in the GitHub repository (https://github.com/qiaobaojuan/ML-model-for-RC-columns.git).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to acknowledge Dr. Wenjie Liao (Tsinghua University) for his constructive suggestions for the preparation of the manuscript. This work was supported by the Youth Research Fund of China Academy of Building Research (No. 20210122331030008); the Beijing Natural Science Foundation (No. 8212019); and the Special Funding of China Academy of Building Research (No. 20220118330730013).

References

[1] R. Park and T. Paulay, *Reinforced concrete Structures*, John Wiley & Sons, New Jersey, USA, 1991.

[2] P. Paultre and F. Legeron, "Confinement reinforcement design for reinforced concrete columns," *Journal of Structural Engineering*, vol. 134, pp. 738–749, 2008.

[3] S. Watson, F. A. Zahn, and R. Park, "Confining reinforcement for concrete columns," *Journal of Structural Engineering*, vol. 120, no. 6, pp. 1798–1824, 1994.

[4] S. A. Sheikh and S. S. Khoury, "A performance-based approach for the design of confining steel in tied columns," *Structural Journal*, vol. 94, no. 4, pp. 421–431, 1997.

[5] O. Bayrak and S. A. Sheikh, "Confinement reinforcement design considerations for ductile HSC columns," *Journal of Structural Engineering*, vol. 124, no. 9, pp. 999–1010, 1998.

[6] K. J. Elwood, J. Maffei, K. A. Riederer, and K. Telleen, "Improving column confinement; Part 1: assessment of design provisions," *Concrete International*, vol. 31, no. 11, pp. 32–39, 2009.

[7] B. Li and R. Park, "Confining reinforcement for high-strength concrete columns," *ACI Structural Journal*, vol. 101, no. 3, pp. 314–324, 2004.

[8] K. Yang, Q Shi, and Q Lin, "Seismic performance and confinement reinforcement design of high-strength concrete columns confined with high-strength stirrups," *Advances in Structural Engineering*, vol. 24, no. 10, pp. 2061–2075, 2021.

[9] B. Kusuma, "Study of Minimum Requirements of Confinement in concrete Columns Confined with WRG in Moment Resisting Frames," *IOP Publishing*, vol. 1088, p. 012088, 2021.

[10] X. Lv, D. Zhou, and H. Jiang, "Deformation capacity and performance-based seismic design method for RC frame columns [J]," *Earthquake Engineering and Engineering Vibration*, no. 06, pp. 53–61, 2005.

[11] H. Sun, H. V. Burton, and H. Huang, "Machine learning applications for building structural design and performance assessment: state-of-the-art review," *Journal of Building Engineering*, vol. 33, p. 101816, 2021.

[12] H. Salehi and R. Burgueño, "Emerging artificial intelligence methods in structural engineering," *Engineering Structures*, vol. 171, pp. 170–189, 2018.

[13] H. Naderpour, M. Mirrashid, and P. Parsa, "Failure mode prediction of reinforced concrete columns using machine learning methods," *Engineering Structures*, vol. 248, p. 113263, 2021.

[14] S. Mangalathu and J. S. Jeon, "Classification of failure mode and prediction of shear strength for reinforced concrete beam-column joints using machine learning techniques," *Engineering Structures*, vol. 160, pp. 85–94, 2018.

[15] S. Mangalathu, H. Jang, S. H. Hwang, and J. S. Jeon, "Data-driven machine-learning-based seismic failure mode identification of reinforced concrete shear walls," *Engineering Structures*, vol. 208, p. 110331, 2020.

[16] H. Huang and H. V. Burton, "Classification of in-plane failure modes for reinforced concrete frames with infills using machine learning," *Journal of Building Engineering*, vol. 25, p. 100767, 2019.

[17] D. C. Feng, W. J. Wang, S. Mangalathu, G. Hu, and T. Wu, "Implementing ensemble learning methods to predict the shear strength of RC deep beams with/without web reinforcements," *Engineering Structures*, vol. 235, p. 111979, 2021.

[18] D. C. Feng, W. J. Wang, S. Mangalathu, and E. Taciroglu, "Interpretable XGBoost-SHAP machine-learning model for shear strength prediction of squat RC walls," *Journal of Structural Engineering*, vol. 147, no. 11, p. 04021173, 2021.

[19] J. S. Jeon, A. Shafieezadeh, and R. DesRoches, "Statistical models for shear strength of RC beam-column joints using machinelearning techniques," *Earthquake Engineering & Structural Dynamics*, vol. 43, no. 14, pp. 2075–2095, 2014.

[20] T. Liu, Z. Wang, Z. Long, J. Zeng, J. Wang, and J. Zhang, "Direct shear strength prediction for precast concrete joints using the machine learning method," *Journal of Bridge Engineering*, vol. 27, no. 5, p. 04022026, 2022.

[21] J. Rahman, K. S. Ahmed, N. I. Khan, K. Islam, and S. Mangalathu, "Data-driven shear strength prediction of steel fiber reinforced concrete beams using machine learning approach," *Engineering Structures*, vol. 233, p. 111743, 2021.

[22] O. B. Olalusi and P. O. Awoyera, "Shear capacity prediction of slender reinforced concrete structures with steel fibers using machine learning," *Engineering Structures*, vol. 227, p. 111470, 2021.

[23] H. Luo and S. G. Paal, "A locally weighted machine learning model for generalized prediction of drift capacity in seismic vulnerability assessments," *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 11, pp. 935–950, 2019.

[24] I. Peruš, K. Poljanšek, and P. Fajfar, "Flexural deformation capacity of rectangular RC columns determined by the CAE method," *Earthquake Engineering & Structural Dynamics*, vol. 35, no. 12, pp. 1453–1470, 2006.

[25] H. Luo and S. G. Paal, "Machine learning–based backbone curve model of reinforced concrete columns subjected to cyclic loading reversals," *Journal of Computing in Civil Engineering*, vol. 32, no. 5, p. 04018042, 2018.

[26] C. L. Ning, L. Wang, and W. Du, "A practical approach to predict the hysteresis loop of reinforced concrete columns failing in different modes," *Construction and Building Materials*, vol. 218, pp. 644–656, 2019.

[27] C. Huang, Y. Li, Q. Gu, and J. Liu, "Machine learning–based hysteretic lateral force-displacement models of reinforced concrete columns," *Journal of Structural Engineering*, vol. 148, no. 3, p. 04021291, 2022.

[28] D. C. Feng, Z. T. Liu, X. D. Wang et al., "Machine learning-based compressive strength prediction for concrete: an adaptive boosting approach," *Construction and Building Materials*, vol. 230, p. 117000, 2020.

[29] M. C. Kang, D. Y. Yoo, and R. Gupta, "Machine learning-based prediction for compressive and flexural strengths of

steel fiberreinforced concrete," *Construction and Building Materials*, vol. 266, p. 121117, 2021.

[30] R. Solhmirzaei, H. Salehi, and V. Kodur, "Predicting flexural capacity of ultrahigh-performance concrete beams: machine learning–based approach," *Journal of Structural Engineering*, vol. 148, no. 5, p. 04022031, 2022.

[31] E. Alotaibi, O. Mostafa, N. Nassif, M. Omar, and M. G. Arab, "Prediction of punching shear capacity for fiber-reinforced concrete slabs using neuro-nomographs constructed by machine learning," *Journal of Structural Engineering*, vol. 147, no. 6, p. 04021075, 2021.

[32] D. Qu and W. Chang, "Design methods for spiral stirrups confined concrete columns by evaluating the lateral performance of transverse reinforcements," *SN Applied Sciences*, vol. 1, no. 12, pp. 1705–1711, 2019.

[33] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[34] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[35] T. Chen and C. Guestrin, "Xgboost: a scalable tree boosting system," *Association for Computing Machinery*, pp. 785–794, 2016.

[36] G. Ke, Q. Meng, and T. Finley, "Lightgbm: a highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[37] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: unbiased boosting with categorical features," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[38] F. Pedregosa, G. Varoquaux, and A. Gramfort, "Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[39] B. Sivaramakrishnan, *Non-linear Modeling Parameters for Reinforced concrete Columns Subjected to Seismic Loads*, Master's thesis. The University of Texas at Austin. Texas, 2010.

[40] W. Ghannoum, B. Sivaramakrishnan, and S. Pujol, *NEES: ACI 369 Rectangular Column Database*, 2015, https://datacenterhub.org/resources/255.

[41] W. Ghannoum, B. Sivaramakrishnan, and S. Pujol, *NEES: ACI 369 Circular Column Database*, 2015, https://datacenterhub.org/resources/254.

[42] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[43] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of Statistics,* pp. 1189–1232, 2001.

[44] W. M. Ghannoum and A. Matamoros, "Nonlinear modeling parameters and acceptance criteria for concrete columns," *ACI Special Publications*, vol. 297, no. 1, pp. 1–24, 2014.