

Article

Interpretable Machine Learning Models for Prediction of UHPC Creep Behavior

Peng Zhu ^{1,2} , Wenshuo Cao ¹, Lianzhen Zhang ³ , Yongjun Zhou ^{4,*}, Yuching Wu ^{1,*}  and Zhongguo John Ma ⁵

¹ College of Civil Engineering, Tongji University, Shanghai 200092, China; pzhu@tongji.edu.cn (P.Z.); 1851808@tongji.edu.cn (W.C.)

² Key Laboratory of Performance Evolution and Control for Engineering Structures, Tongji University, Ministry of Education, Shanghai 200092, China

³ Institute of Bridge Engineering Research, Harbin Institute of Technology, Harbin 150090, China; lianzhen@hit.edu.cn

⁴ Shaanxi Provincial Key Laboratory of Highway Bridge and Tunnel, Chang'an University, Xi'an 710064, China

⁵ Department of Civil and Environmental Engineering, University of Tennessee, Knoxville, TN 37996, USA; zma2@utk.edu

* Correspondence: zyj@chd.edu.cn (Y.Z.); ycwu@tongji.edu.cn (Y.W.)

Abstract: The creep behavior of Ultra-High-Performance Concrete (UHPC) was investigated by machine learning (ML) and SHapley Additive exPlanations (SHAP). Important features were selected by feature importance analysis, including water-to-binder ratio, aggregate-to-cement ratio, compressive strength at loading age, elastic modulus at loading age, loading duration, steel fiber volume content, and curing temperature. Four typical ML models—Random Forest (RF), Artificial Neural Network (ANN), Extreme Gradient Boosting Machine (XGBoost), and Light Gradient Boosting Machine (LGBM)—were studied to predict the creep behavior of UHPC. Via Bayesian optimization and 5-fold cross-validation, the ML models were tuned to achieve high accuracy ($R^2 = 0.9847, 0.9627, 0.9898, \text{ and } 0.9933$ for RF, ANN, XGBoost, and LGBM, respectively). The contribution of different features to the creep behavior was ranked. Additionally, SHAP was utilized to interpret the predictions by the ML models, and four parameters stood out as the most influential for the creep coefficient: loading duration, curing temperature, compressive strength at loading age, and water-to-binder ratio. The SHAP results were consistent with theoretical understanding. Finally, the UHPC creep curves for three different cases were plotted based on the ML model developed, and the prediction by the ML model was more accurate than that by fib Model Code 2010.

Keywords: UHPC; creep; machine learning; interpretability; prediction model



Citation: Zhu, P.; Cao, W.; Zhang, L.; Zhou, Y.; Wu, Y.; Ma, Z.J. Interpretable Machine Learning Models for Prediction of UHPC Creep Behavior. *Buildings* **2024**, *14*, 2080. <https://doi.org/10.3390/buildings14072080>

Academic Editor: Muhammad Junaid Munir

Received: 6 June 2024

Revised: 2 July 2024

Accepted: 5 July 2024

Published: 7 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Ultra-High-Performance Concrete (UHPC) is a promising material for its exceptional mechanical and durability properties [1,2]. Creep is essential for predicting and assessing the mechanical response of concrete structures over both short and long periods under sustained loads. In the prolonged mechanical behavior of concrete structures, creep stands as a primary contributor to significant time-related engineering challenges, including substantial deflection and loss of prestress in bridges [3]. Some researchers summarized the effects of various factors on the creep behavior of UHPC and developed the creep models for UHPC [4–6]. Experimental studies on UHPC have proven to be a viable and effective approach to study the creep behavior [7–16]. However, these methods suffer from drawbacks like high cost, being time-consuming, and discrete material properties. Therefore, alternative methods such as ML have been explored to investigate the creep behavior of UHPC, which can handle massive amounts of more various types of data than theoretical models.

Huang et al. summarized and analyzed the effects of different influencing factors such as fiber type and content, curing conditions, water-to-binder ratio, loading level, and

loading age on the creep of UHPC. An increase in the water-to-binder ratio, strength, elastic modulus, or loading duration can result in an increase in creep, while an increase in fiber and aggregate content or curing temperature can suppress creep development [4,5,17].

Machine learning (ML) algorithms have been used to predict concrete properties such as compressive strength, shear strength, and fracture toughness [18–22]. Nunez et al. conducted a comprehensive review of ML algorithms in predicting the mechanical properties of concrete. Seven ML-related algorithms were discussed: Artificial Neural Network (ANN), Support Vector Machines (SVMs), Fuzzy Logic, Genetic Algorithms, Tree-based Ensembles, Hybrid Procedures, and Deep Learning, and these algorithms were employed to predict the strength of High-Performance Concrete (HPC), Self-Compacting Concrete (SCC), Recycled Aggregate Concrete (RAC), and other types of concrete [23].

There are also studies that utilize ML models to predict the creep characteristics of concrete. Current ML applications in predicting concrete creep focused on Back Propagation (BP) neural networks. Bal et al. established an ANN model based on the NU database to predict the creep of concrete [24]. Karthikeyan et al. predicted the creep coefficient of HPC by computing the CEB 90 creep model results and training a neural network model [25]. Hodhod et al. derived an explicit concrete creep formula and used genetic algorithms to fit the residual between the calculated values from the explicit formula and actual values to obtain more accurate predictions [26]. Gandomi et al. utilized multi-objective genetic programming to derive an explicit expression for concrete creep based on the provided database [27].

Regarding the determination of hyperparameters, Feng et al. employed grid search and k-fold methods to effectively identify the optimal hyperparameters for models like XGBoost and Least Squares Support Vector Machine (LS-SVM) when predicting RAC creep. Particularly, the XGBoost-based prediction model demonstrated accuracy and efficiency for multi-input predictions [28]. Considering the impact of recycled coarse aggregate replacement rates on recycled concrete creep, Xiao et al. constructed a neural network model for predicting the creep of recycled concrete based on the RILEM B3 creep model [29]. Li et al. used a grid search method to adjust existing creep prediction models, and improved prediction performance. Both the fully connected neural network and support vector regression models demonstrated high creep prediction capabilities [30].

The ML model has proven to be promising for predicting various behaviors of concrete materials. However, it remains a black box model. In prior studies, methods such as feature importance analysis, grey relational analysis, and parameter sensitivity analysis were used to explain the interpretability of various ML models. However, these analytical methods only offer importance rankings for individual features and cannot explain the impact of features on outcomes. Some studies have started using SHAP to overcome this limitation. SHAP has been applied in various domains such as ordinary concrete, reinforced concrete structures, infrastructure systems, etc. [31–35]. Wakjira et al. developed eleven ML models to predict the shear capacity of FRP-RC beams and used SHAP to interpret the results. They also developed explainable ML models to predict the flexural capacity of RC beams and the generalization ability between single models and ensemble models was compared [36,37]. Liang et al. built three Ensemble Machine Learning (EML) models for the prediction of creep of concrete. SHAP was adopted to interpretate the predictions of the EML model and validate its reasonability [38]. Feng et al. proposed five typical EML models to establish the surrogate model for predicting the creep behavior of RAC. The input variables included environmental conditions, loading conditions, and concrete mix proportions [39]. This method can sort the contributions of input features for creep behaviors and interpret the prediction results of the best EML model. Wakjira et al. proposed an innovative methodology for predicting the strength of UHPC and optimizing its design and developed ML models for the compressive stress–strain and seismic design of UHPC bridge columns [40–43]. Katlav and Ergen improved the prediction of the compressive strength of UHPC with different algorithms [44]. The interpretable ML model is beneficial for a deeper understanding of UHPC creep behavior. As far as the

authors' knowledge goes, there have been no studies on ML methods to predict UHPC creep behavior up to now.

In this study, a UHPC creep database was established and four UHPC creep prediction ML models (RF, ANN, XGBoost, and LGBM) were developed. Instead of grid search, Bayesian optimization was employed to expedite the search for optimal hyperparameters. The SHAP method was used to elucidate the specific impact of features on the prediction results. Finally, time-dependent creep curves were drawn based on the developed ML model and were compared with the experimental results and also the theoretical predictions.

2. Database Establishment and Data Preprocessing

2.1. Database Establishment

The establishment of the database relies on previous experimental studies [45–49], all of which reported data in graphical form. The figures were imported into Origin and scaled, and the values were extracted from the graphs. For the database establishment, all studies with the water-to-binder ratio (w/b , %), aggregate-to-cement ratio (a/c , %), and ambient conditions recorded were collected. Graphs from the literature were imported into Origin and scaled, and the values were extracted. The creep data were obtained by sampling equally from different experiments, ensuring that the number of points taken from each experiment are the same to avoid the impact of data imbalance. The database contained 560 UHPC creep data. There were no missing values in the database established in this study. Thirteen input parameters were considered initially: water-to-binder ratio (w/b , %), aggregate-to-cement ratio (a/c , %), loading age (t_0 , days), compressive strength at loading age (f_{ct0} , MPa), elastic modulus at loading age (E_{t0} , MPa), loading duration (t , days), steel fiber volume content (*steel fiber*, %), silica fume content (S_f , %), surface-to-volume ratio (V/S , %), curing temperature (T_{cure} , °C), experimental temperature (T , °C), relative humidity (RH , %), and loading stress intensity ratio (σ/f_{ct0} , %). The output parameter was the creep coefficient.

2.2. Feature Selection

The database may contain unnecessary or redundant features, which could increase the complexity and reduce the generalization ability of the models. Via feature selection, the dimension of the feature space could be reduced and the interference from redundant features could be decreased. This allows the model to focus on key features, and thereby reduces the computation cost. Furthermore, feature selection helps to reduce the risk of overfitting. The database was divided into two parts: the training dataset (70% of the data) and the testing dataset (30% of the data). An ML model based on the XGBoost algorithm was established, and the thirteen input features were sorted in descending order of importance through feature importance analysis, as shown in Figure 1. The weight was used as the indicator, which is the number of times a feature is used to split the data across all trees. Pearson correlation coefficient (r) between input features were calculated, as shown in Figure 2. The importance of input parameters to the model and the correlation between parameters can be characterized. For instance, loading duration (t) was the most influential feature and there is a strong negative correlation between curing temperature (T_{cure}) and loading age (t_0).

Based on the analysis of feature importance and Pearson correlation coefficient between input features, the top seven parameters with significant impact between which the correlation was not strong were selected as the input features in the formal study: t (days), f_{ct0} (MPa), w/b (%), E_{t0} (MPa), a/c (%), T_{cure} (°C), and *steel fiber* (%). The data distribution of the input features is shown in Figure 3 and summarized in Table 1.

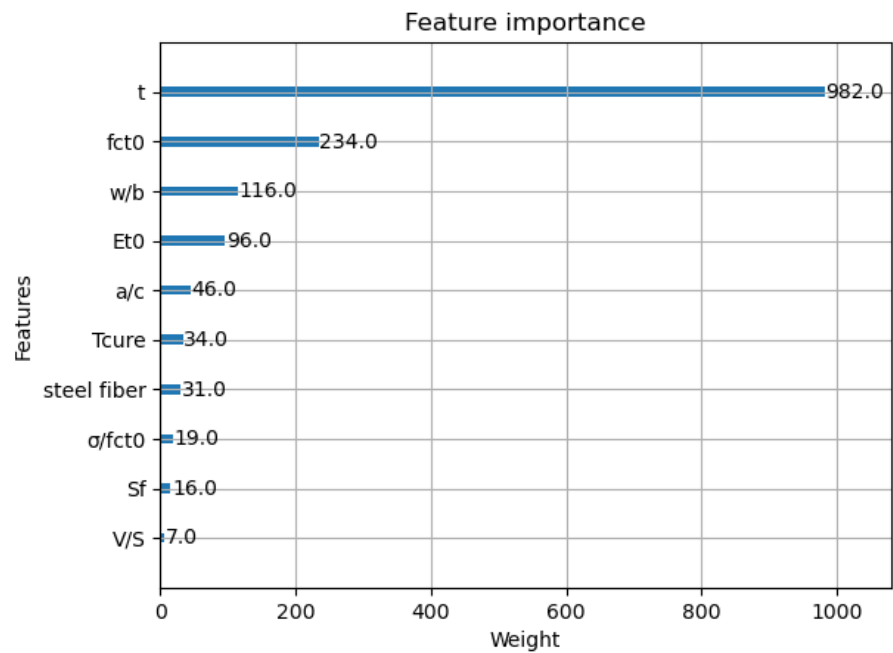


Figure 1. Rank of feature importance based on the XGBoost model.

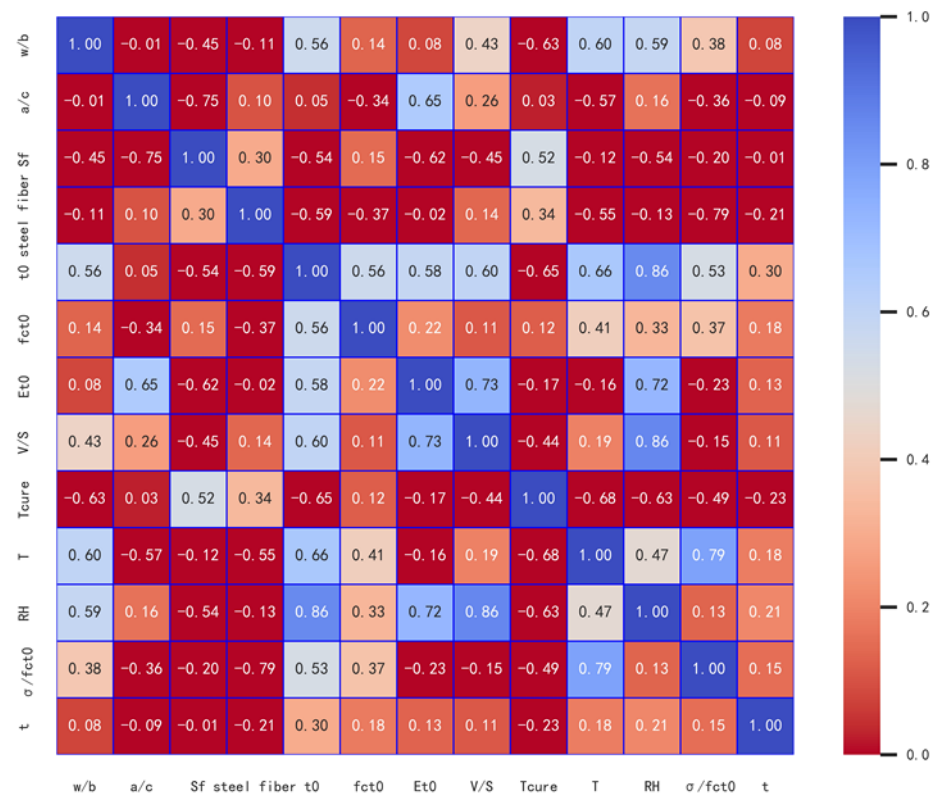


Figure 2. Pearson correlation coefficient between input features.

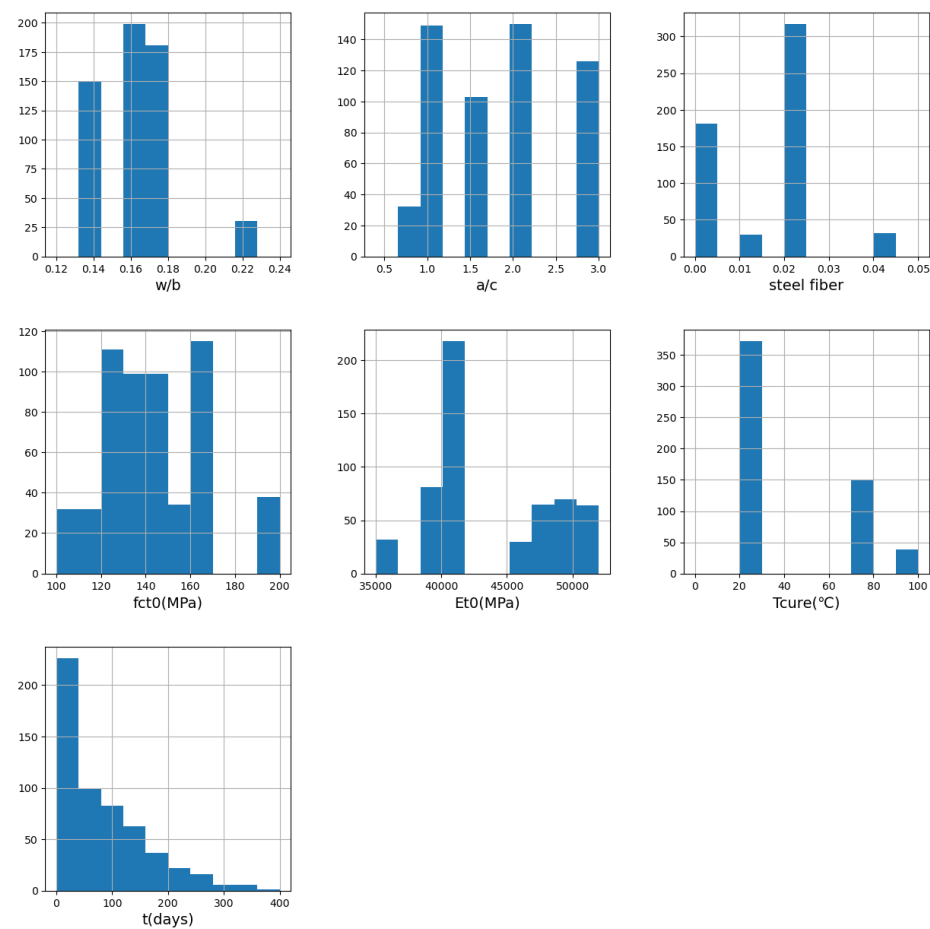


Figure 3. Data distribution of input features.

Table 1. The data distribution of input features.

Input Feature	Minimum	Maximum	Mean	Standard Derivation
w/b	0.14	0.22	0.16	0.02
a/c	0.83	2.92	1.84	0.64
steel fiber (%)	0	4	1.4	0.01
f_{ct0} (MPa)	107.1	198.0	144.8	21.89
E_{t0} (MPa)	36,000	51,000	43,740	4513.22
T_{cure} (°C)	20	90	41	26.64
t (days)	0	364	82	77.55

2.3. Data Standardization

In the ML and statistical modeling, many algorithms are extremely sensitive to the scale of data. Standardization can eliminate the influence of the features' units, allowing the meaningful comparison and analysis of data with different magnitudes. Before the model training, the data were standardized to ensure that the range and variance of different features (or variables) were similar, avoiding the excessive impact of certain features on model training, and thereby improving the performance of the models. In this study, the input data were standardized as per Equation (1).

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

where z is the new data, x is the original data, μ is the mean of the original data, and σ is the standard deviation.

3. Methodology

As a single decision tree model was prone to overfitting and possessed high variance while the ensemble models could reduce the problems [36,37], the single decision tree model was not built. Four commonly used algorithms, (a) RF, (b) ANN, (c) XGBoost, and (d) LGBM, were used to establish the ML models, and the impact of different features was explained.

3.1. Machine Learning Models

3.1.1. Random Forest

RF is a powerful ML algorithm built on the idea of ensemble learning. It is an integrated model composed of multiple decision trees, and the final prediction or classification can be made by integrating the prediction results of each decision tree [50,51]. By using Bootstrap technology for random sampling with dropout and the random selection of features during the training process of each decision tree, each tree has a certain degree of difference and diversity.

The RF algorithm was employed to establish a prediction model for the creep behavior of UHPC. In order to optimize the performance of the model, hyperparameters such as the number of decision trees (n-estimators), the maximum depth of each decision tree (max_depth), the maximum size of each random feature subset (max_features), and the minimum number of samples to split internal nodes (min_samples_split) were selected for optimization. Mean square error was used as the loss function and applies to other models introduced below, as per Equation (2).

$$L = \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2)$$

where n is the number of the training data, y_i is the creep coefficient of i -th sample point, and $\hat{y}_i$ is the prediction value.

3.1.2. Artificial Neural Network

An ANN is a biomimetic mathematical model that solves practical problems by imitating the transmission, storage, and processing of brain signals [52,53]. Similar to biological neural networks, the building blocks of an ANN are simple computing devices, but their interconnections are not as complex as biological neurons. In addition, an ANN has the ability to handle nonlinear relationships between variables and can learn adaptively and autonomously.

A BP algorithm was employed to train the feedforward neural network model. A BP algorithm is an optimization algorithm based on gradient descent and is used to adjust the connection weights in neural networks to minimize the error between predicted and actual values. The network topological structure is shown in Figure 4, which generally includes an input layer, hidden layer(s), and an output layer. The hidden layer size (hidden layer sizes), learning rate (learning_rate_init), and regularization penalty parameter (alpha) were chosen as the hyperparameters for optimization. To prevent overfitting, a regularization penalty parameter was added into the loss function. And the ReLU function was used as the activation function to reduce the potential for overfitting, as per Equation (3) [54].

$$f(x) = \max(0, x) \quad (3)$$

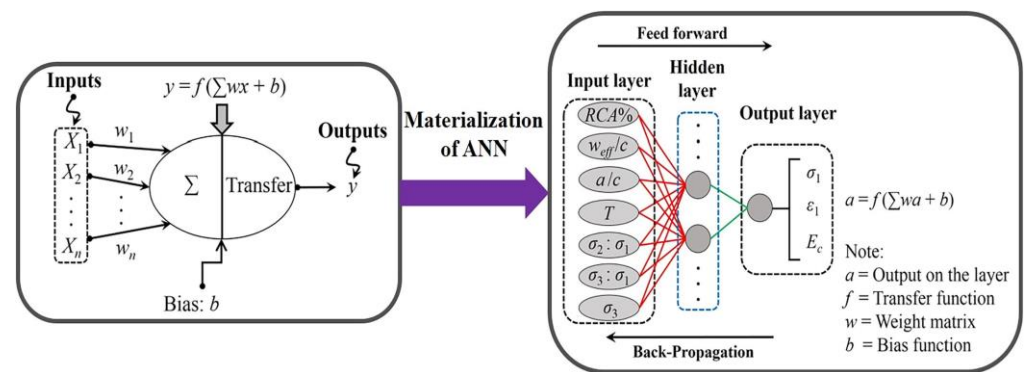


Figure 4. Structure of artificial neuron and BP-ANN [55]. Reprinted with permission from Ref. [55]. 2019, Elsevier.

3.1.3. Extreme Gradient Boosting Machine

XGBoost is an ML algorithm based on Gradient Boosting Decision Trees [56]. It adopts the technique of Gradient Boosting, which trains a new model each time to correct the prediction errors of previous models and thereby continuously optimizes the prediction results.

The number of decision trees (n-estimators), the maximum depth of each decision tree (max_depth), the minimum loss reduction (gamma), and learning rate (learning_rate) were selected as the hyperparameters for optimization.

3.1.4. Light Gradient Boosting Machine

LGBM is a gradient boosting framework that focuses on processing large-scale datasets and efficient training. It is a decision tree-based ensemble learning algorithm that achieves fast training speed and efficient performance by using histogram-based algorithms to accelerate training and reduce memory usage [57].

The number of decision trees (n-estimators), the maximum depth of each decision tree (max_depth), the minimum number of samples for the downward splitting of leaf nodes (mediation-leaf), and learning rate (learning_rate) were selected as the hyperparameters for optimization.

3.2. Bayesian Optimization

The performance of a model often depends largely on the setting of hyperparameters, and traditional hyperparameter tuning methods typically use grid search and random search. These methods may require traversing a large number of parameter combinations, which can be time-consuming and inefficient in practice [58–61]. Bayesian optimization is an iterative optimization method used for global optimization, typically used to find the optimal solution for complex black box functions [62]. It is based on Bayesian inference and infers the potential shape of the objective function. In each iteration, it adaptively adjusts its exploration of the parameter space based on the previous evaluation results, making it more likely to find the global optimal solution. Compared to grid search and random search, Bayesian optimization usually has a higher efficiency.

Bayesian optimization was used to obtain the optimal combination of hyperparameters corresponding to the optimal cross-validation score. Cross-validation is a technique for evaluating model performance and selecting the best hyperparameters [63,64]. It divides the dataset into multiple subsets (for example, k-fold cross-validation divides the dataset into k subsets), and then takes turns using one part as the validation dataset and the other parts as the training dataset to train and validate the model multiple times. This can evaluate the performance of the model on different subsets of data, reduce bias caused by the distribution of specific datasets, and thus better evaluate the model's generalization ability.

3.3. Evaluation Indexes

The models were evaluated with four indexes: coefficient of determination (R^2), mean absolute error (MAE), root mean square error (RMSE), and composite evaluation index (CEI), for a comprehensive comparison of the performance of all models across various metrics [65]. Both MAE and RMSE explicitly describe the residual error at each sample point, providing a precise evaluation of model performance. In contrast, R^2 normalizes the squared residual error with the variance in the database, yielding a dimensionless score between 0 and 1. R^2 is more intuitive and convenient for comparing the performance of different models. CEI was adopted as a comprehensive metric index to obtain a comprehensive comparison of the performance of all models across various metrics, including R^2 , MAE, and RMSE. The closer to zero CEI is, the better the model is. Finally, R^2 was chosen as the primary metric index in the subsequent analysis. Four indexes are as per Equations (4)–(7).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (4)$$

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} \quad (5)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (6)$$

$$CEI = \frac{1}{N} \sum_{j=1}^N \frac{P_j - P_{j,min}}{P_{j,max} - P_{j,min}} \quad (7)$$

where n is the number of the testing data, y_i is the creep coefficient of the i -th sample point in the database, $\hat{y}_i$ is the prediction value for the i -th sample point by the ML models, $\bar{y}$ is the averaged value of the creep coefficient, N is the total number of single metric indexes (3 in this study), P_j is the value of the j -th metric index, $P_{j,min}$ and $P_{j,max}$ are the minimum and maximum values of the j -th metric index.

4. Results and Discussion

4.1. Model Performance

4.1.1. Optimization Results

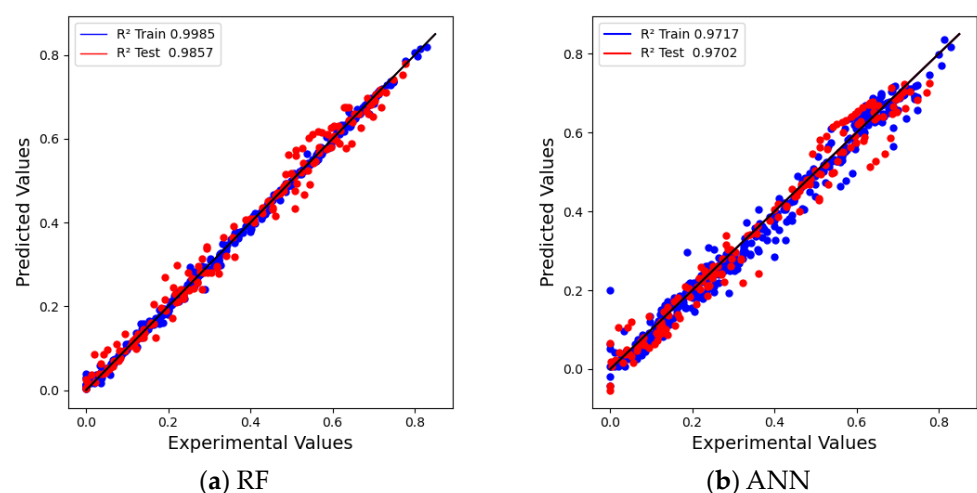
In order to evaluate the generalization ability of the four ML models, this study used a pseudo-random number generator to shuffle the database first, and then all data in the database were randomly divided into two parts in a 7:3 ratio: the training dataset and the testing dataset. The optimal hyperparameters were searched for each ML model through a Bayesian optimization algorithm combined with 5-fold cross-validation. For each ML model, five iterations were firstly conducted to form an initial sample space, which established the initial Gaussian Process (GP) prior. Then, the acquisition function was calculated to select a new sample point and then update the GP prior at the end of each iteration. In general, at least 30 iterations are recommended for most optimization aims [66]. In this study, 30 iterations of Bayesian optimization were performed, and the optimization was stopped when the maximum number of iterations was reached to search for the best combination of hyperparameters, and ideal accuracy was achieved. The training dataset was divided into five sub datasets on average. A single sub dataset was retained as the validation set, and the other four datasets were used for training. The cross-validation was repeated five times, and each sub dataset was validated once. The R^2 was calculated to make an estimation. The range and combination of hyperparameters were shown in Table 2.

Table 2. Optimized hyperparameters of ML models.

ML Model	Hyperparameters	Value Range	Best Value
RF	n_estimators	(10, 600)	362
	max_depth	(5, 30)	19
	max_features	(0.1, 0.999)	0.6867
	min_samples_split	(2, 30)	2
ANN	hidden_layer_sizes	(5, 100)	57
	alpha	(0.0001, 0.999)	0.0758
	learning_rate_init	(0.001, 0.999)	0.0005
XGBoost	n_estimators	(10, 600)	274
	max_depth	(2, 20)	5
	gamma	(0.0001, 0.999)	0.0151
	learning_rate	(0.1, 0.999)	0.5976
LGBM	n_estimators	(10, 600)	355
	max_depth	(2, 20)	8
	min_data_in_leaf	(5, 50)	7
	learning_rate	(0.001, 0.999)	0.1563

4.1.2. Testing results

The determined hyperparameters were assigned to the ML models, the data of the training dataset were trained, and model testing was conducted in the testing dataset. The comparison between the predicted results and the actual values in the training and test datasets are shown in Figure 5. The evaluation indexes of the four ML models in the training and testing datasets were compared, as shown in Table 3 and Figure 6. Radar charts were drawn for each index, as shown in Figure 7. The accuracy of the four models was high in the training and testing datasets. The R^2 of the four models was over 0.97 in the training and testing datasets. For the four evaluation indexes, the ranking of model performance in the training dataset was LGBM > RF > XGBoost > ANN, while the ranking in the testing dataset was LGBM > XGBoost > RF > ANN. The RF model and XGBoost model showed similar accuracy. The ANN model performed the worst as a single learning model while the other three are ensemble models. A Taylor diagram of the testing dataset was plotted to compare the statistical relationships between the predictions of the four ML models and experimental data, including R^2 , standard deviation, and RMSE, as shown in Figure 8. The proximity of the standard deviation of predictions to that of the experimental data was ranked LGBM > ANN > RF > XGBoost.

**Figure 5.** Cont.

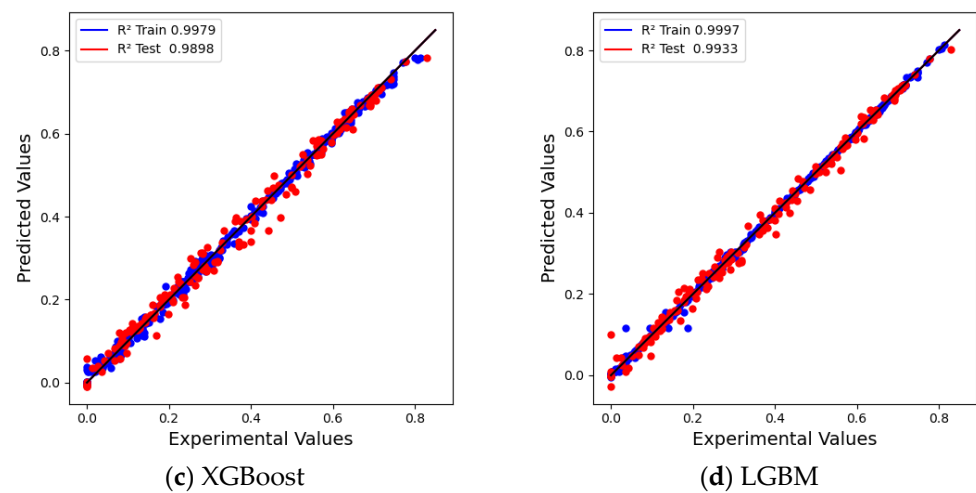


Figure 5. Experimental versus predicted values in training and test sets using four ML models.

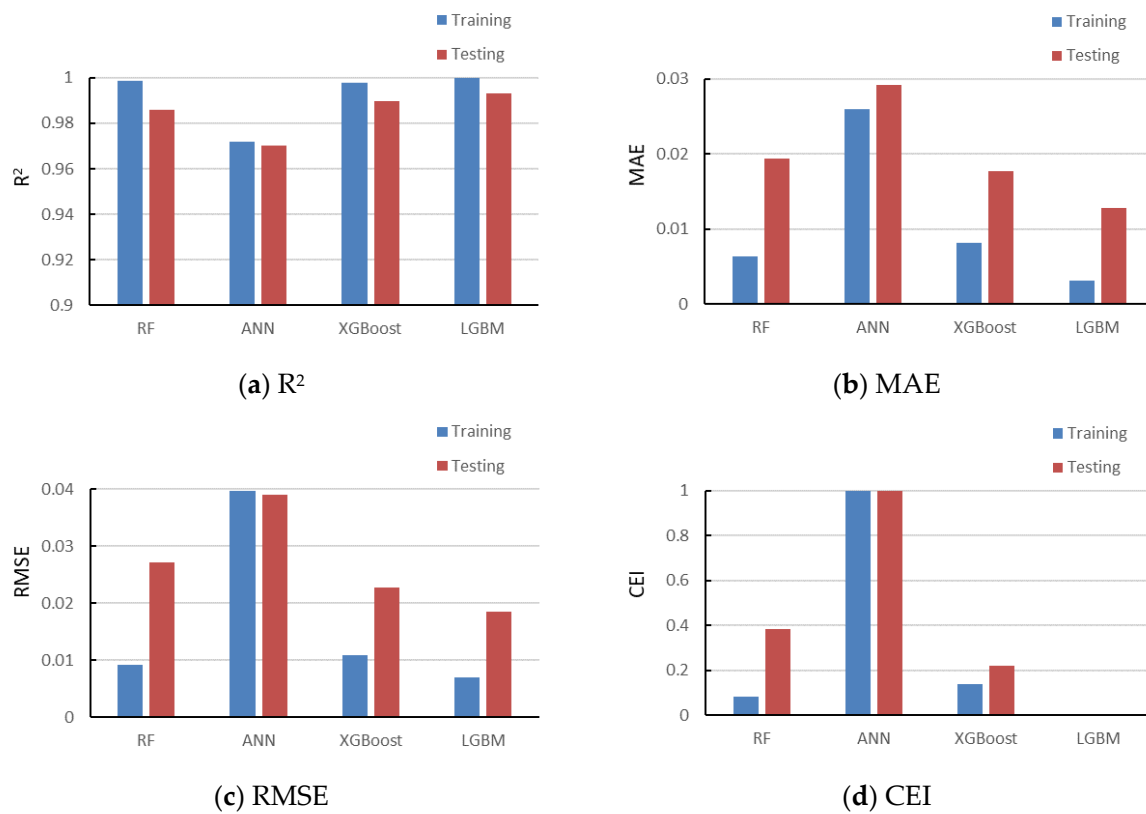


Figure 6. ML model performance in training and testing datasets.

Table 3. Comparison of different ML models' performance.

ML Model	R^2		MAE		RMSE		CEI	
	Train	Test	Train	Test	Train	Test	Train	Test
RF	0.9985	0.9857	0.0063	0.0194	0.0091	0.0271	0.0825	0.3830
ANN	0.9717	0.9702	0.0259	0.0292	0.0397	0.0391	1.0000	1.0000
XGBoost	0.9979	0.9898	0.0082	0.0177	0.0109	0.0228	0.1357	0.2197
LGBM	0.9997	0.9933	0.0031	0.0128	0.0070	0.0185	0.0000	0.0000

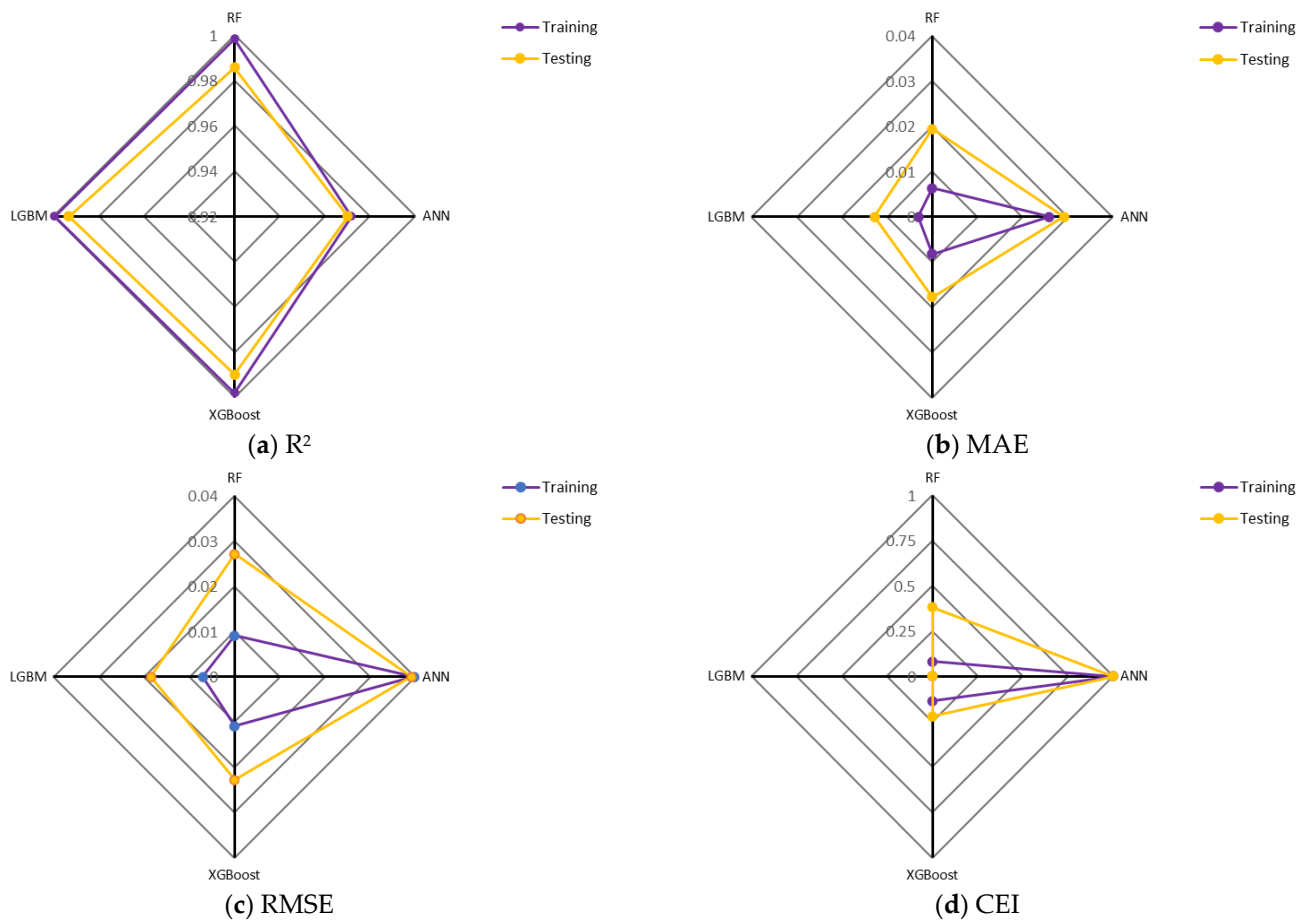


Figure 7. Radar charts for four indexes of ML models.

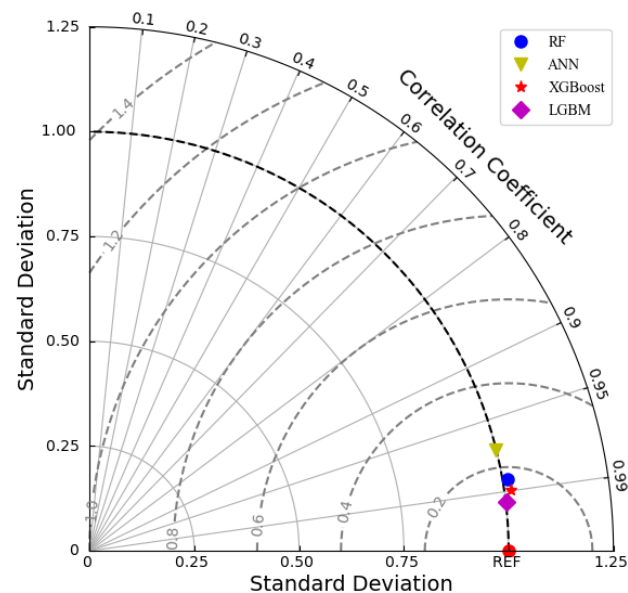


Figure 8. Taylor diagram of the testing dataset based on four ML models.

Sensitivity analysis was conducted on the LGBM model, as shown in Figure 9. Total sensitivity index (ST) is the comprehensive contribution of a single feature and its interaction with others to the output variation. The higher the ST value is, the more sensitive the model is to changes in the input features. For instance, the most sensitive feature is loading

duration (t). The black bar in the figure is the confidence interval at a 95% guarantee rate. The changes in various input features have a significant impact on the model, which indicates the rationality of the selected features.

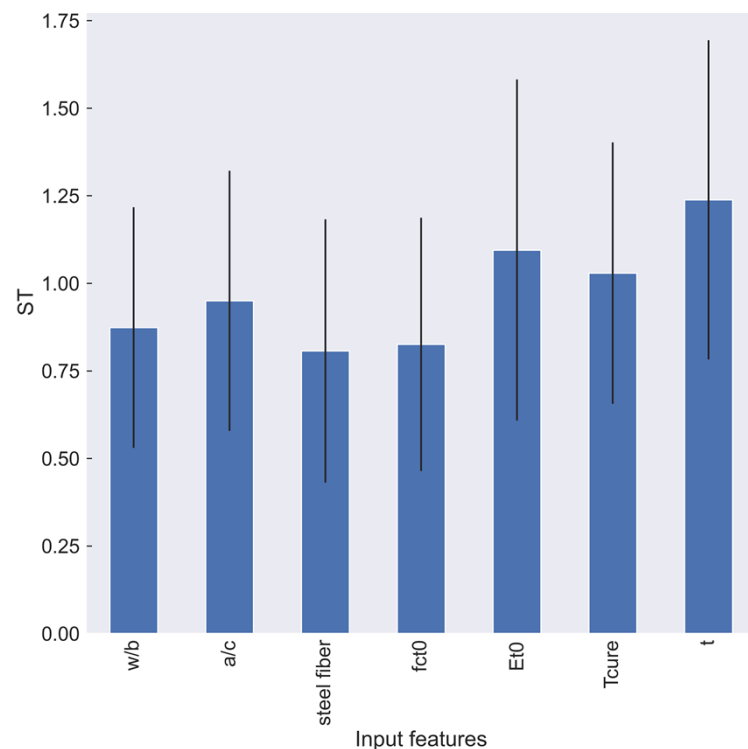


Figure 9. Total sensitivity index of input features on LGBM model.

4.2. Model Interpretability

4.2.1. Overview of SHAP

In order to deal with the lack of explicability of the black box model, Lundberg and Lee developed a new model agnostic technique referred to as SHapley Additive exPlanation (SHAP) [67]. Based on cooperative game theory, SHAP is expressed by the average marginal contribution of an eigenvalue in all possible alliances. Specifically, the SHAP value of a feature is the average predictive value of the sample with the feature minus the average predictive value of other samples without the feature. In order to provide the interpretability of the ML model, the output of the ML model is represented by the linear addition of its input features multiplied by the corresponding SHAP value, as per Equation (8).

$$f(x) = \varphi_0 + \sum_{i=1}^N \varphi_i X'_i \quad (8)$$

where φ_0 is the average value of all predictions; N is the number of input features; φ_i is the SHAP value of the i -th feature; and X'_i is the coalition vector of the i -th feature.

4.2.2. Local Interpretation

The aim of local interpretation is to provide explanations for the predictions of each individual sample. The linear addition of SHAP values constitutes the output of each sample, so the SHAP values of each input parameter can characterize the contribution of each feature. Three typical samples were selected, as shown in Table 4, and their SHAP values were used for model interpretation. Local interpretation diagrams of the three samples are shown in Figure 10. For the LGBM model, the scaling factor was set to 1000 (both the base value and prediction were magnified by 1000 times for observation). The prediction output of the UHPC creep coefficient was the result of the mutual cancellation of

the SHAP values of different features, where the red and blue bars represented the positive and negative contributions to the predicted values, respectively.

Table 4. Details of selected samples for exemplification.

Sample	a/c (%)	w/b (%)	T_{cure} (°C)	f_{ct0} (MPa)	E_{t0} (MPa)	steel fiber (%)	t (days)
A	2.00	0.14	75	137.5	40,837	2	81
B	0.83	0.18	23	107.1	36,000	4	7
C	2.92	0.22	20	137.7	46,000	0	68

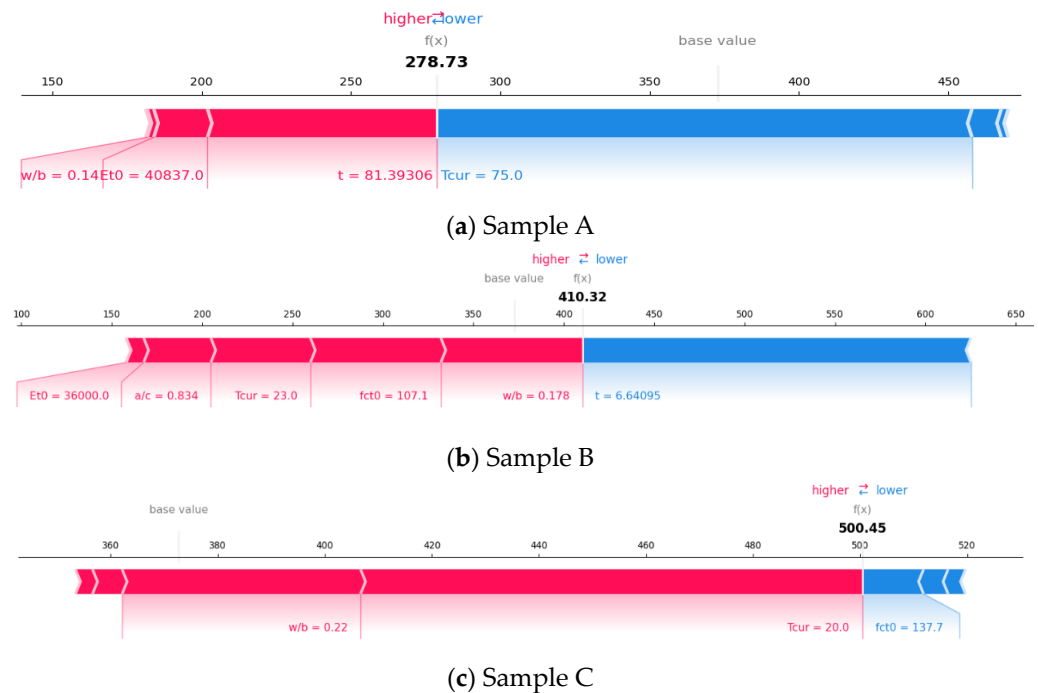


Figure 10. Force plot for local interpretation of three sample points based on LGBM model.

In the local interpretation, the basic value of the creep coefficient was 372.66, which was the mean of the predicted values for all samples. In Sample A, the predicted value of the creep coefficient was 278.73, below the basic value, indicating that creep was suppressed. This was mainly caused by the high curing temperature (T_{cure}). In Sample B, the predicted value was 410.32, above the basic value, indicating that creep was promoted. The high water-to-binder ratio (w/b), low aggregate-to-cement ratio (a/c), low compressive strength at loading age (f_{ct0}), low elastic modulus at loading age (E_{t0}), and short loading duration (t) could explain this phenomenon. In Sample C, the predicted value was 500.45, above the basic value, indicating that creep was also promoted, and the degree was stronger than that in Sample B. The reason might be that the higher water-to-binder ratio (w/b) and lower curing temperature (T_{cure}) played a controlling role.

4.2.3. Global Interpretation

Global interpretation can provide an overview of the SHAP values for all input parameters. The SHAP values of 168 samples in the testing dataset were calculated based on the XGBoost model and LGBM model, as these two models performed well in the testing dataset. Each feature was sorted according to its average SHAP value. As shown in Figure 11, the SHAP values of each parameter in the XGBoost model and LGBM model were calculated, with the influence decreasing from top to bottom. The SHAP heatmap is another way to visualize SHAP values. As shown in Figure 12, SHAP heatmaps based on the XGBoost and LGBM models were drawn. Features were plotted on the y-axis and

records were plotted on the x-axis. A line graph of the predicted values for each sample was plotted at the top, reflecting the degree of deviation between the predicted value and the base value. The red color indicated that features had a promoting effect on the prediction results, while the blue color was the opposite. Based on the results of global interpretation, the loading duration, curing temperature, compressive strength at loading age, and water-to-binder ratio brought the greatest impact on the UHPC creep. The loading duration and water-to-binder ratio promoted the UHPC creep while the curing temperature, aggregate-to-cement ratio, compressive strength, and elastic modulus at loading age and steel fiber volume content suppressed the UHPC creep.

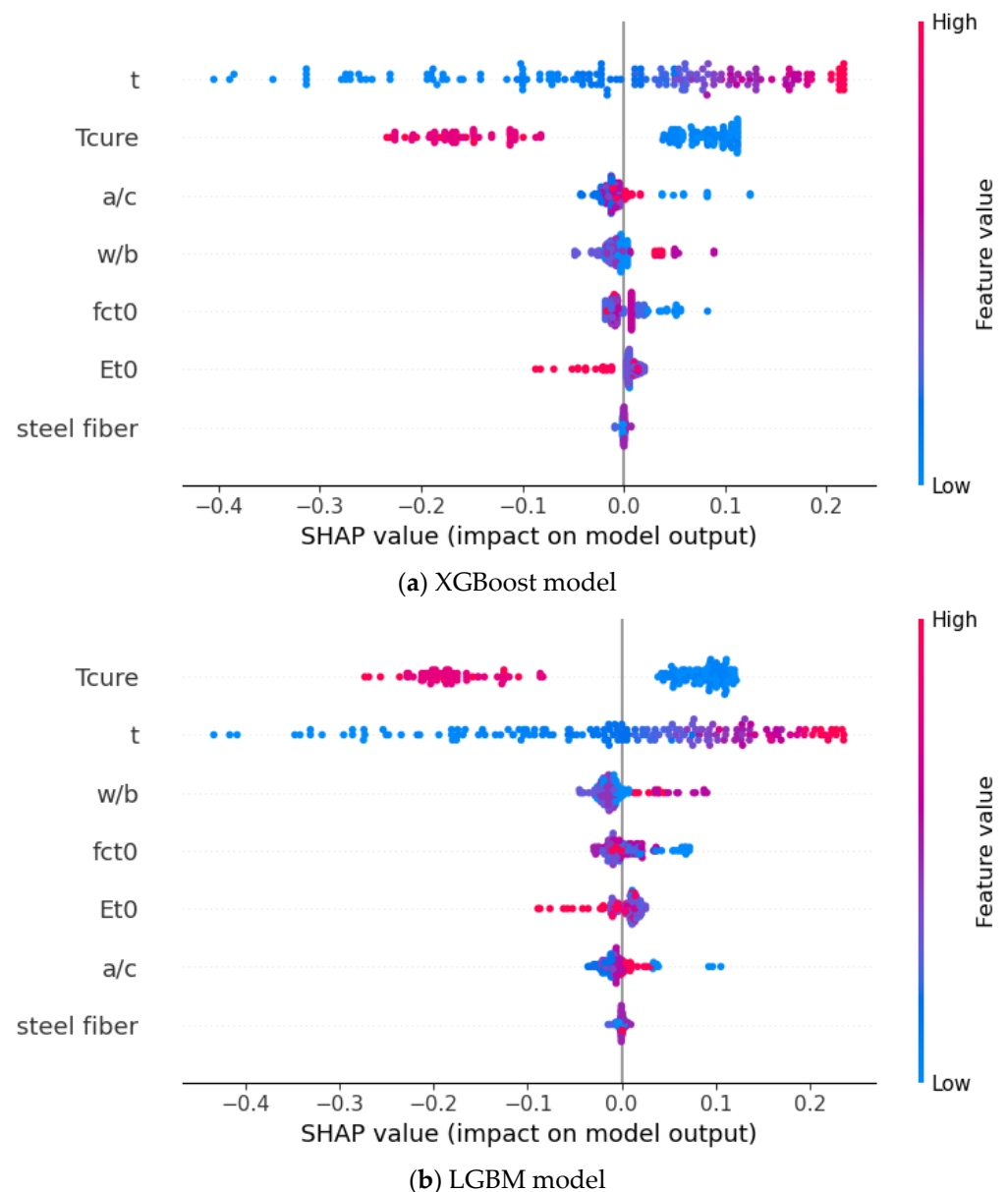
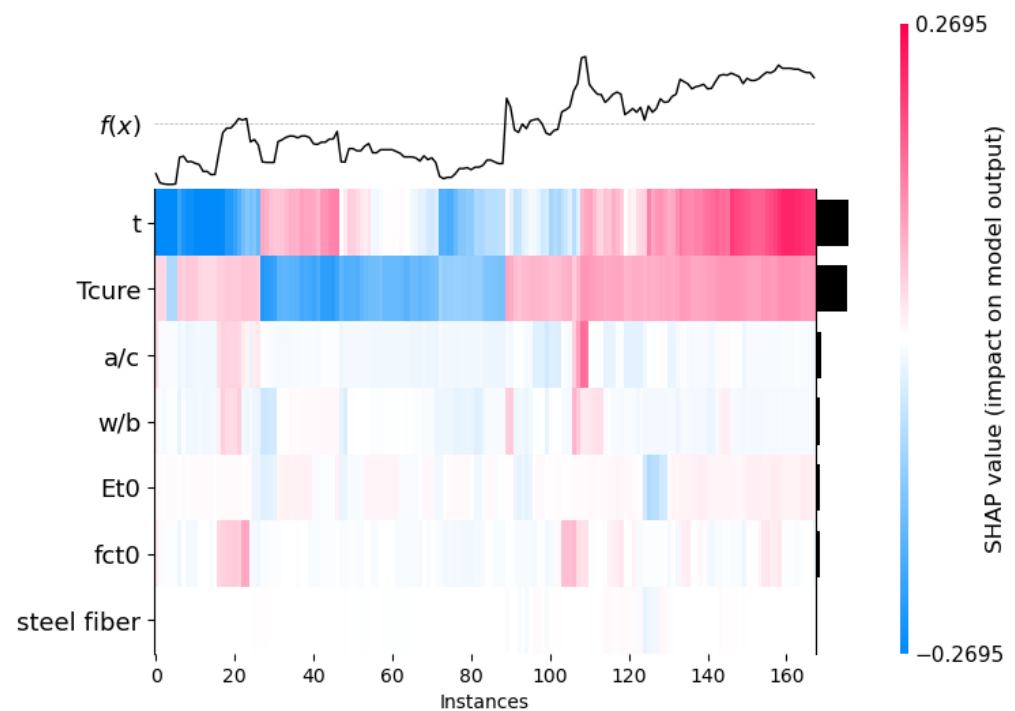
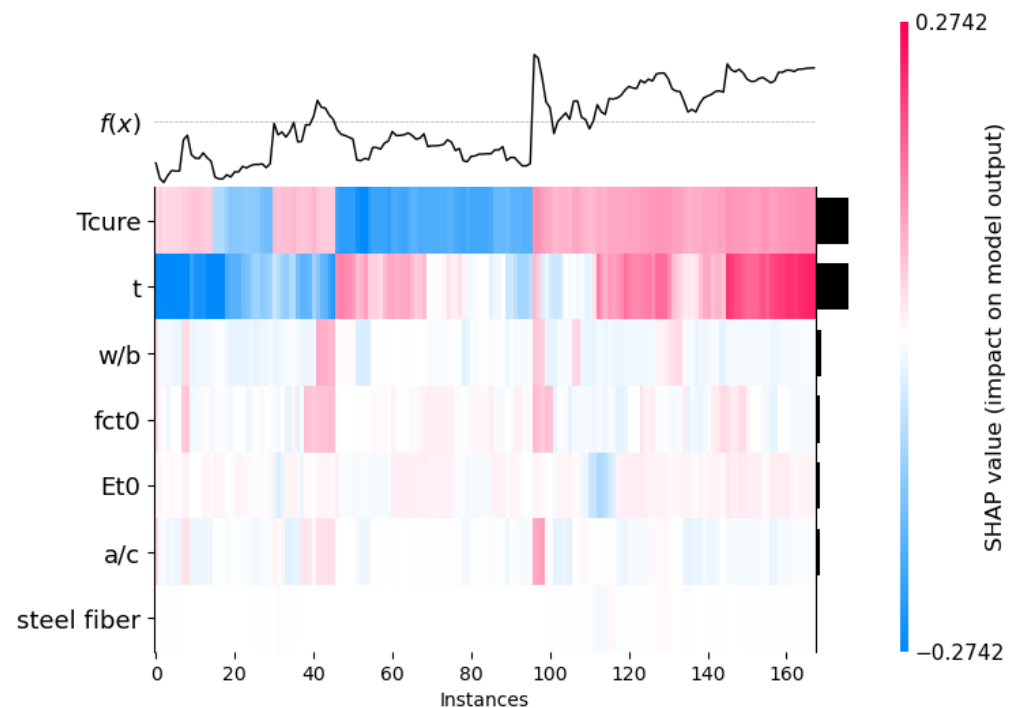


Figure 11. Global SHAP values based on XGBoost and LGBM models.



(a) XGBoost model



(b) LGBM model

Figure 12. SHAP heatmaps based on XGBoost and LGBM models.

The impact of the four most influential parameters on the creep coefficient was demonstrated by the SHAP results of the LGBM, as shown in Figure 13. In general, an increase in the curing temperature or compressive strength at loading age can result in a reduction in the predicted creep coefficient while the increase in the loading duration or water-to-binder ratio can result in an increase in the predicted creep coefficient. Furthermore, the effect of the interaction between the loading duration and other input features, including the water-to-binder ratio, aggregate-to-cement ratio, curing temperature, and steel fiber volume

content, on the UHPC creep based on the results of the SHAP feature dependence analysis was analyzed, as shown in Figure 14. A positive interaction was observed between the loading duration and water-to-binder ratio, while the interaction between the loading duration and aggregate-to-cement ratio, curing temperature, and steel fiber volume content was negative.

According to the local and global interpretation results, the influence of the input features on creep was consistent with current theories on the creep of UHPC, which proved the rationality of the established ML models.

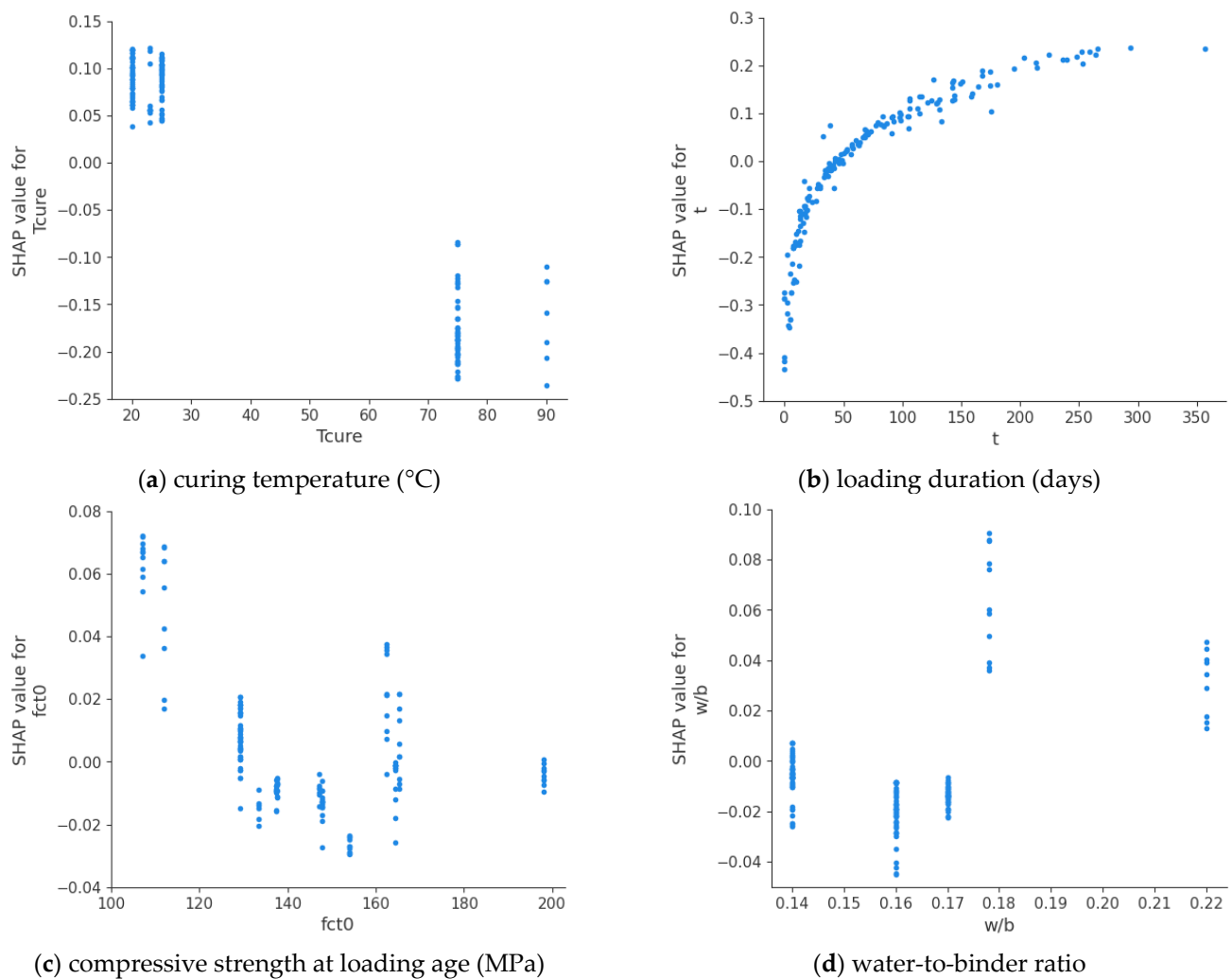


Figure 13. SHAP values of the most influential features based on the LGBM model.

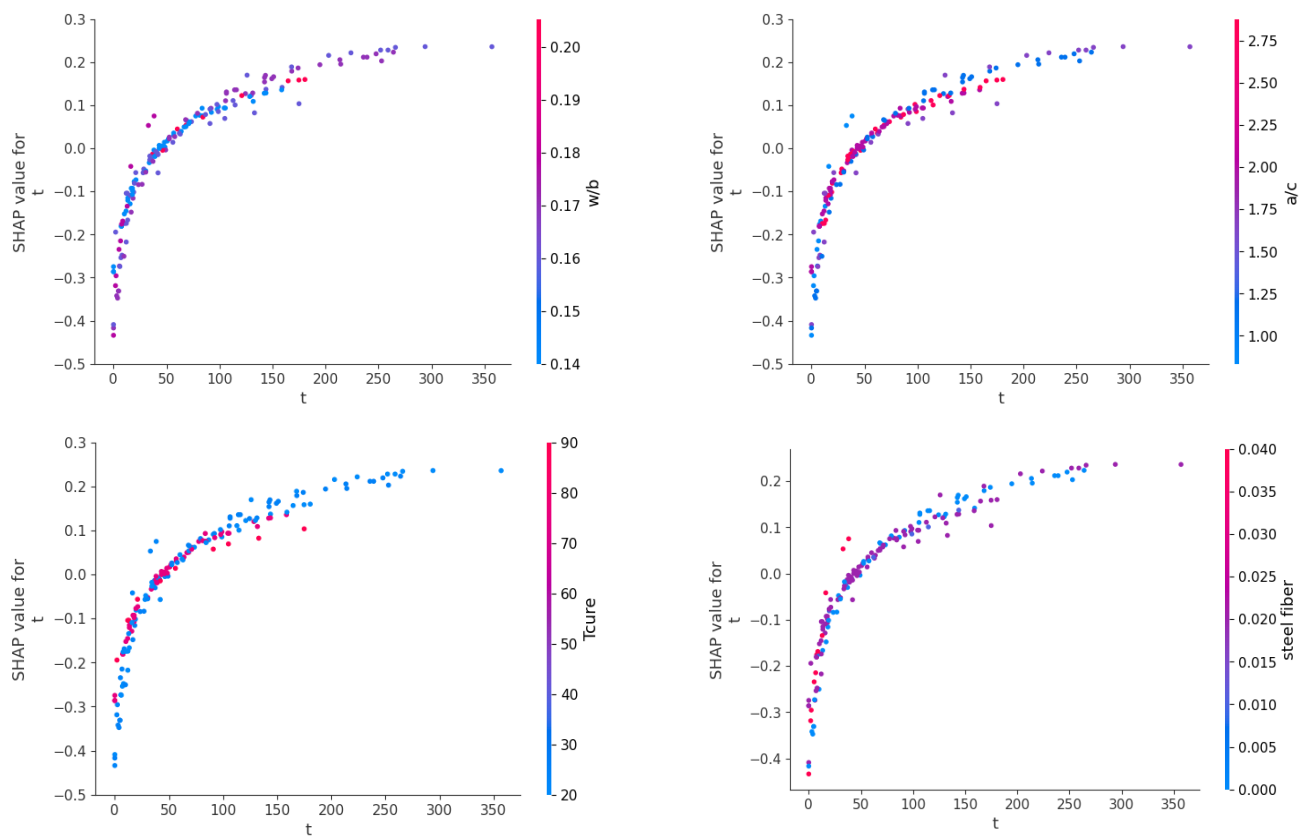


Figure 14. SHAP dependency and interaction plots based on the LGBM model.

4.3. Creep Curves

The LGBM model was selected to predict the creep of UHPC. Three different cases were selected, as shown in Table 5.

Table 5. Details of selected cases for exemplification.

Case	a/c (%)	w/b (%)	T_{cure} (°C)	f_{ct0} (MPa)	E_{t0} (MPa)	steel fiber (%)
C1	2.80	0.16	20	133.4	51,100	1
C2	1.65	0.16	20	147.8	48,000	2
C3	1.18	0.17	25	162.4	41,000	0

The predicted results were compared with the experimental results and the calculated results of MC 2010 [68], as shown in Figures 15 and 16. The prediction MC 2010 only predicted well in Case 1. The LGBM model exhibited a higher accuracy than MC 2010. The R^2 of the LGBM model is 1.01, 1.26, and 2.75 times that of MC 2010 for the three cases, respectively. The influence of the fiber content and curing temperature is not included in MC 2010, while it is considered in the ML models. As there were no missing values in the database established in this study, there was no need to fill the missing data, which might be one of the reasons for the great prediction accuracy.

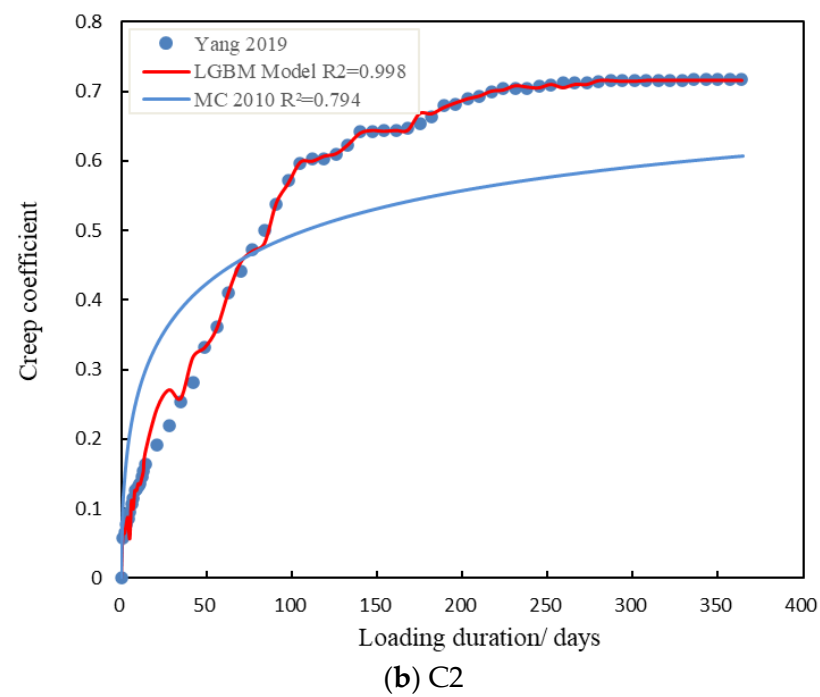
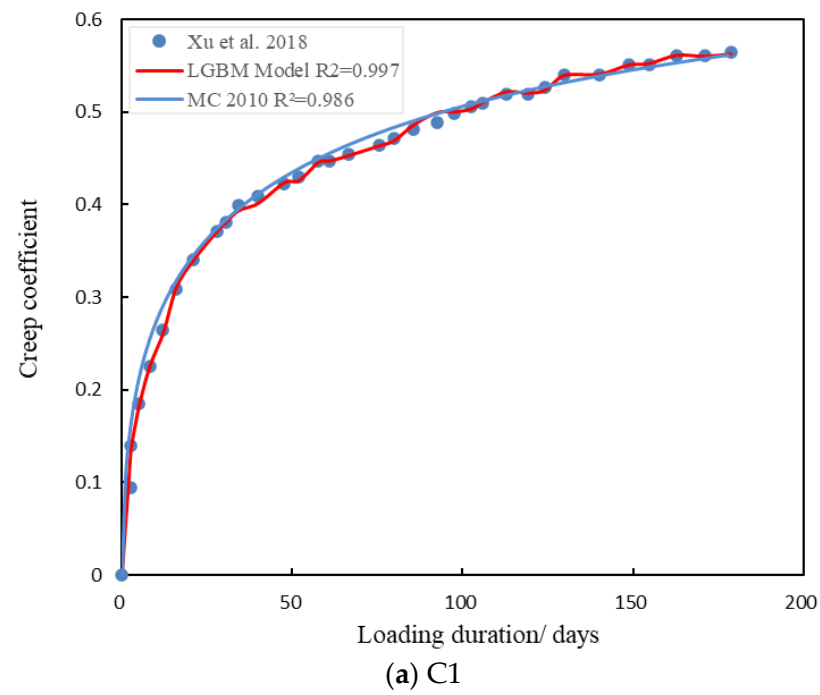


Figure 15. Cont.

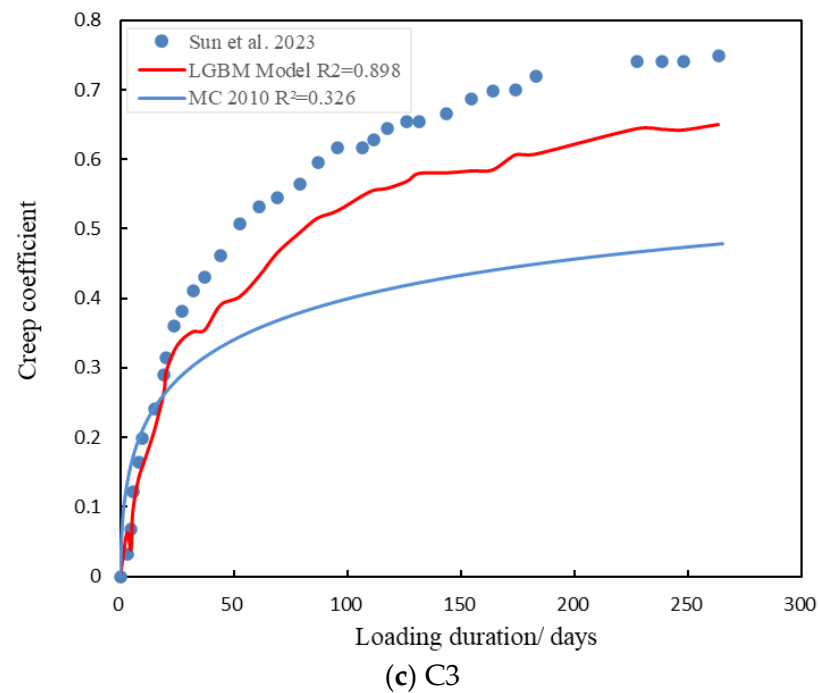


Figure 15. Creep curves from experiment, LGBM model, and MC 2010 [46,48,49].

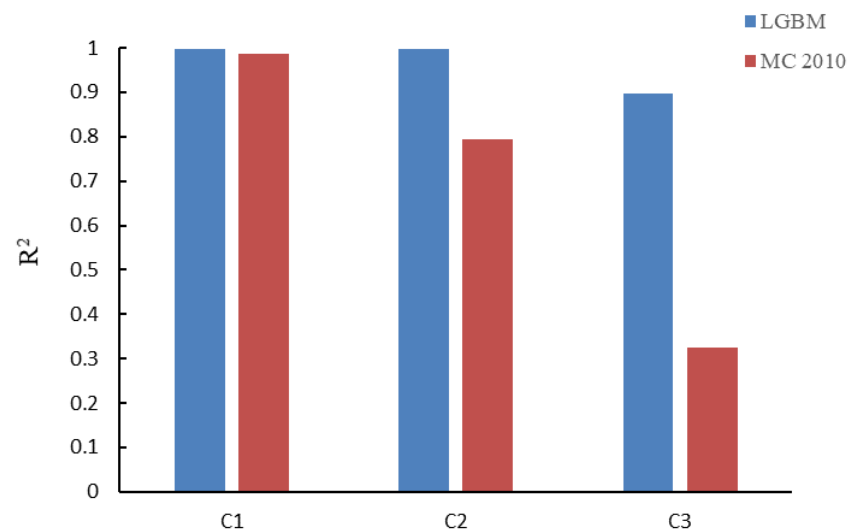


Figure 16. R^2 of LGBM model and MC 2010 for different cases.

5. Conclusions

UHPC creep prediction is a basis for the evaluation of the mechanical response of UHPC structures under sustained loads, and data-driven ML models for UHPC creep prediction are investigated. This study can address problems such as expensive costs, being time-consuming, discreteness of material, and geometry properties in experimental research and the lack of ability to handle complex and high-dimensional data with massive amounts in a numerical study. A UHPC creep database was established; firstly, the influencing parameters were identified, and the database was divided into training and testing datasets. Four ML models (RF, ANN, XGBoost, LGBM) were developed to predict the creep behavior of UHPC. The ML models were trained through 5-fold cross-validation and Bayesian optimization. Then, interpretability analysis was conducted on the models. As far as the authors' knowledge goes, there have been no studies on ML methods to predict UHPC creep behavior up to now. The following conclusions can be obtained:

- (1) All the four ML models developed exhibited high accuracy ($R^2 > 0.97$), and the LGBM model using the Bayesian optimization method and 5-fold cross-validation demonstrated the highest accuracy in the testing dataset. Ensemble learning models had a better predictive performance than a single learning model.
- (2) The ML models can be effectively interpreted by SHAP values both locally and globally. All the ML models arrived at the same conclusion regarding the top four most influential parameters for the UHPC creep coefficient: loading duration, curing temperature, compressive strength at loading age, and water-to-binder ratio.
- (3) The creep curves predicted by the LGBM model were compared with the experimental results and the calculated results of MC 2010, and the LGBM model developed showed a higher accuracy than MC 2010. The R^2 of the LGBM model is 1.01, 1.26, and 2.75 times that of MC 2010 for the three cases, respectively.
- (4) The size of the database is limited in this study due to limited data being available and more input features can be considered. The data cleaning procedure can be enhanced, and a user interface tool should be developed for the application of the ML models.

Author Contributions: Conceptualization, P.Z.; resources, P.Z.; formal analysis, W.C.; writing—original draft preparation, W.C.; writing—review and editing, P.Z. and Z.J.M.; validation, Y.W.; funding acquisition, Y.Z. and L.Z.; supervision, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The research was supported by the National Key R&D Program of China (2022YFC3801100) and the Opening Fund of Shaanxi Provincial Key Laboratory of Highway Bridge and Tunnel at Chang'an University, China (300102213524).

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shi, C.J.; Wang, D.H.; Wu, L.M.; Wu, Z.M. The hydration and microstructure of ultra high-strength concrete with cement-silica fume-slag binder. *Cem. Concr. Compos.* **2015**, *61*, 44–52. [\[CrossRef\]](#)
2. Park, S.H.; Kim, D.J.; Ryu, G.S.; Koh, K.T. Tensile behavior of Ultra High Performance Hybrid Fiber Reinforced Concrete. *Cem. Concr. Compos.* **2012**, *34*, 172–184. [\[CrossRef\]](#)
3. Bazant, Z.P.; Panula, L. Creep and shrinkage characterization for analyzing prestressed concrete structures. *J. Prestress. Concr. Inst.* **1980**, *25*, 86–122.
4. Huang, Y.; Wang, J.; Wei, Q.A.; Shang, H.; Liu, X. Creep behaviour of ultra-high-performance concrete (UHPC): A review. *J. Build. Eng.* **2023**, *69*, 106187. [\[CrossRef\]](#)
5. Liu, Y.; Wang, L.; Wei, Y.; Sun, C.; Xu, Y. Current research status of UHPC creep properties and the corresponding applications—A review. *Constr. Build. Mater.* **2024**, *416*, 135120. [\[CrossRef\]](#)
6. Zhu, Y.; Zhang, Y.; Hussein, H.H.; Xu, Z. Normal concrete and ultra-high-performance concrete shrinkage and creep models: Development and Application. *Adv. Struct. Eng.* **2022**, *25*, 2400–2412. [\[CrossRef\]](#)
7. Ding, Y.; Zeng, B.; Zhou, Z.; Wei, Y.; Huang, Y. Behavior of UHPC columns confined by high-strength transverse reinforcement under eccentric compression. *J. Build. Eng.* **2023**, *70*, 106352. [\[CrossRef\]](#)
8. Lu, K.; Du, L.; Zhang, F.; Zou, X.; Wang, J. Flexural Behavior of Ultra-High Performance Concrete (UHPC) Deck with Joints subjected to an Improved Steel Wire Mesh (SWM) Treatment. *KSCE J. Civ. Eng.* **2023**, *27*, 2163–2169. [\[CrossRef\]](#)
9. Mohebbi, A.; Graybeal, B.; Haber, Z. Time-Dependent Properties of Ultrahigh-Performance Concrete: Compressive Creep and Shrinkage. *J. Mater. Civ. Eng.* **2022**, *34*, 04022096. [\[CrossRef\]](#)
10. Sameer, S.; Fu, C.C.; Graybeal, B. Fabrication and experiment of ultra high performance concrete highway bridge girders. In *Proceedings of the 1st International Conference on Recent Advances in Concrete Technology*, Washington, DC, USA, 19–21 September 2007; p. 253.
11. Soliman, A.A.; Heard, W.F.; Williams, B.A.; Ranade, R. Effects of the tensile properties of UHPC on the bond behavior. *Constr. Build. Mater.* **2023**, *392*, 131990. [\[CrossRef\]](#)
12. Wei, Y.; Guo, W.; Ma, L.; Liu, Y.; Yang, B. Materials, Structure, and construction of a low-shrinkage UHPC overlay on concrete bridge deck. *Constr. Build. Mater.* **2023**, *406*, 133353. [\[CrossRef\]](#)
13. Xu, T.; Zhang, Z.; Liu, Z.; Bian, X.; Zhou, Y.; Deng, K. Linear and nonlinear creep of UHPC under compression: Experiments, modeling, and verification. *J. Build. Eng.* **2023**, *72*, 106566. [\[CrossRef\]](#)

14. Yuan, C.; Fu, W.; Raza, A.; Li, H. Study on Mechanical Properties and Mechanism of Recycled Brick Powder UHPC. *Buildings* **2022**, *12*, 1622. [\[CrossRef\]](#)
15. Zeng, X.; Zhu, S.; Deng, K.; Zhao, C.; Zhou, Y. Experimental and numerical study on cyclic behavior of a UHPC-RC composite pier. *Earthq. Eng. Eng. Vib.* **2023**, *22*, 731–745. [\[CrossRef\]](#)
16. Zhang, R.; Hu, P.; Chen, K.; Li, X.; Yang, X. Flexural Behavior of T-Shaped UHPC Beams with Varying Longitudinal Reinforcement Ratios. *Materials* **2021**, *14*, 5706. [\[CrossRef\]](#)
17. Ullah, R.; Qiang, Y.; Ahmad, J.; Vatin, N.I.; El-Shorbagy, M.A. Ultra-High-Performance Concrete (UHPC): A State-of-the-Art Review. *Materials* **2022**, *15*, 4131. [\[CrossRef\]](#) [\[PubMed\]](#)
18. de-Prado-Gil, J.; Palencia, C.; Silva-Monteiro, N.; Martinez-Garcia, R. To predict the compressive strength of self compacting concrete with recycled aggregates utilizing ensemble machine models. *Case Stud. Constr. Mater.* **2022**, *16*, e01046. [\[CrossRef\]](#)
19. Wang, Q.C.; Hussain, A.; Farooqi, M.U.; Deifalla, A.F. Artificial intelligence-based estimation of ultra-high-strength concrete's flexural property. *Case Stud. Constr. Mater.* **2022**, *17*, e01243. [\[CrossRef\]](#)
20. Xie, T.Y.; Yang, G.S.; Zhao, X.Y.; Xu, J.J.; Fang, C.F. A unified model for predicting the compressive strength of recycled aggregate concrete containing supplementary cementitious materials. *J. Clean. Prod.* **2020**, *251*, 119752. [\[CrossRef\]](#)
21. Xu, J.J.; Chen, Z.P.; Ozbakkaloglu, T.; Zhao, X.Y.; Demartino, C. A critical assessment of the compressive behavior of reinforced recycled aggregate concrete columns. *Eng. Struct.* **2018**, *161*, 161–175. [\[CrossRef\]](#)
22. Xu, S.L.; Wang, Q.M.; Lyu, Y.; Li, Q.H.; Reinhardt, H.W. Prediction of fracture parameters of concrete using an artificial neural network approach. *Eng. Fract. Mech.* **2021**, *258*, 108090. [\[CrossRef\]](#)
23. Nunez, I.; Marani, A.; Flah, M.; Nehdi, M.L. Estimating compressive strength of modern concrete mixtures using computational intelligence: A systematic review. *Constr. Build. Mater.* **2021**, *310*, 125279. [\[CrossRef\]](#)
24. Bal, L.; Buyle-Bodin, F. Artificial neural network for predicting creep of concrete. *Neural Comput. Appl.* **2014**, *25*, 1359–1367. [\[CrossRef\]](#)
25. Karthikeyan, J.; Upadhyay, A.; Bhandari, N.M. Artificial neural network for predicting creep and shrinkage of high performance concrete. *J. Adv. Concr. Technol.* **2008**, *6*, 135–142. [\[CrossRef\]](#)
26. Hodhod, O.A.; Said, T.E.; Ataya, A.M. Prediction of creep in concrete using genetic programming hybridized with ANN. *Comput. Concr.* **2018**, *21*, 513–523. [\[CrossRef\]](#)
27. Gandomi, A.H.; Sajedi, S.; Kiani, B.; Huang, Q. Genetic programming for experimental big data mining: A case study on concrete creep formulation. *Autom. Constr.* **2016**, *70*, 89–97. [\[CrossRef\]](#)
28. Feng, J.; Zhang, H.; Gao, K.; Liao, Y.; Gao, W.; Wu, G. Efficient creep prediction of recycled aggregate concrete via machine learning algorithms. *Constr. Build. Mater.* **2022**, *360*, 129497. [\[CrossRef\]](#)
29. Xiao, J.; Xu, X.; Fan, Y. Shrinkage and Creep of Recycled Aggregate Concrete and Their Prediction by ANN Method. *J. Build. Mater.* **2013**, *16*, 752–757.
30. Li, K.; Long, Y.P.; Wang, H.; Wang, Y.F. Modeling and Sensitivity Analysis of Concrete Creep with Machine Learning Methods. *J. Mater. Civ. Eng.* **2021**, *33*, 04021206. [\[CrossRef\]](#)
31. Ekanayake, I.U.; Meddage, D.P.P.; Rathnayake, U. A novel approach to explain the black-box nature of machine learning in compressive strength predictions of concrete using Shapley additive explanations (SHAP). *Case Stud. Constr. Mater.* **2022**, *16*, e01059. [\[CrossRef\]](#)
32. Tran, V.Q.; Mai, H.V.T.; To, Q.T.; Nguyen, M.H. Machine learning approach in investigating carbonation depth of concrete containing Fly ash. *Struct. Concr.* **2023**, *24*, 2145–2169. [\[CrossRef\]](#)
33. Zhang, S.; Xu, J.; Lai, T.; Yu, Y.; Xiong, W. Bond stress estimation of profiled steel-concrete in steel reinforced concrete composite structures using ensemble machine learning approaches. *Eng. Struct.* **2023**, *294*, 116725. [\[CrossRef\]](#)
34. Liu, T.; Huang, T.; Ou, J.; Xu, N.; Li, Y.; Ai, Y.; Xu, Z.; Bai, H. Modeling the load carrying capacity of corroded reinforced concrete compression bending members using explainable machine learning. *Mater. Today Commun.* **2023**, *36*, 106781. [\[CrossRef\]](#)
35. Alyousef, R.; Nassar, R.-U.-D.; Fawad, M.; Farooq, F.; Gamil, Y.; Najeh, T. Predicting the properties of concrete incorporating graphene nano platelets by experimental and machine learning approaches. *Case Stud. Constr. Mater.* **2024**, *20*, e03018. [\[CrossRef\]](#)
36. Wakjira, T.G.; Al-Hamrani, A.; Ebead, U.; Alnahhal, W. Shear capacity prediction of FRP-RC beams using single and ensemble Explainable Machine learning models. *Compos. Struct.* **2022**, *287*, 115381. [\[CrossRef\]](#)
37. Wakjira, T.G.; Ibrahim, M.; Ebead, U.; Alam, M.S. Explainable machine learning model and reliability analysis for flexural capacity prediction of RC beams strengthened in flexure with FRCM. *Eng. Struct.* **2022**, *255*, 113903. [\[CrossRef\]](#)
38. Liang, M.F.; Chang, Z.; Wan, Z.; Gan, Y.D.; Schlangen, E.; Savija, B. Interpretable Ensemble-Machine-Learning models for predicting creep behavior of concrete. *Cem. Concr. Compos.* **2022**, *125*, 104295. [\[CrossRef\]](#)
39. Feng, J.P.; Zhang, H.W.; Gao, K.; Liao, Y.C.; Yang, J.; Wu, G. A machine learning and game theory-based approach for predicting creep behavior of recycled aggregate concrete. *Case Stud. Constr. Mater.* **2022**, *17*, e01653. [\[CrossRef\]](#)
40. Wakjira, T.G.; Kutty, A.A.; Alam, M.S. A novel framework for developing environmentally sustainable and cost-effective ultra-high-performance concrete (UHPC) using advanced machine learning and multi-objective optimization techniques. *Constr. Build. Mater.* **2024**, *416*, 135114. [\[CrossRef\]](#)
41. Wakjira, T.G.; Alam, M.S. Peak and ultimate stress-strain model of confined ultra-high-performance concrete (UHPC) using hybrid machine learning model with conditional tabular generative adversarial network. *Appl. Soft Comput.* **2024**, *154*, 111353. [\[CrossRef\]](#)

42. Wakjira, T.G.; Abushanab, A.; Alam, M.S. Hybrid machine learning model and predictive equations for compressive stress-strain constitutive modelling of confined ultra-high-performance concrete (UHPC) with normal-strength steel and high-strength steel spirals. *Eng. Struct.* **2024**, *304*, 117633. [\[CrossRef\]](#)
43. Wakjira, T.G.; Alam, M.S. Performance-based seismic design of Ultra-High-Performance Concrete (UHPC) bridge columns with design example—Powered by explainable machine learning model. *Eng. Struct.* **2024**, *314*, 118346. [\[CrossRef\]](#)
44. Katlav, M.; Ergen, F. Improved forecasting of the compressive strength of ultra-high-performance concrete (UHPC) via the CatBoost model optimized with different algorithms. *Struct. Concr.* **2024**. [\[CrossRef\]](#)
45. Rossi, P.; Charron, J.P.; Bastien-Masse, M.; Tailhan, J.L.; Le Maou, F.; Ramanich, S. Tensile basic creep versus compressive basic creep at early ages: Comparison between normal strength concrete and a very high strength fibre reinforced concrete. *Mater. Struct.* **2014**, *47*, 1773–1785. [\[CrossRef\]](#)
46. Xu, Y.; Liu, J.P.; Liu, J.Z.; Zhang, P.; Zhang, Q.Q.; Jiang, L.H. Experimental studies and modeling of creep of UHPC. *Constr. Build. Mater.* **2018**, *175*, 643–652. [\[CrossRef\]](#)
47. Zhu, L.; Wang, J.J.; Li, X.; Zhao, G.Y.; Huo, X.J. Experimental and numerical study on creep and shrinkage effects of ultra high-performance concrete beam. *Compos. Part B-Eng.* **2020**, *184*, 107713. [\[CrossRef\]](#)
48. Yang, Y. *Study on the Effect of Curing Condition on Properties of Reactive Powder Concrete (RPC) with Recycled Powder*; Tongji University: Shanghai, China, 2019.
49. Sun, M.; Visintin, P.; Bennett, T. Basic and drying creep of ultra-high performance concrete. *Aust. J. Civ. Eng.* **2023**, 2197274. [\[CrossRef\]](#)
50. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [\[CrossRef\]](#)
51. Belgiu, M.; Dragut, L. Random forest in remote sensing: A review of applications and future directions. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 24–31. [\[CrossRef\]](#)
52. Movassagh, A.A.; Alzubi, J.A.; Gheisari, M.; Rahimi, M.; Mohan, S.; Abbasi, A.A.; Nabipour, N. Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 6017–6025. [\[CrossRef\]](#)
53. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#) [\[PubMed\]](#)
54. Jiang, X.; Pang, Y.; Li, X.; Pan, J.; Xie, Y. Deep neural networks with Elastic Rectified Linear Units for object recognition. *Neurocomputing* **2018**, *275*, 1132–1139. [\[CrossRef\]](#)
55. Xu, J.; Chen, Y.; Xie, T.; Zhao, X.; Xiong, B.; Chen, Z. Prediction of triaxial behavior of recycled aggregate concrete using multivariable regression and artificial neural network techniques. *Constr. Build. Mater.* **2019**, *226*, 534–554. [\[CrossRef\]](#)
56. Chen, T.Q.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
57. Ke, G.L.; Meng, Q.; Finley, T.; Wang, T.F.; Chen, W.; Ma, W.D.; Ye, Q.W.; Liu, T.Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.
58. Kang, M.C.; Yoo, D.Y.; Gupta, R. Machine learning-based prediction for compressive and flexural strengths of steel fiber-reinforced concrete. *Constr. Build. Mater.* **2021**, *266*, 121117. [\[CrossRef\]](#)
59. Sun, C.; Wang, K.; Liu, Q.; Wang, P.J.; Pan, F. Machine-Learning-Based Comprehensive Properties Prediction and Mixture Design Optimization of Ultra-High-Performance Concrete. *Sustainability* **2023**, *15*, 15338. [\[CrossRef\]](#)
60. Su, M.; Zhong, Q.Y.; Peng, H.; Li, S.F. Selected machine learning approaches for predicting the interfacial bond strength between FRPs and concrete. *Constr. Build. Mater.* **2021**, *270*, 121456. [\[CrossRef\]](#)
61. Nguyen, H.; Vu, T.; Vo, T.P.; Thai, H.T. Efficient machine learning models for prediction of concrete strengths. *Constr. Build. Mater.* **2021**, *266*, 120950. [\[CrossRef\]](#)
62. Shahriari, B.; Swersky, K.; Wang, Z.Y.; Adams, R.P.; de Freitas, N. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* **2016**, *104*, 148–175. [\[CrossRef\]](#)
63. Fushiki, T. Estimation of prediction error by using K-fold cross-validation. *Stat. Comput.* **2011**, *21*, 137–146. [\[CrossRef\]](#)
64. Alippi, C.; Roveri, M. Virtual k-fold cross validation: An effective method for accuracy assessment. In Proceedings of the World Congress on Computational Intelligence (WCCI 2010), Barcelona, Spain, 18–23 July 2010.
65. Cook, R.; Lapeyre, J.; Ma, H.Y.; Kumar, A. Prediction of Compressive Strength of Concrete: Critical Comparison of Performance of a Hybrid Machine Learning Model with Standalone Models. *J. Mater. Civ. Eng.* **2019**, *31*, 04019255. [\[CrossRef\]](#)
66. Brochu, E.; Cora, V.M.; De Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv* **2010**, arXiv:1012.2599.
67. Lundberg, S.M.; Lee, S.-I. A Unified Approach to Interpreting Model Predictions. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.
68. CEB-FIP Fib. *Model Code for Concrete Structures 2010*; Ernst & Sohn: Hoboken, NJ, USA, 2013; p. 402.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.