Smart Structures and Systems, *Vol. 34, No. 6 (2024) 377-406*
https://doi.org/10.12989/sss.2024.34.6.377

Predicting the mechanical properties of high-performance concrete implementing boosting models integrated with metaheuristic algorithms

Fangxiu Wang* and Jiemei Zhao

School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430023, Hubei, China

(Received March 7, 2024, Revised November 3, 2024, Accepted November 13, 2024)

Abstract. Compressive Strength (CS) and Tensile Strength (TS) are vital in designing a reinforced concrete structure. In fact, ensuring structural strength and safety requires both properties. Establishing predictive models for CS and TS yields high advantages, ensuring considerable cost savings by reducing labor-intensive and time-consuming lab experiments. This becomes more critical where high performance is necessary, such as in testing advanced HPC, renowned for its remarkable durability and strength in critical infrastructure and construction undertakings. The use of machine learning has become one of the innovative approaches to predicting these concrete characteristics. Through data-driven analyses on ingredient ratios, curing conditions, and environmental exposure, ML returns quite accurate CS and TS predictions. Applying ML methodologies results in gains in efficiency, cost economy, design refinement, higher-quality control, and increased safety. The contribution of this paper is the realization of an extended comparison between many algorithms. In particular, this work investigates two ML-based models: Histogram Gradient Boosting (HGB) and Light Gradient Boosting (LGB). These models are combined in a structured fashion with three newer optimization algorithms: Snake Optimization Algorithm (SO), Fox Optimization Algorithm (FO), and the Prairie Dog Optimization Algorithm (PDO), along with an ensemble of all three optimizers, namely SO-FO-PDO. From the results represented by the R^2 values, it is obvious that the HGPD model demonstrated far better forecasting performance for the CS, with a resultant R^2 value of 0.9961 during training. Similar to TS, the HGPD model developed as the best estimator in the case of TS with an R^2 value of 0.9947 for the training phase. Besides, from the statistical measures of accuracy w.r.t MAE and RMSE, it has been quite evident that the proposed PDO-based hybrid and ensemble models outperformed the rest by a long margin in estimating concrete mechanical properties.

Keywords: compressive and tensile strength; high-performance concrete; light gradient and histogram gradient boosting; meta-heuristic algorithms; sensitivity analyses

1. Introduction

1.1 Background

Concrete is considered the most consumed material in the building industry. Carbon dioxide is emitted from using fossil fuels and natural resources in cement production (Jafari *et al.* 2023, Qian *et al.* 2018, Zhang *et al.* 2020). The total volume of CO_2 from the abovementioned process poses a danger to this earth's environment. Various researchers have taken a keen interest in finding a solution by using alternative cohesive elements in building concrete with cement (Akbar and Liew 2020, Khalil *et al.* 2021). More recently, alternative materials have been incorporated, such as FA, MS, nano-silicate, etc., which increase concretes' sustainability regarding the reduction of carbon dioxide emissions that correlate with the consumption of by-product materials (Wang *et al.* 2020).

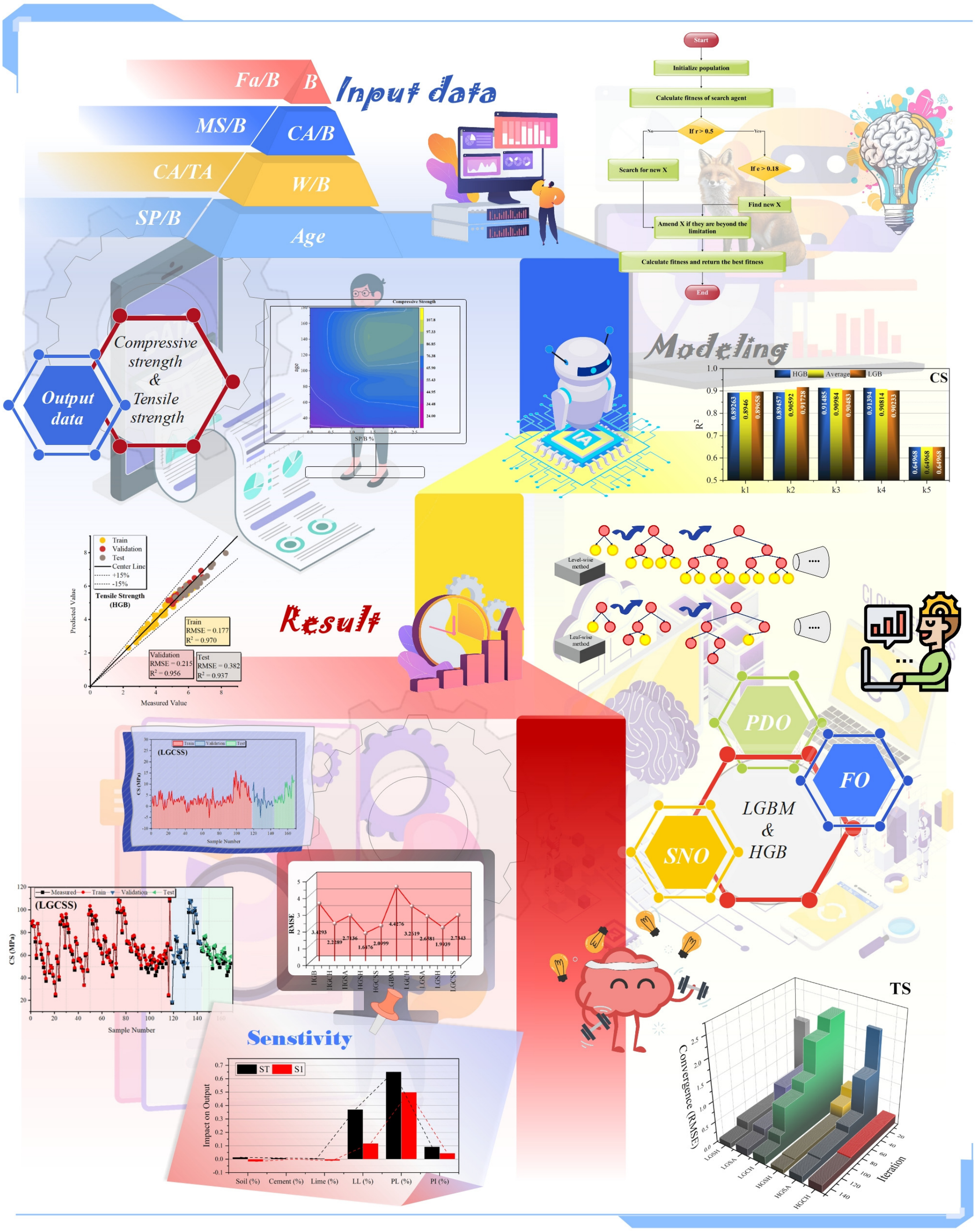
One homogenous concrete is HPC, which has great durability, workability, and strength and has formed from water, high-quality cement, coarse aggregate, and a fine blend of active components. Numerous concrete constructions used HPC, including bridges, skyscrapers, and components. Concrete may be more durable and have a longer structural life by having smaller, lighter, and more minimally constructed concrete buildings (Akbarzadeh *et al.* 2023, Huang *et al.* 2022, Khajeh *et al.* 2021).

To assist the provision of HPC, fibroid materials, mineral additives, and chemical additives can be added to the concrete. The particle size and form of FA are comparable to those of Portland cement. These parallels tend to reduce the amount of water in concrete construction. FA is primarily categorized into classes F and C, with class C's chemical composition showing the most remarkable variances. FA is frequently utilized as a powder ingredient in concrete structures instead of concrete binders. When suitable admixtures, such as superplasticizers, are added, FA is used most effectively to improve the workability, durability, and strength qualities of HPC is the outcome of this optimal FA usage (Algaifi *et al.* 2021, Geng *et al.* 2021, Hu *et al.* 2022). Contrarily, using FA in concrete mix design helps reduce concrete's cost, permeability, sulfate assault, and vulnerability in complex environments. The stiffness of the concrete structure, which is directly influenced by the reduction of permeability, directly influences the increase in concrete strength, particularly the compressive and tensile

---

*Corresponding author, Ph.D., Professor,
E-mail: wfx323@126.com

ISSN: 1738-1584 (Print), 1738-1991 (Online)



Graphical abstract

strength (CS and TS). The partial substitution of FA with binder materials is consolidated by the technological, environmental, and advantageous effects of FA inclusion in the concrete mix design (Ghanbari *et al.* 2023, Hansen 1990, Tavana Amlashi *et al.* 2023)**.**

MS is a filler, and a pozzolan in the concrete mix is another additional admixture because of its tiny particle size. It has been demonstrated that boosting the volume ratio of MS in the concrete mixture improves the concrete's mechanical qualities as it matures. In reinforced structures, it is generally acknowledged that a specific range of MS may significantly increase CS and TS, minimize permeability, and prevent corrosion. According to several experts, the main effects of MS on mechanical qualities happen between 3 and 28 days. Despite this, an ideal MS concentration in the mixture has not yet been documented (Detwiler *et al.* 1996, Haeri *et al.* 2020, Hubertova and Hela 2007, Turk *et al.* 2010). Reza *et al.* (2022) investigated the tensile and flexural properties of the HPC reinforced with micro-carbon fibers (MCF). Due to the inclusion of MS, MCF incorporation efficiency increased dramatically. The modulus of rupture increased with and without MS by 32% and 6%, respectively, when MCF was included at the optimal dosage (0.5-0.75% volume percent). The pozzolanic and micro-filler impact of MS particles improved the binding strength and effectiveness of MCF microfilaments. Under flexural stress, MS addition prevented the fiber-reinforced HPC's fracture.

1.2 Literature review

ML concepts have been employed in different fields for several years to predict various characteristics successfully. Similarly, the civil engineering industry is adopting such methods to overcome cumbersome observed techniques (Sadaghat *et al.* 2024, Geng *et al.* 2021, Sadaghat *et al.* 2024). These methods contain genetic engineering programming (GEP) (Chen 2003), multivariate adaptive regression splines (MARS) (Gholampour *et al.* 2020), Artificial Neural Networks (ANN) (Basheer and Hajmeer 2000), Support Vector Machines (SVM) (Wang 2005), Decision Trees (DT) (Brodley and Utgoff 1995), Adaptive Boosting Algorithm (ABA) (Margineantu and Dietterich 1997), and Adaptive Neuro-Fuzzy Inference (ANFIS) (Chang and Chang 2006). For instance, Javed *et al.* (2020) used GEP to predict the axial behavior of concrete-filled steel pipes (CFSTs) with 227 data points, and in other research, Farjad *et al.* (Iqbal *et al.* 2020), Nour *et al.* (Nour and Güneyisi 2019), and Azim *et al.* (2021) utilized GPE in the case of other typed of the concrete.

Considering other applied ML methods, Pala *et al.* (2007) investigated the effects of MS and FA on the CS of concrete. Comprehensive experiments were performed to analyze the effects of different W/C ratios and MS and FA percentages on concrete performance. Furthermore, ANN was employed to show the effect of additives on concrete strength parameters. Memarzadeh *et al.* (2022) studied the ANN method to assess the concrete's CS. ANN has been shown to reflect observed values accurately. Therefore, it has proven to be an excellent forecasting strategy. Shahmansouri *et al.* (2021) used ANN to estimate the CS of geopolymer concrete containing natural zeolite and silica fume. Thus, ANN determined suitable relationships and provided sustained accuracy in predicting geopolymer concrete. Keshavarz and Torkian *et al.* (Keshavarz and Torkian 2018) use ANN and ANFIS models to estimate concrete CS. Javed *et al.* (2020) predict the CS of sugarcane bagasse ash concrete by conducting observed and literature-based studies. The remaining data were collected from the published literature, and experimental work was conducted to validate the model. Zhang *et al.* (2019) have modeled the CS of self-compacting concrete (SCC), operating a random forest algorithm based on beetle antenna searches. Han *et al.* (2019) used a random forest method to estimate the CS of HPC. Sun *et al.* (2019) utilized a random forest algorithm developed on 138 dataset samples to estimate the CS of rubberized concrete.

1.3 Objective

Previous scholarly investigations have predominantly centered on single and hybrid models, encompassing various methodologies (Erdal *et al.* 2013, Kaloop *et al.* 2020, Kamath *et al.* 2022, Li and Song 2022). Nevertheless, it is important to acknowledge that, to the author's best knowledge, there exists a conspicuous gap in the existing literature concerning research initiatives that specifically tackle the utilization of ensemble learning methodologies with HGB and LGB in conjunction with three innovative optimization algorithms, for the prediction of HPC strengths. In this research, a combination of soft computing techniques, namely the HGB and LGB models, was utilized to predict the CS and TS of HPC. Further enhancement was achieved by integrating optimization algorithms, including SO, FO, PDO, and SO-FO-PDO ensemble algorithms. After this investigation, two primary goals were successfully attained. Initially, predictive models were formulated and trained using a range of input parameters relevant to the HPC mixture. These models underwent evaluation utilizing metrics including R^2, RMSE, MAE, NMSE, and MAPE. Subsequently, a comparative analysis was conducted to gauge the efficacy of the developed models, demonstrating the enhanced accuracy of the novel ensemble method compared to conventional single and hybrid models in forecasting HPC strengths. This validation underscores the practical utility of the developed models within civil engineering applications.

2. Dataset description

Drawing from an experimentally derived dataset comprising 56 data records for TS and 168 for CS, sourced from a published paper, the study aims to evaluate the influence of FA, MS, and superplasticizers on the TS and CS of concrete at different ages (as detailed in Table 1 and Table 2). It is recommended that the collected database be partitioned into separate training and testing datasets to facilitate analysis. Eight potential variables, including age, B, CA/B, CA/TA, FA/B, MS/B, SP/B, and W/B, were identified as factors influencing the CS and TS of concrete, determined as input variables (Lam *et al.* 1998). Tables 1 and 2 outline the statistical attributes of the input and output variables utilized in constructing the models for CS and TS, respectively.

Enhancing comprehension, Fig. 1 elucidates the statistical attributes of both input and output variables through contour plots. Upon reviewing Table 1 alongside Fig. 1, it is observed that variable B ranges from a minimum of 394 Kg to a maximum of 500 Kg, with FA/B% averaging at 0.241 and a standard deviation of 0.148.

Notably, the target variable, compressive strength, spans from a minimum of 24 MPa to a maximum of 107.8 MPa, with an average of 64.03 MPa and a standard deviation of 15.35. Similarly, Table 2 and Fig. 1 facilitate a parallel examination, depicting the same metrics as Table 1, with the target variable being TS in MPa. Detailed statistical properties of each indicator and the target variable are presented. Table 2 and Fig. 1 reveal that variable B ranges from a minimum of 394 Kg to a maximum of 500 Kg, with FA/B% averaging at 0.237 and a standard deviation of 0.127. The target variable, tensile strength, displays a minimum value of 2.32 MPa, a maximum of 8.28 MPa, an average of 5.199 MPa, and a standard deviation of 1.154. These insights offer the statistical characteristics' comprehensive overview of the TS dataset.

Table 1 The variables of the dataset used for models' input of compressive strength

Variables	Category	Unit	Statistical properties			
			Max.	Min.	Avg.	St. Dev.
B	Input	(Kg/m^3)	500	394	418.38	40.136
FA/B	Input	(%)	0.55	0	0.241	0.148
MS/B	Input	(%)	0.109	0	0.0542	0.045
CA/B	Input	(%)	2.90	2.172	2.736	0.279
CA/TA	Input	(%)	0.678	0.6	0.628	0.020
W/B	Input	(%)	0.5	0.3	0.4	0.081
SP/B	Input	(%)	2.6	0	1.033	0.624
age	Input	(Day)	180	28	68.57	49.99
CS	Output	(MPa)	107.8	24	64.03	15.35

Table 2 The variables of the dataset used for models' input of tensile strength

Variables	Category	Unit	Statistical properties			
			Max.	Min.	Avg.	St. Dev.
B	Input	(Kg/m^3)	500	394	411.06	35.443
FA/B	Input	(%)	0.55	0	0.237	0.127
MS/B	Input	(%)	0.109	0	0.068	0.044
CA/B	Input	(%)	2.906	2.172	2.787	0.247
CA/TA	Input	(%)	0.679	0.6	0.624	0.018
W/B	Input	(%)	0.5	0.3	0.4	0.082
SP/B	Input	(%)	2.6	0	1.023	0.523
age	Input	(Day)	56	28	42	14.058
CS	Output	(MPa)	8.28	2.32	5.199	1.154

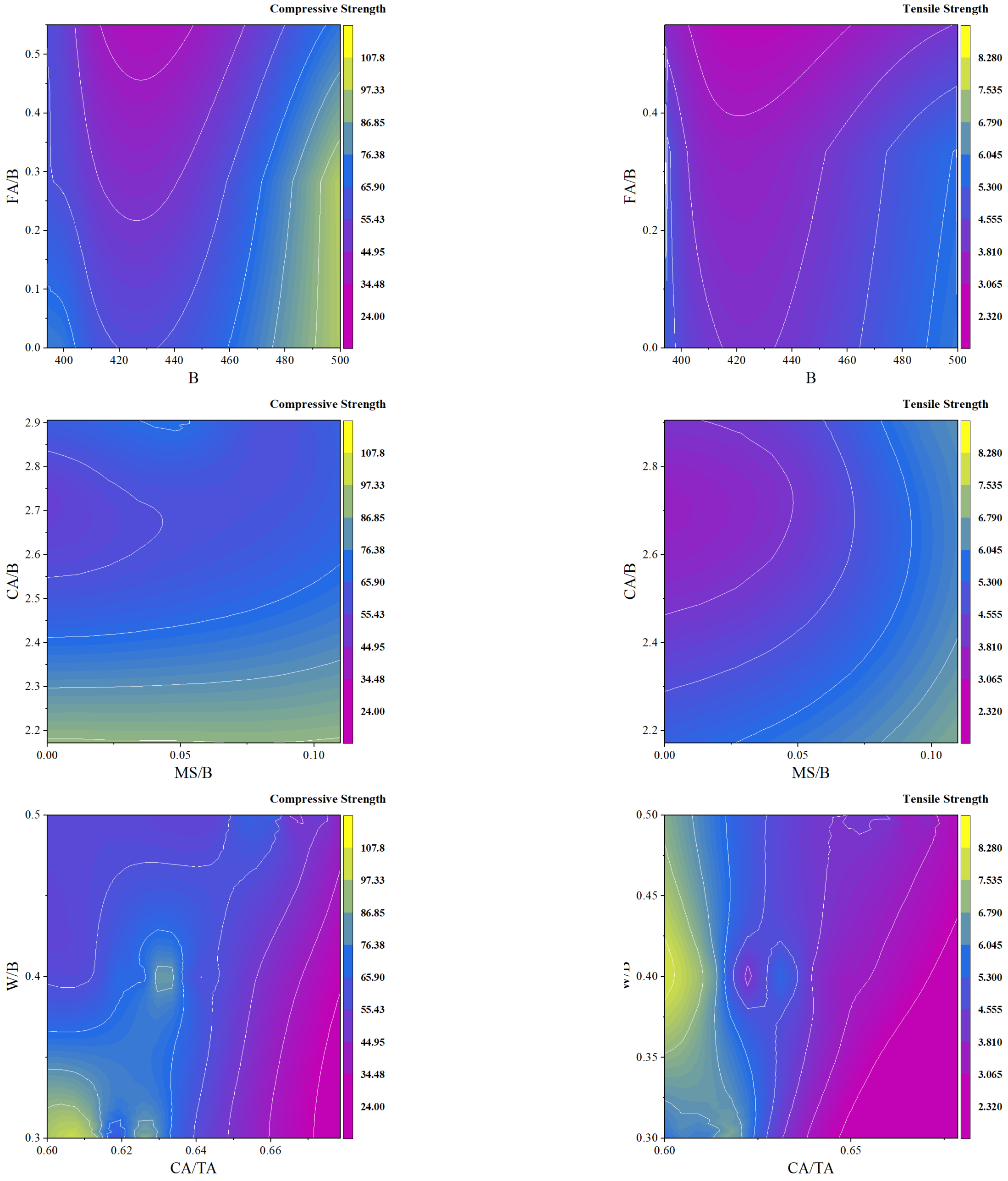


Fig. 1 The Contour plot for the correlation between input and outputs of CS and TS

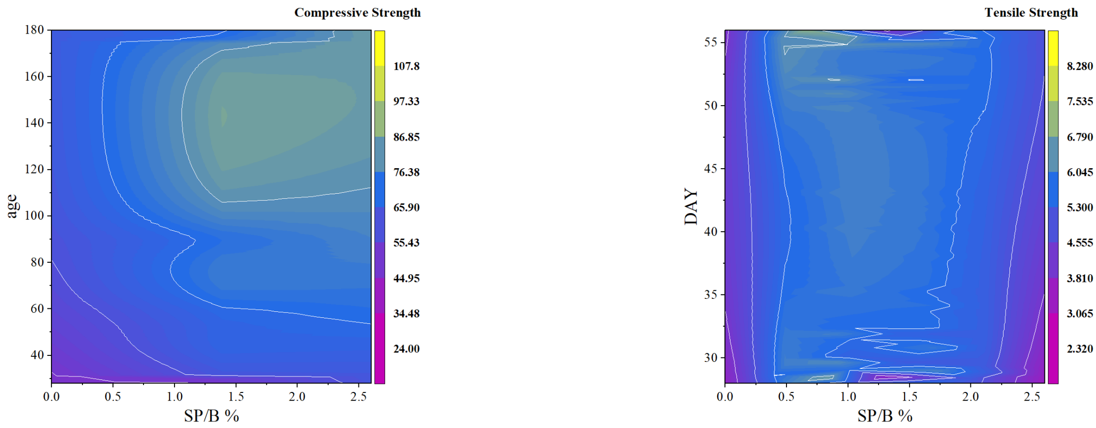


Fig. 1 Continued

3. Research methodology

3.1 Machine-Larning techniques

3.1.1 Light gradient boosting (LGB)

Utilizing a gradient learning framework, LGBM merges boosting principles with decision tree algorithms (Ekanayake *et al.* 2022, Fan *et al.* 2019, Ke *et al.* 2017). Renowned for its focus on training efficiency and speed enhancement, LGBM distinguishes itself from similar methods like Extreme Gradient Boosting (EGB) by leveraging histogram-based techniques. These techniques expedite training while reducing memory usage. Additionally, LGBM adopts a leaf-wise tree growth strategy, contrasting with traditional level-wise tree growth in decision trees. This approach efficiently identifies leaves with the highest branching gain during each cycle, optimizing algorithm performance.

LGBM's calculation procedures entail a detailed breakdown of numerical operations, systematically presented step-by-step, as delineated in Sun *et al.* (2020).

The primary aim of LGBM is to ascertain an estimated function $\hat{f}(x)$ that closely approximates the function $f^{*}(x)$, leveraging the provided training dataset $X = \{(x_i, y_i)\}_{i=1}^{m}$. The objective is to minimize the expected values of specific loss functions $L(y, f(x))$.

$$\hat{f}(x)\arg\min_{f} E_{y,x}\ L\ (y, f\ (x)) \tag{1}$$

To approximate the final model, LGBM employs multiple regression trees, symbolized as $\sum_{t=1}^{T} f_t(x)$.

$$f_T(X) = \sum_{t=1}^{T} f_t(X) \tag{2}$$

$asw_q\ (x)$ represents the regression trees, w signifies the weights assigned to the leaf node, and q denotes the decision rule of trees. N refers to the count of leaves present in each tree.

At every iteration t, the model undergoes additive training, depicted by the subsequent expression

$$\Gamma_t \cong \sum_{j=1}^{N} L(y_i, F_{t-1}(x_i) + f_t(x_i)) \tag{3}$$

Newton's method swiftly estimates the objective function, eliminating the constant term from Eq. (3) for simplification.

$$\Gamma_t \cong \sum_{j=1}^{N} (g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)) \tag{4}$$

g_i and h_i stand for the first-order and second-order gradient statistics of the loss functions, correspondingly. If the sample set of leaf j as I_j is denoted, Eq. (4) can be converted to Eq. (5)

$$\Gamma_t = \sum_{j=1}^{J} \left(\left(\sum_{i \epsilon I_j} g_i \right) \omega_j + \frac{1}{2} \left(\sum_{i \epsilon I_j} h_i + \lambda) \right) \omega_j^2 \right) \tag{5}$$

The optimal weights assigned to the leaves are denoted as ω_j^{*}, and the extreme values of Γ_K are calculated using Eq. (6) and Eq. (7), correspondingly.

$$\omega_j^{*} = -\frac{\sum_{i \epsilon I_j} g_i}{\sum_{i \epsilon I_j} h_i + \lambda} \tag{6}$$

$$\Gamma_T^{*} = -\frac{1}{2} \sum_{j=1}^{J} \frac{\left(\sum_{i \epsilon I_j} g_i\right)^2}{\sum_{i \epsilon I_j} h_i + \lambda} \tag{7}$$

Ultimately, the objective function is achieved with aggregating the splits as shown in Eq. (8)

$$G = \frac{1}{2} \left(\frac{\left(\sum_{i \epsilon I_l} g_i\right)^2}{\sum_{i \epsilon I_l} h_i + \lambda} + \frac{\left(\sum_{i \epsilon I_r} g_i\right)^2}{\sum_{i \epsilon I_r} h_i + \lambda} + \frac{\left(\sum_{i \epsilon I} g_i\right)^2}{\sum_{i \epsilon I} h_i + \lambda} \right) \tag{8}$$

I_r and I_l show the samples located in the right and left branches, alternatively. Fig. 2 shows the scheme of LGB.

3.1.2 Histogram Gradient Boosting (HGB)

HGB is a modified version of gradient-boosted decision trees. It aims to make training faster and use less memory by fixing problems that come with regular gradient-boosted decision trees (Guryanov 2019). Traditional models use many resources when checking each data sample to find the best split, which causes delays in memory use and slow training speeds. The cost of training the model depends on

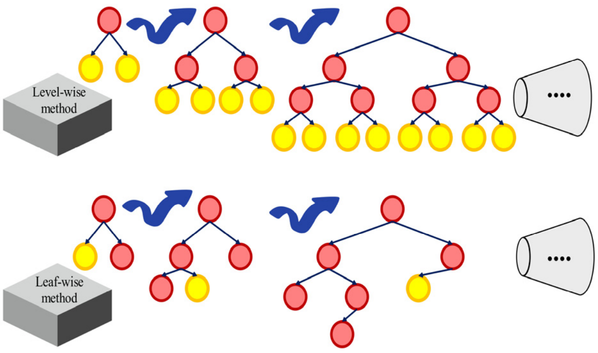


Fig. 2 Scheme of LGB

the size of examples and features. On the other hand, the histogram-based method in HGB breaks continuous variables into groups and creates a histogram of feature values while training (Nhat-Duc and Van-Duc 2023). This method greatly limits the number of options to find the best split when building a decision tree, which helps save time and memory during training. So, the model works better and is more beneficial for big data sets.

3.2 Metaheuristic algorithms

3.2.1 Snake optimization algorithm

By modeling diverse snake behaviors during mating, the classic snake optimization method optimizes model hyperparameters (Hashim and Hussien 2022). The snake optimization method has different steps and ways to work. In the beginning, the method creates a group of possible solutions quite simply, which makes the group less varied. The algorithm's capacity to do global searches steadily declines as the number of iterations rises, while its ability to perform local optimum escapes significantly declines. Later stages see a significant increase in the pace of population convergence, but because of the population's propensity for aggregation, the algorithm's best search performance suffers. When the snake optimization algorithm was being developed, the population position update equations looked like this

$$X_{i.j}(t+1) = X_{food} \pm c_2 \cdot Temp \cdot r \cdot \left(X_{food} - X_{i.j}(t)\right) \quad (9)$$

$$Q \in (0.25, 1] \; and \; Temp \in \; (0.6, 1] \quad (10)$$

$$Q = c_1 \cdot exp\left(\frac{t-T}{T}\right) \quad (11)$$

$$Temp = exp\left(-\frac{t}{T}\right) \quad (12)$$

$X_{i.j}(t+1)$ in Eq. (9) denote the snake's ideal location after $t+1$ iterations. The target food's ideal location is indicated by X_{food}. c_1 and c_2 are constants, and r defines a random number between [0,1]. X_{ij} shows where the snake is positioned. T determines the iterations' maximum number, and t indicates the current iterations' number.

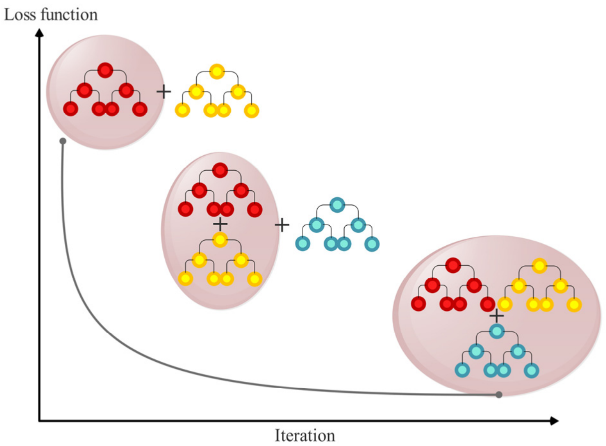


Fig. 3 Structure of HGB

The following equations describe how to update the population located in the snake optimization algorithm's mating mode as it is being developed

$$X_{i.m}(t+1) = X_{i.m}(t) + c_2 \cdot A_m \cdot r \cdot \left(Q \cdot X_{i.f}(t) - X_{i.m}(t)\right) \quad (13)$$

$$Q \in (0.25, 1] \quad and \quad Temp \in (0, 0.6) \quad (14)$$

$$A_m = exp\left(-\frac{f_{i.f}}{f_{i.m}}\right) \quad (15)$$

$$X_{i.f}(t+1) = X_{i.f}(t) + c_2 \cdot A_f \cdot r \cdot \left(Q \cdot X_{i.m}(t) - X_{i.f}(t)\right) \quad (16)$$

$$Q \in (0.25, 1] \quad and \quad Temp \in (0, 0.6) \quad (17)$$

$$A_f = exp\left(-\frac{f_{i.m}}{f_{i.f}}\right) \quad (18)$$

$X_{i.m}(t+1)$ and $X_{i.f}(t+1)$ denote the new generation of individual positions of the $i-th$ male and female snakes, respectively. r is a random number within [0,1]. The present locations of the $i-th$ male and female snakes are $X_{i.f}$(t) and $X_{i.m}$(t), respectively. The capacity of the male and female snakes to mate is designated as A_m and A_f, respectively. The fitness values of the placements of the $i-th$ male and female snakes are represented, respectively, by $F_{i.m}$ and $F_{i.f}$ (Hashim and Hussien 2022).

The flowchart and pseudo-code of SO have been shown in Fig. 4 and Algorithm 1.

3.2.2 Fox optimization algorithm

The proposed optimization algorithm, named the Snow Optimization Algorithm, draws inspiration from the hunting habits of red foxes. Emulating the red fox's behavior of diving into the snow to seek prey, the algorithm incorporates the primary methods red foxes employ to pursue the best prey (Połap and Woźniak 2021).

In order to examine $BestFitness$ and $BestX$, the fitness values of different search agents were compared,

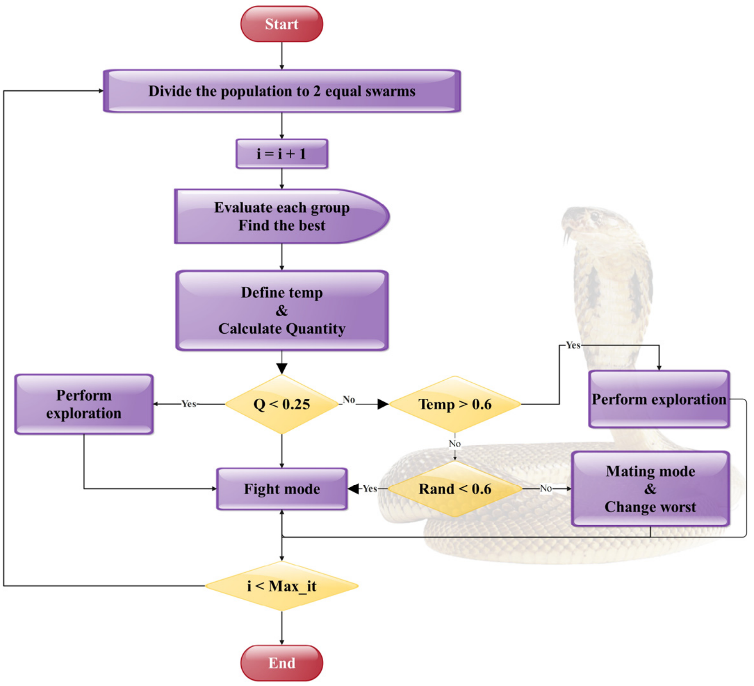


Fig. 4 Flowchart of SO

Algorithm 1 Snake optimizer algorithm

```
Initialize Issue Setting
Initialize the population randomly
  while $r \le R$ do
  Find the best agent
  Determine temperature
  Determine the quality of food $F$
    If $F < 0.25$ then
    Perform a search
    else if $F < 0.6$ then
    Perform exploitation
    else
      If $rand > 0.6$ then
        The snake in Fight mode
    End if
  End if
  End while
  Return the best solution
```

which are represented by rows in an X matrix. The iterative processes of $BestFitness$ and $BestX$ are carried out with the help of a condition that compares the current row's fitness $(fitness_{i+1})$ with the previous row's fitness $(fitness_i)$.

The proposed method balances exploration and exploitation in FOX by utilizing a random variable and a condition statement. This random variable, denoted as 'r,' assigns a 50% probability for either exploration or exploitation during iterations. The suggested approach is crucial for avoiding local optima, as it evenly allocates iterations for exploration and exploitation. The condition statement ensures this balance, and the variable 'BestX' is employed to reduce search performance gradually. Motivated by the fitness value, search agents steer clear of local optima, and location updating is driven by the exploration phase deactivation, allowing subsequent phases to engage if the new position remains unchanged. The subsequent sections elaborate on exploration and exploitation.

- *Exploitation*

A condition exists regarding the potential for capturing prey during the exploitation phase. In the range [0,1], the value of the random variable p determines. In this case, the red fox's new home must be found if the random number p is greater than 0.18. For finding a new position, distance sound travels $Dist\_S\_T_{it}$, the distance of the red fox from the prey $Dist\_Fox\_Prey_{it}$ and jumping value Jumpit must be computed. A random value between 0 and 1 is created for the sound travel period $Time\_S\_T_{it}$.

A condition exists regarding the potential for capturing prey during the exploitation phase. The value of the random variable p is in the range [0,1]. In this case, the red fox's new home must be found if the random number p is more significant than 0.18. To find a new position, distance sound travels $Dist\_S\_T_{it}$, the distance of the red fox from the prey $Dist\_Fox\_Prey_{it}$ and jumping value Jumpit must be computed. A random value between 0 and 1 is created for the sound travel period $Time\_S\_T_{it}$. The distance of the sound from the red fox is found by multiplying the speed sound in the air Sp_S with time sound travels time $Time\_S\_T_{it}$

$$Dist\_S\_T_{it} = Sp\_S \times Time\_S\_T_{it} \tag{19}$$

The sound's speed in air is 343, and $Time\_S\_T_{it}$ is a random value between 0 and 1. The number of iterations ranges from 1 to 500. An additional equation is formulated to calculate Sp_S, derived from the optimal location discovered by dividing the time for sound to travel between the fox and prey. $Time\_S\_T_{it}$ shows a random and $BestPositionit$ is the most effective search agent in the population. Eq. (20) demonstrates the determination of the speed of sound, Sp_S, depending on the optimal position.

$$Sp_S = \frac{BestPosition}{Time\_S\_T_{it}} \tag{20}$$

Eq. (20) is used to find the distance sound travels, which means that distance $(Dist\_S\_T_{it}) = velocity\ (Sp\_S) * time\ (Dist\_S\_T_{it})$. Consequently, the distance of the fox from the prey; $Dist\_Fox\ \_Prey_{it}$ can be calculated by halving $Dist\_S\ \_T_{it}$. To calculate the distance between the sensor and object in physics, divide the distance sound travels by 2 since it is half the distance the sound wave has traversed. A sensor doubles the sound's travel time by 0.5 or 12 the distance. Because just half of the sound wave is considered, multiplying by 0.5 and dividing by 12 have the same effect. Eq. (21), which calculates the distance between two objects using sensors, is derived from these methods

$$Dist\_Fox)Prey_{it} = Dist\_S\_T_{it} \times 0.5 \tag{21}$$

The red fox must choose a new position based on the distance from the prey to make a successful leap and catch the prey. The fox must thus determine the leap height of $Jumpit$. Thus, the following equation may be used to compute $Jumpit$

$$Jump_{it} = 0.5 \times 9381 \times t^2 \tag{22}$$

The formula includes the acceleration due to gravity (9.81) and the average time it takes for sound to travel (t), squared to account for the up and down movements during the leap. By dimensions, the value of time transition is determined by dividing the total of $Time\_S\_T_{it}$. The average time (t) is calculated by dividing it by 2. The gravity and average time are halved because the leap value involves two separate durations for descending and ascending. Therefore, the time of gravity and average need to be multiplied by 0.5. The value of jump is then multiplied by $Dist\_Fox\_Preyit$ and c_1. The c_1 ranges from 0 to 0.18 when the red fox jumps northeast. The following equation determines the red fox's updated location when the variable p, a random value between 0 and 1, exceeds 0.18.

$$X_{(it+1)} = Dist\_Fox\_Prey_{it} \times Jump_{it} \times c_1 \tag{23}$$

The red fox is moved to a new position using Eqs. (23) and (24). Due to the usage of the p condition, only one is carried out in each iteration. The second section of the p condition in Eq. (23). If the value of p is less than or equal to 0.18, the Eq. (23) is multiplied by c_2 instead of c_1 if the otherwise condition of the p is different. The equation defines the new location if the p-value is bigger than 0.18 using Eq. (23). However, the equation defines the new position in Eq. (24) if the result is less than 0.18. c_2's range lies between [0.19,1]

$$X_{(it+1)} = Dist\_Fox\_Prey_{it} \times Jump_{it} \times c_2 \tag{24}$$

The value of $c_1 and\ c_2$ are 0.18 and 0.82 respectively. The numbers pertain to the jumping behavior of a red fox, which either leaps to the northeast or in the opposite direction. Consequently, if the p-value exceeds 0.18, the red fox is seen to jump in the northeast direction. Consequently, to find a new position, both $Dist\_Fox\_Prey_{it}$ and $Jump_{it}$ are multiplied by c_1. The red fox is likely to seize fresh opportunities and move towards achieving the best possible outcome on a worldwide scale. The red fox jumps northeast in the other direction. A $p - value$ below 0.18 indicates a poor probability of successfully capturing prey (18%). Therefore, both $Dist\_Fox\_\ Prey_{it}$ and $Jump_{it}$ are multiplied by c_2.

- *Exploration*

In this phase, the fox randomly searches based on the fox's best position discovered thus far to govern the random walk. Because it must move haphazardly to examine potential food in the search region, the fox lacks a leaping skill during this period. $MinT$ and variable an are utilized to regulate the fox's random movement toward the optimal spot throughout the search process. Eqs. (25) and (26) demonstrate the computation of the $MinT$ and variables. The minimal value of tt is used to determine $MinT$.

$$tt = \frac{sum\ (Time_{S_{T_{it}}}(i.:))}{dimension}.MinT = Min(tt) \tag{25}$$

The summation of $Time\_S\_T_{it}\ (i.:)$ is separated by the problem's size to determine the least average time tt.

$$a = 2 \times \left(it - \left(\frac{1}{Max_{it}}\right)\right) \tag{26}$$

The Max_{it} represents the maximum number of iterations. Calculating the $MinT$ and a variable impacts the search phase, guiding it towards a solution that closely approximates the optimal answer. Utilizing $rand(1.dimension)$ to ensure that the fox moves randomly in order to search for the prey. To enhance the searchability of FOX, both $MinT$ and a variable are utilized. r, a random number, is utilized to maintain equilibrium between the exploration and exploitation stages. The most effective solution, $BestX_{it}$, significantly influences the exploration phase. Eq. (27) illustrates the strategy used by the fox to find a new position in the search area $X_{(\mathrm{it}+1)}$. Existing algorithms can improve their performance by adjusting the equations at this step. They can also be utilized to suggest novel metaheuristic algorithms.

$$X_{(it+1)} = BestX_{it} \times rand(1.dimension) \times MinT \times a \tag{27}$$

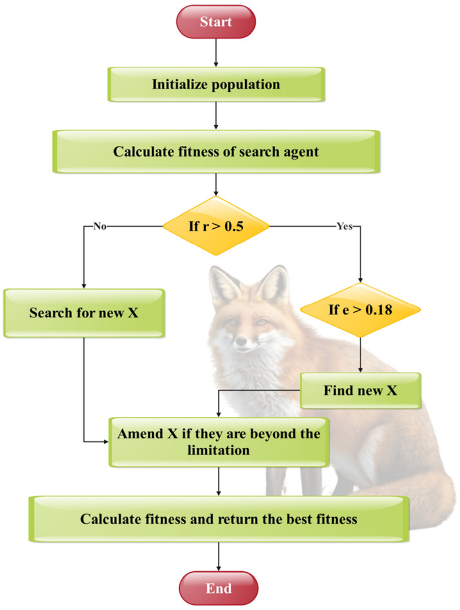


Fig. 5 Flowchart of FO

While FOX is employed for solving the multi-dimensional space issue, the equations in both stages do not require any modifications; instead, they must be adapted to a particular problem. In general, FOX starts by randomly initializing the red fox population. The population is then examined in the first iteration to determine whether or not each red fox's position is inside the benchmark function's boundary. The benchmark function's fitness value is then computed based on the population's row. The $BestFitness$ value and the best position ($BestX$) are chosen by following these procedures. Then, by comparing the random number r, a condition is begun. It is triggered to enter the exploitation phase if it is more than or equal to 0.5. In the exploitation phase, there is also a p condition. Eqs. (19)-(23) are used to determine the red fox's new position if the $p-value$ is larger than 0.18 can use Eq. (23). But only if the $p-value$ is lower than or equal to 0.18 using Eq. (26). The exploration phase is initiated if r in the otherwise condition is smaller than 0.5.

Therefore, the optimal location, a random number, and multiplying $MinT$ variable by variable are used to get a new position. As a result, the first iteration returns the $BestFitness$. After altering the population, the same methods are performed in the second iteration to identify the optimal location and fitness.

FOX's computational complexity per iteration is $O(SearchAgents * D * it)$, where $SearchAgents$ represents the population size, D is the problem's dimension, and it is the number of iterations. Thus, it may be said that FOX has a temporal complexity of O (n^2). The space complexity of FOX is determined by the matrices and vectors included in Algorithm 1. FOX has a space complexity of O (n^2) for every iteration. Fig. 5 shows the flowchart of FOX.

3.2.3 Prairie dog optimization algorithm

Prairie dogs, specifically the black-tailed species, are prevalent in the Great Plains, southwest desert grasslands of the United States, and the surrounding plains and plateaus in Canada and Mexico (Ezugwu *et al.* 2022). These herbivorous burrowing rodents belong to the genus Cynomys and are linked to the Sciuridae family, which includes ground squirrels and chipmunks. Prairie dogs exhibit physical adaptations, like short, muscular limbs and large toenails, enabling them to dig tunnels and run at speeds up to 35 mph for short distances. These adaptations are crucial for their survival and defense against predators in their habitats (Hoogland 1995).

Each prairie dog (PD) in a coterie is a member of one of the n coteries. A vector may determine the position of each prairie dog inside a certain coterie since the PD exists and functions as a unit or coterie. The location of each coterie in a colony is represented by the matrix provided in Eq. (28)

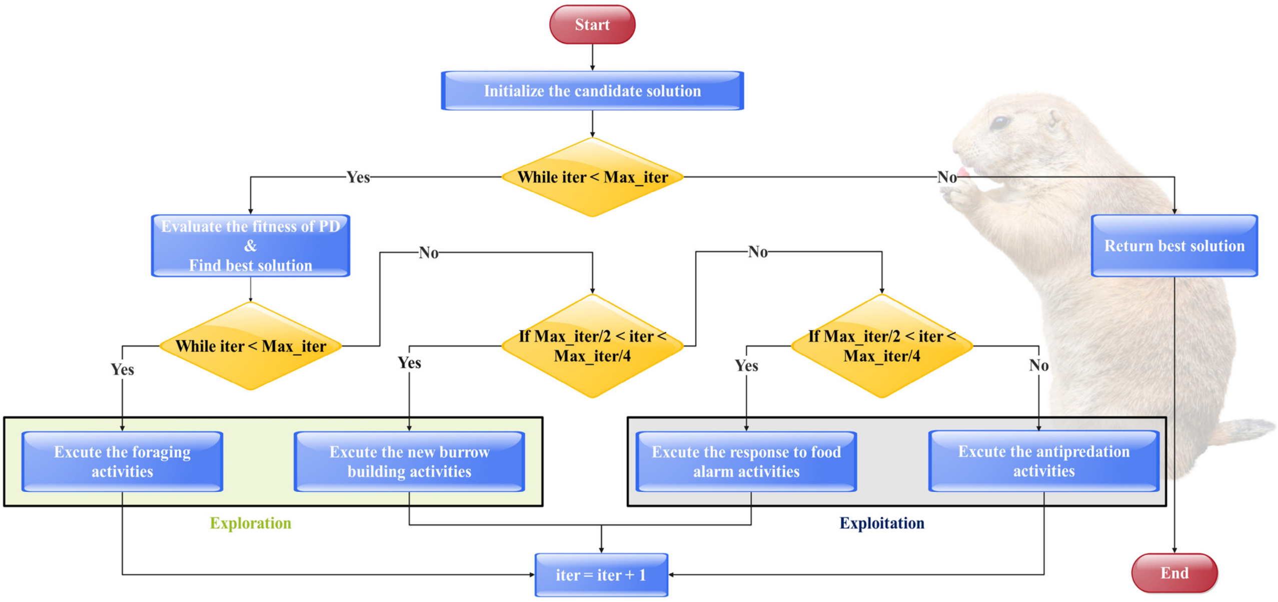


Fig. 6 Flowchart of PDO

$$CT = \begin{bmatrix} CT_{1,1} & CT_{1,2} & \cdots & CT_{1,D-1} & CT_{1,D} \\ \vdots & \vdots & CT_{i,j} & \vdots & \vdots \\ CT_{k,1} & CT_{k,2} & \cdots & CT_{k,D} & CT_{k,D} \end{bmatrix} \tag{28}$$

Where $CT_{i,j}$ represents the jth dimension of the ith coterie in a colony. Eq. (29) represents the location of all the prairie dogs in a coterie

$$PD = \begin{bmatrix} PD_{1,1} & PD_{1,2} & \cdots & PD_{1,D-1} & PD_{1,D} \\ \vdots & \vdots & PD_{i,j} & \vdots & \vdots \\ PD_{k,1} & PD_{k,2} & \cdots & PD_{k,D} & PD_{k,D} \end{bmatrix} \tag{29}$$

$$CT_{i.j} = U(0.1) \times (UB_j - LB_j) + LB_j \tag{30}$$

$$PD_{i.j} = U(0.1) \times (ub_j - lb_j) + lb_j \tag{31}$$

Where UB_j and LB_j show the upper and lower bounds of the $j-th$ dimension of the optimization problem, $ub_j = \frac{UB_j}{m}$ and $lb_j = \frac{LB_j}{m}$, and $U(0.1)$ iindicates a random number with a uniform distribution between 0 and 1.

Fig. 6 shows the flowchart of PDO.

3.3 Hybridization

This study utilizes a novel hybrid methodology that integrates the SO, FO, PDO, and an ensemble of three optimization algorithms for enhancing the accuracy of estimating CS and TS values. The hybridization effectively combines the unique advantages of the specified algorithms inside the HGB and LGBM, creating ten distinct models, including two single models, six hybrid models, and two ensemble models.

This strategic integration seeks to use the combination of algorithms, consequently enhancing and diversifying the predictive powers of the HGB and LGBM models in CS and TS prediction. The hybridization process occurred through the steps as follows:

Optimized parameters were integrated from the SO, FO, PDO, and SO-FO-PDO ensemble models into the HGB and LGBM models during the first stage. This critical phase in the hybridization process entailed integrating the distinctive advantages of optimization algorithms. These tuned parameters were then used to develop hybrid models called HGSO, HGFO, HGPD, LGSO, LGFO, and LGPD,

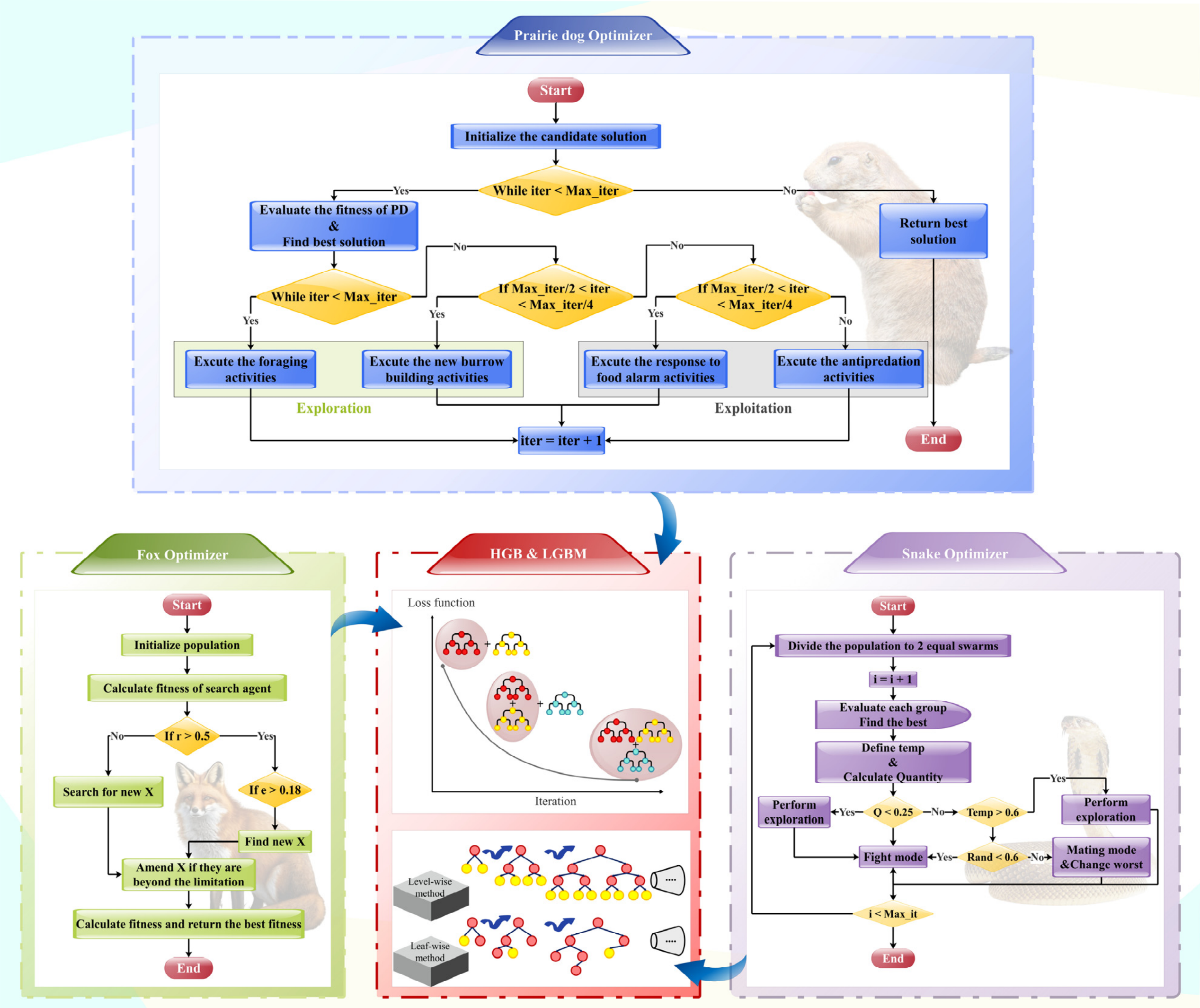


Fig. 7 Flowchart of hybrid model

representing the new synergy between the contributing algorithms. Hybridizing SO, FO, and PDO algorithms with HGB and LGBM models have several advantages. SO is well-documented for its ability to simulate the hunting strategy of snakes and enables a step further in efficiently exploring high-dimensional search spaces. This allows full exploration of the combination of parameters in the models. FO, inspired by the hunting behavior of foxes, excels in balancing exploration and exploitation, enabling effective fine-tuning of model parameters while avoiding premature convergence.

Additionally, PDO, drawing inspiration from the collaborative foraging behavior of prairie dogs, enhances the diversity of solutions and promotes global search, thereby complementing the optimization process by preventing stagnation in local optima. By hybridizing these algorithms with HGB and LGBM models, we harness their strengths to facilitate robust parameter optimization, improving model performance and generalization across diverse datasets and tasks. A detailed dataset was carefully prepared in the second phase to train the specified models. This dataset contains a wide range of input variables and concrete properties parameters, carefully selected from various literature sources to capture real-world complexities and strengthen the hybrid models' ability to predict CS and TS in civil engineering applications.

Also, the flowchart of hybrid models has been shown in Fig. 7.

3.4 K-Fold Cross Validation

Overfitting occurs when a model fits too closely to a particular dataset, limiting its generalization ability to unseen data. While this can result in high predictive accuracy on the training data, it often leads to poor performance when the model encounters new data. This challenge is particularly significant when working with smaller or limited validation data, where merely increasing the validation set size may not address overfitting.

In such conditions, finding the correct underlying pattern also becomes increasingly challenging. The study attempts to address the problem of overfitting by following the K-fold cross-validation method, which is the most effective way to evaluate model performance with minimal risk of overfitting (Kohavi 1995). For example, when this

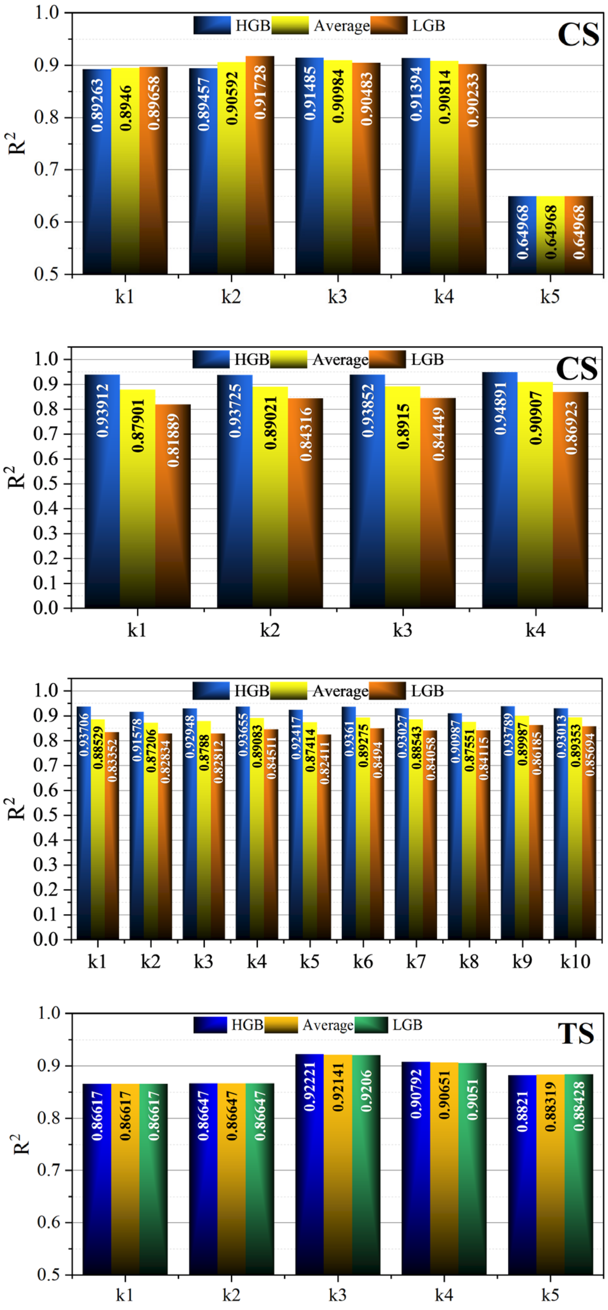


Fig. 8 Result of Model prediction performance on each cross-validation fold

Table 3 The result of developed K-Fold

HPC feature	Indicator	Model	Number of K-Fold									
			K1	*K2*	*K3*	*K4*	*K5*	*K6*	*K7*	*K8*	*K9*	*K10*
CS	R^2	HGB	0.9391	0.9373	0.9385	0.9489	-	-	-	-	-	-
		LGBM	0.8189	0.8432	0.8445	0.8692	-	-	-	-	-	-
	R^2	HGB	0.8926	0.8946	0.9149	0.9139	0.6497	-	-	-	-	-
		LGBM	0.8966	0.9173	0.9048	0.9023	0.6497	-	-	-	-	-
	R^2	HGB	0.9371	0.9158	0.9295	0.9365	0.9242	0.9361	0.9303	0.9099	0.9379	0.9301
		LGBM	0.8335	0.8283	0.8281	0.8451	0.8241	0.8494	0.8406	0.8411	0.8618	0.8569
TS	R^2	HGB	0.8662	0.8665	0.9222	0.9079	0.8821	-	-	-	-	-
		LGBM	0.8662	0.8665	0.9206	0.9051	0.8843	-	-	-	-	-

Table 4 The results of hyperparameters for HGB

HPC features	Models	Hyperparameter			
		max_iter	max_depth	learning_rate	max_leaf_nodes
CS	HGSO	274	65	0.07	68
	HGFO	349	11	0.04	132
	HGPD	174	3	0.12	284
TS	HGSO	135	3	0.21	325
	HGFO	33	5	0.1	51
	HGPD	348	1	0.18	39

Table 5 The results of hyperparameters for LGBM

HPC features	Models	Hyperparameter			
		num_leaves	max_depth	learning_rate	n_estimators
CS	LGSO	541	57	0.05	211
	LGFO	124	12	0.1	487
	LGPD	41	23	0.46	314
TS	LGSO	102	54	0.6	214
	LGFO	251	127	0.26	139
	LGPD	53	11	0.38	218

approach splits the dataset into five subsets (K1, K2, K3, K4, and K5), four subsets are used for training each iteration, while the remaining subset is used for testing. This process is repeated until every subset has been used for testing, resulting in five distinct models and corresponding cross-validation assessments. R² values for each fold are calculated to measure the models' performance in predicting CS and TS. Table 3 and Fig. 8 summarize the results by comparing model performance across the five folds. HGB and LGB had some predictive solid power in the third fold. For TS prediction, the 5-fold cross-validation showed consistent results without notable variation, indicating no signs of overfitting in the prediction process. However, in the case of CS prediction, the five-fold cross-validation exhibited some variability across the evaluation metrics. For CS, additional cross-validation experiments were conducted using 4-fold and 10-fold divisions to identify the optimal configuration. Based on the analysis, the 4-fold cross-validation was selected for the subsequent modeling process, as it provided the most reliable and stable results. Across these folds, K4 and K3 were identified as the most effective subsets for model predictions.

4. Results and discussion

4.1 The evaluation criteria (Performance evaluation methods)

The following statistical measures were used to assess the performance and accuracy of the chosen optimizers: mean absolute percentage error (MAPE), root means square error (RMSE), mean absolute error (MAE), Normalized Mean Square Error (NMSE), and coefficient of determination (R^2). As a result, the selected optimizers that are considered for training are carefully compared. The statistical criteria may be calculated as shown in Eqs. (32)-(36)

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{32}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{33}$$

$$MAE = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{n} \tag{34}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{35}$$

$$NMSE = \frac{1}{n}\sum_{i=1}^{n}\frac{(y_i - \hat{y}_i)^2}{y_i . \hat{y}_i} \tag{36}$$

Here, y_i and $\hat{y}_i$ are the actual and predicted load values at the corresponding time of y_i, alternatively, the bar shows the mean of variables, and n denotes the total amount of data. In addition,

4.2 Results of hyperparameters and convergence curves

This study examines the impact of critical hyperparameters on both the HGB and LGB models, crucial components in machine learning that significantly influence model performance, including optimizer ensembling. Specifically, the HGB model is analyzed using four key hyperparameters: max_iter, max_depth, learning_rate, and

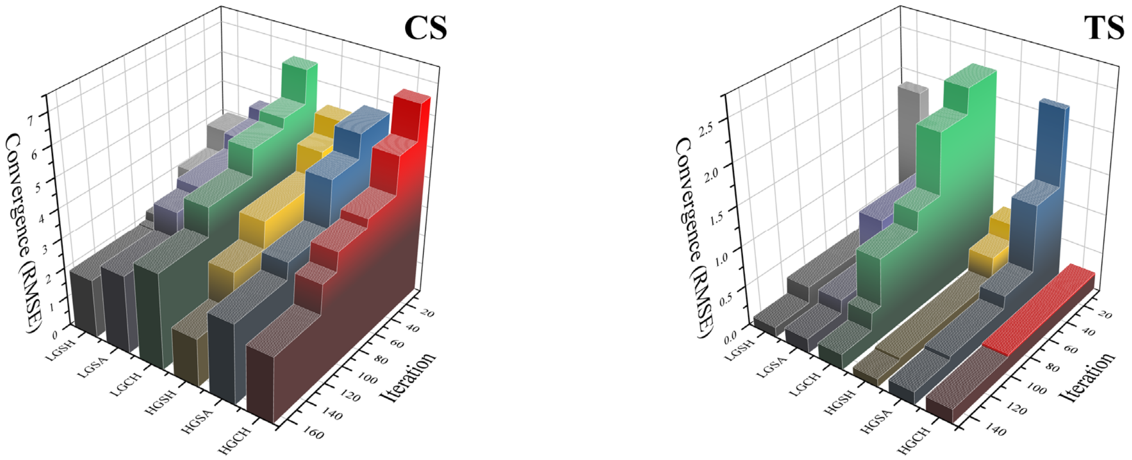


Fig. 9 The 3D convergence Plot for the output of hybrid models

max_leaf_nodes. Similarly, the research investigates the sensitivity of the LGB model to four important hyperparameters: max_depth, learning_rate, num_leaves, and n_estimators. The study has emphasized the importance of tuning hyperparameters for the best performance. By carefully selecting and adjusting these hyperparameters, the study tries to attain high accuracy with the best generalization performance of the hybrid models while, on the one hand, trying to avoid overfitting and reducing the training time.

In this research, the convergence curve is a graphical representation showing the optimization process used to train the different machine learning models. This analysis has considered six different models: HGSO, HGFO, HGPD, LGSO, LGFO, and LGPD. All of them were analyzed adequately regarding their convergence process while estimating the CS and TS of HPC, as shown in Fig. 9. All the models started with an RMSE value of more than four at the commencement of the CS prediction process. In contrast, the initial values of RMSE within the TS prediction process

Table 6 The outcomes generated by employing LGBM and HGB models in the training process focus on predicting CS

Model	Group	Part	Indicator values				
			RMSE	R^2	MAE	NMSE	MAPE
HGB	Single	Train	3.7604	0.9546	4.8533	2.9780	10.3774
		Validation	3.0643	0.8509	4.5207	2.6527	8.4542
		Test	2.3798	0.9304	3.2416	2.0085	6.4224
		All	3.4889	0.9492	4.5640	2.7853	9.6530
HGSO	Hybrid	Train	2.6578	0.9768	3.5446	2.1244	7.3669
		Validation	2.3905	0.9217	3.4921	1.9683	6.5583
		Test	2.4588	0.9171	3.0266	2.0110	6.8087
		All	2.5907	0.9715	3.4597	2.0843	7.1807
HGFO	Hybrid	Train	3.1090	0.9688	3.9283	2.4761	8.5903
		Validation	2.4843	0.8776	3.5897	2.1089	6.8857
		Test	2.2520	0.9349	2.8959	1.8297	6.0661
		All	2.9090	0.9644	3.7243	2.3253	8.0587
HGPD	Hybrid	Train	2.1283	0.9851	2.9696	1.7623	5.8965
		Validation	1.8273	0.9430	2.6011	1.5336	4.9319
		Test	1.7502	0.9646	1.9928	1.2558	4.8433
		All	2.0333	0.9826	2.7694	1.6529	5.6257
HGSFP	Ensemble	Train	1.5079	0.9927	1.9230	1.1893	4.1688
		Validation	1.0642	0.9780	1.4003	0.8329	2.9481
		Test	0.9693	0.9878	1.2403	0.7961	2.6710
		All	1.3804	0.9919	1.7436	1.0778	3.8218

Table 6 Continued

Model	Group	Part	Indicator values				
			RMSE	R^2	MAE	NMSE	MAPE
LGBM	Single	Train	6.2954	0.8822	8.7016	5.3081	17.3982
		Validation	5.5290	0.6547	8.3141	4.7844	15.2856
		Test	4.5206	0.7965	5.8013	3.7026	12.5305
		All	5.9520	0.8673	8.2124	4.9912	16.4609
LGSO	Hybrid	Train	4.7083	0.9326	6.3646	3.7928	13.0355
		Validation	4.0142	0.8040	5.0268	3.0357	11.0967
		Test	2.9956	0.8817	3.7380	2.5328	8.2744
		All	4.3936	0.9253	5.7747	3.4926	12.1601
LGFO	Hybrid	Train	5.3502	0.9094	7.2588	4.4116	14.8268
		Validation	4.2183	0.7009	6.3780	3.6652	11.6796
		Test	3.7558	0.8396	5.0482	3.0260	10.0828
		All	4.9852	0.8991	6.7988	4.0944	13.8161
LGPD	Hybrid	Train	3.1305	0.9709	4.4313	2.6475	8.6088
		Validation	2.5557	0.8953	3.8612	2.2832	7.0732
		Test	2.5276	0.9338	3.5802	2.2231	6.5085
		All	2.9675	0.9661	4.2198	2.5301	8.1527
LGSFP	Ensemble	Train	2.4500	0.9805	3.2731	1.9709	6.7907
		Validation	1.6775	0.9497	2.3052	1.3580	4.6442
		Test	2.0685	0.9466	2.5765	1.5887	5.5330
		All	2.2960	0.9779	3.0254	1.8228	6.3597

ranged from 0.2 for HGSO to 2.3 for LGSO. There was a gradual reduction of the RMSE values throughout the models in the convergence process; this reflects the iterative improvement of predictive performances for all the models. According to the last values of RMSE, the range was different for CS prediction, indicating up to 3 for LGSO, while it had 1.5 for HGPD, while for the TS prediction process, the final RMSE values have been closer to each other with intervals lower than 0.5. Overall, the convergence curves for both CS and TS targets underlined that the models could learn from the data effectively and refine predictive accuracy across 170 iterations, showcasing robustness, and adaptability in predictive modeling tasks

4.3 Comparing prediction results

In this section, the results of the relevant models have been evaluated in all sections, training, validation, and testing for CS and TS. All sections have allocated 100% of the samples, while 70%, 15%, and 15% are allocated to the training, validation and testing sections. The models have undergone evaluation based on the metrics introduced in the preceding section. As per the evaluators, the results are documented and visually represented in Table 6 and Fig. 10 for CS and Table 7 and Fig. 11 for TS. These models are categorized into three groups: single, hybrid, and ensemble. Single models encompass HGB and LGBM. Hybrid models are represented as follows: HGB+SO (HGSO), HGB+FO(HGFO), HGB+PDO (HGPD), LGB+SO (LGSO), LGB+FO (LGFO), and LGB+PDO (LGPD). Finally, the ensemble models consist of HGSFP and LGSFP.

- *CS*

Based on the CS values illustrated in Table 6 and Fig. 10, it is evident that among the two ensemble models (HGSFP and LGSFP), the HGSFP model stands out with the highest R^2 value, reaching 0.9960 during the training phase. Regarding RMSE and MAE, the HGSFP model exhibits the lowest values at 1.5378 and 1.2343, respectively. Furthermore, regarding two other evaluators (NMSE and MAPE), the HGSFP model demonstrates the most favorable values, with 0.0246 and 1.9305, respectively.

The PDO optimization algorithm showcased superior performance among the six hybrid models (HGSO, HGFO, HGPD, LGSO, LGFO, and LGPD). Upon comparing the two models utilizing the PDO algorithm (HGPD and LGPD), it became apparent that HGPD (which combines the HGB model and PDO algorithm) achieved the highest R^2 value of 0.9961 and the lowest RMSE value of 1.1553 during the training phase of modeling.

Overall, the results highlight that the ensemble-hybrid HGSFP model has demonstrated notably superior performance to the other models, except for HGPD. It is noteworthy that the PDO optimizer showed a remarkable ability to enhance the performance when paired with the models.

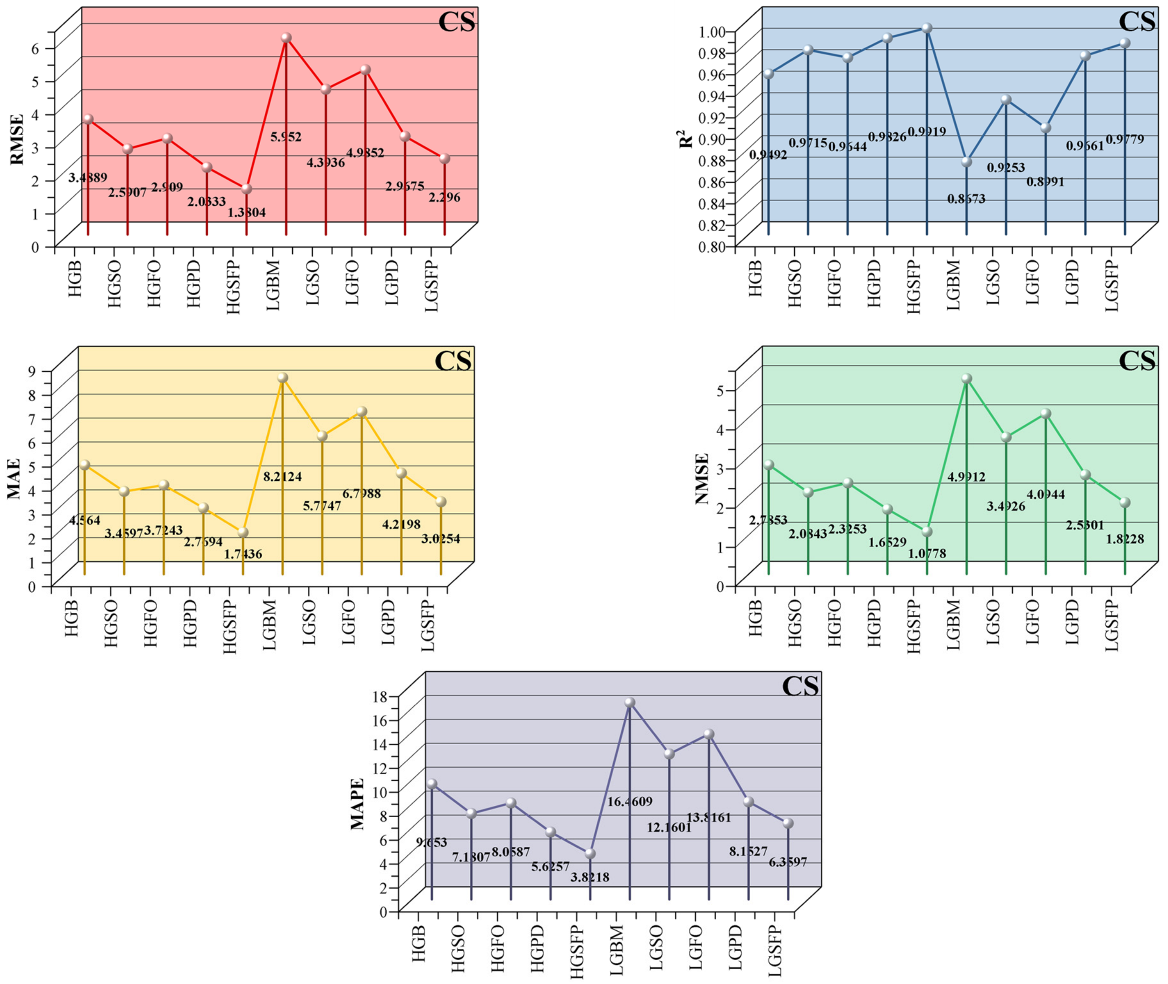


Fig. 10 The comparison of metrics for CS

Table 7 The outcomes generated by employing LGBM and HGB models in the training process focus on predicting TS

Model	Group	Part	Indicator values				
			RMSE	R^2	MAE	NMSE	MAPE
HGB	Single	Train	0.1776	0.9702	0.1464	0.0003	3.0749
		Validation	0.2153	0.9561	0.1936	0.0039	3.2657
		Test	0.3821	0.9376	0.3454	0.0122	5.4225
		All	0.2109	0.9695	0.1710	0.0004	3.3287
HGSO	Hybrid	Train	0.1387	0.9816	0.1183	0.0002	2.5931
		Validation	0.2275	0.9368	0.1902	0.0043	3.1954
		Test	0.3622	0.9558	0.3365	0.0109	5.2228
		All	0.1835	0.9782	0.1473	0.0003	2.9163
HGFO	Hybrid	Train	0.1103	0.9886	0.0941	0.0001	2.0556
		Validation	0.2411	0.9520	0.1998	0.0048	3.3140
		Test	0.3539	0.9734	0.3382	0.0104	5.2315
		All	0.1675	0.9851	0.1291	0.0002	2.4991
HGPD	Hybrid	Train	0.0766	0.9947	0.0594	0.0001	1.2666
		Validation	0.1762	0.9476	0.1510	0.0026	2.4982
		Test	0.1742	0.9284	0.0303	0.0145	0.0217
		All	0.1041	0.9918	0.0773	0.0001	1.4789

Table 7 Continued

Model	Group	Part	Indicator values				
			RMSE	R^2	MAE	NMSE	MAPE
HGSFP	Ensemble	Train	0.1030	0.9947	0.0857	0.0001	1.8349
		Validation	0.0824	0.9835	0.0731	0.0006	1.2205
		Test	0.1489	0.9769	0.1148	0.0018	1.7113
		All	0.1067	0.9936	0.0873	0.0001	1.7611
LGBM	Single	Train	0.1809	0.9684	0.1501	0.0003	3.3531
		Validation	0.3384	0.9348	0.3056	0.0095	5.3060
		Test	0.4995	0.9457	0.4736	0.0208	7.6059
		All	0.2501	0.9623	0.1980	0.0005	3.9737
LGSO	Hybrid	Train	0.1647	0.9760	0.1427	0.0003	3.1801
		Validation	0.2451	0.9155	0.1881	0.0050	3.2013
		Test	0.3428	0.9565	0.3141	0.0098	4.8105
		All	0.1986	0.9730	0.1644	0.0003	3.3453
LGFO	Hybrid	Train	0.1200	0.9862	0.1020	0.0002	2.1911
		Validation	0.2650	0.9336	0.2283	0.0059	3.8644
		Test	0.4022	0.9551	0.3785	0.0135	5.8803
		All	0.1863	0.9809	0.1422	0.0003	2.7273
LGPD	Hybrid	Train	0.0905	0.9925	0.0772	0.0001	1.6779
		Validation	0.2003	0.9225	0.1638	0.0033	2.7719
		Test	0.2384	0.9710	0.2122	0.0047	3.2360
		All	0.1275	0.9907	0.0993	0.0001	1.9431
LGSFP	Ensemble	Train	0.1195	0.9932	0.0976	0.0001	2.0078
		Validation	0.1569	0.9376	0.1311	0.0021	2.2165
		Test	0.1965	0.9847	0.1794	0.0032	2.6885
		All	0.1332	0.9888	0.1091	0.0001	2.0968

- *TS*

Table 7 and Fig. 11 present the results detailing the performance of predictive models in predicting TS values. Similarly, in the TS section, the behavior of models resembled that observed in the CS section, leading to PDO forming more effective combinations with the models in this context, while SO exhibited the weakest performance. Notably, HGPD displayed the most optimal performance, boasting an R^2 value of 0.9947, surpassing LGPD among the six hybrid models. HGPD achieved the lowest RMSE and MAE values, at 0.0766 in the training section and 0.0303 in the testing section, respectively, representing the most favorable outcomes among all models. Regarding the two ensemble models, the HGSFP model demonstrated superior applicability to the LGSFP model, with an R^2 value of 0.9947, matching that of the HGPD model. However, considering other metric values, the HGPD model emerged as the stronger performer in predicting TS values. Therefore, the hybridized HGPD model, particularly when paired with SO, concludes with more reliable results than other models.

4.4 Discussion

In assessing predictive accuracy for compressive strength values over 28-day and 180-day samples, the HGPD and LGPD models showcase the first and second lowest measurement errors, as illustrated in Figs. 12 and 13. This highlights their superior performance, attributed to their high R^2 values. Conversely, analysis reveals that the LGFO model exhibits the highest errors for both sample durations, suggesting that the fusion of LGBM and FO leads to inaccuracies. Notably, among the remaining models, HGPD emerges as the top performer.

Based on the analysis of tensile strength data depicted in Figs. 14 and 15, the HGPD model outperforms others by showcasing the lowest measurement error in predicting both the 28-day and 56-day samples. Conversely, the LGSO and LGFO models display the most significant differences between the measured and projected values for the 28-day and 56-day samples, respectively. This disparity suggests that these models are unsuitable for accurately predicting tensile strength. However, among the remaining models, the combination of HGB and PDO emerges as the top performer, exhibiting superior accuracy and reliability in predicting TS values.

- *CS*

In Fig. 16, a scatter plot illustrates the relationship between predicted and measured samples for CS. The samples are categorized into three phases: training, validation, and testing. Each sample point is determined

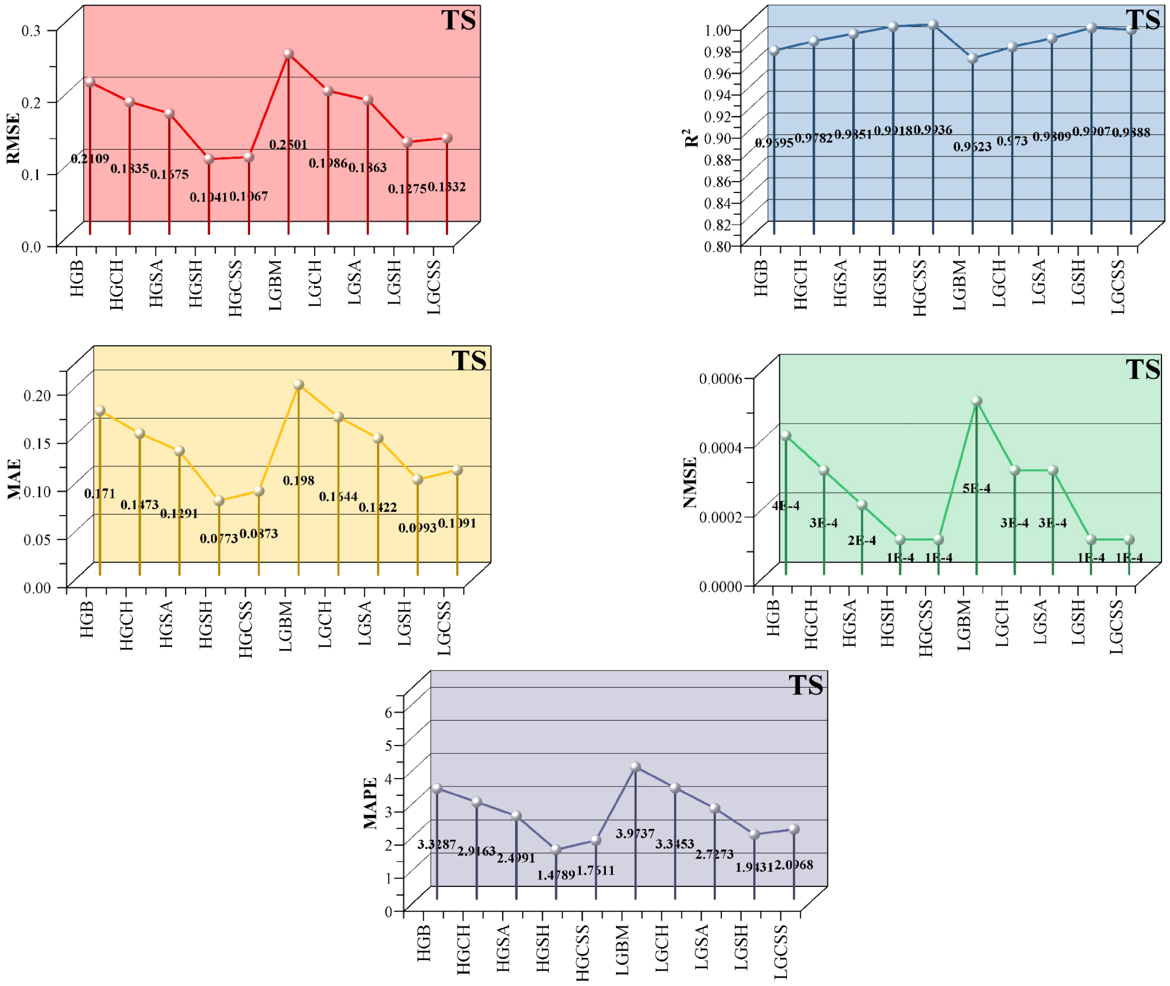


Fig. 11 The comparison of metrics for TS

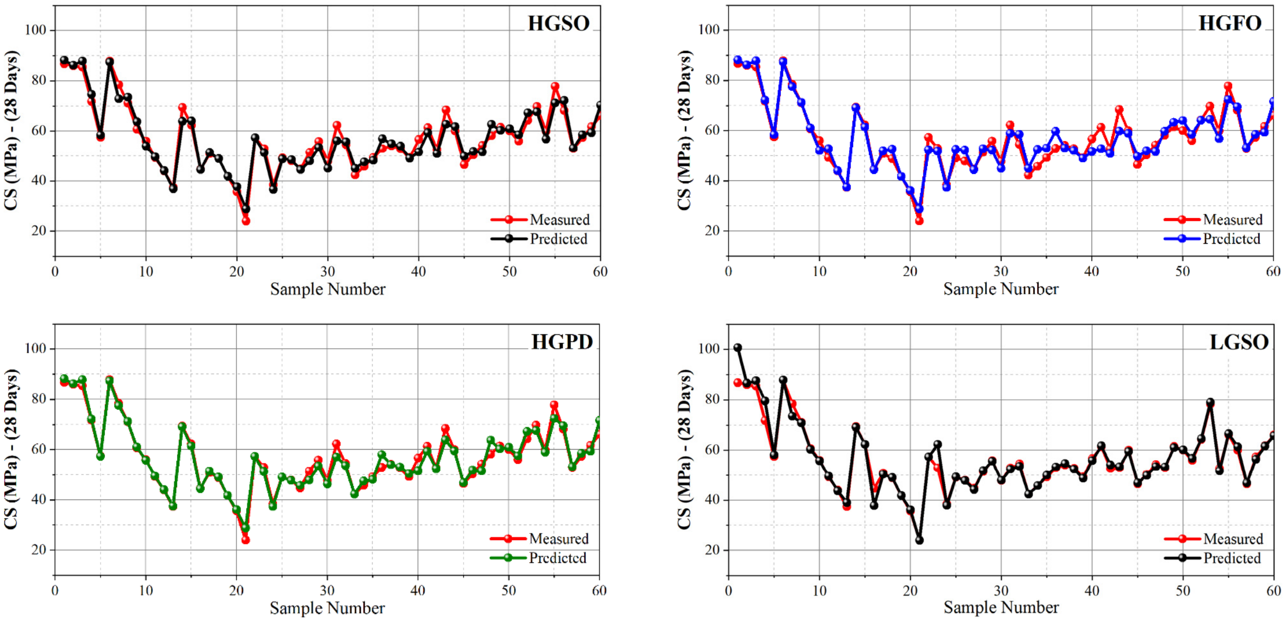


Fig. 12 Comparison of predicted values and measured values based on 28 days for CS

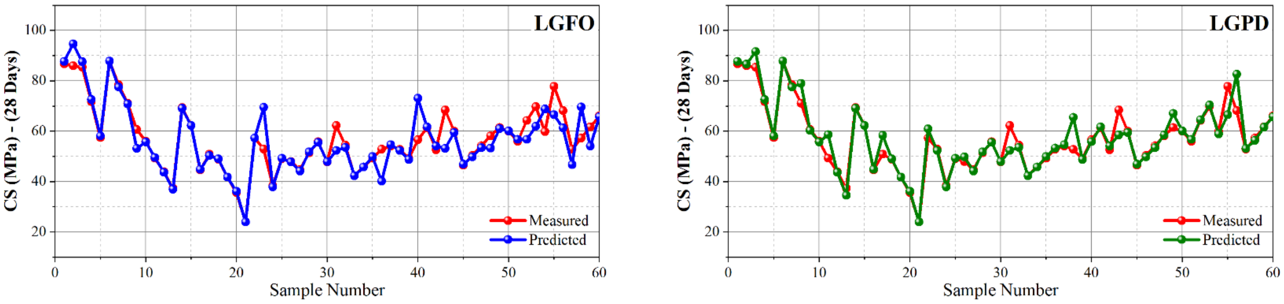


Fig. 12 Continued

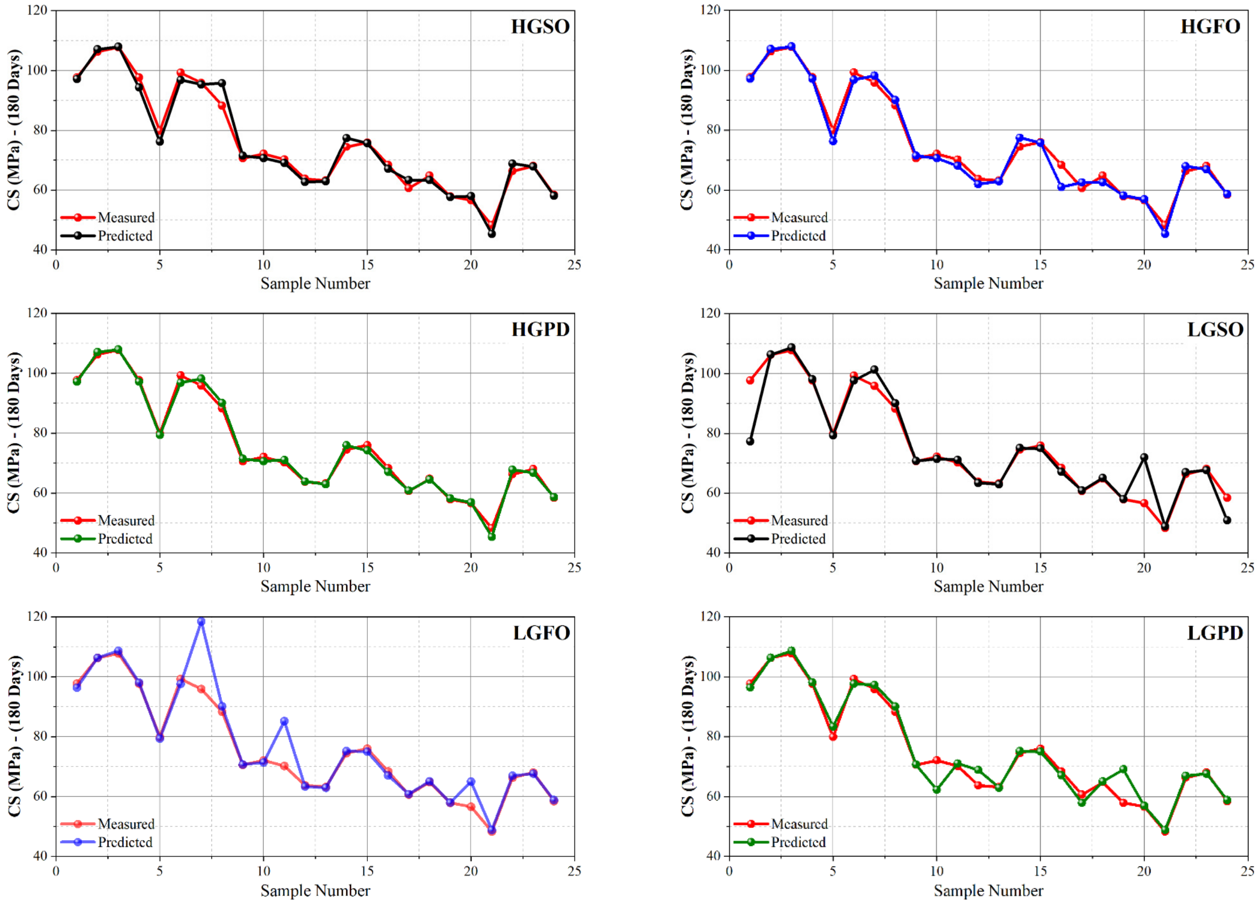


Fig. 13 Comparison of predicted values and measured values based on 180 days for CS

based on two key metrics: RMSE, which indicates the scatter in the figure, and R², which reflects the collinearity of the sample points. A high R² value and a low RMSE value denote the ideal state of prediction accuracy. The center line, represented by X = Y, signifies that samples closer to this line indicate better performance, while greater dispersion from this line suggests poorer performance.

Additionally, two dashed lines are drawn 15% above and below the X = Y line to visually highlight overestimation and underestimation tendencies in the data. In the case of HGB hybrid models, the section dedicated to PDO optimization shines with exceptional performance, showcasing the most commendable metrics among all optimizers. With a deficient RMSE value of 1.15 and an impressive R² value of 0.996, it outperforms other optimization methods. According to LGBM-based models, PDO has crafted a well-suited hybrid model alongside the corresponding one (LGPD). This symbiosis is evident as sample points closely align with or fall precisely on the central line, while the two fitting lines exhibit minimal angular disparity (RMSE = 1.63 and R² = 0.992). Transitioning to the ensemble-hybrid model, remarkable strides are evident, with both training and testing sections revealing R² values exceeding 0.99. In summary, the HGPD model emerges as the top performer, ensuring that sample points neither surpass nor fall short of estimations, thus underscoring its robust accuracy. Conversely, the LGMB single model displayed the weakest applicability in CS prediction.

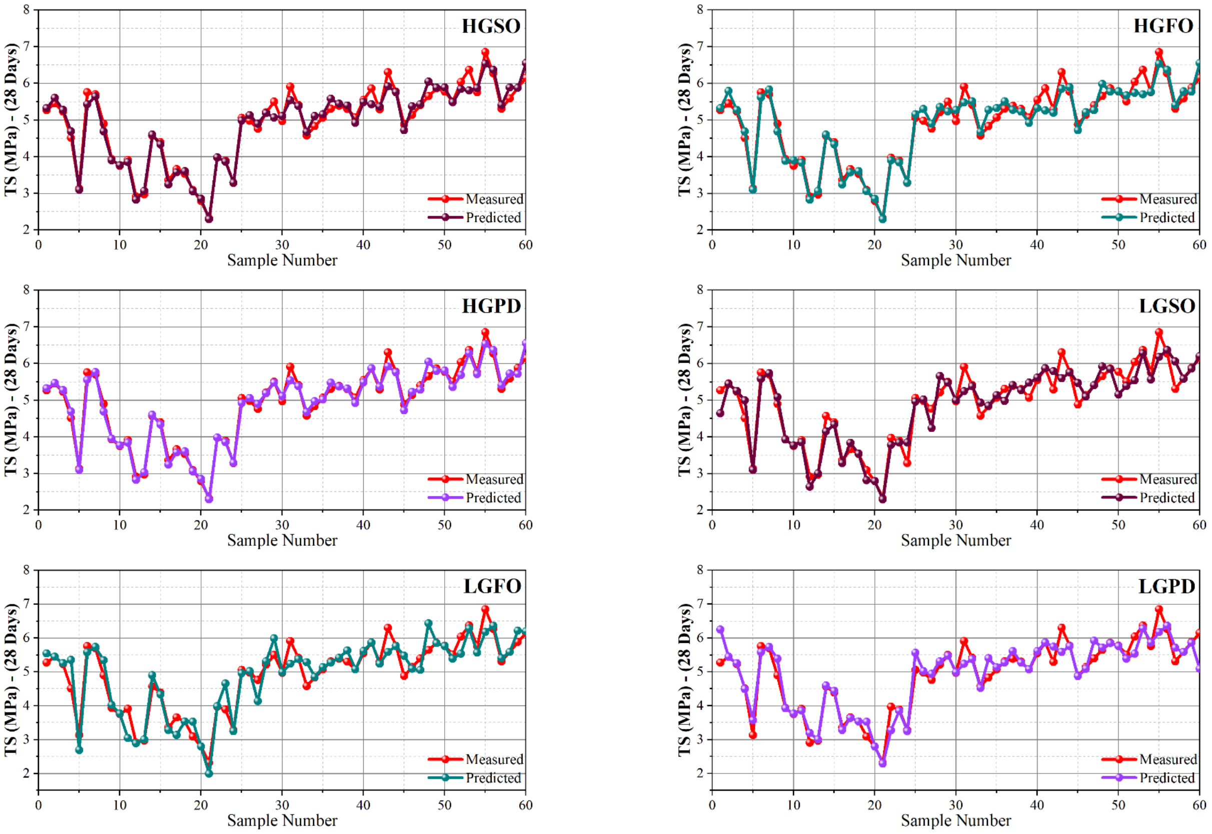


Fig. 14 Comparison of predicted values and measured values based on 28 days for TS

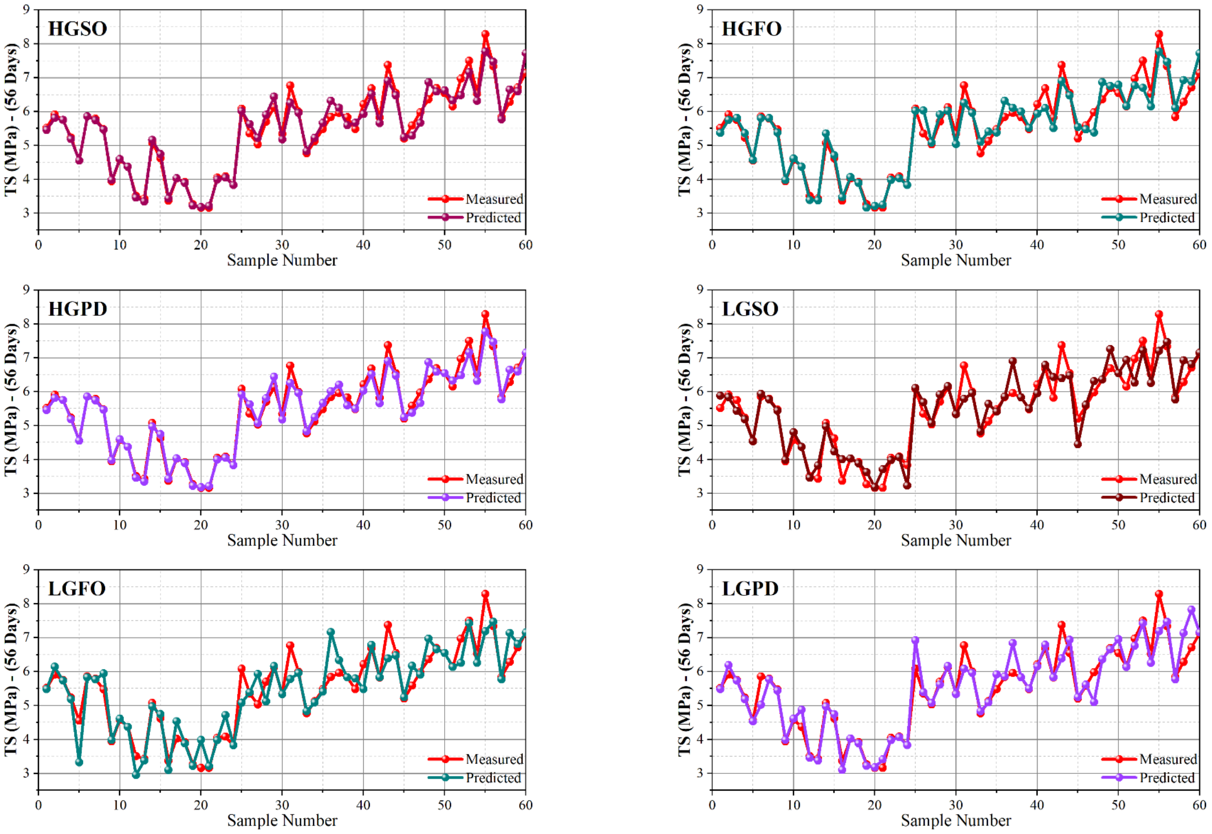


Fig. 15 Comparison of predicted values and measured values based on 56 days for TS

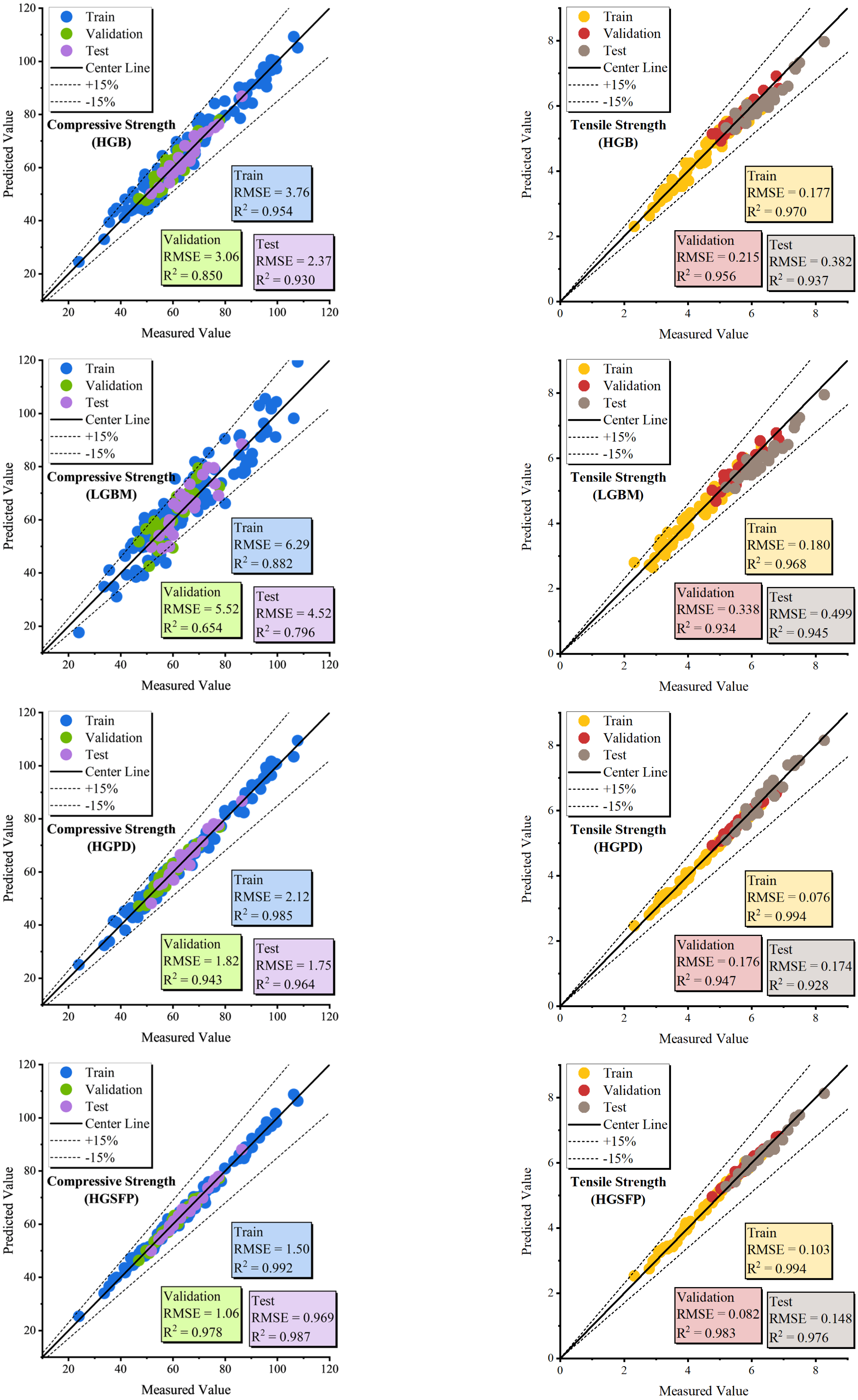


Fig. 16 The scatter plot of predicted models for CS and TS

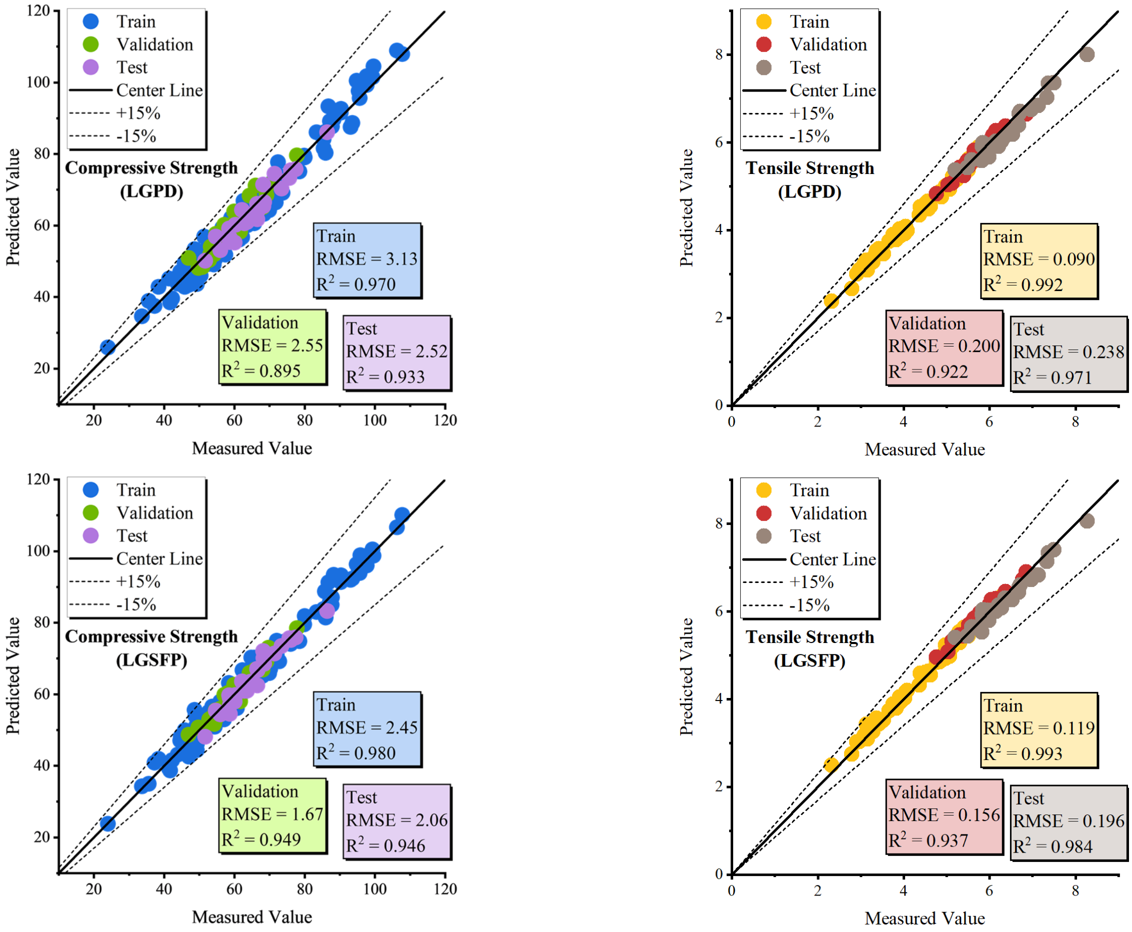


Fig. 16 Continued

- *TS*

Fig. 16 provides scatterplots that visually represent the predicted and measured TS values. Notably, within the LGBM hybrid model section, PDO showcases outstanding performance, evident from its achievement of the lowest RMSE value (0.09) and the highest R² value (0.992) among all optimizers. Transitioning to the HGB-based models,

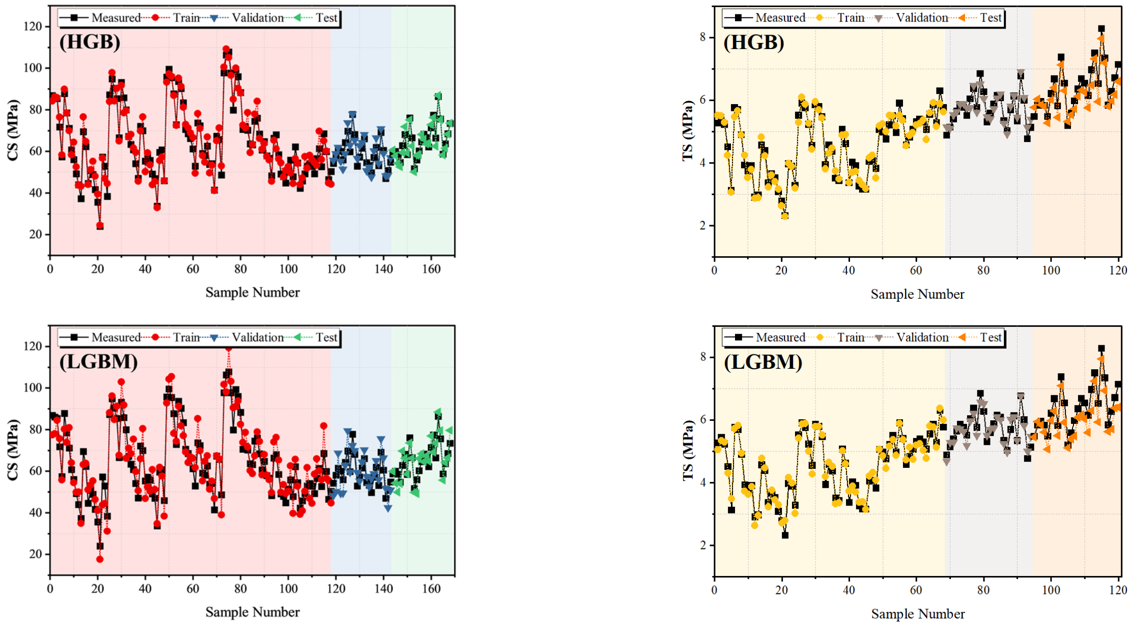


Fig. 17 Line and Symbol for the correlation between the predicted and measured models in CS and TS

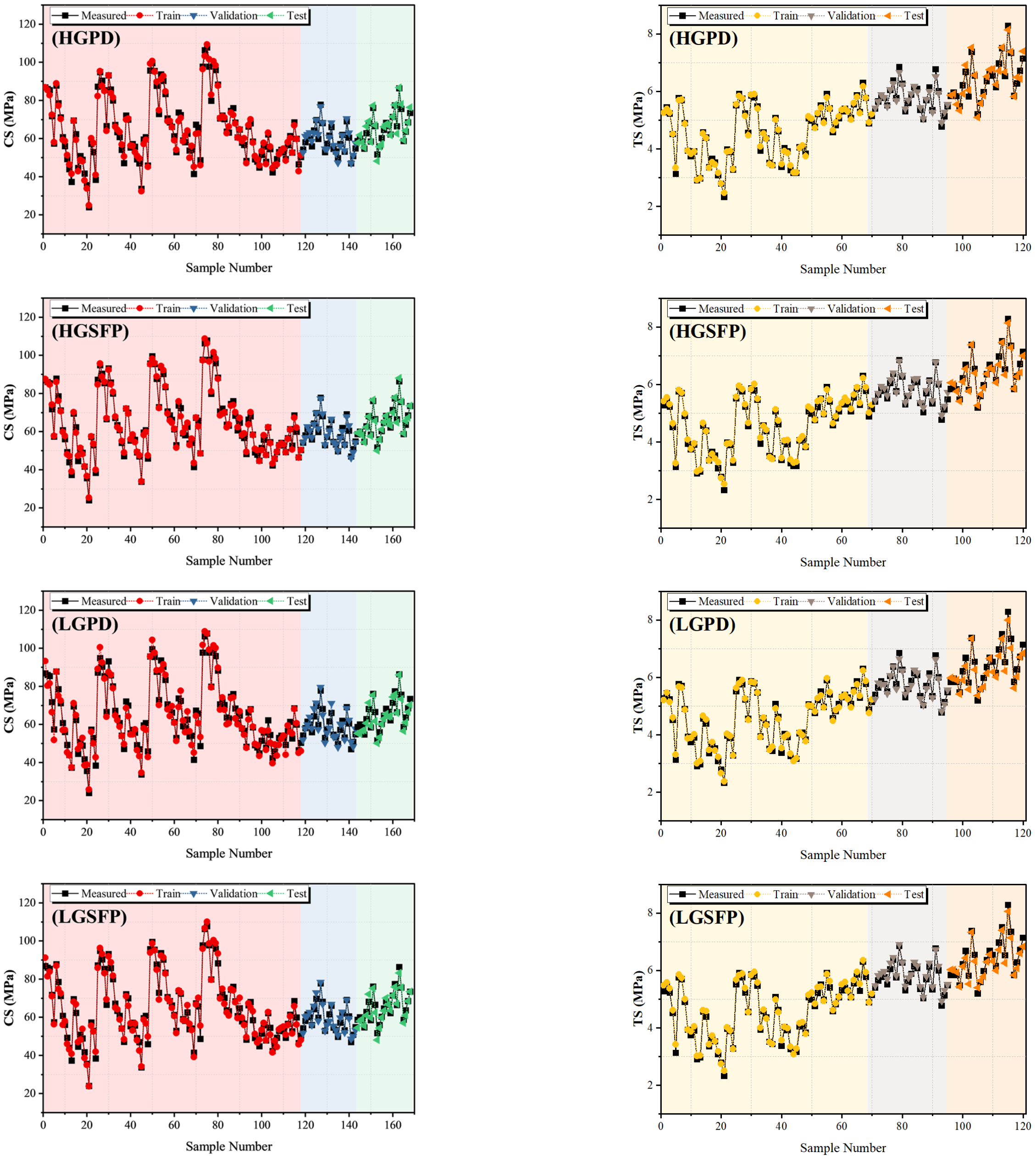


Fig. 17 Continued

PDO skillfully establishes a robust coupling model with its corresponding counterpart. This integration ensures that sample points closely adhere to or align near the central line, effectively minimizing disparities between predicted and measured values. Furthermore, the HGSFP ensemble-hybrid model signifies a significant advancement in model refinement, with R^2 values nearing perfection (0.994) and RMSE values approaching zero (0.103). In a broader context, the HGPD model emerges as the superior performer, demonstrating precision in estimation without overestimation or underestimation for any given point.

In Fig. 17, a comparison chart presents predicted and measured samples of both CS and TS. Ideally, the size of the predicted charts should match that of the measured charts. Upon examining the figures pertaining to HGB-based models, it becomes apparent that samples in the training section outperform those in the test section, displaying a smaller difference than the measured values. Across both CS and TS predictive results, among the three HGB-based models, HGPD demonstrates the lowest difference, while HGB single models exhibit the weakest performance in both phases. Similarly, the most favorable performance is associated with PDO within the LGB context. Evaluating the performance of hybrid models, it is observed in the test section that LGPD shows the lowest difference, while LGBM demonstrates the highest. The amalgamation of each model with an ensemble of three optimizers (HGSFP and LGSFP) results in improvement, achieving the lowest difference between the two components. Among ensemble-hybrid models, HGSFP showcases a more acceptable performance than other models. In summary, considering the results of predictors, it is evident that the combination of HGB and PDO proves to be more reliable in predicting both CS and TS values.

Fig. 18 displays the error percentage plots for CS and TS values. As mentioned, the samples are divided into training, validation, and testing phases, with proportions of 70%, 15%, and 15%, respectively. In the HGB-based models, HGPD demonstrates consistent performance across both stages, with an error rate of approximately 12% for CS and 7% for TS. Conversely, in the case of HGSFP, the highest error rate increases slightly to 13% in the training phase of CS and 9% in the training phase of TS prediction. On the other hand, HGB exhibits the least favorable performance, with error rates of 25% for CS and 10% for TS. This suggests that among the HGB models, the combination of HGB and PDO delivers acceptable performance. In the LGBM-based models, the lowest error rates are observed in LGPD, with 10% for CS and 5% for TS. LGSFP performs well, with error rates of 17% for CS and 9% for TS prediction values. Shifting focus to LGBM, the training phase of CS shows the highest error at 60%, while the training phase of TS displays a lower rate of 20%, with the maximum validation error recorded at 8%. In summary, the comprehensive error analysis highlights the superior performance of the ensemble-hybrid HGSFP and hybrid HGPD models compared to their counterparts for both CS and TS values.

Figs. 19 and 20 depict Error-Frequency and Box with Normal Distribution plots, respectively, showcasing the errors of the models mentioned above. These diagrams provide insights into error percentages and data distribution. When focusing on the CS section, it becomes evident that the weakest performance is associated with LGBM and

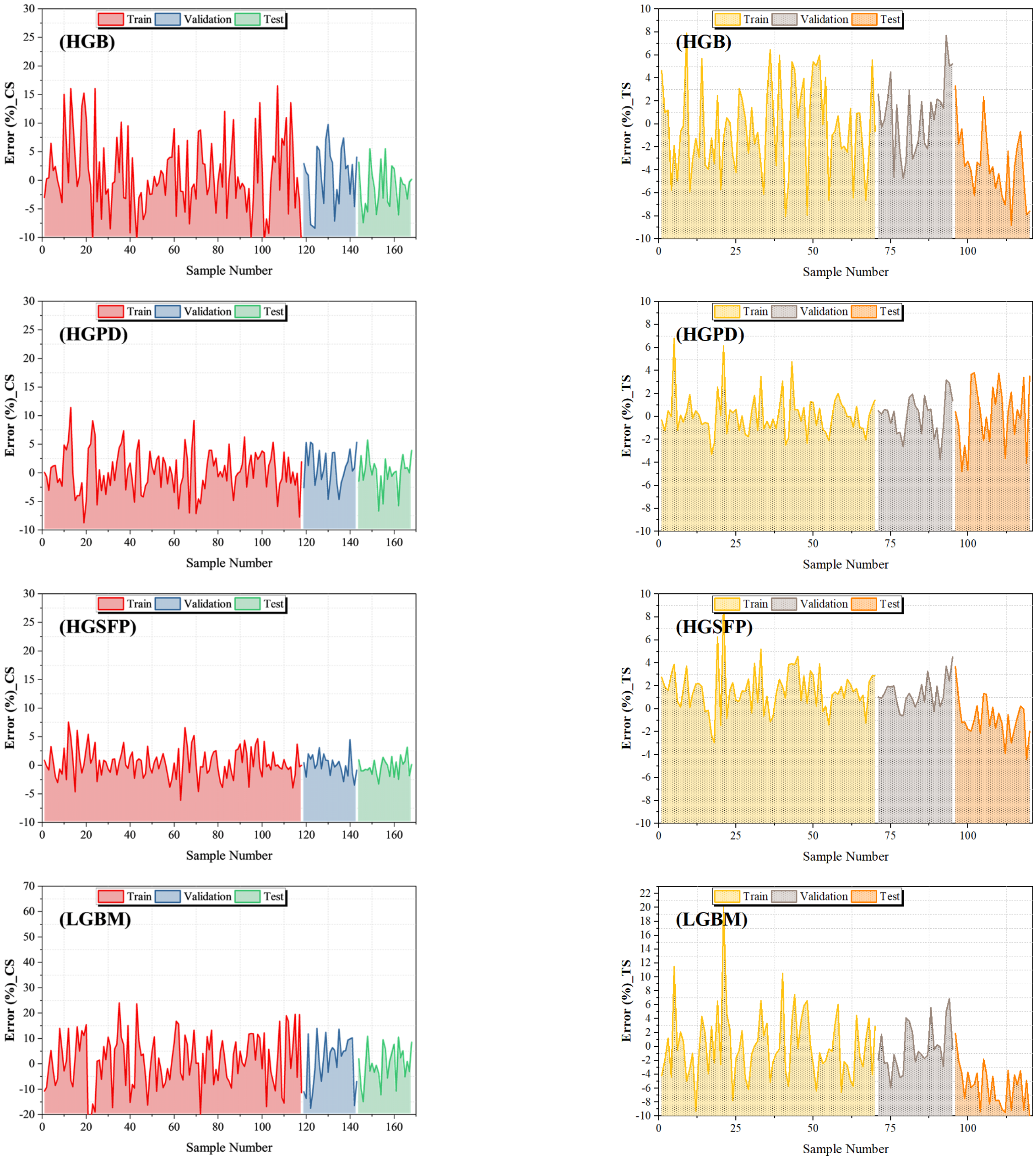


Fig. 18 Error percentage of presented models based on Line plot

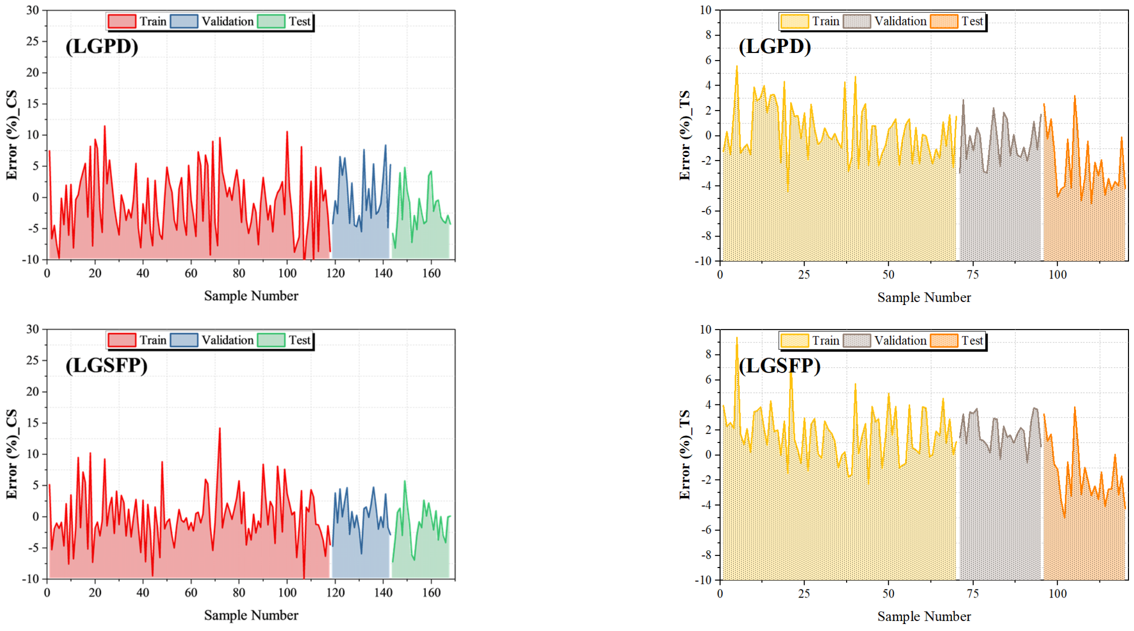


Fig. 18 Continued

HGB models, characterized by the highest dispersion and error percentages ranging from -10% to 60% for LGBM and -10% to 25% for HGB. In contrast, HGPD demonstrates the best performance, with most samples clustering close to zero and errors ranging between -6% and 14%. Shifting focus to plots related to TS, it is notable that LGBM exhibits the highest dispersion and error percentage compared to other models. The most favorable performance

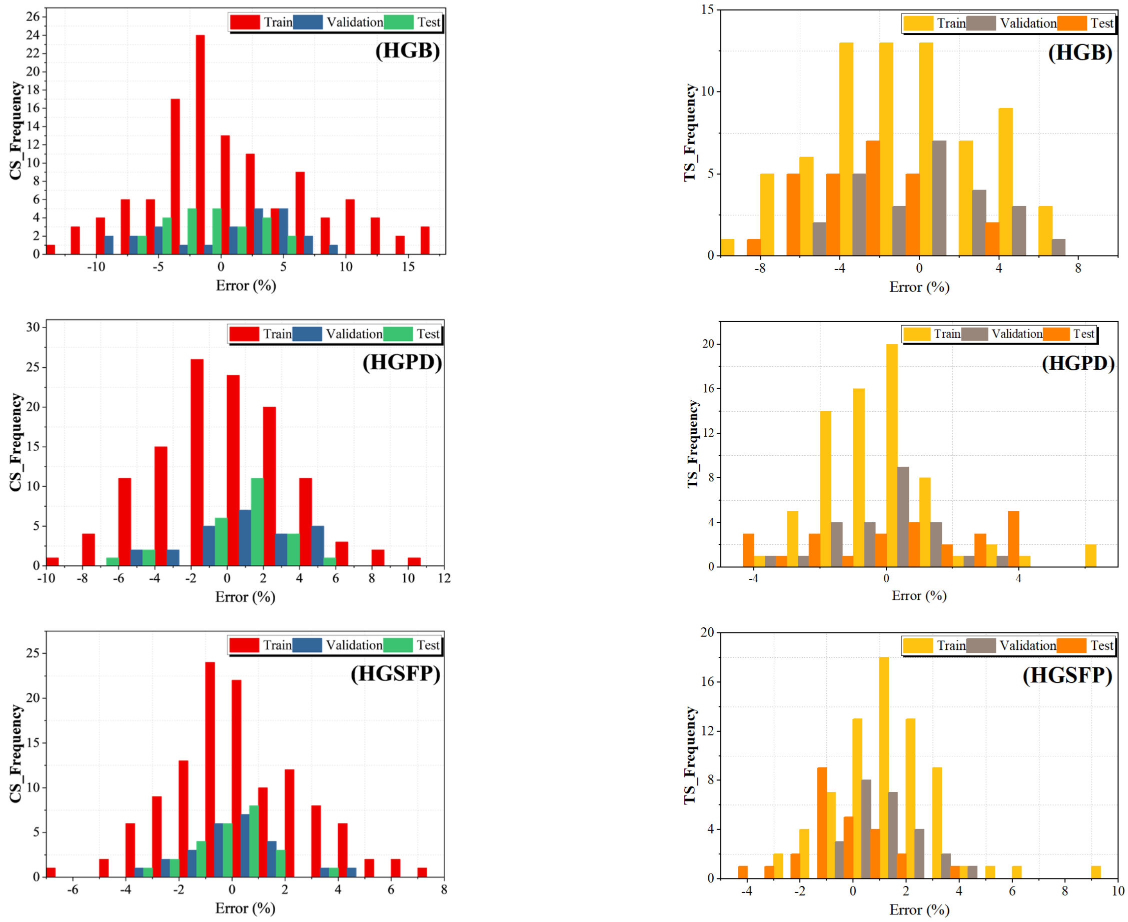


Fig. 19 Error frequency of presented models based on histogram plot

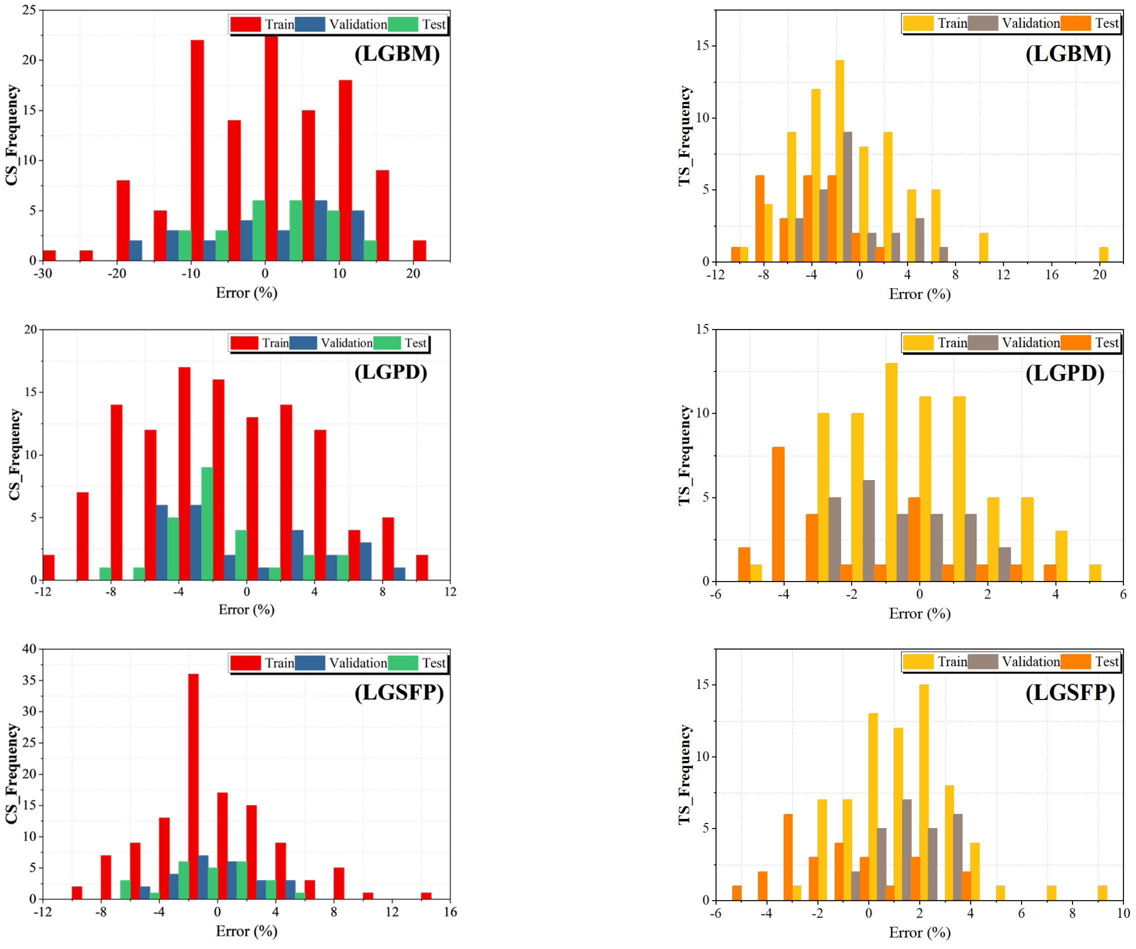


Fig. 19 Continued

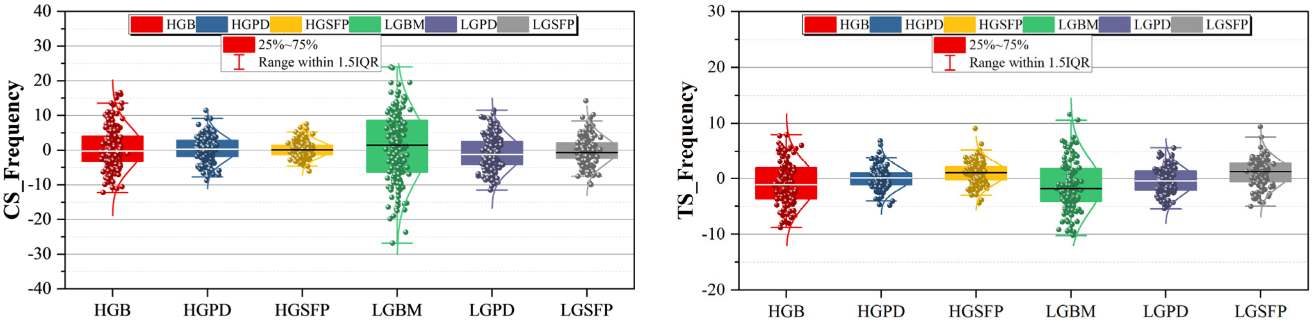


Fig. 20 Box with normal distribution for the error frequency of the developed models

is attributed to HGPD and HGSFP, which showcase the lowest error values. Notably, the 25%-75% of errors in HGPD fall within the range of [-2,2], while for the LGBM single model, it is [-5,5]. In conclusion, HGPD emerges as the most suitable model for achieving the least error across both CS and TS sections.

Fig. 21 illustrates the Taylor diagram used to compare the suggested models according to the correlation coefficient (R) and the standard deviation of projected and observed compressive strength and tensile strength. This diagram effectively showcases the fluctuation between expected and measured compressive strength. According to the analysis, the anticipated standard deviations of the HGPD and HGSFP models closely align with the experimental standard deviation, indicating their robust performance. Meanwhile, LGBM and HGB record poorer performances among the two models discussed in the paper. Further, Fig. 21 Taylor diagram shows TS prediction values. Notably, it is noted that the projected standard deviations of HGSFP and HGPD models fall very close to the experimental standard deviation, thereby ensuring reliability for the TS prediction. Among all the models presented, the worst fit was obtained by LGBM, while the best results were produced in both tensile strength predictions by HGPD.

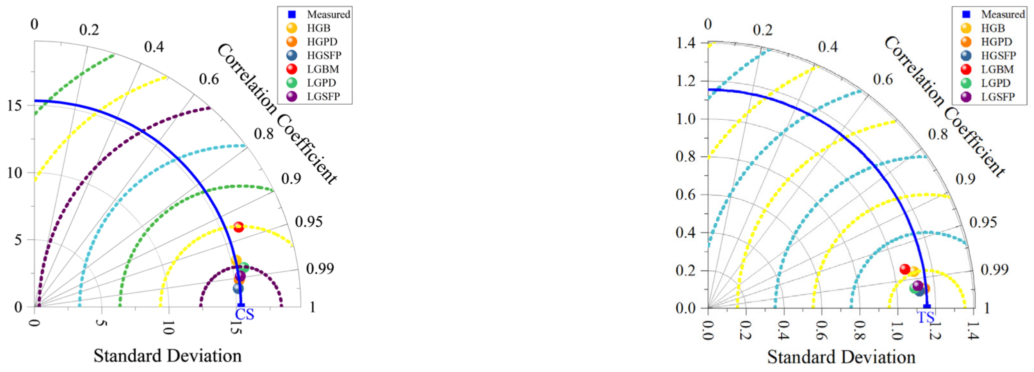


Fig. 21 Taylor diagram of related models for CS and TS

Table 8 Comparing the results of the present study with previous studies

Articles	Models	Models' performance	
		RMSE	R^2
(Bui *et al.* 2018) (CS)	ANN+MFA	4.85	0.95
(Nguyen *et al.* 2021) (CS)	MLP	4.52	0.96
(Nguyen *et al.* 2021) (CS)	GPR	3.77	0.97
(Nguyen *et al.* 2021) (CS)	XGBoost	2.47	0.97
(Nguyen *et al.* 2021) (CS)	SVR	5.11	0.95
Present Study (CS)	**HGP+SO+FO+PDO (HGSFP)**	**1.3804**	**0.9919**
(Bui *et al.* 2018) (CS)	ANN+MFA	0.38	0.96
(Nguyen *et al.* 2021) (CS)	MLP	0.39	0.96
(Nguyen *et al.* 2021) (CS)	GPR	0.39	0.96
(Nguyen *et al.* 2021) (CS)	XGBoost	0.38	0.96
(Nguyen *et al.* 2021) (CS)	SVR	0.39	0.96
Present Study (TS)	**HGB+PDO (HGPD)**	**0.076**	**0.994**

4.3 Comparison between the present study and previous research

Table 8 provides a comprehensive overview of previous investigations in the context of CS and TS prediction, which have employed various models such as Artificial Neural Networks (ANN) (Bui *et al.* 2018), Multilayer Perceptron (MLP) (Nguyen *et al.* 2021), Support Vector Regression (SVR) (Nguyen *et al.* 2021), Gaussian Process Regression (GPR) (Nguyen *et al.* 2021), among others.

These studies have been compared to those results obtained from the current study. In this research, the HGPD model emerges as a clear winner of the study, since it outperforms the outcome acquired from all of the above models. The HGSFP model achieves unmatched accuracy and precision in prediction while training for the CS and TS values. The HGPD model significantly outperforms its predecessors with an R^2 value of 0.9919 and a root mean square error of 1.3804 in CS prediction, and its R^2 value is 0.9947 with a root mean square error of 0.0766 regarding TS prediction. These excellent performance metrics set the HGPD model as the gold standard in CS and TS prediction predictive accuracy. This reflects excellent effectiveness and reliability in supremacy over models in vogue, leaving it more desirable in arriving at accurate and precise predictive values of CS and TS over various practical applications.

4.6 Sensitivity analysis

The SHAP (SHapley Additive exPlanations) analysis contributes to the input parameters on the prediction models regarding CS and TS. This approach highlights the contribution of every individual input in that aspect, which should show how parameters change as their impact on model outputs; higher SHAP values indicate greater importance, showing whether a parameter increases or decreases the strength via positive or negative shift. Indications of a critical factor in CS and TS predictions follow the W/B ratio. It might be observed from the SHAP plots that for higher W/B values corresponding to a blue color forecast, and strengths are positively influenced, while for the lower values (in red), strengths are reduced. FA/B also makes an essential contribution in this regard, particularly in the TS model, where changes in its value affect the predictions. Other features, like the CA/B ratio, are more spread out in their SHAP values, showing a weaker and less consistent impact. For the TS predictions, the W/B, FA/B, and MS/B ratios have high contributions from more than one variable. This postulates that slight

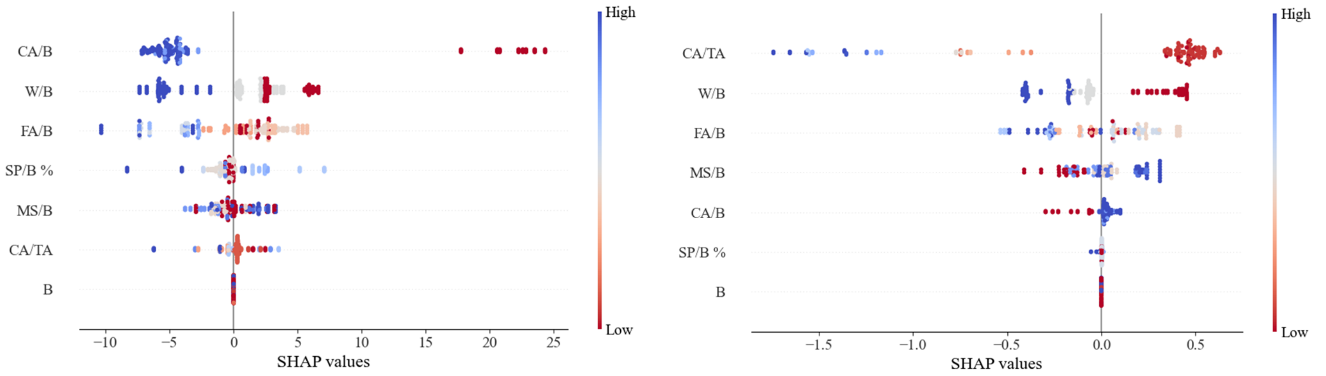


Fig. 22 Sensitivity of inputs on CS and TS

changes in these ratios will change the output from the model. The CA/TA ratio has a minimal effect since the SHAP values congregate around zero, showing a lower impact on the TS predictions. The W/B ratio, therefore, plays a significant role for the CS since higher SHAP values confirm its critical importance.

On the other hand, it is observed that the MS/B ratio also has a meaningful impact and contributes positively to strength predictions. In contrast, the CA/B and CA/TA ratios have relatively low SHAP values, indicating a limited role in CS outcomes. This sensitivity analysis highlights the most important input variables and provides insight into model behavior for different input conditions. This indicates that W/B and MS/B ratios are the most important in making more accurate strength predictions, while CA/B and CA/TA ratios bear less predictive power. Such an understanding of relationships allows for targeted optimization, focusing on the most influential parameters to enhance the model's performance and reliability. The SHAP-based sensitivity analysis also agrees with the nature of the mechanical properties of concrete, as it believes water content, binder ratios, and fine aggregates play a primary role in the strength determination. The present in-depth analysis can help identify the critical input parameters and assist in practical adjustments for better predictions regarding different datasets and conditions.

5. Conclusions

The research mainly aimed to apply advanced ML techniques to predict CS and TS for high-performance concrete accurately. To achieve this, complicated models like the Lightweight Boosting algorithm, henceforth LGB, and Hist Gradient Boosting, henceforth HGB, were used. Besides, in order to improve the predictive performance and strengths of the models, hybrid models were developed by incorporating the optimization algorithms: Snake Optimization Algorithm (SO), Fox Optimization Algorithm (FO), and Prairie Dog Optimization Algorithm (PDO). Additionally, ensemble models that integrated all three mentioned optimizers, namely SO-FO-PDO, were developed, which can take full advantage of different optimization methods. This holistic approach was adopted in the present study to ensure the development of sound and trustworthy models for the predictions of CS and TS for applications in HPC. The primary outcomes of this work are summarized below, separately for CS and TS:

- ➢ Compressive strength (CS):
  - Results have emphasized the model's outstanding accuracy in estimating CS values. In this respect, including SO, FO, PDO, and combining these methods (SO-FO-PDO) into hybrid models, HGB and LGB increased the R-squared value. Indeed, increases of 1.89% for HGSO, 1.07% for HGFO, 2.5% for HGPD, 2.48% for HGSFP, 0.63% for LGSO, 1.64% for LGFO, 2.44% for LGPD, and finally 2.51% for LGSFP are registered. This enhancement underscores the effectiveness of integrating these optimization techniques within the HGB and LGBM frameworks to augment the accuracy of CS value predictions.
  - The HGPD model distinguishes itself with exceptional precision and performance compared to alternative models such as LGPD and HGSFP. It attained a low RMSE value of 1.1553 and the highest R-squared (R^2) value of 0.9961, showcasing its remarkable accuracy in estimating CS values. Conversely, the LGBM model exhibited poor performance, with the highest error value (RMSE of 8.1352) and the lowest R^2 value (0.9763).
- ➢ Tensile Strength (TS):
  - By incorporating the aforementioned optimization techniques into the HGB and LGBM models, notable improvements (reductions) in RMSE values were achieved, with reductions of 13% for HGSO, 20.6% for HGFO, 50.64% for HGPD, 49.41% for HGSFP, 20.56% for LGSO, 25.51% for LGFO, 75.48% for LGPD, and finally, 46.74% for LGSFP. Based on these results, it can be inferred that the HGPD model exhibited the most precise performance in predicting TS values of HPC.
  - During the testing phase, the LGBM model emerged as the least effective predictor, exhibiting the highest error values (RMSE of 0.4995 and MAE of 0.4736) and the lowest R-squared (R^2) value of 0.9348.
  - The HGPD model's outstanding precision and performance distinguish it from all other models. It attained low values in error-based metrics such as

RMSE (0.0766) and MAE (0.0594), along with the highest R-squared (R^2) value of 0.9947. These results underscore its accuracy in estimating TS values.

In summary, it is noteworthy that the HGB and LGBM single models encountered difficulties in accurately predicting both CS and TS values, as evidenced by their high error rates and relatively low R^2 values. In contrast, the HGPD model, which combines HGB with the PDO optimizer, provides precise predictions for both CS and TS. Additionally, the HGSFP model, a hybridized model based on HGB and an ensemble of SO-FO-PDO optimizers, demonstrates the significant impact of incorporating additive optimization techniques such as SO, FO, PDO, and SO-FO-PDO in enhancing the accuracy and overall performance of the HGB and LGBM models.

References

Akbar, A. and Liew, K.M. (2020), "Assessing recycling potential of carbon fiber reinforced plastic waste in production of eco-efficient cement-based materials", *J. Cleaner Product.*, **274**, 123001. https://doi.org/10.1016/j.jclepro.2020.123001

Akbarzadeh, M.R., Ghafourian, H., Anvari, A., Pourhanasa, R. and Nehdi, M.L. (2023), "Estimating compressive strength of concrete using neural electromagnetic field optimization", *Materials*, **16**(11), p. 4200. https://doi.org/doi.org/10.3390/ma16114200

Algaifi, H.A., Bakar, S.A., Alyousef, R., Sam, A.R.M., Alqarni, A.S., Ibrahim, M., Shahidan, S., Ibrahim, M. and Salami, B.A. (2021), "Machine learning and RSM models for prediction of compressive strength of smart bio-concrete", *Smart Struct. Syst.*, *Int. J.*, **28**, 535-551. https://doi.org/10.12989/sss.2021.28.4.535

Azim, I., Yang, J., Iqbal, M.F., Mahmood, Z., Javed, M.F., Wang, F. and Liu, Q.F. (2021), "Prediction of catenary action capacity of RC beam-column substructures under a missing column scenario using evolutionary algorithm", *KSCE J. Civil Eng.*, **25**, 891-905. https://doi.org/10.1007/s12205-021-0431-0

Basheer, I.A. and Hajmeer, M. (2000), "Artificial neural networks: fundamentals, computing, design, and application", *J. Microbiol. Methods*, **43**(1), 3-31. https://doi.org/10.1016/S0167-7012(00)00201-3

Brodley, C.E. and Utgoff, P.E. (1995), "Multivariate decision trees", *Machine Learn.*, **19**(1), 45-77. https://doi.org/10.1007/BF00994660

Bui, D.-K., Nguyen, T., Chou, J.-S., Nguyen-Xuan, H. and Ngo, T.D. (2018), "A modified firefly algorithm-artificial neural network expert system for predicting compressive and tensile strength of high-performance concrete", *Constr. Build. Mater.*, **180**, 320-333. https://doi.org/10.1016/j.conbuildmat.2018.05.201

Chang, F.-J. and Chang, Y.-T. (2006), "Adaptive neuro-fuzzy inference system for prediction of water level in reservoir", *Adv. Water Resour.*, **29**(1), 1-10. https://doi.org/10.1016/j.advwatres.2005.04.015

Chen, L. (2003), "Study of applying macroevolutionary genetic programming to concrete strength estimation", *J. Comput. Civil Eng.*, **17**(4), 290-294. https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(290)

Detwiler, R.J., Bhatty, J.I. and Battacharja, S. (1996), *Supplementary cementing materials for use in blended cements*, No. R&D Bulletin RD112T.

Ekanayake, I.U., Meddage, D.P.P. and Rathnayake, U. (2022), "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.*, **16**, p. e01059. https://doi.org/10.1016/j.cscm.2022.e01059

Erdal, H.I., Karakurt, O. and Namli, E. (2013), "High performance concrete compressive strength forecasting using ensemble models based on discrete wavelet transform", *Eng. Applicat. Artif. Intell.*, **26**(4), 1246-1254. https://doi.org/10.1016/j.engappai.2012.10.014

Ezugwu, A.E., Agushaka, J.O., Abualigah, L., Mirjalili, S. and Gandomi, A.H. (2022), "Prairie dog optimization algorithm", *Neural Comput. Applicat.*, **34**(22), 20017-20065. https://doi.org/10.1007/s00521-022-07530-9

Fan, J., Ma, X., Wu, L., Zhang, F., Yu, X. and Zeng, W. (2019), "Light Gradient Boosting Machine: An efficient soft computing model for estimating daily reference evapotranspiration with local and external meteorological data", *Agricult. Water Manage.*, **225**, p. 105758. https://doi.org/10.1016/j.agwat.2019.105758

Geng, X., Moayedi, H., Pan, F. and Foong, L.K. (2021), "Predicting the concrete compressive strength through MLP network hybridized with three evolutionary algorithms", *Smart Struct. Syst.*, *Int. J.*, **28**(5), 711-725. https://doi.org/10.12989/sss.2021.28.5.711

Ghanbari, S., Shahmansouri, A.A., Akbarzadeh Bengar, H. and Jafari, A. (2023), "Compressive strength prediction of high-strength oil palm shell lightweight aggregate concrete using machine learning methods", *Environ. Sci. Pollut. Res.*, **30**(1), 1096-1115. https://doi.org/10.1007/s11356-022-21987-0

Gholampour, A., Mansouri, I., Kisi, O. and Ozbakkaloglu, T. (2020), "Evaluation of mechanical properties of concretes containing coarse recycled concrete aggregates using multivariate adaptive regression splines (MARS), M5 model tree (M5Tree), and least squares support vector regression (LSSVR) models", *Neural Comput. Applicat.*, **32**(1), 295-308. https://doi.org/10.1007/s00521-018-3630-y

Guryanov, A. (2019), "Histogram-based algorithm for building gradient boosting ensembles of piecewise linear decision trees", In: *Analysis of Images*, *Social Networks and Texts*: *8th International Conference*, *AIST 2019*, Kazan, Russia, July. Revised Selected Papers 8, pp. 39-50.

Haeri, H., Mirshekari, N., Sarfarazi, V. and Marji, M.F. (2020), "Numerical simulation of compressive to tensile load conversion for determining the tensile strength of ultra-high performance concrete", *Smart Struct. Syst.*, *Int. J.*, **26**(5), 605-617. https://doi.org/10.12989/sss.2020.26.5.605

Han, Q., Gui, C., Xu, J. and Lacidogna, G. (2019), "A generalized method to predict the compressive strength of high-performance concrete by improved random forest algorithm", *Constr. Build. Mater.*, **226**, 734-742. https://doi.org/10.1016/j.conbuildmat.2019.07.315

Hansen, T.C. (1990), "Long-term strength of high fly ash concretes", *Cement Concrete Res.*, **20**(2), 193-196. https://doi.org/10.1016/0008-8846(90)90071-5

Hashim, F.A. and Hussien, A.G. (2022), "Snake Optimizer: A novel meta-heuristic optimization algorithm", *Knowled.-Based Syst.*, **242**. https://doi.org/10.1016/j.knosys.2022.108320

Hoogland, J.L. (1995), *The black-tailed prairie dog*: *social life of a burrowing mammal*, University of Chicago Press.

Hu, P., Moradi, Z., Ali, H.E. and Foong, L.K. (2022), "Metaheuristic-reinforced neural network for predicting the compressive strength of concrete", *Smart Struct. Syst.*, *Int. J.*, **30**(2), 195-207. https://doi.org/10.12989/sss.2022.30.2.195

Huang, L., Jiang, W., Wang, Y., Zhu, Y. and Afzal, M. (2022), "Prediction of long-term compressive strength of concrete with admixtures using hybrid swarm-based algorithms", *Smart*

Struct. Syst., Int. J., **29**(3), 433-444. https://doi.org/10.12989/sss.2022.29.3.433

Hubertova, M. and Hela, R. (2007), "The effect of metakaolin and silica fume on the properties of lightweight self consolidating concrete", *Special Publication*, **243**, 35-48. https://doi.org/10.14359/18741

Iqbal, M.F., Liu, Q., Azim, I., Zhu, X., Yang, J., Javed, M.F. and Rauf, M. (2020), "Prediction of mechanical properties of green concrete incorporating waste foundry sand based on gene expression programming", *J. Hazard. Mater.*, **384**, p. 121322. https://doi.org/10.1016/j.jhazmat.2019.121322

Jafari, A., Ma, L., Shahmansouri, A.A. and Dugnani, R. (2023), "Quantitative fractography for brittle fracture via multilayer perceptron neural network", *Eng. Fract. Mech.*, **291**, p. 109545. https://doi.org/https://doi.org/10.1016/j.engfracmech.2023.109545

Javed, M.F., Amin, M.N., Shah, M.I., Khan, K., Iftikhar, B., Farooq, F., Aslam, F., Alyousef, R. and Alabduljabbar, H. (2020a), "Applications of gene expression programming and regression techniques for estimating compressive strength of bagasse ash based concrete", *Crystals*, **10**(9), p. 737. https://doi.org/10.3390/cryst10090737

Javed, M.F., Farooq, F., Memon, S.A., Akbar, A., Khan, M.A., Aslam, F., Alyousef, R., Alabduljabbar, H. and Rehman, S.K.U. (2020b), "New prediction model for the ultimate axial capacity of concrete-filled steel tubes: An evolutionary approach", *Crystals*, **10**(9), p. 741. https://doi.org/10.3390/cryst10090741

Kaloop, M.R., Kumar, D., Samui, P., Hu, J.W. and Kim, D. (2020), "Compressive strength prediction of high-performance concrete using gradient tree boosting machine", *Constr. Build. Mater.*, **264**, p. 120198. https://doi.org/10.1016/j.conbuildmat.2020.120198

Kamath, M.V., Prashanth, S., Kumar, M. and Tantri, A. (2022), "Machine-Learning-Algorithm to predict the High-Performance concrete compressive strength using multiple data", *J. Eng. Des. Technol.*, ahead-of-p(ahead-of-print). https://doi.org/10.1108/JEDT-11-2021-0637

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y. (2017), "Lightgbm: A highly efficient gradient boosting decision tree", *Adv. Neural Inform. Process. Syst.*, **30**.

Keshavarz, Z. and Torkian, H. (2018), "Application of ANN and ANFIS models in determining compressive strength of concrete", *J. Soft Comput. Civil Eng.*, **2**(1), 62-70. https://doi.org/10.22115/scce.2018.51114

Khajeh, A., Ebrahimi, S.A., MolaAbasi, H., Jamshidi Chenari, R. and Payan, M. (2021), "Effect of EPS beads in lightening a typical zeolite and cement-treated sand", *Bull. Eng. Geol. Environ.*, **80**(11), 8615-8632. https://doi.org/10.1007/s10064-021-02458-1

Khalil, M.J., Aslam, M. and Ahmad, S. (2021), "Utilization of sugarcane bagasse ash as cement replacement for the production of sustainable concrete–A review", *Constr. Build Mater.*, **270**, p. 121371. https://doi.org/10.1016/j.conbuildmat.2020.121371

Kohavi, R. (1995), "A study of cross-validation and bootstrap for accuracy estimation and model selection", In: *Ijcai*, Vol. 14, pp. 1137-1145, Montreal, Canada.

Lam, L., Wong, Y.L. and Poon, C.-S. (1998), "Effect of fly ash and silica fume on compressive and fracture behaviors of concrete", *Cement Concrete Res.*, **28**(2), 271-283. https://doi.org/10.1016/S0008-8846(97)00269-X

Li, Q.-F. and Song, Z.-M. (2022), "High-performance concrete strength prediction based on ensemble learning", *Constr. Build. Mater.*, **324**, p. 126694. https://doi.org/10.1016/j.conbuildmat.2022.126694

Margineantu, D.D. and Dietterich, T.G. (1997), "Pruning adaptive boosting", In: *ICML*, Vol. 97, pp. 211-218, Citeseer.

Memarzadeh, A., Shahmansouri, A.A. and Poologanathan, K. (2022), "A novel prediction model for post-fire elastic modulus of circular recycled aggregate concrete-filled steel tubular stub columns", *Steel Compos. Struct., Int. J.*, **44**(3), 309-324. https://doi.org/10.12989/scs.2022.44.3.309

Nguyen, H., Vu, T., Vo, T.P. and Thai, H.-T. (2021), "Efficient machine learning models for prediction of concrete strengths", *Constr. Build. Mater.*, **266**, p. 120950. https://doi.org/10.1016/j.conbuildmat.2020.120950

Nhat-Duc, H. and Van-Duc, T. (2023), "Comparison of histogram-based gradient boosting classification machine, random Forest, and deep convolutional neural network for pavement raveling severity classification", *Automat. Constr.*, **148**, p. 104767. https://doi.org/10.1016/j.autcon.2023.104767

Nour, A.I. and Güneyisi, E.M. (2019), "Prediction model on compressive strength of recycled aggregate concrete filled steel tube columns", *Compos. Part B: Eng.*, **173**, p. 106938. https://doi.org/10.1016/j.compositesb.2019.106938

Pala, M., Özbay, E., Öztaş, A. and Yuce, M.I. (2007), "Appraisal of long-term effects of fly ash and silica fume on compressive strength of concrete by neural networks", *Constr. Build. Mater.*, **21**(2), 384-394. https://doi.org/10.1016/j.conbuildmat.2005.08.009

Połap, D. and Woźniak, M. (2021), "Red fox optimization algorithm", *Expert Syst. Applicat.*, **166**. https://doi.org/10.1016/j.eswa.2020.114107

Qian, X., Lee, S., Soto, A. and Chen, G. (2018), "Regression model to predict the higher heating value of poultry waste from proximate analysis", *Resources*, **7**(3), p. 39. https://doi.org/10.3390/resources7030039

Raza, S.S., Amir, M.T., Azab, M., Ali, B., Abdallah, M., El Ouni, M.H. and Elhag, A.B. (2022), "Effect of micro-silica on the physical, tensile, and load-deflection characteristics of micro fiber-reinforced high-performance concrete (HPC)", *Case Stud. Constr. Mater.*, **17**, p. e01380. https://doi.org/10.1016/j.cscm.2022.e01380

Sadaghat, B., Ebrahimi, S.A., Souri, O., Yahyavi Niar, M. and Akbarzadeh, M.R. (2024), "Evaluating strength properties of Eco-friendly Seashell-Containing Concrete: Comparative analysis of hybrid and ensemble boosting methods based on environmental effects of seashell usage", *Eng. Applicat. Artif. Intell.*, **133**, p. 108388. https://doi.org/doi.org/10.1016/j.engappai.2024.108388

Shahmansouri, A.A., Yazdani, M., Ghanbari, S., Bengar, H.A., Jafari, A. and Ghatte, H.F. (2021), "Artificial neural network model to predict the compressive strength of eco-friendly geopolymer concrete incorporating silica fume and natural zeolite", *J. Cleaner Product.*, **279**, p. 123697. https://doi.org/10.1016/j.jclepro.2020.123697

Sun, Y., Li, G., Zhang, J. and Qian, D. (2019), "Prediction of the strength of rubberized concrete by an evolved random forest model", *Adv. Civil Eng.*, **2019**, 1-7. https://doi.org/10.1155/2019/5198583

Sun, X., Liu, M. and Sima, Z. (2020), "A novel cryptocurrency price trend forecasting model based on lightGBM", *Finan. Res. Lett.*, **32**, p. 101084. https://doi.org/10.1016/j.frl.2018.12.032

Tavana Amlashi, A., Mohammadi Golafshani, E., Ebrahimi, S.A. and Behnood, A. (2023), "Estimation of the compressive strength of green concretes containing rice husk ash: a comparison of different machine learning approaches", *Eur. J. Environ. Civil Eng.*, **27**(2), 961-983. https://doi.org/10.1080/19648189.2022.2068657

Tejani, G.G., Sadaghat, B. and Kumar, S. (2023), "Predict the Maximum Dry Density of soil based on Individual and Hybrid Methods of Machine Learning", *Adv. Eng. Intell. Syst.*, **2**(03), 98-109. https://doi.org/10.22034/aeis.2023.414188.1129

Turk, K., Turgut, P., Karatas, M. and Benli, A. (2010), "Mechanical properties of selfcompacting concrete with silica fume/fly ash", In: *The 9th International Congress on Advances in Civil Engineering*, Trabzon, Turkey, September, pp. 27-30.

Wang, L. (2005), *Support Vector Machines*: *Theory and Applications*, Vol. 177, Springer Science & Business Media.

Wang, L., Chen, L., Tsang, D.C., Guo, B., Yang, J., Shen, Z., Hou, D., Ok, Y.S. and Poon, C.S. (2020), "Biochar as green additives in cement-based composites with carbon dioxide curing", *J. Cleaner Product*., **258**, p. 120678. https://doi.org/10.1016/j.jclepro.2020.120678

Zhang, J., Ma, G., Huang, Y., Aslani, F. and Nener, B. (2019), "Modelling uniaxial compressive strength of lightweight self-compacting concrete using random forest regression", *Constr. Build. Mater*., **210**, 713-719. https://doi.org/10.1016/j.conbuildmat.2019.03.189

Zhang, N., Duan, H., Miller, T.R., Tam, V.W.Y., Liu, G. and Zuo, J. (2020), "Mitigation of carbon dioxide by accelerated sequestration in concrete debris", *Renew. Sustain. Energy Rev*., **117**, p. 109495. https://doi.org/10.1016/j.rser.2019.109495