Gong and Zhang
Journal of Engineering and Applied Science (2023) 70:107
https://doi.org/10.1186/s44147-023-00274-w

Journal of Engineering and Applied Science

RESEARCH Open Access

Predict the compressive strength of ultra high-performance concrete by a hybrid method of machine learning

Nana Gong[1] and Naimin Zhang[1*]

*Correspondence: sunlight2017@163.com

[1] School of Artificial Intelligence, Hebei Oriental University, Langfang 065000, China

Abstract

Ultra-high performance concrete (UHPC) benefits the construction industry due to its improved flexibility, high workability, durability, and performance compared to normal concrete. Some investigators have conducted observed papers on the UHPC's mechanical properties for establishing a reliable analytical approach for calculating the compressive strength, tensile strength, slump, etc. However, most of these studies were performed with limited samples because of the UHPC's high cost. This study aims to predict the compressive strength (CS) of UHPC through hybrid machine-learning approaches. The model is included Adaptive-Network Fuzzy Inference System (ANFIS). Moreover, three meta-heuristic algorithms were employed to improve the developed model's accuracy, including the Generalized Normal Distribution Optimization, the COOT optimization algorithm, and the Honey Badger Algorithm. Several metrics were used to compare and assess the performance of the hybrid models in the framework of ANGN, ANCO, and ANHB. A comparison of the predicted and measured results generally shows that the proposed developed models can reasonably estimate the mechanical properties of UHPC. The results indicated that the ANHB model could estimate the CS of UHPC with the most suitable accuracy.

Keywords: Ultra high-performance concrete, Compressive strength, Adaptive-Network Fuzzy Inference System, Generalized normal distribution optimization, COOT optimization algorithm, Honey Badger Algorithm

Introduction

Ultra-high-performance concrete (UHPC) is a new development in concrete technology. UHPC is a durable cement-based composite with high tensile and compressive strength [1]. Improved mechanical properties increase the shear strength, flexural strength, and concrete structures' durability. UHPC is currently utilized in some concrete structures, typically containing precast waffle panels for bridge decks, precast/prestressed bridge girders, and connecting materials between precast concrete deck slabs and beams [2, 3]. In 2001, the USA began using UHPC for highway infrastructure. In addition, replacing normal concrete with UHPC saves materials and reduces labor costs and installation [4]. Regardless, the benefits have not been commonly identified due to the specific needs

regarding the material variables employed for producing UHPC blends and the UHPC's high cost [5].

But getting the right mixture for UHPC sampling is tedious and time-consuming. For this reason, artificial intelligence (AI) has replaced laboratory work to predict the mechanical properties of UHPC [6–8]. Machine learning (ML) algorithms have been widely utilized to assess estimative results that nearly match the experiment, like artificial neural networks (ANNs). Nevertheless, an investigation may contain a complete test matrix with many parameters, of which the majority show little assistance to the test results. These computer scientists for developing new selection algorithms according to data-driven models [9]. Demand for software calculating tools in estimating engineering components, systems, and materials continues to grow.

Therefore, ANN has appeared as one of the most popular software computational models successfully used in many engineering problems [10]. In general, ANN has been executed in pattern and character recognition prediction and approximation, classification, image processing, prediction, optimization, and control of corresponding issues. This has prompted investigators have been offered ANN models for solving many civil engineering problems [11–13]. Moreover, wide applications in modeling the ANN behavior of specific texture elements have been reported in several studies. Investigator interests have turned to using various ANN models to solve predictive building materials challenges in recent years consisting of concrete, steel, and composite [14–16].

Most of the issues corresponding to concrete properties, such as new properties and hardening, have been solved employed ANN models according to the collected observed dataset. In addition, the compressive strength (CS) estimation of concrete by the ANN model is a topic of continuous investigation. This has prompted investigators to use the ANN calculation to evaluate the CS of light-weight, normal-weight, and recycled concrete [17–19]. Other researchers have investigated different predictive models to explain high-performance concrete's compressive strength, employing various ML methods. Subsequently, the emergence of UHPC has caused the fundamental development of the ANN model toward prediction. Investigators have generated ANN to simulate the UHPC performance accurately [20, 21].

Awodiji et al. [22] trained a series of ANN models to examine the relationship between CS and the ratio of material mass to set an age for various hydrated lime-cement concrete. Kasperkiewicz et al. [23] used ANN to optimize silica, cement, fine and coarse aggregates in superplasticizer, and water high-performance concrete (HPC) regardless of data complexity, incompleteness, and consistency predicted an excellent mixing ratio. They showed a significant correlation between the observed actual and estimative values, and ANN models can be used to approximate optimal mixtures. Ghafari et al. [24] studied a backpropagation neural network (BPNN) implementation and statistical mixture designed for estimating the UHPC needed performance. They aim to use BPNN and statistical blend design for assessing the CS and consistency of UHPC in two various curing modes, including primarily wet and steam curing. The 53 concrete samples were designed according to a statistical mixture design sizing matrix, and the components that create the mixture were accepted as separated parameters in the BPNN model. The results showed that BPNN can predict CS and slump more accurately than the statistical mixed method.

Regardless, these black box models provide very little information about what happens during the process of the ANN calculator. Thus, when evaluating the performance of UHPC blends, resolving this equivocation will be the next step in driving the motion to deploy intelligent algorithms while proving it mathematically. Deep ML applications have indicated promising work when optimization strategies during the ANN training phase are exploited to iteratively choose parameters that affect the model's accuracy [22, 25]. Then the selected parameters can be used in ANN or any other intelligent regression algorithm to improve the accuracy of the prediction model while understanding the physical phenomenon behind these selections [26].

UHPC is a material with complex and nonlinear behavior that poses a challenge for modeling using conventional analytical techniques. However, using an Adaptive neuro-fuzzy inference system (ANFIS) can provide a solution for developing a predictive model for the compressive strength (CS) of UHPC. ANFIS can capture the intricate nonlinear relationships between the input variables, such as mixed design parameters, and the output variable, CS. The study aims to present ML models containing ANFIS to identify critical parameters affecting the accuracy of UHPC CS estimation. ANFIS uses a set of fuzzy rules to represent the mapping between the input and output variables. Comprehensive data of multi-parameter experimental results have been compiled from publicly available CS of UHPC analysis. In addition, when combined with the model, three innovative algorithms increase the accuracy of the prediction and reduce the error of the results. Algorithms that form a hybrid model by combining with the corresponding model include generalized normal distribution optimization (GNDO), COOT optimization algorithm (COA), and Honey Badger Algorithm (HBA). The hybrid models' framework consists of ANGN, ANCO, and ANHB. To evaluate the models, some metrics have been used to select the most appropriate model are discussed in the following sections.

Methods

Dataset

Table 1 shows the constitutive variables of the UHPC samples based on empirical tests from the published paper [27]. In Table 1, the minimum (Min), maximum (Max), average (Mean), and standard deviation (St. Dev.) of the variables are specified that the inputs are cement, silica fume, fly ash, sand, steel fiber, quartz powder, water, and admixture,

Table 1 The properties of data set components engaged in the modeling process

Components	*Units*	*Properties*			
		Max	**Min**	**Mean**	**St. Dev**
C	(Kg/m^3)	1600	383	879.707	331.28
SF/C	(%)	0.332	0	0.214	0.0848
FA/C	(%)	1.011	0	0.053	0.1323
S/C	(%)	4.699	0	1.447	1.1751
STF/C	(%)	0.447	0	0.0395	0.076
QP/C	(%)	0.937	0	0.0469	0.157
W/C	(%)	0.514	0.0375	0.238	0.0636
Ad/C	(%)	0.281	0	0.0385	0.0399
CS	(MPa)	240	95	152.223	31.603

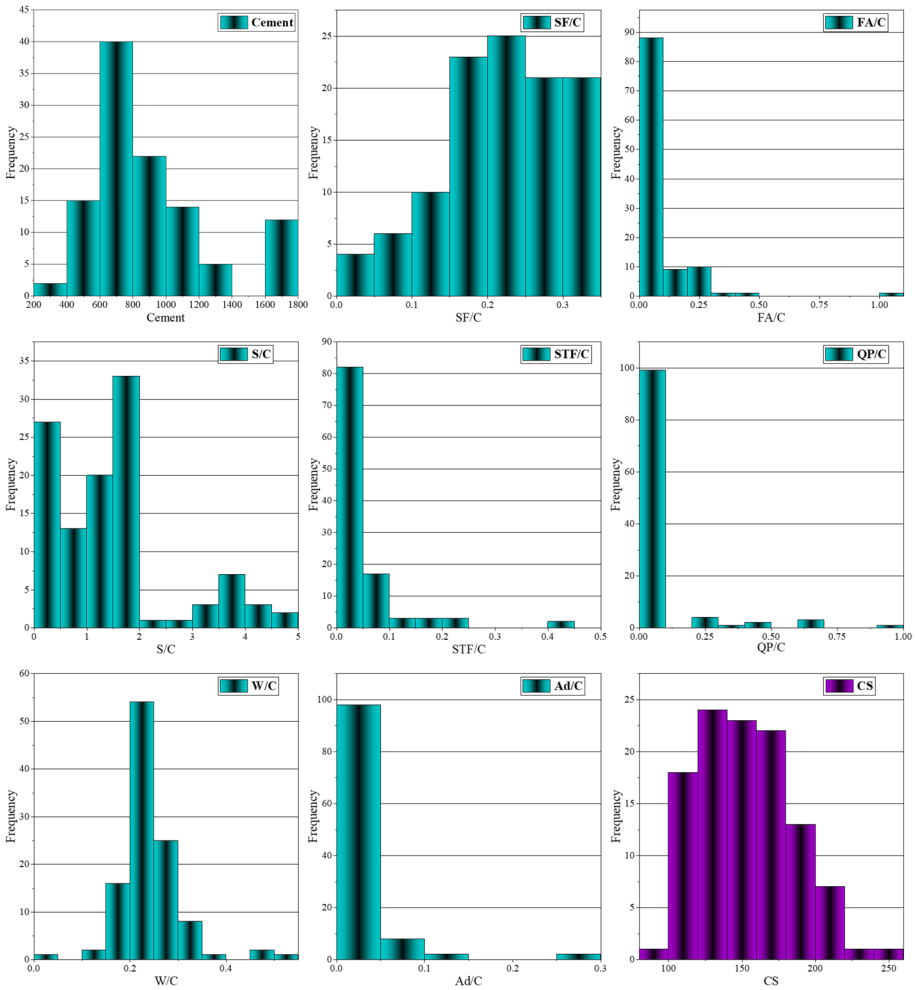

Fig. 1 The histogram for the input and output variables

Table 2 The correlation between the input and output variables

	C	SF/C	FA/C	S/C	STF/C	QP/C	W/C	Ad/C	CS
C	1	0.215	-0.313	-0.761	0.169	-0.084	-0.518	-0.154	-0.086
SF/C	0.215	1	-0.323	-0.418	0.274	0.096	-0.278	0.079	-0.118
FA/C	-0.313	-0.323	1	0.339	-0.197	-0.121	0.328	-0.100	-0.075
S/C	-0.761	-0.418	0.339	1	-0.215	-0.112	0.565	0.095	0.112
STF/C	0.169	0.274	-0.197	-0.215	1	0.313	-0.153	0.158	-0.092
QP/C	-0.084	0.096	-0.121	-0.112	0.313	1	0.012	0.425	-0.027
W/C	-0.518	-0.278	0.328	0.565	-0.153	0.012	1	0.111	0.058
Ad/C	-0.154	0.079	-0.100	0.095	0.158	0.425	0.111	1	-0.051
CS	-0.086	-0.118	-0.075	0.112	-0.092	-0.027	0.058	-0.051	1

and the output is compressive strength. In addition, the dataset contains 132 samples, of which 92 belonged to the training and 40 to the testing phase. Also, the distribution of the dataset is indicated in Fig. 1 [28].

Furthermore, Table 2 shows the correlation between the input and output variables. The values in the matrix indicate a negative correlation between the compressive strength (CS) of UHPC and variables such as C, SF/C, QP/C, and Ad/C. In contrast,

a positive correlation exists between CS and variables such as FA/C, S/C, STF/C, and W/C. Moreover, the correlation matrix reveals interesting interdependencies between some independent variables, including a robust negative correlation between C and S/C and a strong positive correlation between S/C and W/C.

Adaptive neuro-fuzzy inference system

A fuzzy set consists of elements with different membership levels. The degree of membership offers flexibility in modeling fuzzy collections [29]. Several inference approaches like Mamdani and Sugeno are created for fuzzy rule-based systems [30]. Distinguish the output of fuzzy rule from sharp function. In Sugeno's system, a typical representation of fuzzy rules is represented by when $x_1, x_2,$ andx_N are A_1, A_2, and A_N, alternatively, then $y = f(x)$ here $A_1, A_2, ...,$ andA_N represents fuzzy sets, and y represents the hash function. The result of each rule is a weighted average used to calculate the results of all the rules and a separate value in this technique. The explanation of a nonlinear map of the system is like a Kanno-type system (f_{FS}) can be defined as follows:

$$f_{FS} = \sum_{i}^{N} w_i f_i = \frac{\sum_i^N h_i f_i}{\sum_i^N h_i} \tag{1}$$

Here, N denotes the number of rules and h_i denotes the membership function of fuzzy collection. From ANFIS, membership functions have been repeatedly determined to produce the correct output. Many membership functions exist, such as Bell, trigonometric, trapezoidal, and Gaussian. The functions of Gaussian membership were employed in this analysis. The function of Gaussian used as

$$f(x, m, s) = e^{\frac{-(x-s)^2}{2m^2}} \tag{2}$$

In Eq. (2), s and m indicate the standard deviation and the dataset's mean. Training techniques are generally performed via two strategies containing hybrid learning algorithms and backpropagation in the ANFIS methodology (Appendix 1).

Generalized normal distribution optimization

GNDO inspired the theory of normal distribution [31]. A normal distribution can be determined by expecting a random variable x to follow a probability distribution with location parameter (μ) and scale parameter (δ). Its probability density function can be determined as follows:

$$f(x) = \frac{1}{\sqrt{2\pi\,\delta}}.\exp\left(-\frac{(x-\mu)^2}{2\delta}\right) \tag{3}$$

In Eq. (3), x indicates a normal random variable, and the normal distribution, μ and δ show the position of parameters and the scale parameter utilized to define the mean and standard variance of the random variable alternatively. Based on the relationship between the normal distribution and the distribution of individuals within the

population in Eq. (4), a generalized normal distribution model can be constructed for optimization:

$$V_i^t = \mu_i + \delta_i \times p, i = 1, 2, 3, \ldots, N \tag{4}$$

Here V_i^t represents the tracking vector of the $i-th$ individual at time t, μ_i represents the overall mean position of the i-th individual, δ_i represents the generalized standard variance, and p shows the indicative penalty coefficient. In addition, μ_i, δ_i, and p can be defined as follows:

$$\mu_i = \frac{1}{3}(x_i^t + x_{\text{best}}^t + a) \tag{5}$$

$$\delta_i = \sqrt{\frac{1}{3}[(x_i^t - \mu)^2 + (x_{\text{best}}^t - \mu)^2 + (a - \mu)^2]} \tag{6}$$

$$p = \begin{cases} \sqrt{-\log(\lambda_1)} \times \cos(2\pi\lambda_2), ifr \leq 1 \\ \sqrt{-\log(\lambda_1)} \times \cos(2\pi\lambda_2 + \pi), ifr > 1 \end{cases} \tag{7}$$

In the above equations, r, λ_1, and λ_2 represent random numbers between 0 and 1, x_{best}^t represents the current best position, and a represents the current average position of the population. Furthermore, the a is determined as

$$a = \frac{\sum_{i=1}^{N} x_i^t}{N} \tag{8}$$

Global Exploration finds promising regions in language regions around the world. GNDO's global scan assumes three randomly chosen people, which can be given in Eq. (9):

$$V_i^t = x_i^t + b \times (|\lambda_3| \times V_1) + (1 - b) \times (|\lambda_4| \times V_2) \tag{9}$$

In Eq. (9), λ_3 and λ_4 represent two random numbers that follow a normal distribution, b shows the adjustment parameter representing a random number between 0 and 1, and V_1 and V_2 represent two tracking vectors. Alternatively, V_1 and V_2 can be calculated as follows:

$$V_1 = \begin{cases} x_i^t - x_{p1}^t, \ if\ f\left(x_i^t\right) < f\left(x_{p1}^t\right) \\ x_{p1}^t - x_i^t, \qquad otherwise \end{cases} \tag{10}$$

$$V_2 = \begin{cases} x_{p2}^t - x_{p3}^t, \ \ if\ f\left(x_{p2}^t\right) < f\left(x_{p3}^t\right) \\ x_{p3}^t - x_{p2}^t, \qquad otherwise \end{cases} \tag{11}$$

Here $p1$, $p2$, and $p3$ show the three random integers selected from 1 to N.

COOT optimization algorithm

Coots are small aquatic birds of the family Rallidae. They form the genus Fulica, which in Latin means "coot". The algorithm begins with a first-order random population of

$x = \{x_1, x_2, \ldots, x_n\}$ [32]. A random population is often evaluated by the target function to determine the target values $V = \{V_1, V_2, \ldots, V_n\}$. The population is calculated in visible space as

$$P(i) = r(1, d) \times (ub - lb) + lb \tag{12}$$

where $P(i)$ is the coot position, d shows the dimension of the problem and ub, lb shows the upper and lower bounds of the search space.

Furthermore, the fitness of each solution must be computed using the objective function $Oi = f(x)$ after the initial population is generated and given the position of each agent. Choosing some coot to be the team leaders. To find a random position based on Eq. (13), move the coot to an arbitrary position in the search room.

$$G = r(1, d) \times (ub - lb) + lb \tag{13}$$

Coot motions explore different parts of the search distance. This movement takes the algorithm out of the local optimal point when the algorithm gets stuck in the local optimal point. The new position of the "coot" is calculated according to Eq. (14)

$$P(i) = P(i) + J \times r \times (G - P(i)) \tag{14}$$

In Eq. (14), r is a random number between 0 and 1, and J can be calculated as:

$$J = 1 - T \times \left(\frac{1}{Max_{Iter}}\right) \tag{15}$$

where T shows the current iteration and $\mathrm{Max}_{\mathrm{Iter}}$ shows the maximum iteration.

Chain development can be performed utilizing the average position of two coots. Another way to realize chain movement is that first calculate the distance vector between the two coots, then bring the larger coot closer together by about half the vector distance. Utilizing the primary strategy, the new position of the coot is calculated as follows:

$$P(i) = 0.5 \times (P(i - 1) + P(i)) \tag{16}$$

where $P(i - 1)$ shows the second coot.

The remaining coots may have to control their position and approach according to the group leader, and several coots manage the group in front of the group. The idea is to maintain its position depending on the Leader. The Leader's average position can be considered, and the coot can upgrade it according to this average position. Expecting mean position leads to premature convergence. Using a mechanism for the motion implementation as

$$I = 1 + (cN) \tag{17}$$

where I is the index number of the Leader, c is the current coot number, and N is the number of leaders.

Depending on the Leader's position, the coot should upgrade its position. The coot's next position, according to the chosen Leader, can be determined as follows:

$$P(i) = p + 2 \times r \times \cos(2\pi r_1) \times (l - P(i)) \tag{18}$$

where $P(i)$ indicates the coot's current position, p determines the chosen Leader's position, and r_1 shows a random number in the interval [− 1,1].

Groups need to be aligned themselves with their goals, so leaders need to update their positions on purpose. Equation (19) suggests upgrading the leader position as the formula finds a suitable location around the current sweet point. Managers must step away from their best fit to find the correct position. This equation provides a great way to get closer to or farther from the optimal position.

$$p = \begin{Bmatrix} K \times r \times \cos(2\pi r_1) \times (L - p) \; if & r < 0.5(a) \\ K \times r \times \cos(2\pi r_1) \times (L - p) - L \; if & r \geq 0.5(b) \end{Bmatrix} \tag{19}$$

where L indicates the best location found so far, and D can be determined as

$$K = 1 - T \times \left(\frac{1}{Max_{Iter}} \right) \tag{20}$$

In addition, the COA pseudo-code has been shown in Algorithm 1.

Honey Badger Algorithm

The HBA imitates the Honey Badger search method [33].

Initialize each position based on the number of badgers (N) as

$$x_i = lb_i + r_1 \times (ub_i - lb_i) \tag{21}$$

In Eq. (21), x_i shows the honey badger's ith position associated with the candidate solution for the N population, r_1 indicates a random number between 0 and 1, lb_i and ub_i determine the explore region's lower and upper bounds, alternatively.

Intensities included space between prey concentrations and prey and honey badgers. I_i indicates the intensity of the prey's odor. The motion is slow, and vice versa when the odor is strong, described by the inverse square law [34] expressed as

$$\begin{aligned} I_i &= r_2 \times \frac{s}{4\pi S_i^2} \\ s &= (x_i - x_{i+1})^2 \\ S_i &= x_{prey} - x_i \end{aligned} \tag{22}$$

where r_2 shows the random number between 0 and 1, s indicates the concentration's strength, S_i is the space between the ith badger and the prey.

The density factor manages time-varying randomness from Exploration to exploitation to allow a smooth transition. In addition, updating the density factor, which decreases with iteration, to account for randomness over time can be determined as follows:

$$a = C \times \exp\left(\frac{-t}{t_{\max}} \right) \tag{23}$$

Here, C indicates a constant ≥ 1 (default = 2) and $t_{\max}$ shows a maximum iteration number.

The output of the local optimal step and the position of the agent's actions are used to exit the locally optimal region. In the explore space, the developed algorithm exploits an indicator to change the direction of discovery to benefit from significant opportunities for tight roaming agents.

HBA position (x_{new}) update techniques are divided into two sections containing the "mining phase" and the "crypt phase." During the digging phase, a badger indicates an action similar to cardioid conformation [35]. Cardioid movement can be calculated as follows:

$$x_{new} = x_{prey} + e \times c \times S \times x_{prey} + e \times r_3 \times a \times d_i \times |\cos(2\pi r_4) \times [1 - \cos(2\pi r_5)]| \tag{24}$$

In Eq. (24), x_{prey} indicates the position of prey, this is the best position found so far, in other words, the best overall position. $c \geq 1$ (default = 6) is the ability of the badger to achieve food, r_3, r_4, and r_5 shows various random numbers between 0 and 1, and e acts as a explore direction change flag, which can be calculated as

$$\mathrm{I} = \begin{cases} 1 & if\ r_6 \leq 0.5 \\ -1 & else \end{cases} \tag{25}$$

During the digging phase, honey badgers strongly depend on prey odor intensity, space between prey and badger, and time-varying food-influence factors. Additionally, badgers can detect every F sound, making it easier to locate prey when foraging.

If the honey badger follows the honeyguide to achieve the hive, this is presented as:

$$x_{new} = x_{prey} + e \times r_7 \times a \times S_i \tag{26}$$

Algorithm 2 has indicated the HBA pseudo-code.

Performance evaluation methods

Evaluating the performance of the hybrid models during the training and testing sections is an essential step in ensuring that the model performs well against future unpublished datasets in terms of robustness, accuracy, and generalizability. Specifically, statistical metrics can be used to assess the ML model's error in estimating the target. This paper used the coefficient of determination (R^2), mean squared error (RMSE), the median of absolute percentage error (MDAPE), mean absolute error (MAE), and uncertainty 95% (U95) to assess the predictive accuracy of each model in the following:

$$R^2 = \left(\frac{\sum_{i=1}^{n} (p_i - \overline{p})(r_i - \overline{r})}{\sqrt{\left[\sum_{i=1}^{n} (p_i - \overline{p})^2\right]\left[\sum_{i=1}^{n} (r_i - \overline{r})^2\right]}} \right)^2 \tag{27}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (r_i - p_i)^2} \tag{28}$$

$$\mathrm{MAE} = \frac{1}{n} \sum_{i=1}^{n} |p_i - r_i| \tag{29}$$

$$\mathrm{MDAPE} = \mathrm{medain}\left(\left\|\frac{r_i - \overline{r}}{r_i}\right\| \times 100\right) \tag{30}$$

$$U95 = \frac{1.96}{n}\sqrt{\sum_{i=1}^{n}(r_i - p_i)^2 + \sum_{j=1}^{n}(r_j - p_j)^2} \tag{31}$$

In the above equations, n determines the sample number, r_i and p_i are actual and predicted values, $\overline{p}$ and $\overline{r}$ are the mean values of predicted and actual, alternatively.

Results and discussion

This section will be discussed the results obtained from the model in two parts containing training and testing, in which 70% of the sample evaluation involved the training phase and 30% were assigned to the test. In addition, the models are evaluated and compared with each other to choose the model with the highest accuracy and most minor error. The models were assessed by the evaluators introduced in "Performance evaluation methods" section. Table 3 has been shown the results obtained from the proposed models. The ideal values of the results in the evaluator are that except for R^2, the remaining metrics should get the lowest value and close to zero due to the indicating error of models. If the values obtained during the test phase are better than the training, it indicates that the learning of the samples has been done suitable in the training section, which shows the model's power.

In R^2, where values are specified as percentages, models should get values close to 100%. As shown in Table 3, the models obtained better values during the testing phase. Comparing between models, it can be seen that ANHB reached the highest value, equivalent to 99.58%, during the test phase, but not much different from the rest of the models. In RMSE, $\mathrm{ANHB_{train}} = 2.112$ (MPa) has the lowest value and weakest performance of both parts of ANGN, and the differences between ANHB with ANCO and ANGN were 29% and 43%, respectively. In MDAPE and MAE, the lowest values obtained during the ANHB test phase were equal to 1.153 and 1.845, respectively. Finally, for U95, the lowest value equivalent to 5.901 (MPa) was obtained for the $\mathrm{ANHB_{test}}$, which reached 29 and 43% differences with ANCO and ANGN, alternatively. In general, the strongest to the weakest performance of the evaluation of models in two phases are related to ANHB, ANCO, and ANGN, respectively.

Table 4 compares our present study and previously published articles that explored similar fields. It serves as a reference to assess the performance and workability of

Table 3 The results obtained from the proposed models

Models	Optimizers	Hybrid models	Section	Statistic evaluator				
				R^2	RMSE	MDAPE	MAE	U95
ANFIS	HBA	ANHB	Train	98.93	3.241	1.418	2.435	8.963
			Test	99.58	2.112	1.153	1.845	5.901
ANFIS	COA	ANCO	Train	97.32	5.148	1.677	3.449	14.21
			Test	99.20	2.973	1.589	2.669	8.271
ANFIS	GNDO	ANGN	Train	95.31	6.750	1.701	3.909	18.71
			Test	98.74	3.719	1.824	3.048	10.33

Table 4 Comparison of present study results with recently published articles with similar datasets

Work ID	Model	R^2	RMSE
Wu [36]	FDA-RBF	0.916	9
R. Abuodeh et al. [27]	BPFNN	0.8	-
Alabduljabbar et al. [37]	Gene expression	0.969	6.5
Present work	ANHB	0.9958	2.11

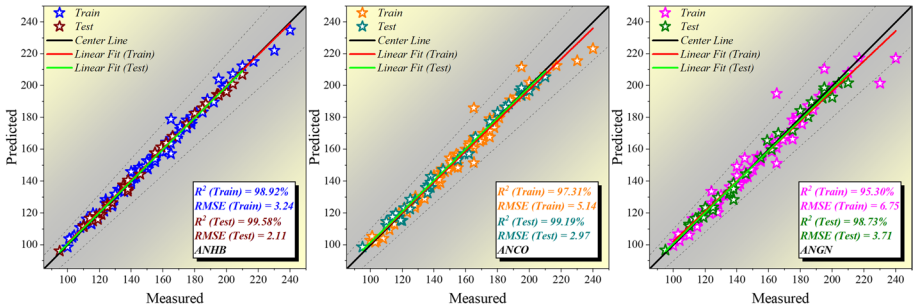

Fig. 2 The scatter plot in the training and testing phase of developed models

our developed hybrid model concerning recent studies. The results from the ANHB model demonstrate its superior ability to predict the compressive strength of UHPC compared to the other models studied.

Figure 2 shows the scatter plot in the developed models' training and testing phase. The corresponding figure is based on the R^2 and RMSE evaluators, which specify the dispersion and density of points. In addition, the center line is determined in $X = Y$ coordinates, and the angle between the linear fit and the center line indicates the performance of the models. The points related to ANHB are close to or on the center line, which is not observed in the overestimated or underestimated points. On the other hand, ANGN had more dispersion due to the high RMSE and low R^2, and the angle difference was high between the linear fit of ANGN with center compared to other models. In addition, the high density and accuracy of ANHB can be seen in Fig. 3, which shows the comparison between predicted and measured samples. ANHB had a low difference between the predicted and measured. The scatter of points in the training section in ANCO is more due to the low R^2 and high RMSE than the test.

Furthermore, it is possible to find a significant difference in some points of the training phase, but improving the performance in the test has minimized it. On the other hand, for ANGN, the dispersion of points in training is such that the points are over and underestimated, and as can be seen in Fig. 3, the difference between predicted and measured is higher than in other models. In general, it can be concluded that the ANHB model has been able to have high accuracy with the density of points and the slight difference between predicted and measured.

Figure 4 presents the scatter error plot for the developed models during the training and testing phases. In an ideal scenario, the error values should be close to zero, indicating accurate predictions. During the ANHB model's training phase, most predictions exhibited less than 5% errors, signifying its robust performance. However, a few samples,

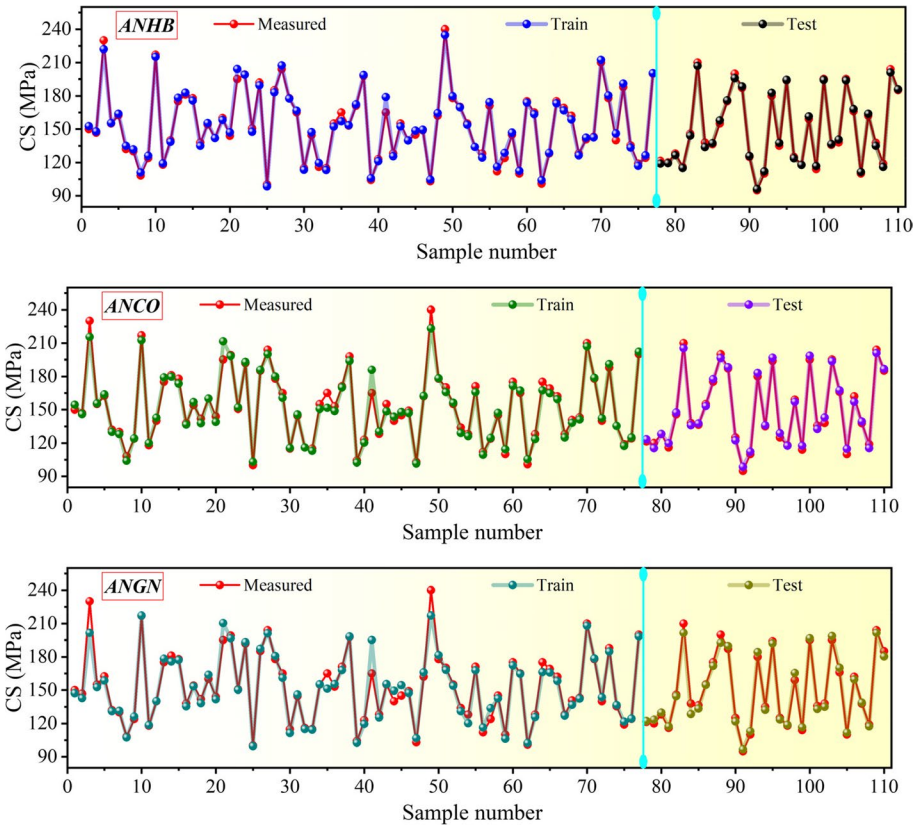

Fig. 3 The comparison between predicted and measured samples

like sample 42, exhibited increased dispersion and were identified as outlier data, as demonstrated in Fig. 5. During the testing phase, the ANHB model showed no particular distribution of errors, and most data points fell within the range of 0%. As a result, the mean error was nearly zero, demonstrating the model's capability to generalize well to unseen data. In the case of the ANCO model, the dispersion of errors increased, leading to the identification of four outliers in both negative and positive ranges. Despite this, the ANCO model significantly improved, reducing its error from 13% during the training phase to 5% during testing. This reduction in error showcased the model's ability to enhance its performance and better handle diverse datasets. Contrastingly, the ANGN model demonstrated higher error values than the other two models during the training phase, achieving an error rate of 18%. This higher error rate can be attributed to the presence of outlier data. Figure 5 highlighted outlier data points, further underscoring a performance weakness in the ANGN model. However, the ANGN model showed remarkable improvement during the testing phase, outperforming the other two. No outlier data was observed during this phase, and the error rate was reduced to 10%, demonstrating the model's adaptability and ability to overcome its initial limitations.

The scatter error plot provided valuable insights into the models' performance during training and testing. While the ANHB model performed well with some outlier data during training, it demonstrated robustness in testing. The ANCO model improved performance during testing, despite encountering increased dispersion during training. On the other hand, the ANGN model initially suffered from higher

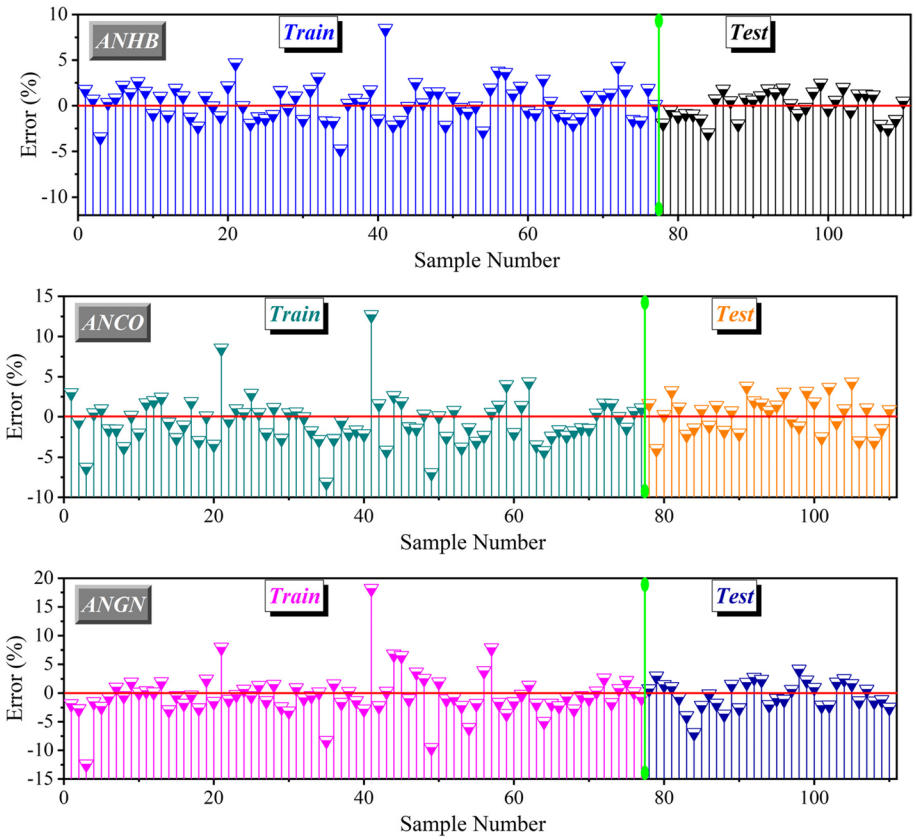

Fig. 4 The scatter error plot of presented models based on the training and testing phase

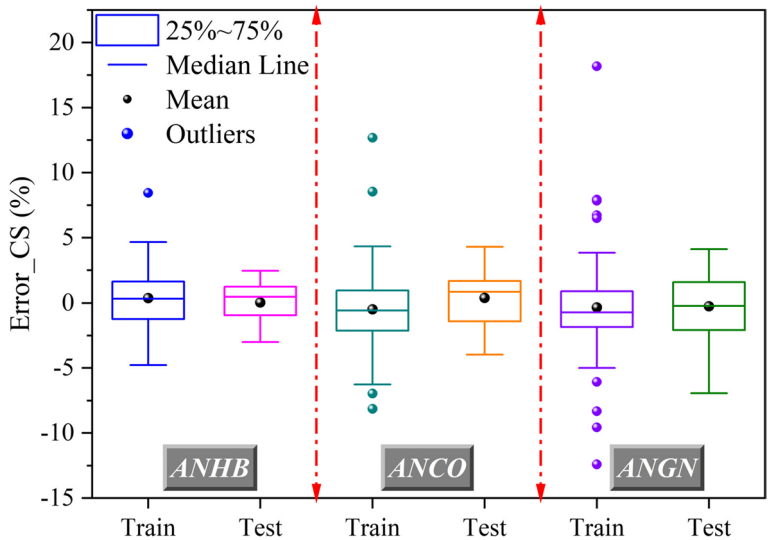

Fig. 5 The error box plot for developed models in the training and testing phase

errors and outlier data during training but exhibited remarkable improvement and outperformed the other two models during testing. These observations underscore the strengths and weaknesses of each model, guiding future refinements and optimizations for enhanced performance.

Applying the predictive model, comprising ANFIS with GNDO, COA, and HBA, offers practical engineering benefits for UHPC construction.

Summary of benefits:

1. Enhanced quality control: the model estimates UHPC's CS before testing, optimizing mix designs, and reducing material waste and costs.
2. Improved structural design: accurate CS predictions enable precise structural design, ensuring compliance with safety standards and regulations.
3. Cost and time savings: the model's predictions reduce the need for physical tests, saving time and resources during construction.
4. Early issue detection: early assessment of CS helps identify and address potential issues in mixed design and curing processes.
5. Optimal construction scheduling: accurate CS predictions facilitate efficient scheduling of construction activities for enhanced project management.
6. Risk mitigation: engineers can assess risks related to specific concrete batches or conditions, making informed decisions to avoid problems.
7. Research advancements: the model supports UHPC research by exploring the effects of different mixtures, additives, or curing methods on CS.

On the other vise, the developed model has several limitations:

1. It is computationally complex and requires significant processing power, making it challenging for real-time applications or low-resource environments.
2. The model's interpretability is reduced due to incorporating complex algorithms, which can be a concern in domains where transparency is essential.
3. While it performs well on the training data, its generalization to unseen data is limited, presenting a challenge for broader applications.
4. The model's success depends heavily on access to large, high-quality datasets, which can be resource-intensive and difficult to obtain in specific fields.
5. Integrating multiple algorithms introduces numerous hyperparameters that require careful tuning, making the optimization process time-consuming and hindering deployment.

Conclusions

Ultra-High-Performance Concrete (UHPC) is a new development in concrete technology. UHPC is a durable cement-based composite with high tensile and compressive strength. However, getting the right mixture for UHPC sampling is tedious and time-consuming. For this reason, artificial intelligence (AI) has replaced laboratory work to predict the mechanical properties of UHPC. This study aimed to forecast the CS of UHPC employing ANFIS with the most significant influential concrete mixing factors. The results of the research are the following:

- In R^2, the most appropriate value in the training and testing phase belonged to ANHB, which did not differ significantly from the other two models.
- The lowest RMSE value is related to ANHB, which had a difference of 29% and 43% with ANCO and ANGN, respectively.
- In MDAPE, the most appropriate value was obtained by ANHB equal to 1.153 MPa, which had a difference of 27 and 32% with ANCO and ANGN, respectively.

- In MAE and U95, the most appropriate values and other metrics belonged to ANHB and were equaled to 1.845 and 5.901 Mpa, alternatively.
- Machine learning models are reliable for predicting the mechanical properties of UHPC and can replace laboratory methods to save time and energy.

Appendix 1

Initialize the coots' first population randomly by Eq. (12)
Initialize the parameters of N with random positions.
Compute the fitness of coots and leaders
Compute J and K parameters based on Eq. (15) and Eq. (20)
if $r < 0.5$ **do**
for $i = 1$ to the coots' number, **do**
Calculate the I by Eq. (17)
if $r > 0.5$ **then**
Update the coots' position by Eq. (18)
else
if $r < 0.5$ **then**
Update the coots' position by Eq. (16)
else
Update the coots' position by Eq. (14)
end if
end for
Compute the coot's fitness
if $the\ coot's\ fitness \leq the\ leader\ coot$ **then**
For leaders' number
if r < 0.5
Update the Leader's position by Eq. (19)(a)
else
Update the Leader's position by Eq. (19)(b)
end if
end for
Stop criteria met.

Algorithm 1. COOT Optimization Algorithm

Set N, t_{max}, c, C parameters.

Initialize the population with random positions.

Assign and evaluate the goodness of fit of each honey badger location x_i employing the objective function $f_i, i \in [1, 2, \ldots, N]$

Save the best x_{prey} position and assign a fitness

while $t \leq t_{max}$ **do**

Update the decreasing factor employing Eq. (23).

for $i = 1$ to N **do**

Compute the intensity by Eq. (22)

if $r < 0.5$ **then**

Update the x_{new} by Eq. (24)

else

Update the x_{new} by Eq. (26)

end if

Evaluate new position and assign to new fitness

if $new\ fitness\ \leq f_i$ **then**

Set $f_i = new\ fitness$ and $x_i = x_{new}$

end if

end for

end while

Stop criteria met.

Algorithm 2. Honey Badger Algorithm

Abbreviations

UHPC	Ultra-High-performance concrete
ANFIS	Adaptive Network Fuzzy Inference System
COA	COOT optimization algorithm
R^2	Coefficient of correlation
MDAPE	Median absolute percentage error
S	Sand
FA	Fly ash
STS	Steel fiber
W	Water
CS	Compressive Strength
GNDO	Generalized Normal Distribution Optimization
HBA	Honey Badger Algorithm
MAE	Mean Absolute error
RMSE	Root mean square error
C	Cement
SF	Silica fume
QP	Quartz powder
AD	Admixture

Acknowledgements
I would like to take this opportunity to acknowledge that there are no individuals or organizations that require acknowledgment for their contributions to this work.

Authors' contributions
Nana GONG (NG)'s active and diligent contributions across methodology, software, validation, and formal analysis significantly elevated the quality and rigor of the research project. Naimin ZHANG (NZ) diverse contributions in writing, conceptualization, supervision, and project administration were instrumental in the successful execution of the research endeavor. All authors have read and approved the manuscript.

Funding
No funding.

Availability of data and materials
The authors do not have permissions to share data.

Declarations

Ethics approval and consent to participate
This option is not necessary due to that the data were collected from the references.

Competing interests
The authors declare that they have no competing of interests.
Research involving human participants and/or animals.
The observational study conducted on medical staff needs no ethical code. Therefore, the above study was not required to acquire ethical code.

Received: 6 June 2023 Accepted: 26 August 2023
Published online: 12 September 2023

References

1. Wille K, Naaman AE, El-Tawil S, Parra-Montesinos GJ (2012) Ultra-high performance concrete and fiber reinforced concrete: achieving strength and ductility without heat curing. Mater Struct 45:309–324
2. Graybeal B (2011) Ultra-high performance concrete
3. Russell HG, Graybeal BA, Russell HG (2013) Ultra-high performance concrete: a state-of-the-art report for the bridge community. Federal Highway Administration. Office of Infrastructure Research and Development, United States
4. Tang MC (2004) High performance concrete—past, present and future. Proc. Int. Symp. UHPC, Kassel, Ger. pp 3–9
5. Alsalman A, Dang CN, Prinz GS, Hale WM (2017) Evaluation of modulus of elasticity of ultra-high performance concrete. Constr Build Mater 153:918–928. https://doi.org/10.1016/j.conbuildmat.2017.07.158
6. Yin H, Liu S, Lu S, Nie W, Jia B (2021) Prediction of the compressive and tensile strength of HPC concrete with fly ash and micro-silica using hybrid algorithms. Adv Concr Constr 12:339–354
7. Huang L, Jiang W, Wang Y, Zhu Y, Afzal M (2022) Prediction of long-term compressive strength of concrete with admixtures using hybrid swarm-based algorithms. Smart Struct Syst 29:433–444
8. Nurlan Z (2022) A novel hybrid radial basis function method for predicting the fresh and hardened properties of self-compacting concrete. Adv Eng Intell Syst 1
9. Kamath MV, Prashanth S, Kumar M, Tantri A (2022) Machine-learning-algorithm to predict the high-performance concrete compressive strength using multiple data. J Eng Des Technol ahead-of-p. https://doi.org/10.1108/JEDT-11-2021-0637
10. Haykin S (2009) Neural networks and learning machines, 3/E. Pearson education India
11. Lu P, Chen S, Zheng Y (2012) Artificial Intelligence in Civil Engineering. Math Probl Eng 2012:1–22. https://doi.org/10.1155/2012/145974
12. Abdalla JA, Attom M, Hawileh R (2012) Artificial neural network prediction of factor of safety of slope stability of soils. Proc. 14th Int. Conf. Comput. Civ. Build. Eng. pp 27–9
13 Shaban WM, Elbaz K, Yang J, Shen S-L (2021) A multi-objective optimization algorithm for forecasting the compressive strength of RAC with pozzolanic materials. J Clean Prod 327:129355. https://doi.org/10.1016/j.jclepro.2021.129355
14. Abdalla JA, Hawileh RA (2010) Energy-based predictions of number of reversals to fatigue failure of steel bars using artificial neural network. 13th Int. Conf. Comput. Civ. Build. Eng., vol 108
15. Abdalla JA, Hawileh R (2011) Modeling and simulation of low-cycle fatigue life of steel reinforcing bars using artificial neural network. J Franklin Inst 348:1393–1403
16. Pujol JCF, Pinto JMA (2011) A neural network approach to fatigue life prediction. Int J Fatigue 33:313–322
17. Naderpour H, Rafiean AH, Fakharian P (2018) Compressive strength prediction of environmentally friendly concrete using artificial neural networks. J Build Eng 16:213–219
18. Heidari A, Hashempour M, Tavakoli D (2017) Using of backpropagation neural network in estimation of compressive strength of waste concrete. J Soft Comput Civ Eng 1:54–64
19. Sobhani J, Najimi M (2014) Numerical study on the feasibility of dynamic evolving neural-fuzzy inference system for approximation of compressive strength of dry-cast concrete. Appl Soft Comput 24:572–584
20. Shaban WM, Yang J, Elbaz K, Xie J, Li L (2021) Fuzzy-metaheuristic ensembles for predicting the compressive strength of brick aggregate concrete. Resour Conserv Recycl 169:105443. https://doi.org/10.1016/j.resconrec.2021.105443
21. Shaban WM, Elbaz K, Amin M, Ashour AG (2022) A new systematic firefly algorithm for forecasting the durability of reinforced recycled aggregate concrete. Front Struct Civ Eng 16:329–346. https://doi.org/10.1007/s11709-022-0801-9
22. Awodiji CTG, Onwuka DO, Okere C, Ibearugbulem O (2018) Anticipating the compressive strength of hydrated lime cement concrete using artificial neural network model. Civ Eng J 4:3005–3018
23. Kasperkiewicz J, Racz J, Dubrawski A (1995) HPC strength prediction using artificial neural network. J Comput Civ Eng 9:279–284
24. Ghafari E, Bandarabadi M, Costa H, Júlio E (2012) Design of UHPC using artificial neural networks. Brittle Matrix Compos. 10. Elsevier, pp 61–9
25. Cheng H, Kitchen S, Daniels G (2022) Novel hybrid radial based neural network model on predicting the compressive strength of long-term HPC concrete. Adv Eng Intell Syst 1

26. Masoumi F, Najjar-Ghabel S, Safarzadeh A, Sadaghat B (2020) Automatic calibration of the groundwater simulation model with high parameter dimensionality using sequential uncertainty fitting approach. Water Supply 20:3487–3501. https://doi.org/10.2166/ws.2020.241
27. Abuodeh OR, Abdalla JA, Hawileh RA (2020) Assessment of compressive strength of ultra-high performance concrete using deep machine learning techniques. Appl Soft Comput 95:106552
28 Shi C, Wu Z, Xiao J, Wang D, Huang Z, Fang Z (2015) A review on ultra high performance concrete: Part I. Raw materials and mixture design. Constr Build Mater 101:741–51
29 Peizhuang W (1983) Pattern recognition with fuzzy objective function algorithms (James C. Bezdek). Siam Rev 25:442
30. Bhattacharya M, Das A (n.d.) Identi ication and classi ication o tumor/cancer lesion appearing in brain using CT and MR images: study on adaptive neuro uzzy systems
31. Zhang Y, Jin Z, Mirjalili S (2020) Generalized normal distribution optimization and its applications in parameter extraction of photovoltaic models. Energy Convers Manag 224:113301
32. Naruei I, Keynia F (2021) A new optimization method based on COOT bird natural life model. Expert Syst Appl 183:115352
33. Hashim FA, Houssein EH, Hussain K, Mabrouk MS, Al-Atabany W (2022) Honey Badger Algorithm: new metaheuristic algorithm for solving optimization problems. Math Comput Simul 192:84–110
34. Kapner DJ, Cook TS, Adelberger EG, Gundlach JH, Heckel BR, Hoyle CD et al (2007) Tests of the gravitational inverse-square law below the dark-energy length scale. Phys Rev Lett 98:21101
35. Akopyan AV (2015) Geometry of the cardioid. Am Math Mon 122:144–150
36. Wu M (n.d.) Using the two optimization algorithms (BBO and FDA) coupling with radial basis neural network to estimate the compressive strength of high-ultra-performance concrete. J Intell Fuzzy Syst 1–11
37. Alabduljabbar H, Khan M, Awan HH, Eldin SM, Alyousef R, Mohamed AM (2023) Predicting ultra-high-performance concrete compressive strength using gene expression programming method. Case Stud Constr Mater 18:e02074

Publisher's Note