https://doi.org/10.3311/PPci.41258

Periodica Polytechnica Civil Engineering, 69(4), pp. 1124–1148, 2025

Metaheuristic-optimized Machine Learning for Mechanical Property Prediction in Eco-friendly Rubberized Concrete

Ali Kaveh[1]*, Amir Eskandari[1], Mahroo Piri[1]

[1] School of Civil Engineering, Iran University of Science and Technology, P. O. B. 16765-163, Narmak, 16846-13114 Tehran, Iran
* Corresponding author, e-mail: alikaveh@iust.ac.ir

Received: 12 June 2025, Accepted: 28 August 2025, Published online: 18 September 2025

Abstract

The disposal of discarded tires presents a global environmental challenge, but recycling them into rubber particles for concrete offers a sustainable solution, reducing waste while creating eco-friendly construction materials. However, experimental studies in this field are resource-intensive, requiring significant time and financial investment. This research develops 99 machine learning (ML) models, optimized using three improved metaheuristic algorithms, to predict the mechanical properties of rubberized concrete. The eight ML models include Adaptive Boosting, Artificial Neural Networks, Decision Tree, Gradient Boosting, K-Nearest Neighbors, Random Forest, Support Vector Machines, and Extreme Gradient Boosting, refined through the Improved Hybrid Growth Optimizer, Improved Ray Optimization, and Improved Sand Cat Swarm Optimization algorithms. A dataset of 315 experimental cases, incorporating six input variables, rubber size, rubber weight, cement content, water content, coarse aggregate, and fine aggregate, was analyzed. In addressing missing data for certain mechanical properties, a two-level sequential artificial neural network was employed. The study revealed that the XGBoost-IRO hybrid model excelled in predicting compressive and tensile strength, while the AdaBoost-ISCSO ensemble was best for flexural strength, and the XGBoost-IHGO model performed optimally for modulus of elasticity. Partial Dependence Plots and SHAP analysis highlighted the complex relationships between input variables and mechanical properties, confirming the significance of all input features. Validation through Taylor diagrams and error distribution further confirmed the reliability of the models in predicting all mechanical properties.

Keywords

rubberized concrete, mechanical properties, eco-friendly material, machine learning, hyperparameters, metaheuristic algorithms, optimization

1 Introduction

As population growth and urbanization accelerate, the demand for infrastructure planning rises [1, 2], driving construction sector expansion [1]. In this regard, concrete is widely used due to its strength, durability, cost-effectiveness, and versatility [2]. Concrete consists of coarse/fine aggregates, water, cement, and additives, irregularly dispersed. Nonetheless, the manufacture of concrete carries environmental costs, including habitat destruction and soil erosion from quarrying natural aggregates [3]. To address these impacts, researchers have explored sustainable alternatives to natural aggregates with recycled materials like plastics, glass, and tire rubber [4].

End-of-life tire disposal is a critical global environmental issue [5]. Millions of non-biodegradable tires accumulate annually, threatening ecosystems and human health through improper disposal and long-term ecological damage [6]. There are many methods to get rid of scrap tires, including recycling them into rubber products, using them as filler materials, or converting them to fuel. While incineration was historically preferred for its cost-effectiveness, it is now recognized to cause significant fire risks and environmental contamination of air, water, and soil [6]. These concerns have driven research into safer alternatives.

In recent decades, rubberized concrete, incorporating crushed tire particles as partial aggregate replacements, has gained prominence. Many researchers have demonstrated that the inclusion of waste rubber in concrete mixtures diminishes the compressive, tensile, and flexural strength and elastic modulus [7]. The complete substitution of natural coarse aggregates with coarse rubber aggregates resulted

Cite this article as: Kaveh, A., Eskandari, A., Piri, M. "Metaheuristic-optimized Machine Learning for Mechanical Property Prediction in Eco-friendly Rubberized Concrete", Periodica Polytechnica Civil Engineering, 69(4), pp. 1124–1148, 2025. https://doi.org/10.3311/PPci.41258

in an 85% reduction in compressive strength, while the replacement of fine aggregate with fine crumb rubber led to a 65% reduction [8]. A decline of 9%, 25%, 2.46%, and 13% in compressive, tensile, and flexural strength and modulus of elasticity, respectively, was observed by 5% rubber replacement [7]. A decrease in the mechanical properties of rubberized concrete is due to the inadequate rubber-cement bonding, the softer nature of crumb rubber, rubber's low density, and void formation at the interface [9].

Conventional experimental techniques to evaluate the mechanical properties of rubberized concrete are time-consuming, costly, and sensitive to experimental variability. Therefore, there is a need for a cost-effective and efficient technique to assess the strength of rubberized concrete. Machine learning (ML) offers an efficient alternative, analyzing complex data to predict performance, and various studies demonstrate ML's effectiveness in various fields, especially civil engineering [10, 11].

Concerning ML and rubberized concrete, Abdollahzadeh et al. [12] developed an ANN model for predicting the strength of rubberized concrete, showing superior efficiency compared to MLR. Also, Topçu and Sarıdemir [13] utilized a database comprising 36 datasets to analyze eco-friendly rubber concrete's properties via ANN and fuzzy logic models, with results closely matching the models' predictions. Dat et al. [14] indicated ANN's superior performance in comparison to RF and MLP in predicting rubberized concrete's compressive strength. Similarly, Gao et al. [15] found ANN, RG, and SVR effective for frost resistance prediction compared to ensemble methods (RF, XGBoost, Stacking). Habib and Yildirim [16] showed that XGBoost and FFBP were the most robust among multiple ML models for analyzing crumb rubber concrete properties. Table 1 [12, 14, 17–24] presents a list of related studies that have used soft computing techniques with details of their models and datasets for the mentioned purpose.

Table 1 Application of different ML models to rubberized concrete

Year	Ref.	Datasets	Model used	Input variables	Output variables
2024	[17]	135	ANN; MEP; MARS; NLR	8 inputs: Cement; water; superplasticizer; coarse aggregate; fine aggregate; rubber (%); maximum rubber size; curing time	compressive strength
2024	[18]	590	CAT_def GB; HGB; XGB; LGB; RF; ADA; ANN2; CAT_opt	10 inputs: Water; cement; cementitious replacement; coarse aggregate; fine aggregate; crumb rubber (%); chipped rubber (%); superplasticizer; ages; CS Sample Shape	compressive strength
2021	[19]	89	SVM; GPR	7 inputs: Cement; fine aggregate; coarse aggregate; aggregate pre-treatment; *w/c* ratio; fine aggregate replacement (%); coarse aggregate replacement (%)	compressive strength
2020	[14]	129	ANN; MLP; RF	9 inputs: water, cement; supplementary cementitious materials; coarse aggregate; coarse rubber aggregate; fine aggregate; fine rubber aggregate; superplasticizer; age using	compressive strength
2019	[20]	457	ANN	6 inputs: cement; *w/c* ratio; fine natural aggregate (%); coarse natural aggregate (%); fine rubber (%); coarse rubber (%);	compressive strength
2019	[21]	138	BRF	9 inputs: water, cement, Superplasticizer (%); supplementary cementitious materials, coarse aggregate, coarse rubber aggregate, fine aggregate, fine rubber aggregate, ages	compressive strength
2018	[22]	287	ANN	5 inputs: coarse aggregate; fine aggregate; *w/c* ratio; fine rubber; coarse rubber;	compressive strength
2013	[23]	50	ANN	2 inputs: epoxy resin; tire powder;	compressive strength, flexural strength, tensile strength
2011	[12]	20	ANN-MLP (BP)	3 inputs: Recycled rubber (%); Rubber powder size; Fresh concrete unit weight;	compressive strength
2010	[24]	70	NN; GEP; LR	8 inputs: Cement; Silica fume; Water; Superplasticizer; Coarse aggregate; Fine aggregate; Crumb rubber; Tire chips	compressive strength; tensile strength; static elastic modulus

The employment of metaheuristic algorithms as the hyperparameter optimizer of ML models has proven to be significantly influential for predictive models in various studies [25]. Therefore, this study aims to develop different ML models with hyperparameter optimization through improved metaheuristic algorithms to effectively predict the mechanical properties of rubberized concrete. The remainder of this article is structured as follows: Section 2 illustrates the research methodology, in Section 3, a summary of the utilized ML models is provided, Section 4 explains the metaheuristic algorithms, in Section 5, the performance evaluation of ML models is discussed, Section 6 presents the results, and in Section 7, the conclusion is highlighted.

2 Research methodology

2.1 Data collection

In this research, a total of 315 different samples from existing literature, as provided in Table S1 (available in the Supplement file), were utilized to analyze models aimed at estimating the mechanical properties of rubberized concrete at 28 days. The dataset comprises six distinct input variables: Rubber Size (RS, mm), Rubber Weight (RW, kg/m^3), Cement Content (C, kg/m^3), Water Content (W, kg/m^3), Coarse Aggregate (CA, kg/m^3) and Fine Aggregate (FA, kg/m^3). Compressive Strength (CS), Tensile Strength (TS), Flexural Strength (FS), and Modulus of Elasticity (ME) were included as the output parameters separately. The gathered data underwent statistical analysis, and the outcomes of the training and test sets from the dataset, including maximum and minimum values, mean, mode, median, standard deviation, variance, skewness, kurtosis, and range, are concisely presented in Table 2.

The data variability was assessed through the calculation of the standard deviation (SD). A smaller standard deviation (SD) indicates that numerous variables are closely clustered around the mean. Conversely, a bigger standard deviation signifies a wider distribution range among the variables [26]. We also harmonized mix design parameters by excluding inconsistent/outlier data and re-scaling inputs via Min-Max normalization. Strength values were unified at 28-day curing age, ensuring comparability across studies. Testing standards were cross-checked; inconsistent cases were corrected or omitted.

Skewness and kurtosis are employed to assess data symmetry and shape about a normal distribution. Positive skewness indicates an extension of the tail to the right, whereas negative skewness suggests an elongation to the left. A skewness value of zero denotes a perfectly symmetrical distribution. When the kurtosis is below zero, it demonstrates a flatter data distribution curve relative to the normal distribution, whereas a positive kurtosis illustrates a more peaked curve. Zero kurtosis value shows that the data distribution is neither excessively peaked nor flat, closely exhibiting a normal distribution [26].

2.2 Visualization of data

The distribution of input variables is meticulously presented in Section 6. The histogram aids in visualizing the concentrated values and identifying any outliers within the data [27]. The input features display approximately great distribution motifs within each of the ranges. Both rubber particle size and weight parameters exhibited right-skewed distributions, suggesting that lower rubber content values are more frequently utilized. The rubber size and rubber weight are predominantly concentrated in the

Table 2 Overview of the statistical analysis of the input and output features

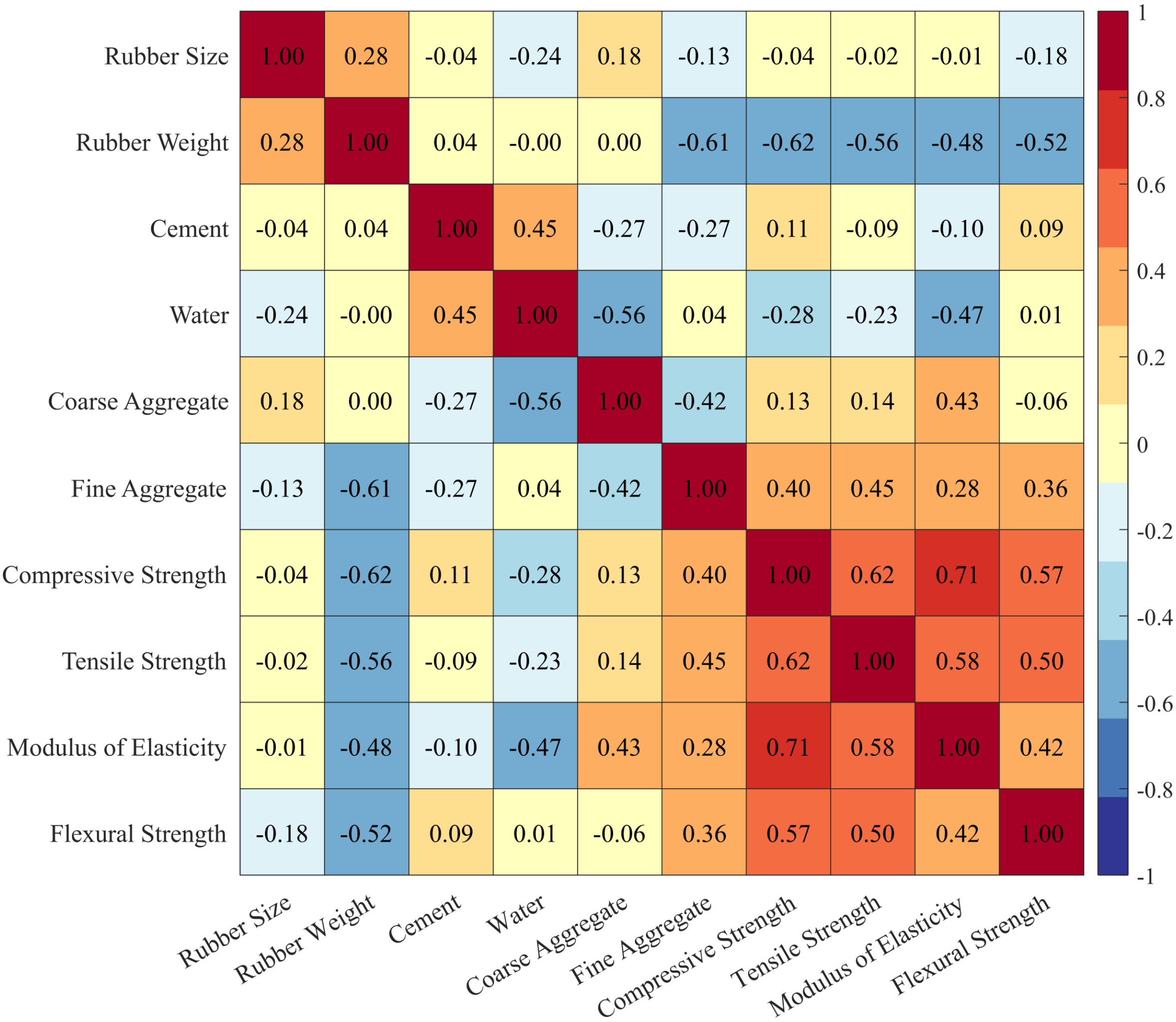
	Inputs						Outputs			
	RS	RW	C	W	CA	FA	CS	TS	ME	FS
Max	5	350	656.8	258	1296	1065	77.7	7.3	52.3496	10.35
Min	0.15	0	213.188	117.93	610	0	2.5	0.1939	0.4379	0.64
Mean	1.5321	49.1586	426.936	194.665	1031.06	604.274	29.6416	2.83037	25.1043	4.59799
Mode	1	0	388	209	1202.8	577	33.7	3.2	12	4.8
Median	1	34.66	420	195	1104	611	29.6	2.75	25.9970	4.5340
SD	1.0838	54.6599	77.0095	34.0382	186.243	169.062	12.3438	1.1190	8.9808	1.4856
Var	1.1745	2987.702	5930.461	1158.602	34686.34	28581.79	152.3691	1.2522	80.6555	2.2069
Skew	1.1230	2.2824	0.9623	−0.1064	−0.6774	−0.7241	0.3696	0.8587	−0.3310	0.6314
Kur	0.8803	6.7759	1.6804	−0.4259	−0.7019	1.7960	0.3308	1.8138	0.1170	1.6538
Range	4.85	350	443.612	140.07	686	1065	75.2	7.1061	51.9116	9.71

range of approximately 0.15 to 2.5 mm and 0 to 150 kg/m^3, respectively. Cement and fine aggregate parameters demonstrated approximately normal distributions, indicating a consistent material utilization pattern throughout the experimental dataset, which employed C dosages of 320–500 kg/m^3 and FA contents of 350–850 kg/m^3. The input parameters for coarse aggregate and water content exhibited wide distribution ranges across the experimental dataset.

2.3 Sensitivity analysis

Selecting appropriate input parameters is essential to achieve accurate results [27]. Sensitivity analysis was employed to identify input-output relationships and to recognize key factors influencing the outputs. Correlation analysis evaluated parameter relationships using Pearson coefficients, as illustrated in Fig. 1. Values near ±1 show robust positive/negative correlations, while values close to 0 suggest weak correlations [26]. This approach aids in choosing features in machine learning. Fine aggregate exhibits moderate positive correlations with compressive strength, with the values of 0.40 and 0.45 for tensile strength, demonstrating its contribution to improving structural robustness.

Subsequently, coarse aggregate demonstrates a positive coefficient of 0.43 with the modulus of elasticity, highlighting its role in enhancing rigidity. The data reveals that the weak associations between rubber size and all output variables suggest its negligible direct influence. Conversely, it is observed that rubber weight displays the highest negative correlations with all output features, with the values of −0.62, −0.56, and −0.52 for compressive strength, tensile strength, and flexural strength, respectively. It has been noticed that an increase in the proportion of rubber content leads to a decline in the mechanical properties of concrete. In addition, water exhibits a negative influence on elastic modulus and compressive strength, which can be attributed to the rise of the air voids in concrete.



	Rubber Size	Rubber Weight	Cement	Water	Coarse Aggregate	Fine Aggregate	Compressive Strength	Tensile Strength	Modulus of Elasticity	Flexural Strength
Rubber Size	1.00	0.28	-0.04	-0.24	0.18	-0.13	-0.04	-0.02	-0.01	-0.18
Rubber Weight	0.28	1.00	0.04	-0.00	0.00	-0.61	-0.62	-0.56	-0.48	-0.52
Cement	-0.04	0.04	1.00	0.45	-0.27	-0.27	0.11	-0.09	-0.10	0.09
Water	-0.24	-0.00	0.45	1.00	-0.56	0.04	-0.28	-0.23	-0.47	0.01
Coarse Aggregate	0.18	0.00	-0.27	-0.56	1.00	-0.42	0.13	0.14	0.43	-0.06
Fine Aggregate	-0.13	-0.61	-0.27	0.04	-0.42	1.00	0.40	0.45	0.28	0.36
Compressive Strength	-0.04	-0.62	0.11	-0.28	0.13	0.40	1.00	0.62	0.71	0.57
Tensile Strength	-0.02	-0.56	-0.09	-0.23	0.14	0.45	0.62	1.00	0.58	0.50
Modulus of Elasticity	-0.01	-0.48	-0.10	-0.47	0.43	0.28	0.71	0.58	1.00	0.42
Flexural Strength	-0.18	-0.52	0.09	0.01	-0.06	0.36	0.57	0.50	0.42	1.00

Fig. 1 Pearson correlation of the input and output variables

2.4 Preprocessing

Data preprocessing encompasses data generation, data cleaning, data integration, data reduction, data transformation, data normalization, and K-fold cross-validation in ML. All datasets underwent verification to ensure the accuracy of their data types.

2.4.1 Data generation

A comprehensive examination of the dataset revealed partial missing data in the output variables for tensile strength (TS), modulus of elasticity (ME), and flexural strength (FS). To address incomplete data for TS, FS, and ME, a two-level sequential artificial neural network was implemented. In the first phase, three optimized artificial neural networks, as proven to be reliable in our previous studies [28–30], (using IHGO), were trained on available inputs and complete CS data to estimate the missing TS, MS, and FS values, separately. Table 3 presents the statistical analysis (which will be explained in Section 5.2) and optimized structures of the developed models, which demonstrate the accuracy and efficiency of the developed models for data completion.

Upon completion of the dataset, the second phase focused on predicting all target mechanical properties of missing data points through a hybrid computational framework integrating machine learning models. Fig. 2 illustrates a high degree of accuracy in estimating missing mechanical property values, reinforcing the robustness of the predictive approach. TS exhibited an optimal correlation coefficient of 0.98783, with the fit line ($Y = T$) validating the negligible deviation between predicted and actual values. ME and FS demonstrated superior agreement with

Table 3 Performance and structure of optimized ANN models

ANN Model	Hyperparameters			Set	MSE	RSME	NMSE	NRMSE	Actual/Predicted Value		
	Name	Range	Optimal						R^2	Std	VAR
	Activation	{sigmoid,tanh,relu}	relu	Training	0.0024	0.0491	0.0010	0.0051	0.9991	1.0054	1.0108
	NumLayers	[1, 4]	3	Testing	0.3534	0.5945	0.0858	0.0746	0.9228	0.9919	0.9839
TS	NumNeurons	[5, 60]	23	All	0.0718	0.2679	0.0249	0.0276	0.9758	0.9937	0.9874
	Standardize	[0, 1]	1								
	Lambda	[0.0001, 0.1]	4.5E-03								
	Activation	{sigmoid,tanh,relu}	relu	Training	0.1310	0.3620	0.0018	0.0099	0.9982	1.0063	1.0126
	NumLayers	[1, 4]	4	Testing	4.4838	2.1175	0.0421	0.0628	0.9587	0.9889	0.9778
ME	NumNeurons	[5, 60]	37	All	1.0016	1.0008	0.0126	0.0260	0.9874	1.0027	1.0054
	Standardize	[0, 1]	1								
	Lambda	[0.0001, 0.1]	1.0E-01								
	Activation	{sigmoid,tanh,relu}	tanh	Training	0.0002	0.0143	0.0001	0.0017	0.9999	1.0004	1.0008
	NumLayers	[1, 4]	2	Testing	0.2178	0.4667	0.0587	0.0481	0.9405	1.0362	1.0738
FS	NumNeurons	[5, 60]	60	All	0.0432	0.2078	0.0150	0.0214	0.9850	1.0077	1.0154
	Standardize	[0, 1]	1								
	Lambda	[0.0001, 0.1]	1.0E-04								

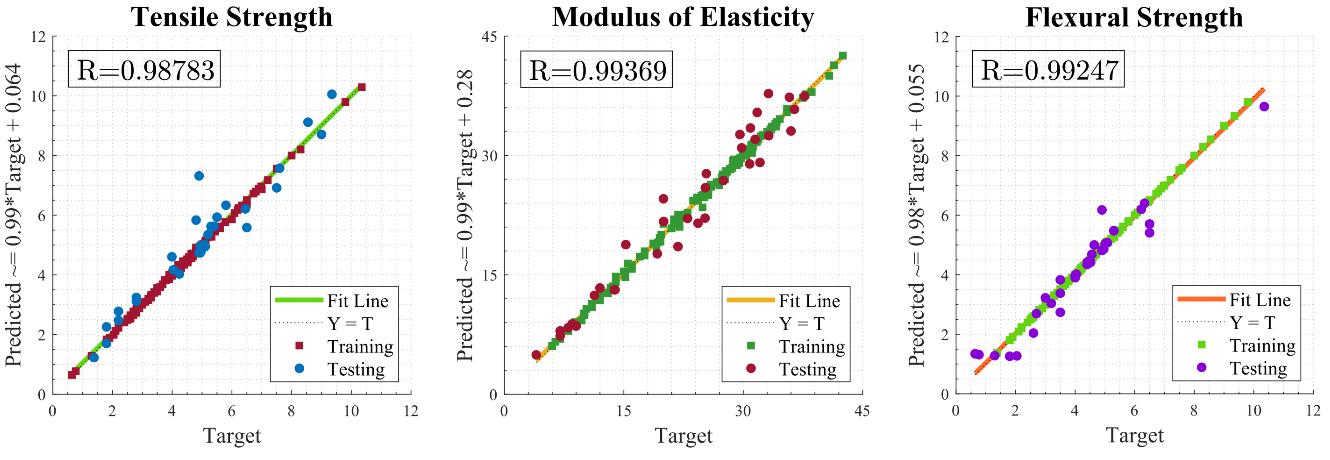


Fig. 2 Regression of the optimized ANN models for data generation

experimental values, as reflected by *R*-values of 0.99369 and 0.99247, respectively, underscoring the model's strong generalization capability. The findings affirm the effectiveness of a two-level sequential approach, demonstrating that inputs and compressive strength (CS)-based relationships can accurately reconstruct missing TS, ME, and FS data.

2.4.2 Normalization

Normalizing data is an essential step in preprocessing, especially with variables of differing scales [2]. This study employed the Min-Max normalization within the range of 0 to 1 to standardize their dimensions, decrease training time, and maintain accuracy [22]. The exact scaling function is represented by Eq. (1).

$$X_n = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \left(\mathrm{new}X_{\max} - \mathrm{new}X_{\min} \right) + \mathrm{new}X_{\min} \quad (1)$$

X_n illustrates the normalized value, X denotes the initial data value (input or output variables), and the variables $X_{\min}$ and $X_{\max}$ represent the lowest and highest values of X, respectively.

2.4.3 K-fold cross-validation

The dataset is classified into training, validation, and test sets, with training employed for model learning, validation for performance evaluation, and testing for generalization assessment [15]. Cross-validation estimates model performance, with K-fold CV (which in this study is 5-fold) optimizing accuracy, mitigating overfitting, and refining hyperparameters [31]. This research employed 5-fold cross-validation integrated with metaheuristic algorithms, utilizing 80% of the dataset for training and 20% for testing. Fig. 3 shows the full procedure for gathering data and model development.

3 Summary of machine learning models

ML constitutes a significant sub-discipline within the broader field of artificial intelligence. Machine learning models can automatically perform the forecasting procedure when they are trained, enabling the rapid and effective prediction of the mechanical properties of concrete without requiring manual computations or comprehensive knowledge [32]. In this study, various ML techniques, including AdaBoost, ANN, DT, GB, KNN, RF, SVM, and XGBoost, were employed to predict the mechanical properties of rubberized concrete. A summarized description of the aforementioned models is provided in Sections 3.1–3.8.

3.1 AdaBoost

AdaBoost (Adaptive Boosting) is a meta-algorithm for statistical classification and regression. It employs a forward stagewise additive approach for optimization [33]. At each iterative step, the algorithm increases the weight of instances misclassified by previous weak learners, whereas correctly classified samples experience reduced weights, thereby reducing the overall training error. Thus,

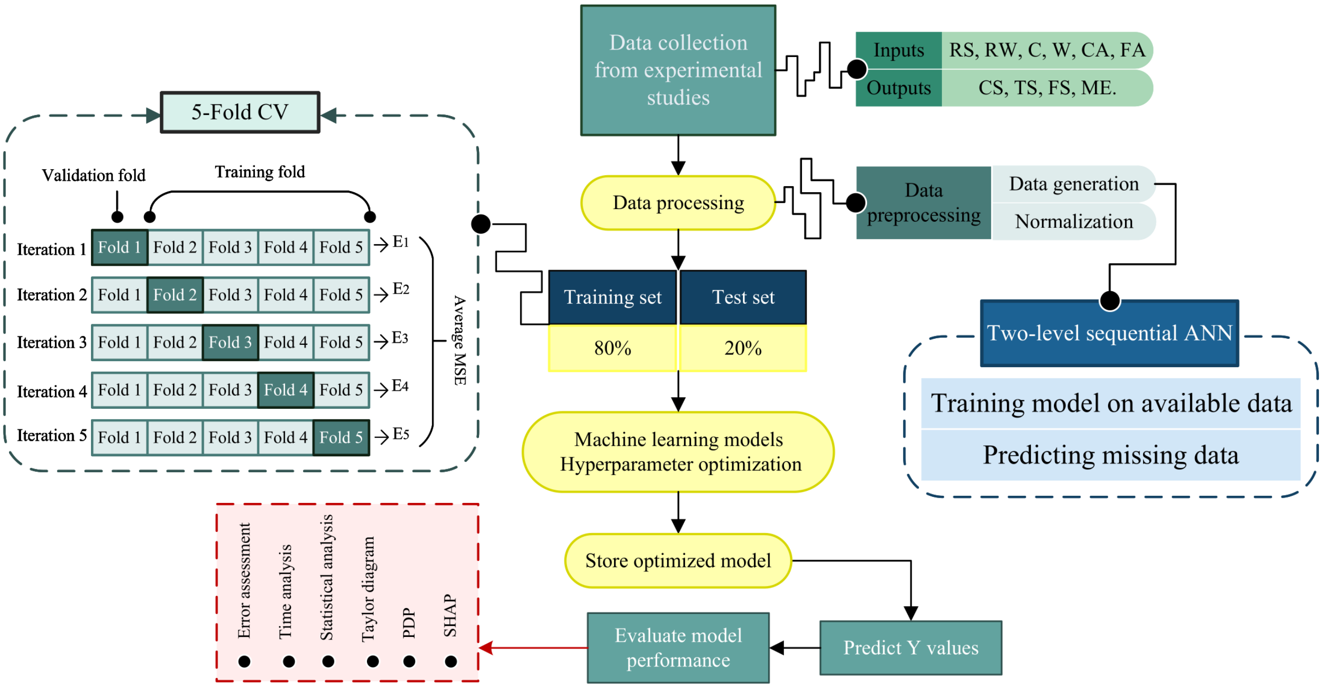
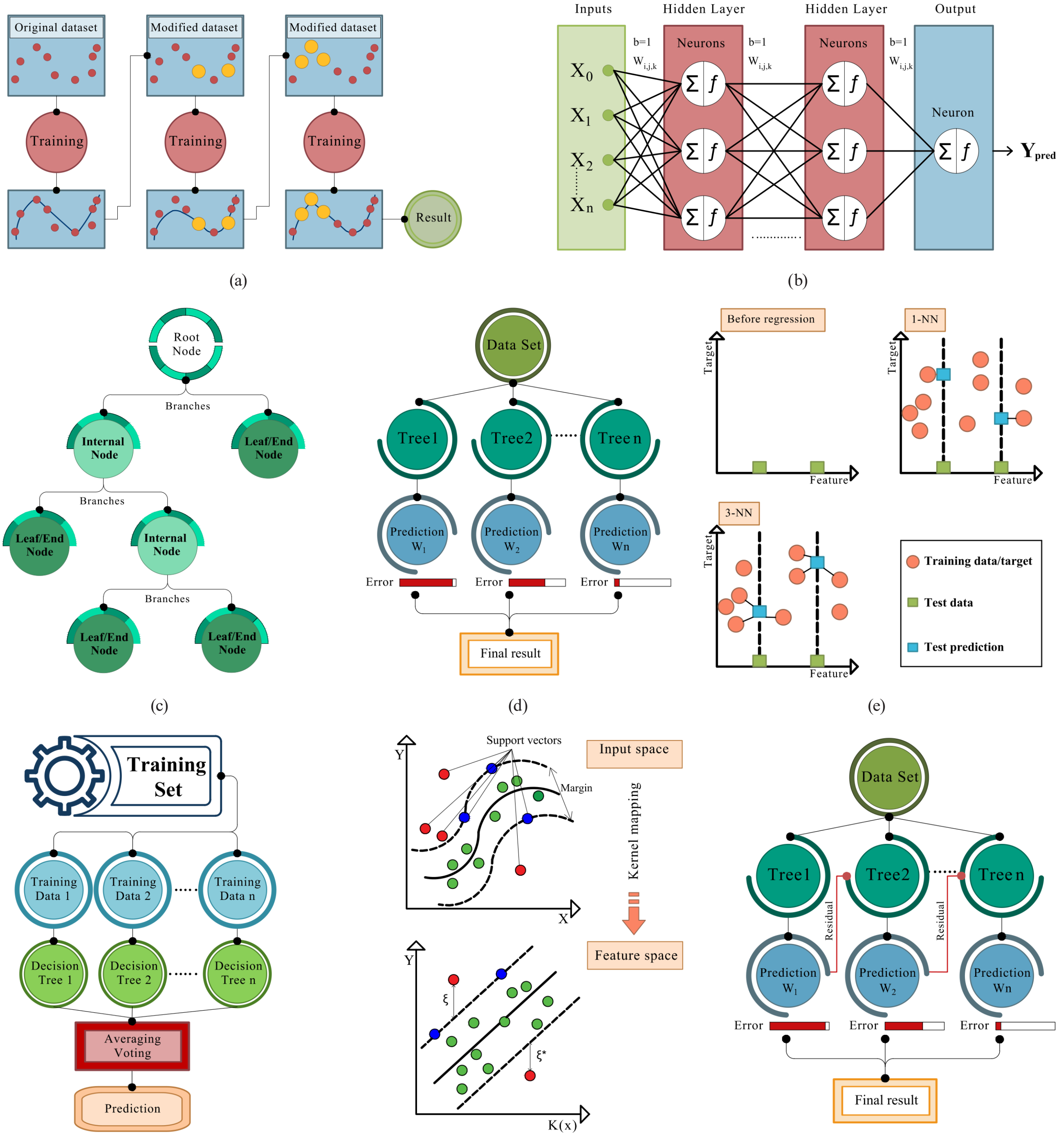


Fig. 3 Flow diagram of ML process

weak learners are combined into a weighted ensemble model through an adaptive integration process [26, 33]. The framework of the model is presented in Fig. 4 (a).

3.2 Artificial neural network (ANN)

An artificial neural network analyzes data in a method similar to the human brain, comprising a single input, hidden, and output layers, as is depicted in Fig. 4 (b). Hidden layers are responsible for data analysis via neurons (weights and biases, activation function), iteratively refining predictions [26]. A Multilayer Perceptron (MLP), a feed-forward ANN, can address a wide range of complicated issues, employing the sigmoid activation function in the hidden layer and linear activation function in the output layer [18], with structure optimized through trial-and-error.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 4 The structure of (a) AdaBoost, (b) ANN, (c) DT, (d) GB, (e) KNN, (f) RF, (g) SVM, and (h) XGBoost

3.3 Decision Tree (DT)

The decision tree is prevalent for classification/regression tasks. As illustrated in Fig. 4 (c), DT consists of the root node (first choice), internal nodes (feature evaluations), branches (possible results), and leaf/end nodes (ultimate targets) [3]. However, the Decision Tree model is susceptible to overfitting, especially with complex trees or noisy data. To address the problem, pruning or establishing minimum leaf samples.

3.4 Gradient Boost (GB)

Gradient Boosting is a powerful technique for both classification and regression problems, joining several weak learners, like decision trees, into a robust predictor. Each iteration reduces previous prediction errors via gradient-based optimization [3]. Achieving optimized performance effectiveness depends on two parameters: n-trees (number of trees) and learning rate. Insufficient trees limit predictive power, while excessive learning rates cause instability and inaccuracy [26]. The architecture of gradient boosting is shown in Fig. 4 (d).

3.5 K-nearest neighbors (KNN)

The k-nearest neighbors algorithm is a supervised machine learning technique that demonstrates high performance for datasets of limited size. During the prediction step of the KNN algorithm, feature similarity is utilized to assign an output value based on the proximity of the input data point to the points within the training dataset. As a result, the parameter (K) plays a pivotal role in influencing the performance of the KNN algorithm [26]. Fig. 4 (e) displays the KNN architecture.

3.6 Random Forest (RF)

Random Forest is a bagging ensemble of multiple decision trees (DTs) [34], each constructed through a process of random generation. This randomness within the forest enhances the accuracy of predictions by diminishing tree variance and overfitting because the models do not depend only on a single DT [22]. Final outputs are determined by majority vote (classification) or averaging (regression). Fig. 4 (f) shows the RF structure.

3.7 Support vector machine (SVM)

Support vector machine is a supervised learning model for non-linear classification, regression, and outlier detection [35]. It can form the best hyperplanes within higher dimensions via maximization of the functional margin, as illustrated in Fig. 4 (g). In regression tasks, SVM strive to reduce the gap between predictions and targets within a specified threshold (ε), whilst simultaneously increasing flatness. The kernel trick maps observations to a higher-dimensional space for separation, with the choice of various kernel functions that impact the performance of the final model. Fig. 4 (g) depicts that each data point is squared and subsequently plotted.

3.8 Extreme gradient boosting (XGBoost)

Extreme gradient boosting constitutes an enhanced version of conventional gradient boosting [36]. XGB is capable of managing missing values, applying regularization techniques, and automatically handling feature relationships to reduce over-fitting and enhance performance. It constructs regression trees using the Similarity Score and Gain metrics to optimize node splitting. XGBoost is viewed in Fig. 4 (h).

4 Metaheuristic algorithms

Optimization has been employed in numerous applications in civil engineering. In the field of computational optimization, metaheuristics are utilized as powerful problem-solving tools [37]. Since ML models have some hyperparameters that significantly influence the performance of the models, their optimization plays an integral role. Hyperparameter optimization is considered a mixed-integer optimization problem because a combination of continuous and discrete variables is included [38].

To enhance the predictive accuracy of the employed machine learning models, the hyperparameter search space was explored using three state-of-the-art metaheuristic algorithms: the Improved Hybrid Growth Optimizer (IHGO) [39], Improved Ray Optimization (IRO) [40], and Improved Sand Cat Swarm Optimization (ISCSO) [41]. These algorithms, each grounded in distinct natural or physical metaphors, are adept at navigating the complex, high-dimensional, and often non-convex landscapes characteristic of engineering optimization tasks. The pseudo-code formulations of each algorithm are provided to illustrate their respective workflows in a reproducible and algorithmically transparent manner (Pseudocode 1).

4.1 Improved Hybrid Growth Optimizer (IHGO)

IHGO represents a robust hybridization of the Growth Optimizer (GO) and the Improved Arithmetic Optimization Algorithm (IAOA), designed to address GO's limitations in parameter sensitivity and exploitation depth [39].

Pseudocode 1 Employed metaheuristic algorithms

Improved Hybrid Growth Optimizer (IHGO)

Inputs: *Population size, dimensions, Max evaluations, Lower and upper bounds.*
Output: *Optimal solution.*

```
Initialize population randomly within search space boundaries
Evaluate initial population
Set evaluation counter to initial population size
while termination criteria not met do:
    Sort individuals by fitness
    Identify the best individual
    for each agent:
        Determine agents (elite, worse, random) for gap calculation
        Compute four knowledge gaps
        if sum of gaps equals zero:
            Execute improved exploration step (from IAOA)
        else:
            Compute learning factors and acquire knowledge from gaps
        Update agents' position by learning knowledge
        Round positions for discrete variables if needed
        Boundary check and corrections
        Replace agents if improved, keep the best solutions
        Increment evaluation counter
    end for
    Sort agents again by fitness
    for each agent:
        Perform reflection operation to refine solutions
        Round positions for discrete variables if needed
        Boundary check and corrections
        Replace agents if improved, keep the best solutions
        Increment evaluation counter
    end for
    Calculate convergence criteria
end while
Return the best solution found
```

Improved Ray Optimization (IRO)

Inputs: *Population size, dimensions, Max evaluations, Lower and upper bounds.*
Output: *Optimal solution.*

```
Initialize agents' positions and velocities randomly within search space
Evaluate fitness of agents
Identify and update Local Best Memory (LBM) and Global Best (GB)
while termination criteria not met do:
    for each agent:
        Update agent position by adding current position to movement vector
        Check and correct boundary violations using intersection method
        Evaluate new positions
    end for
        Update LBM and GB based on new evaluations
    for each agent:
        Determine new direction and magnitude of movement vectors
        Normalize movement vectors
    end for
end while
Return the best solution found
```

Improved Sand Cat Swarm Optimization (ISCSO)

Inputs: *Population size, dimensions, Max evaluations, Lower and upper bounds.*
Output: *Optimal solution.*

```
Initialize sand cat positions randomly within search bounds
Evaluate initial population fitness and identify best individual
while termination criteria not met do:
    for each agent:
        Update control parameter ($R$) to determine exploration or exploitation
        if $|R| > 1$ (Exploration phase):
            Execute low-frequency noise search strategy:
            - Update positions based on prey location frequencies
            - Move sand cats toward promising positions
        else (Exploitation phase):
            Execute spiral contraction walk strategy:
            - Calculate spiral correction factors
            - Update positions in spiral around current best position
        Apply stochastic opposition-based learning strategy:
        - Generate new positions randomly opposed to current ones
        - Replace individuals if fitness is improved
        Apply restart strategy if stagnation is detected:
        - Re-initialize stagnated individuals randomly
        - Reset stagnation counters
        Evaluate updated positions
        Update best individual and fitness memory
    end for
end while
Return the best solution found
```

Pseudocode 1 Employed metaheuristic algorithms (continued)

Improved Hybrid Growth Optimizer (IHGO)	**Improved Ray Optimization (IRO)**	**Improved Sand Cat Swarm Optimization (ISCSO)**
	Parameters	
Max Iteration = 50 (stopping criterion) Population Size = 20 **% IHGO Parameters** P_1 = round(0.25 * popsize); P_2 = 0.001; P_3 = 0.3;	Max Iteration = 50 (stopping criterion) Population Size = 20 **% IRO Parameters** stoch = 0.35; d = 7.5; a = sqrt(sum(($lb - ub$)^2));	Max Iteration = 50 (stopping criterion) Population Size = 20 **% ISCSO Parameters** p = 1:360; C = 0.01; β = 2.77;

The algorithm introduces four pivotal enhancements:

1. the infusion of IAOA's exploratory dynamics into GO's learning stage to boost global search efficacy,
2. a revised elitism-aware replacement mechanism to preserve high-quality solutions,
3. structural refinements that augment the algorithm's hierarchical behavior, and
4. a reformulated reflection phase to intensify local search precision.

Collectively, these augmentations empower IHGO with heightened convergence stability, superior solution quality, and increased adaptability to discrete design spaces, making it an exceptional tool for real-world predictive modeling in civil engineering contexts.

4.2 Improved Ray Optimization (IRO)

Building upon the foundational Ray Optimization (RO) algorithm, originally inspired by the physical principles of light refraction governed by Snell's Law, IRO incorporates methodological advancements to bolster solution quality and convergence performance [40]. By eliminating the need for dimensional grouping, redefining the boundary correction mechanism, and introducing an efficient memory structure for local optima retention, IRO achieves a more fluid and scalable optimization trajectory. Its physics-inspired navigation of the search domain ensures a refined balance between diversification and intensification, rendering it highly effective for constrained optimization scenarios involving complex structural systems.

4.3 Improved Sand Cat Swarm Optimization (ISCSO)

The ISCSO algorithm draws inspiration from the predatory behavior of sand cats, uniquely characterized by their use of low-frequency acoustic sensitivity to detect subterranean prey [41]. ISCSO advances the original SCSO by embedding three innovative strategies:

1. a low-frequency noise-based search to dynamically control exploratory movement and maximize population diversity,
2. a spiral contraction walk, allowing sand cat agents to perform focused exploitation around promising regions with geometric precision, and
3. stochastic opposition-based learning and a restart mechanism that inject adaptive randomness to prevent stagnation and enhance global optima discovery.

These mechanisms synergize to enable ISCSO to efficiently traverse rugged optimization terrains while maintaining rapid convergence and robust exploration capacity.

5 Model performance evaluation

5.1 Hyperparameter tuning

In machine learning, there are two primary categories of parameters: model parameters (learned from data) and hyperparameters (preset before training) [26]. Optimal hyperparameters are vital for efficiently improving ML models because they enable the development of precise prediction models that avoid both overfitting and underfitting [38]. Hyperparameter tuning is conducted exclusively on the training dataset [3, 38]. Therefore, the ideal ML models and their related hyperparameters were selected following the K-fold CV process [22].

5.2 Assessment metrices

Statistical metrics are employed to evaluate the predictive accuracy and computational efficiency of the developed models, such as the correlation coefficient (R), coefficient of determination (R^2), mean square error (MSE), and the root mean square error (RMSE) [42]. R and R^2 values are utilized to examine the discrepancy between the predicted and actual data, whereas MSE is employed to calculate the average squared disparity between the predicted and experimental data [31]. The superior performance of the ML models was indicated by bigger R^2 values (towards one) and the reduced values of MSE and RMSE (towards zero). Statistical metrics are illustrated in Eqs. (2)–(5):

$$R = \frac{\sum_{i=1}^{n}\left(a_i - \bar{a}_i\right)\left(e_i - \bar{e}_i\right)}{\sqrt{\sum_{i=1}^{n}\left(a_i - \bar{a}_i\right)^2 \sum_{i=1}^{n}\left(e_i - \bar{e}_i\right)^2}} \tag{2}$$

$$R^2 = \frac{\sum_{i=1}^{n}\left(a_i - e_i\right)^2}{\sum_{i=1}^{n}\left(a_i - \bar{a}_i\right)^2} \tag{3}$$

$$\text{MSE} = \frac{\sum_{i=1}^{n}\left(a_i - e_i\right)^2}{n} \tag{4}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}\left(a_i - e_i\right)^2}{n}} \tag{5}$$

where a_i shows the actual value, e_i is the predicted value, $\bar{a}_i$ denotes the mean of actual values, $\bar{e}_i$ is the mean of predicted values, and n represents the number of points.

5.3 Regression analysis

The efficacy criterion for model selection is established based on the slope parameters of regression lines, determined by plotting experimentally measured values (x-axis) against corresponding model predictions (y-axis). For an optimal predictive model, a regression slope (RS) value exceeding 0.8 and approaching 1 represents the ideal criterion for model performance evaluation [26]. In linear regression, the intercept (or bias term) represents the baseline output value when all input variables are zero, working with the slope to define the prediction line. An intercept near zero suggests minimal bias, while a large intercept indicates systematic prediction errors. Together, the intercept and slope determine how well the model fits the observed data. The linear regression relationship is expressed in Eq. (6):

$$\hat{Y} = \beta_1 X + \beta_0 \tag{6}$$

where $\hat{Y}$ denotes the predicted value, β_1 is the slope coefficient, X is the predictor variable, and β_0 is the intercept term.

5.4 Taylor diagram

The Taylor diagram serves as a graphical method for comparing the performance of different models [43]. It represents key statistical metrics, including correlation coefficient (R), root mean square error (RMSE), and normalized standard deviation. Individual models are plotted as discrete points; their displacement from the reference point (standard deviation = 1) reflects predictive performance. Closer proximity exhibits strong correlation coefficients and reduced error metrics, signifying optimal model accuracy [15].

5.5 Partial dependence plot (PDP)

PDP analysis is widely utilized for the intuitive interpretation of machine learning models [44]. The plots are generated by selectively altering one or two input variables while maintaining the other variables constant, revealing the correlation patterns between input variables and the predictor. The benefit of PDP is that it allows for the straightforward calculation of how features in the actual values impact the outputs [15]. It enables us to comprehend ML model mechanics, data extraction patterns, and real-world correlation replication [44].

5.6 SHAP analysis

Input variables exhibit interdependence, whereby modifications in one variable can lead to alterations in another, so the impact of altering a single variable. PDPs can potentially present deceptive structures by isolating the effect of one variable while maintaining other features constant. To tackle this problem, Shapley Additive Explanations is used to measure the influence of each variable by accounting for its contribution alongside other variables [45].

6 Results and discussion

All the considered models were developed and trained in the MATLAB environment (MATLAB R2024a) [46], and just the XGBoost models were developed in MATLAB-Python Integration (Python 3.10.11) running at Operating System: Windows 10 Pro, version 22H2; Processor (CPU): 11^{th} Gen Intel(R) Core(TM) i7-1165G7 @ 2.80 GHz, 2.80 GHz; Memory (RAM): Total installed memory 16 GB DDR4; Storage: 1TB NVMe SSD; Graphics Card (GPU): NVIDIA GeForce MX450.

6.1 Hyperparameter optimization

In this study, the hyperparameters of eight machine learning models (AdaBoost, ANN, DT, GB, KNN, RF, SVM, and XGBoost) were optimized using three metaheuristic algorithms (IHGO, IRO, and ISCSO). Optimization via these metaheuristics significantly boosted the models' predictive performance for rubberized concrete, with each displaying distinct sensitivity to its key parameters.

Table 4 presents the search ranges and optimal values of these hyperparameters, along with the best-performing metaheuristic algorithm for each model. Optimization

Table 4 Performance metrics and optimized hyperparameters of developed ML models

Model	Hyperparameters			Optimizer	Set	MSE	RSME	NMSE	NRMSE	Actual/Predicted Value		
	Name	Range	Optimal							R^2	Std	VAR
AdaBoost	MinLeafSize	[1, 50]	15	ISCSO	Training	0.0028	0.0531	0.0013	0.0058	0.9987	1.0018	1.0037
	MaxNumSplits	[1, 100]	54		Testing	0.1328	0.3644	0.0639	0.0426	0.9374	1.0091	1.0184
	NumLearningCycles	[10, 500]	500		All	0.0288	0.1698	0.0131	0.0175	0.9870	1.0029	1.0059
	LearnRate	[0.001, 1]	0.2050									
ANN	Activation	{sigmoid, tanh,relu}	tanh	IRO	Training	0.0099	0.0992	0.0050	0.0102	0.9950	1.0050	1.0101
	NumLayers	[1, 4]	3		Testing	0.1623	0.4029	0.0518	0.0455	0.9496	1.0580	1.1193
	NumNeurons	[5, 60]	37		All	0.0403	0.2009	0.0183	0.0207	0.9818	1.0204	1.0412
	Standardize	[0, 1]	1									
	Lambda	[0.0001, 0.1]	5.3E-04									
DT	MinLeafSize	[1, 50]	1	IHGO	Training	0.4170	0.6457	0.0028	0.0086	0.9972	1.0014	1.0028
	MinParentSize	[1, 100]	1		Testing	19.8920	4.4600	0.1163	0.0730	0.8877	0.9980	0.9960
					All	4.3120	2.0765	0.0283	0.0276	0.9719	1.0002	1.0004
GB	MinLeafSize	[2, 10]	3	IHGO	Training	0.1200	0.3464	0.0015	0.0067	0.9985	1.0038	1.0077
	MaxNumSplits	[1, 10]	7		Testing	7.9732	2.8237	0.0993	0.0700	0.9010	1.0834	1.1738
	Unlearning Cycles	[10, 200]	200		All	1.6906	1.3002	0.0210	0.0250	0.9791	1.0169	1.0342
	LearnRate	[0.01, 0.3]	0.300									
KNN	NumNeighbors	[1, 50]	4	IHGO	Training	0.0497	0.2230	0.0006	0.0043	0.9994	1.0037	1.0073
	DistanceWeight	{equal, inverse, squaredinverse}	inverse		Testing	11.7170	3.4230	0.1967	0.0984	0.8088	1.0076	1.0153
	DistanceMetric	{euclidean, cityblock, chebychev, minkowski, cosine}	cityblock		All	2.3832	1.5438	0.0295	0.0297	0.9705	1.0045	1.0089
RF	NumTrees	[50, 500]	81	IHGO	Training	5.2899	2.3000	0.0350	0.0306	0.9724	1.1118	1.2361
	MinLeafSize	[1, 30]	1		Testing	22.5307	4.7467	0.1420	0.0871	0.8853	1.2987	1.6867
	MaxNumSplits	[1, 50]	51		All	8.7380	2.9560	0.0573	0.0393	0.9529	1.1438	1.3082
	NumPredictorsToSample	[1, 50]	2									

Table 4 Performance metrics and optimized hyperparameters of developed ML models (continued)

Model	Hyperparameters			Optimizer	Set	MSE	RSME	NMSE	NRMSE	Actual/Predicted Value		
	Name	Range	Optimal							R^2	Std	VAR
SVM	Kernel	{linear, polynomial, rbf}	rbf	IRO	Training	0.0217	0.1472	0.0095	0.0161	0.9910	1.0265	1.0537
	Regularization Param	[0.001, 100]	90.4717		Testing	1.2675	1.1258	0.6420	0.1229	0.3587	1.9226	3.6964
	Kernel Coefficient	[0.001, 10]	9.6247		All	0.2708	0.5204	0.1227	0.0536	0.8776	1.0989	1.2076
	Epsilon	[0.001, 2]	0.0384									
XGBoost	MaxDepth	[1, 10]	5	IHGO	Training	0.0372	0.1929	0.0004	0.0037	0.9996	1.0006	1.0013
	Num-Estimator	[50, 200]	195		Testing	6.8190	2.6113	0.0960	0.0618	0.9036	1.0792	1.1648
	SubSample	[0.5, 1]	0.57067		All	1.3936	1.1805	0.0173	0.0227	0.9827	1.0134	1.0270
	LearnRate	[0.01, 0.3]	0.2287									

results indicated that ISCSO outperformed IHGO and IRO in tuning AdaBoost's hyperparameters, achieving superior model performance. The NumLearningCycles parameter was optimized to 500 (upper bound), suggesting complex ensemble learning was necessary for optimal performance. IRO showed optimal performance for both ANN and SVM models. ANN's tanh activation outperformed sigmoid and ReLU alternatives, indicating effective nonlinear processing, while the high optimal KernelCoefficient (9.6247) of SVM demanded precise kernel regularization. For the remaining machine learning models, IHGO demonstrated superior optimization performance compared to both ISCSO and IRO algorithms.

6.2 Error assessment

Fig. 5 presents the mean squared error (MSE) values for models optimized via three different methods (IHGO, IRO, ISCSO) for estimating the 28-day compressive strength (CS), tensile strength (TS), modulus of elasticity (ME), and flexural strength (FS) of rubberized concrete. All models exhibited a reduction in MSE values as the number of function evaluations (NFE) increased, demonstrating prediction accuracy with more training iterations.

Initial MSE values for CS and ME predictions were significantly higher, approximately 80 and 40, respectively, compared to TS and FS. The IHGO algorithm illustrated optimal performance for CS estimation, achieving minimum MSE values more rapidly than IRO and ISCSO. XGB, GB, ANN, and AdaBoost achieved MSE values of about 20 by about 400 NFE. Statistical analysis reveals that ISCSO-enhanced GB attained the lowest MSE of around 0.2 within around 500 NFE for TS estimation.

For TS, the IHGO and ISCSO metaheuristics exhibited superior characteristics, enabling GB to achieve minimal prediction error (MSE ≈ 0.2) within 400 function evaluations. IRO also optimized GB and XGBoost with a minimal MSE of around 0.25. Notably, RF, DT, KNN, and SVM models showed significantly higher MSE across all metaheuristic optimizations, suggesting less effective.

For ME, the IHGO and ISCSO algorithms enabled GB, XGB, ANN, and AdaBoost models to attain MSE values below 10 within approximately 300 NFE. In contrast, the same models required nearly 700 NFE to reach an MSE value relatively below 10 under IRO optimization. Other models demonstrated weaker performance across all optimization frameworks.

Moreover, for FS prediction, both IHGO and ISCSO demonstrated efficacy as optimization techniques, enabling substantially lower and faster error, conversely, the IRO and ISCSO algorithms enhanced the performance of XGB, GB, AdaBoost, and ANN models, achieving near-zero MSE values with fewer NFE. Besides, trends in MSE reduction highlight that ensemble-based methods (GB, XGB, AdaBoost) converge faster and with lower error, reflecting their robustness against data heterogeneity typical in recycled concrete. The anomalies in DT/KNN reflect overfitting and sensitivity to noise, warning engineers of limited applicability for heterogeneous eco-materials.

Table 4 presents some statistical performance metrics for the optimized model configurations. AdaBoost-ISCSO exhibited robust predictive performance, MSE of 0.0028 on training data and 0.1328 on testing data, with corresponding RMSE values of 0.0531 and 0.3644, respectively.

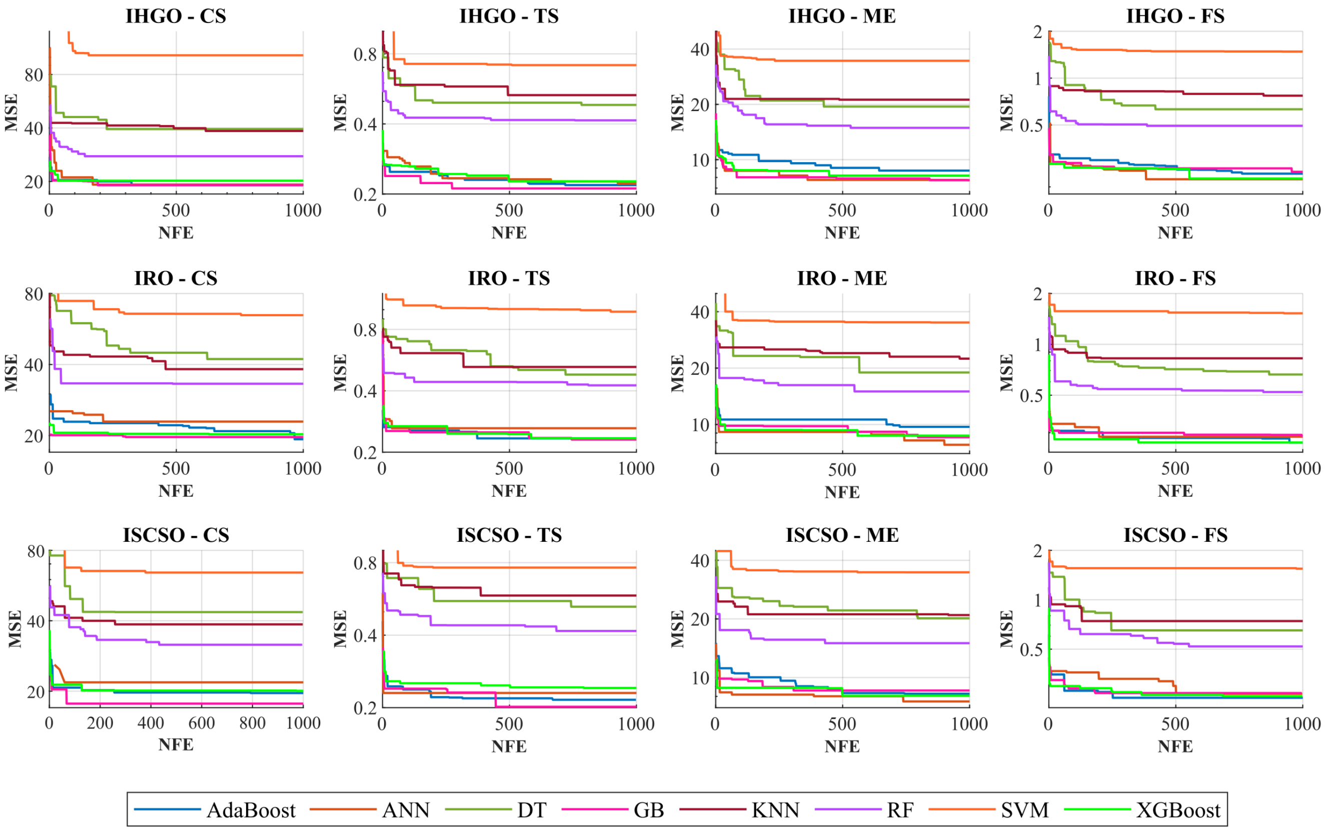


Fig. 5 MSE and NFE performance of optimized ML models for CS, TS, FS, and ME prediction

The minimal normalized MSE (NMSE) value of 0.0131 was attained by AdaBoost. In contrast, DT-IHGO displayed pronounced overfitting, with high testing MSE (19.8920) dramatically exceeding training MSE (0.4170) and the overfitting patterns observed across KNN and RF.

6.3 Computational time analysis

Fig. 6 presents the cumulative optimization time, measured in seconds, for the various machine learning models and metaheuristic algorithms utilized in this study. The analysis of cumulative optimization time demonstrated that IRO achieved the lowest total computational cost (11039.9 s), surpassing IHGO (13204.7 s) and ISCSO (14108.2 s). The findings underscored the IRO's efficiency, demonstrating superior scalability and time optimization across machine learning models. Among the IRO-optimized models, DT (10.87 s), KNN (19.55 s), and RF (710.43 s) were the first three models to achieve the shortest processing time, while AdaBoost required the longest optimization duration of 3333.83. For IHGO optimization, DT (16.86 s), KNN (19.79 s), and GB (1369.43 s) yielded the lowest computational times, while ISCSO optimization exhibited similarly efficient results for DT (12.04 s), KNN (20.50 s), and XGB (1233.74 s). In contrast, AdaBoost-IHGO (3214.14 s) and ANN-ISCSO (3893.02 s) required the highest computational durations. These results emerged IRO as the most appropriate metaheuristic for optimizing machine learning models when minimizing training time is a crucial consideration.

6.4 Statistical analysis

The predictive performance of each model employing three different metaheuristic optimization structures for estimating the 28-day compressive strength (CS), tensile strength (TS), modulus of elasticity (ME), and flexural strength (FS) of rubberized concrete was evaluated using the correlation coefficient (R) and regression equation as assessment metrics.

Statistical evaluation demonstrated that the XGBoost-IRO and XGBoost-ISCSO models achieved a strong predictive capability in predicting CS (as depicted in Fig. 7), with R-values of 0.99096 and 0.98914, respectively. Furthermore, a small degradation of R-value was observed in the IRO-optimized ANN, DT, and RF models compared to the IHGO-optimized versions of these algorithms. While AdaBoost demonstrated slightly improved accuracy over XGBoost in the IHGO optimization, achieving a correlation coefficient of 0.98988. The XGB-ISCSO

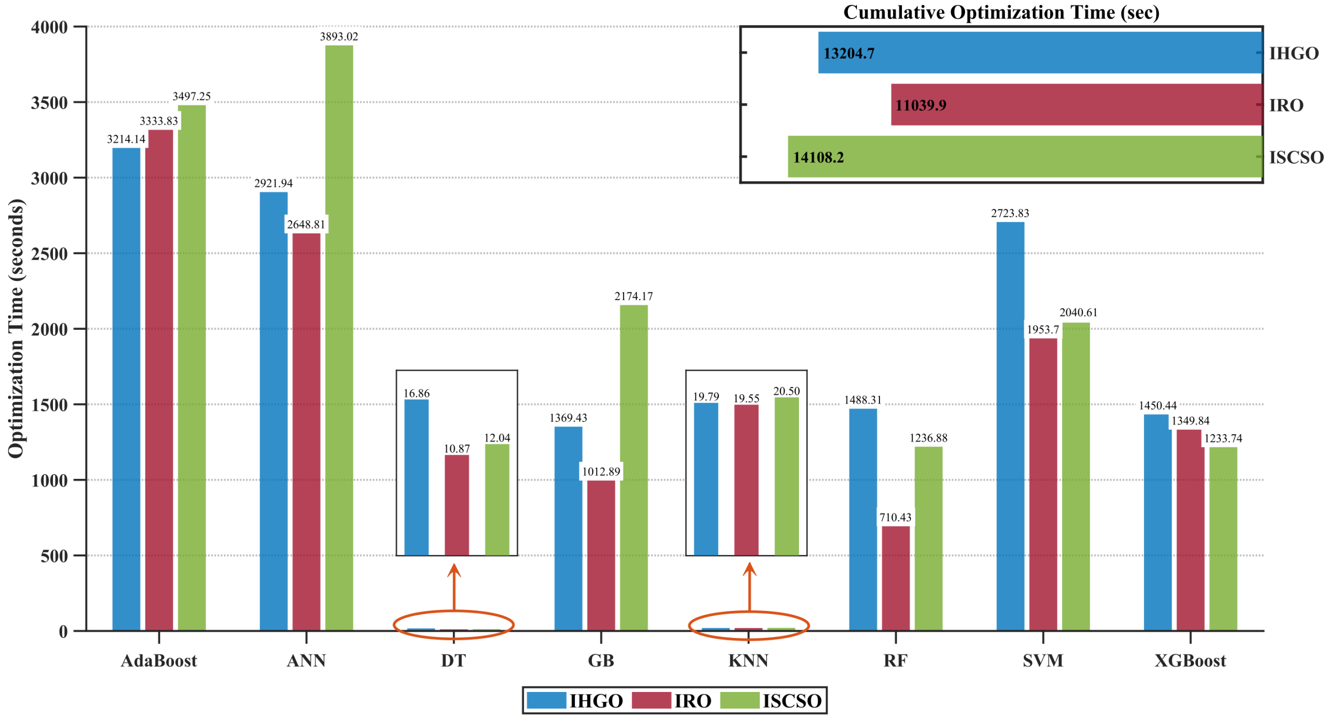


Fig. 6 Optimization time comparison of metaheuristics for ML models

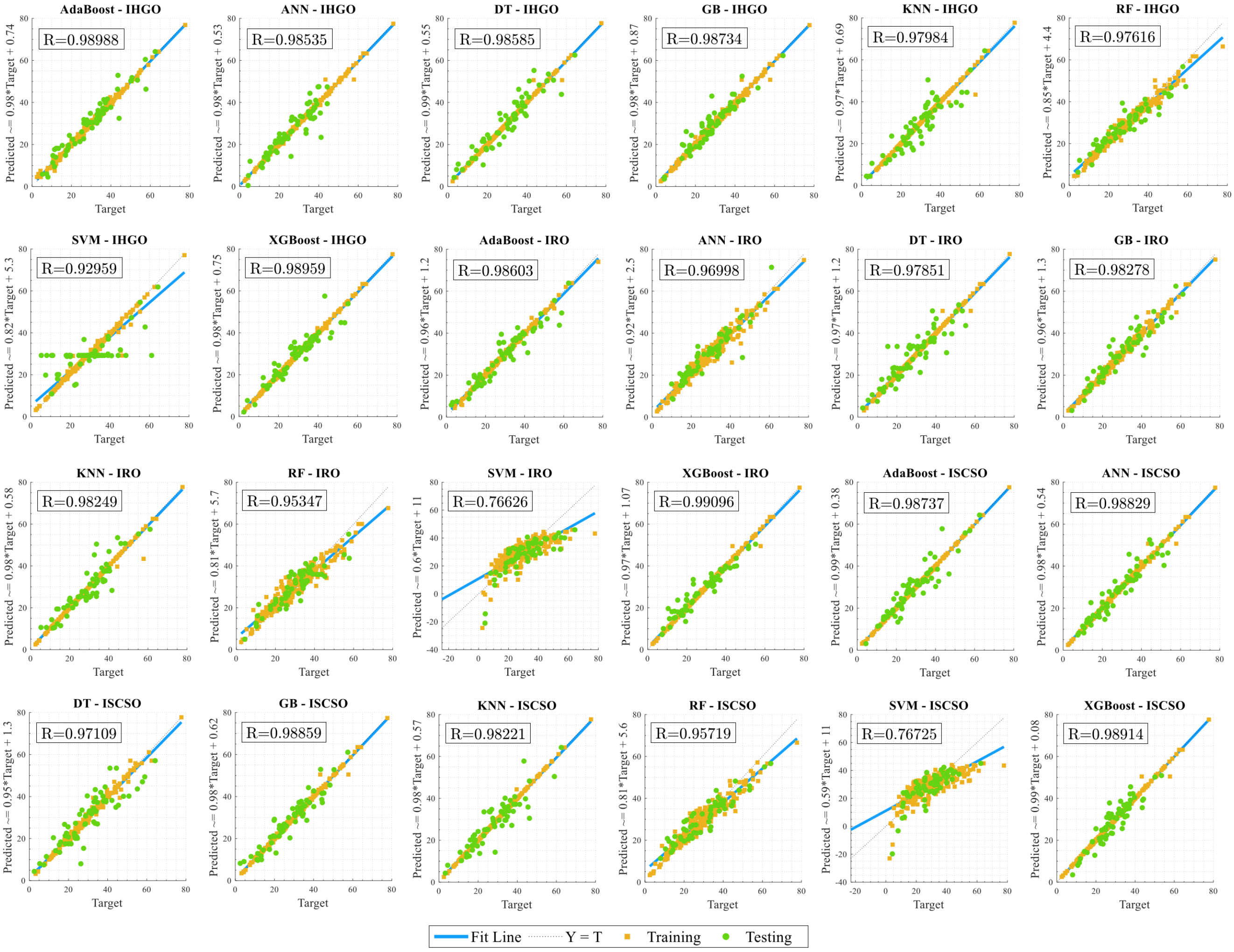


Fig. 7 Regression analysis of the developed compressive strength prediction model

model illustrated the greatest regression slope performance of 0.99 and the most favorable intercept of 0.08 in estimating CS, followed by AdaBoost-ISCSO, which also achieved an RS-value of 0.99 but with a less optimal intercept of 0.38.

For predicting TS, XGBoost achieved a good performance regardless of the optimization method, with a high *R*-value of approximately 0.986, RS-value of about 0.97, and a minimal intercept of 0.08 to 0.17, as shown in Fig. 8. While AdaBoost and GB also yielded strong predictive results, XGB indicated the highest reliability in its estimations.

Among all optimization algorithms, the XGBoost algorithm demonstrated the highest predictive accuracy for ME, achieving correlation coefficients of 0.99131 (IHGO), 0.99057 (IRO), and 0.99017 (ISCSO), according to Fig. 9. Additionally, ANN-ISCSO showed exceptional regression performance, with an RS value of 0.98 and a low intercept of 0.38, indicating strong generalization capability. The RF algorithm yielded moderate predictive performance across all optimization methods, achieving correlation coefficients in the range of 0.95-0.96 and regression slopes approximating 0.8.

For FS prediction (as presented in Fig. 10), the ISCSO-optimized AdaBoost and XGBoost provide an excellent predictive model, with superior correlations of 0.99347 and 0.99149, slopes of 0.99 and 0.97, and perfect intercept values of 0.031 and 0.13, respectively. Notably, ANN in the IHGO and IRO optimization and XGBoost-IHGO represented *R*-values above 0.99. Conversely, SVM exhibited lower correlation coefficients ranging from 0.65 to 0.93 compared to other models. Its regression equations had lower slopes of 0.4 to 0.86 and higher intercepts of 0.48 to 11, indicating potential deviations in prediction accuracy. RF revealed relatively higher accuracy, but its poor slopes and high intercepts made it less reliable. Therefore, both algorithms may be less suitable for predicting the mechanical properties of rubberized concrete.

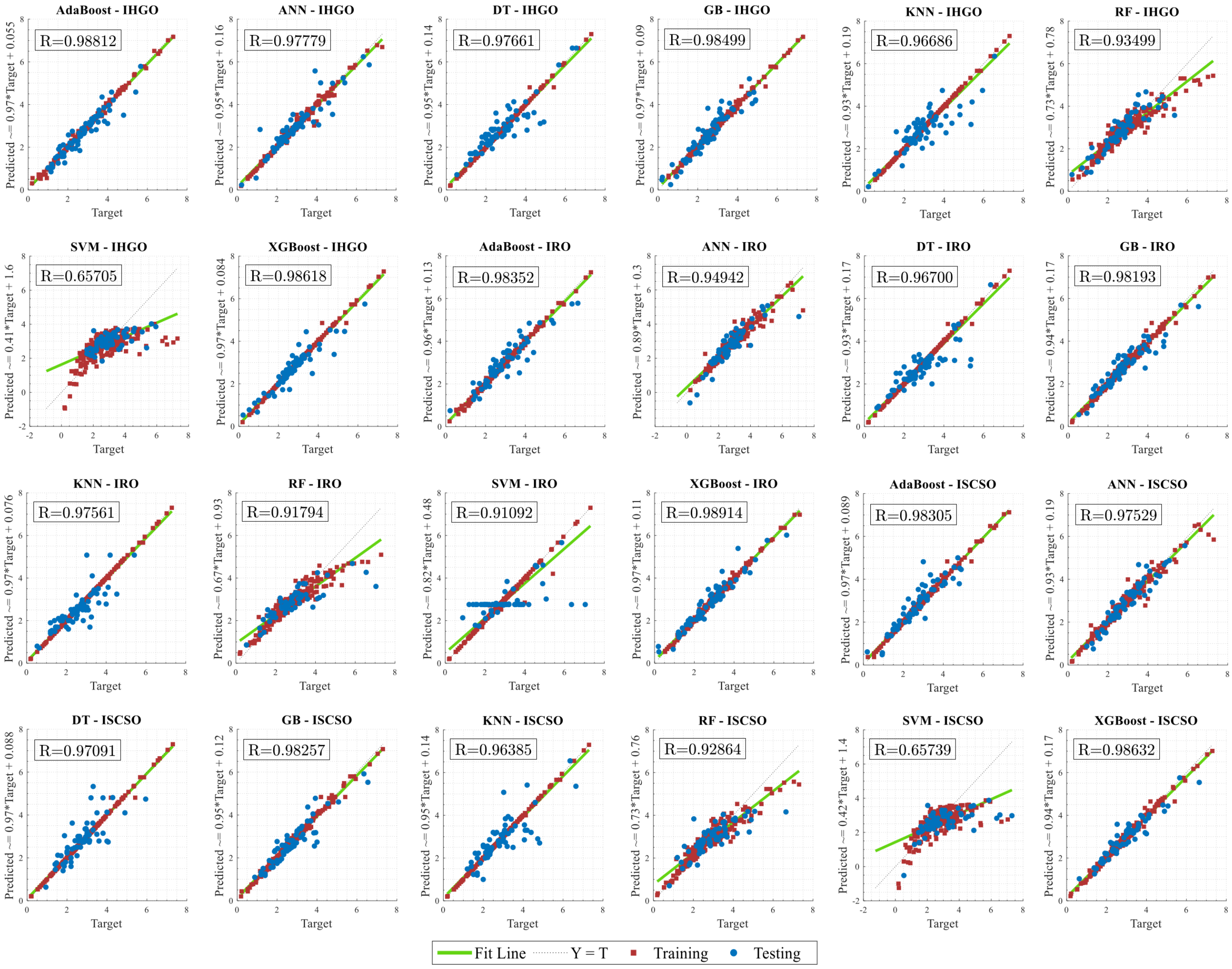


Fig. 8 Regression analysis of the developed tensile strength prediction model

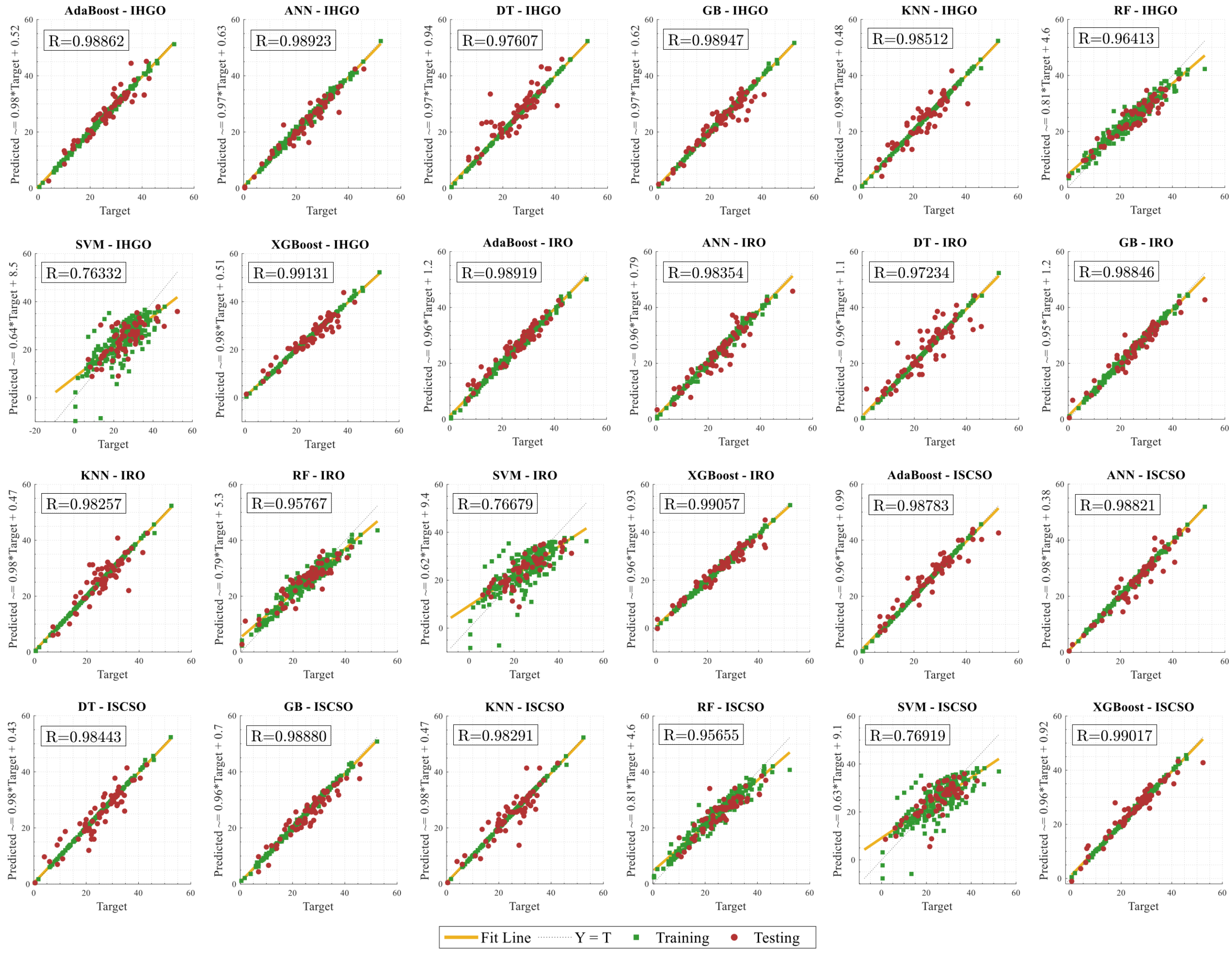


Fig. 9 Regression analysis of the developed modulus of elasticity prediction model

Furthermore, Table 4 shows that AdaBoost-ISCSO suggested a strong R^2 (0.9987 training, 0.9374 testing), standard deviation (Std) close to 1 (1.0018 training, 1.0091 testing), and ideal variance ratios (VAR) (1.0037 training, 1.0184 testing), demonstrating accuracy and stability. The R^2-based performance ranking (AdaBoost-ISCSO > XGB-IHGO > ANN-IRO > GB-IHGO) presented high predictive accuracy, while Models such as SVM-IRO (R^2 = 0.9910 training, 0.3587 testing) and RF-IHGO (R^2 = 0.9724 training, 0.8853 testing) exhibited substantial performance degradation in testing, accompanied by elevated Std (SVM: 1.9226; RF: 1.2987) and VAR (SVM: 3.6964; RF: 1.6867), indicating poor generalization and unstable predictions.

6.5 Models' Taylor diagram interpretation

Fig. 11 presents the Taylor diagram comparing the statistical performance of multiple ML algorithms utilizing three distinct metaheuristic optimization frameworks for predicting the mechanical properties of rubberized concrete. The predictive models for CS exhibited a more concentrated distribution near the reference point (denoted by the purple circle), indicating enhanced predictive accuracy and reduced variance. The XGBoost-IRO hybrid model exhibited strong performance, attaining a correlation coefficient of 0.99096 with a normalized standard deviation of 0.975 and RMSE 0.167, reflecting an almost perfect correspondence with experimental results, which can be attributed to IRO's efficient exploration of XGBoost's hyperparameter space, specifically in optimizing, including learning rate and tree depth parameters.

The TS models similarly showed XGBoost-IRO sustaining peak performance indicators. The correlation coefficient of AdaBoost, GB, and XGB exceeded 0.98 across all three metaheuristic algorithms in the evaluation of TS, demonstrating a high degree of predictive accuracy and reliability. In contrast, SVM within the IHGO and ISCSO optimization models illustrated poorer performance, characterized by

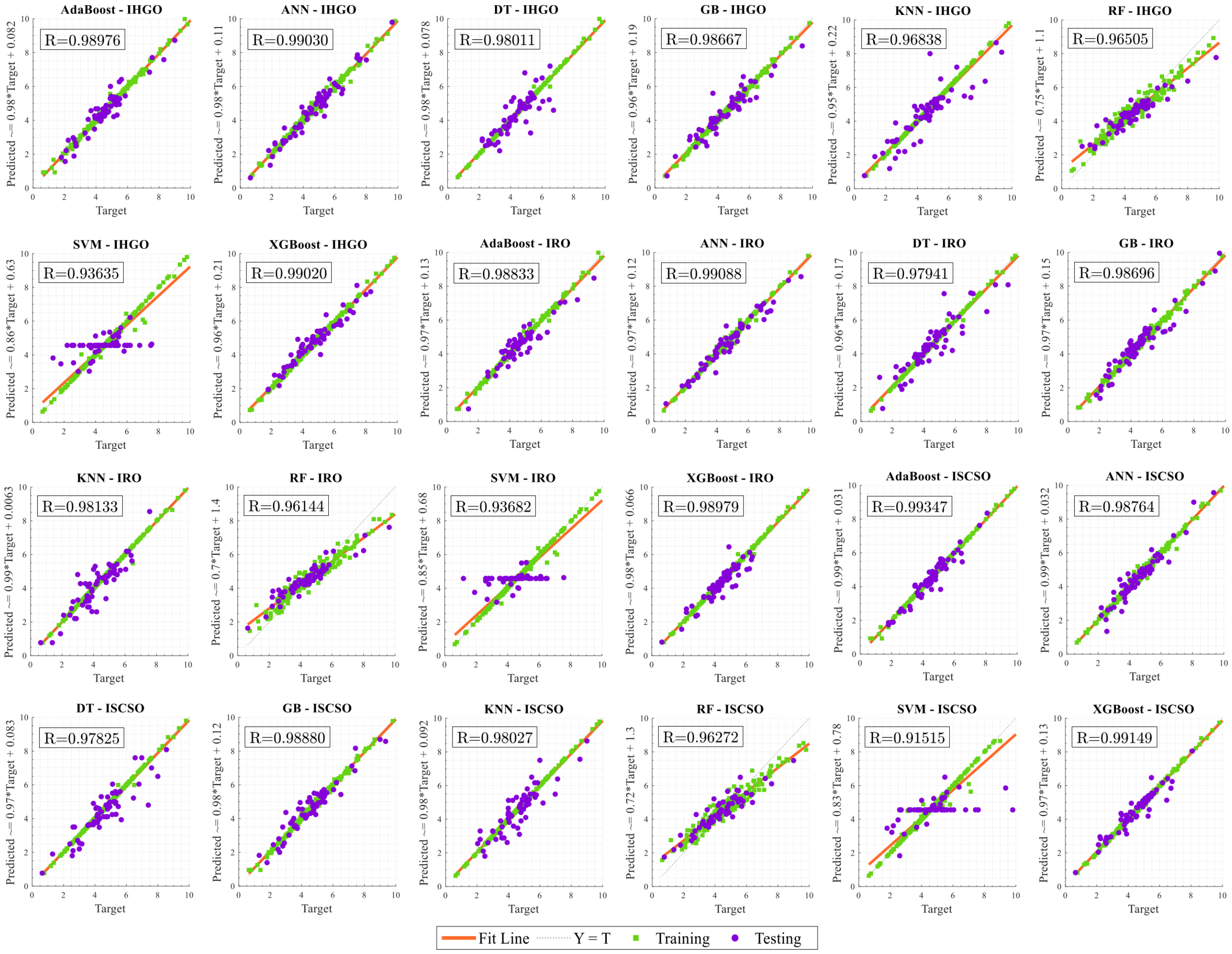


Fig. 10 Regression analysis of the developed flexural strength prediction model

low *R*-values of 0.65705 and 0.65739, a normalized SD of 0.489 and 0.493, respectively. SVM-IRO and RF optimized through IHGO, IRO, and ISCSO frameworks demonstrated intermediate predictive performance, with correlation coefficients ranging from 0.91 to 0.93.

For ME, XGBoost maintained its performance advantage across optimization algorithms, indicating an almost perfect correlation (XGB-IHGO > XGB-IRO > XGB-ISCSO > 0.99), normalized SD (ranging from approximately 0.9 to 1). The Taylor diagram analysis of FS predictions revealed optimal model performance for AdaBoost-ISCSO, XGB-ISCSO, ANN-IRO, and ANN-IHGO methods, demonstrating an ideal balance between minimal dispersion ranging from 0.9 to 1 and strong correlation exceeding 0.99.

6.6 Partial dependency analysis

Partial Dependence Plots (PDPs) of the best models were employed to examine the potential positive or negative impact of six input parameters (rubber size, rubber weight, cement content, water content, coarse aggregate, and fine aggregate) on CS, TS, ME, and FS, as illustrated in Fig. 12. The PDP curve represents the mean feature effects on the target variable [45], while the 95% confidence interval indicates the uncertainty associated with the estimated partial dependence.

The analysis revealed a negative correlation between rubber weight and the predicted mechanical strength of concrete. When the rubber weight increased, the model predicted a considerable decline. An increased rubber weight leads to rising air voids and weaker interfacial transition zones (ITZs) [47], both of which affect the mechanical performance. In addition, the initial steep decline showed that minor increases in rubber weight had an immediate impact on the target variable. Rubber particle size has a significant influence on the determination of mechanical properties. The mechanical properties exhibited slight improvements with particle size reduction. Smaller rubber particles may function as a filler,

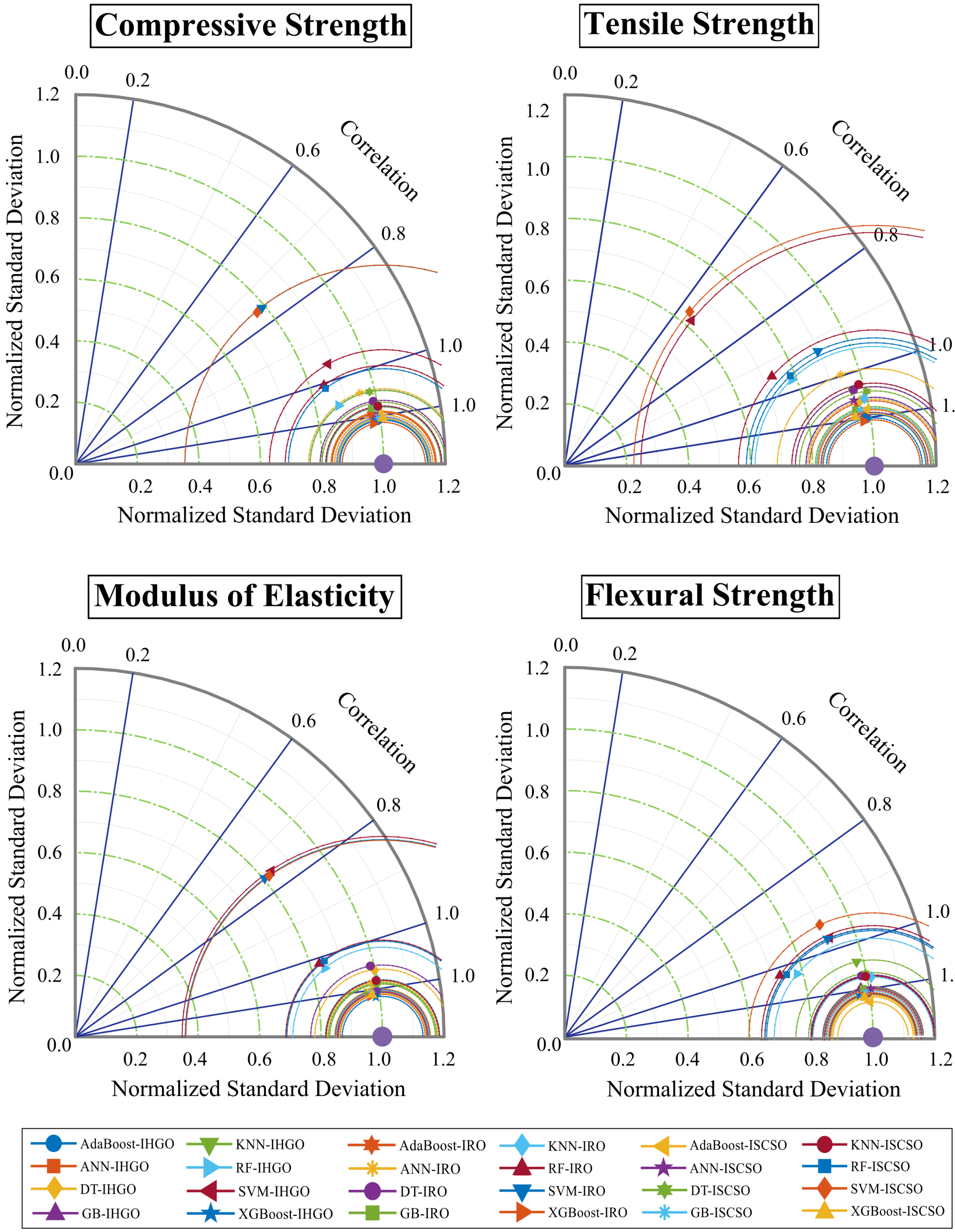


Fig. 11 Taylor diagram of model performance for CS, TS, FS, and ME

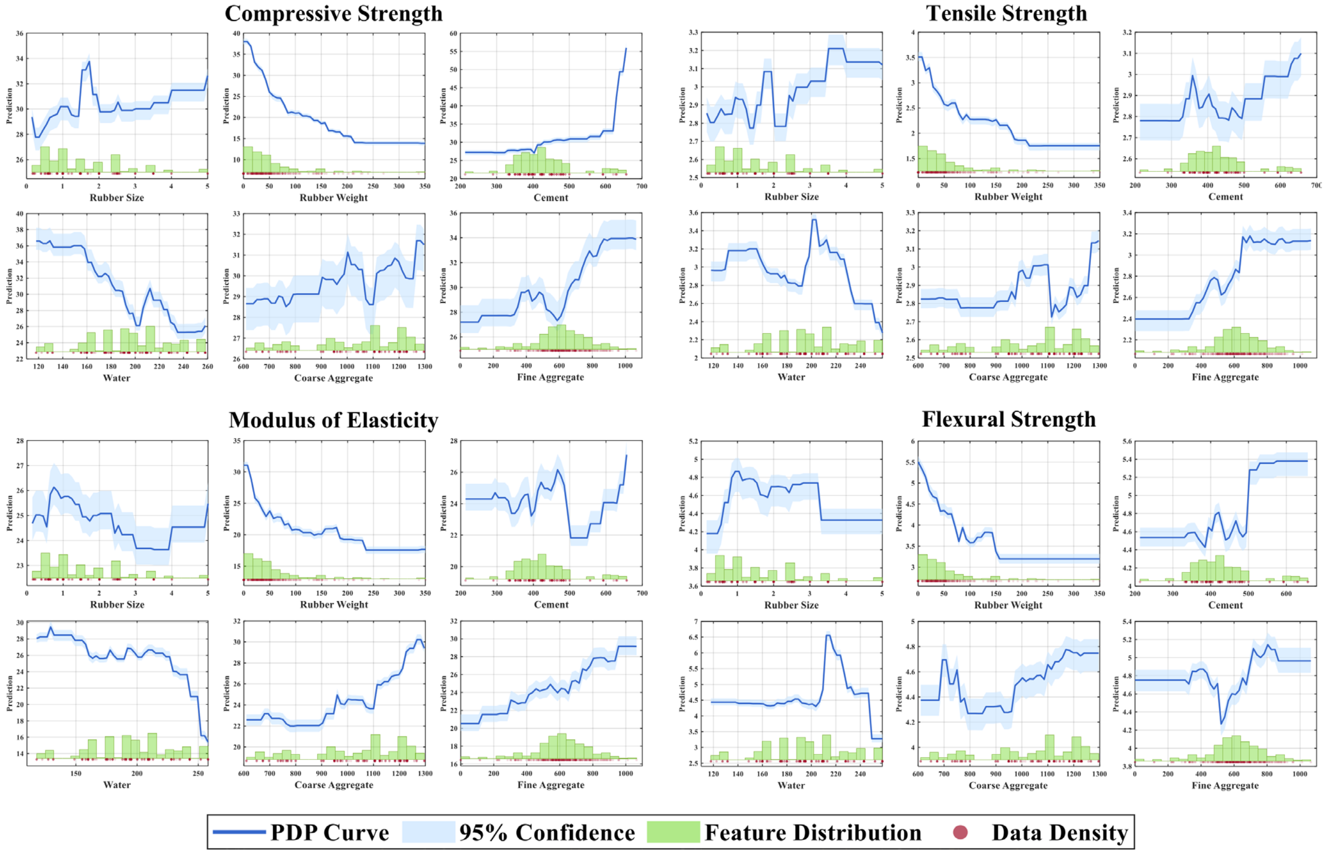


Fig. 12 Partial dependence plot evaluation of input variables for CS, TS, FS, and ME

enhancing the compactness of the concrete while minimizing stress singularities at internal voids. This effect contributes to a reduced probability of fracture, thereby improving the material's overall mechanical integrity [47].

It is evident that as the rubber size surpassed a specific threshold, the curve flattened or exhibited a slight fall, indicating a negative contribution to strength properties. Cement exhibited a robust positive relationship, where increased cement content resulted in enhanced strength characteristics. An increased cement content expands the formation of calcium-silicate-hydrate (C-S-H) gel [48], leading to a higher microstructural density in the composite. Thereby, this densification is directly associated with enhanced compressive and tensile strength.

Water demonstrated nonlinear interactions with the target variables, wherein an optimal range maximized performance, while deviations resulted in deterioration. At low water contents, the mixture exhibited reduced mechanical properties due to insufficient hydration. Beyond a critical threshold, increasing the water content to the optimal range enhanced strength by improving cement hydration efficiency. However, excessive water content degraded strength through porosity formation.

Both coarse and fine aggregates exhibited a positive contribution to mechanical performance, especially at elevated quantities, highlighting their essential function in stiffness enhancement within the composite matrix. The graphical representation reveals that the thickness of the 95% confidence intervals varied across the comparative subplots. Wider intervals appeared in data-sparse regions, highlighting the necessity for supplementary data in these areas. In contrast, reduced interval widths indicate a greater reliability increase in the relationship between the feature and the target variable. In addition, the model demonstrates robust performance in high-density areas of the data distribution, as shown in Fig. 12; the highest density of rubber weight occurred below nearly 100.

6.7 SHAP analysis

Fig. 13 displays a comprehensive visualization of SHAP values for all input features on model outputs throughout the entire feature space. The horizontal axis represents the SHAP value, indicating the feature's influence on the model prediction, either as a positive or negative deviation from the baseline [15]. The vertical axis displays features ranked by their importance, and the color gradient reflects feature values, with red and blue showing high values and low values, respectively.

Across the CS, TS, and FS models, rubber weight exhibited a significantly negative effect on strength,

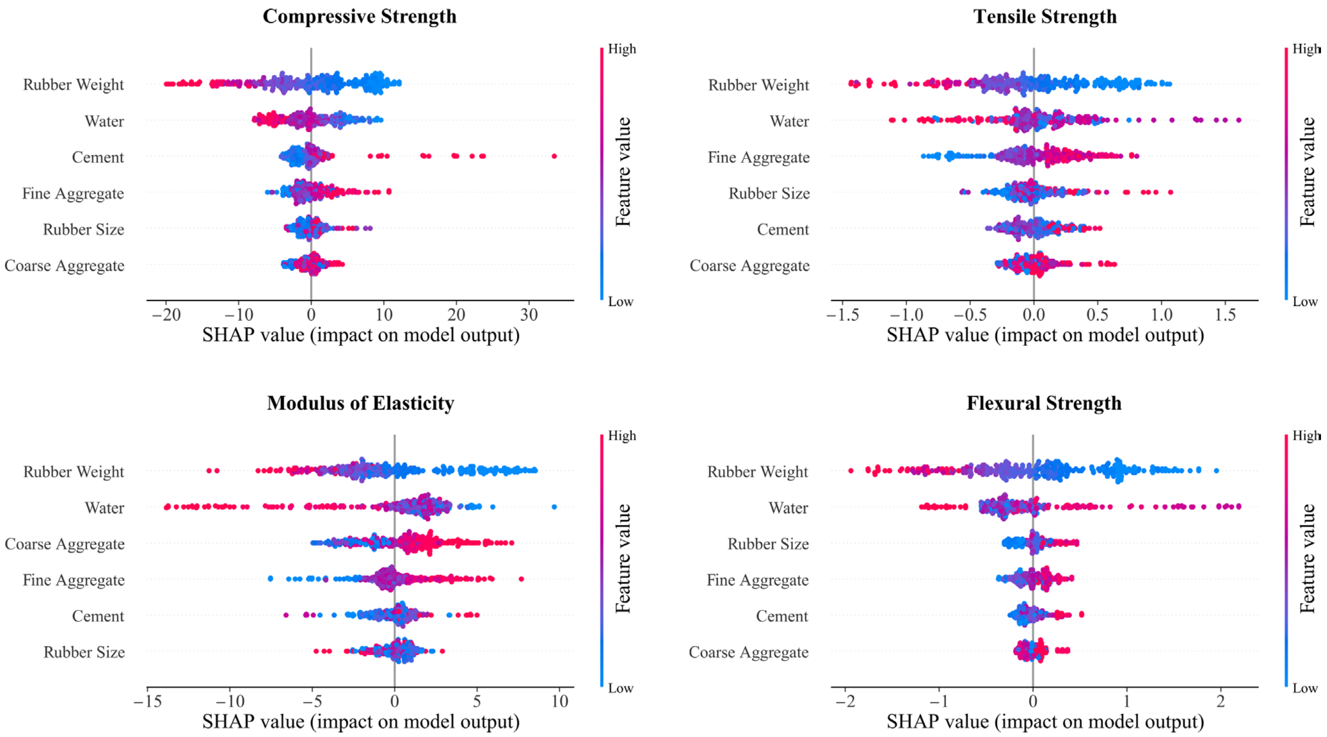


Fig. 13 SHAP value analysis of feature impacts on mechanical properties

followed by water content, which had a slightly less unfavorable impact than rubber weight. However, in the case of ME, water content emerged as the dominant negative factor, followed secondarily by rubber weight. Besides, negative SHAP values were observed for higher water content. In contrast, the optimal water content had a positive impact on the strength properties, supporting cement hydration, as evidenced by the right side of the axis.

For FS, both rubber size and coarse aggregate demonstrated relatively modest SHAP value distributions centered near zero, with coarse aggregate exhibiting particularly negligible influence on model predictions. For the TS and FS models, rubber size and fine aggregate generally exhibited a positive influence. The reduction in strength was influenced by the content, size, and shape of the rubber granules.

Raghavan et al. [49] revealed that fine rubber caused a smaller decrease in mortar strength compared to coarse rubber. Despite its weak bonding properties, fine rubber provided sufficient restraint to limit the propagation of shorter cracks. Additionally, the samples exhibited notable elastic deformation, which persisted upon unloading, significantly enhancing flexibility and elastic deformability [9]. Both coarse and fine aggregates emerged as the most significant predictive parameters for the ME model, demonstrating the positive influence on model outputs. The strength of concrete is largely governed by the characteristics of its coarse and fine aggregates, including their density, size, hardness, and stiffness [26].

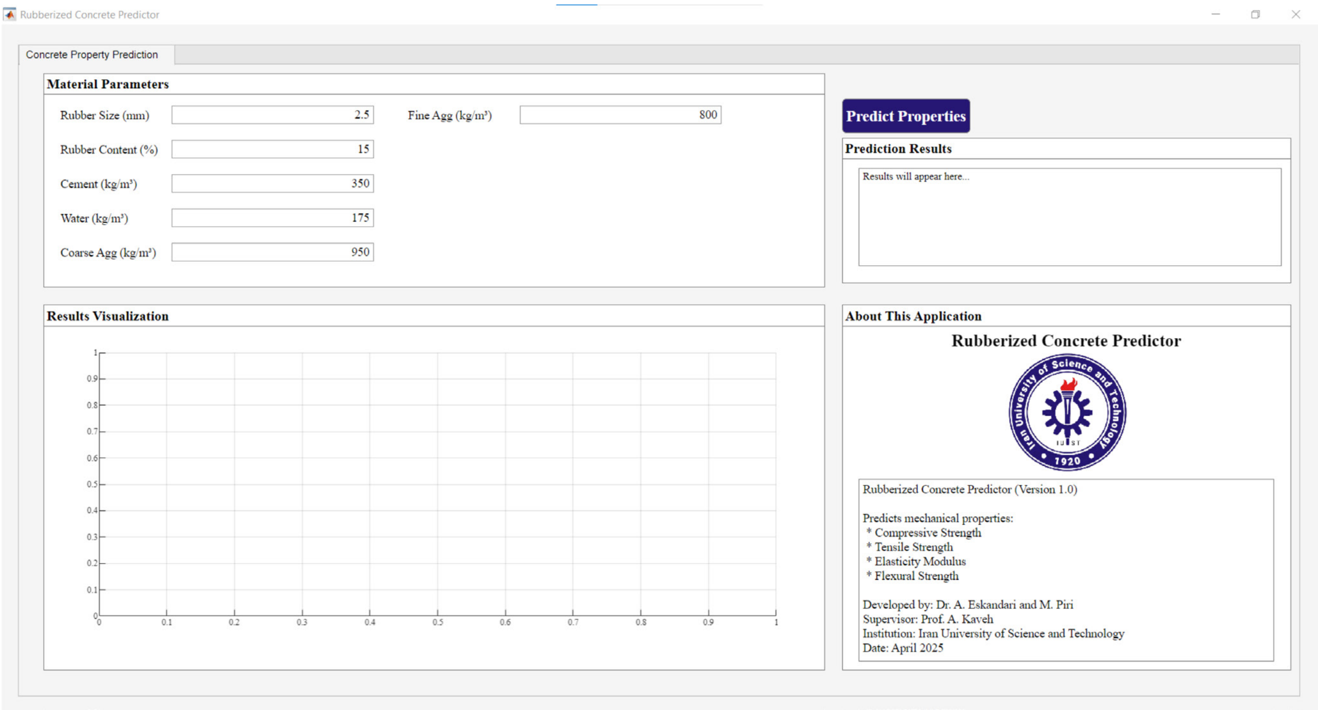
6.8 Graphical User Interface (GUI)

The study also presents a MATLAB-based graphical user interface (GUI) designed to predict key mechanical properties of rubberized concrete, including compressive strength, tensile strength, modulus of elasticity, and flexural strength, as represented in Fig. 14. The GUI integrates a hybrid machine learning (ML) architecture, combining AdaBoost (for flexural strength prediction) and XGBoost (for remaining properties) models, optimized via metaheuristic algorithms to ensure high predictive accuracy. The interface employs a modular framework, enabling seamless integration of Python-based XGBoost models via MATLAB-Python interoperability, while preserving computational efficiency.

7 Conclusions

This study utilized an experimentally derived dataset on the substitution of fine aggregate with fine rubber particles to develop an innovative predictive framework that integrates machine learning algorithms (AdaBoost, ANN, DT, GB, KNN, RF, SVM, and XGBoost) with metaheuristic optimization techniques (IHGO, IRO, and ISCSO) for accurate estimation of rubberized concrete's mechanical properties at 28 days. The main findings are as follows:

- For compressive strength prediction, the XGB-IRO algorithm outperformed all alternative models, attaining an exceptional correlation coefficient of 0.99096.



Material Parameters

Rubber Size (mm): 2.5
Rubber Content (%): 15
Cement (kg/m³): 350
Water (kg/m³): 175
Coarse Agg (kg/m³): 950
Fine Agg (kg/m³): 800

Prediction Results

Results will appear here...

About This Application

Rubberized Concrete Predictor

Rubberized Concrete Predictor (Version 1.0)

Predicts mechanical properties:
* Compressive Strength
* Tensile Strength
* Elasticity Modulus
* Flexural Strength

Developed by: Dr. A. Eskandari and M. Piri
Supervisor: Prof. A. Kaveh
Institution: Iran University of Science and Technology
Date: April 2025

Fig. 14 The developed GUI for mechanical properties prediction of rubberized concrete

- In terms of predictive accuracy for tensile strength, the XGB-IRO algorithm demonstrated the highest correlation coefficient, attaining an R-value of 0.98914.
- The XGBoost model demonstrated optimal performance for predicting modulus of elasticity across all optimization algorithms, achieving correlation coefficients exceeding 0.99.
- The AdaBoost-ISCSO model demonstrated superior predictive capability for flexural strength (FS), achieving the highest correlation coefficient (R = 0.99347) among all evaluated algorithms. This near-perfect linear correlation indicates exceptional agreement between predicted and experimental FS values.
- The support vector machine (SVM) model demonstrated limited predictive capability in all optimization scenarios, consistently underperforming relative to alternative models.
- IRO emerged as the most effective optimizer, as it produced two of the top-performing models while also demonstrating the fastest computational efficiency.
- The Partial Dependence Plot (PDP) reveals a consistent decline in mechanical properties as rubber weight (RW) increases, with most data concentrated in the 10–60 kg/m^3 range. Smaller rubber particles had a stronger impact than larger ones.
- SHAP-based feature importance analysis revealed that rubber particles negatively impact rubberized concrete's mechanical properties, potentially reducing structural performance. Conversely, optimal proportions of fine aggregates, coarse aggregates, and cement, indicated by positive SHAP values, were key to superior mechanical properties.

Limitations and future directions

Although the proposed framework achieved high predictive accuracy, some limitations must be acknowledged. The dataset, while carefully curated, consisted of 315 samples compiled from multiple sources, which may restrict the statistical depth of certain analyses. The parameter ranges were bounded (e.g., rubber content ≤ 350 kg/m^3, 28-day curing age only), limiting extrapolation to extreme mix designs or long-term durability predictions. Furthermore, while simpler baseline models such as linear regression were tested and confirmed to perform poorly, their detailed results were omitted for brevity. External validation using independent datasets demonstrated deviations of less than 10%, but further large-scale experimental testing is needed to fully confirm generalizability across different eco-concrete systems. Finally, although residual and parity plots confirmed unbiased performance for representative models, comprehensive residual diagnostics are provided only in Supplement.

Future research should therefore expand the dataset with broader parameter ranges, include durability and time-dependent properties, and integrate additional baseline models for benchmarking. Moreover, transfer learning and hybrid frameworks combining physics-based and data-driven models offer promising directions for strengthening predictive reliability in sustainable concrete applications.

Declaration of competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The database used in this study is in the Supplement, which is available online.

References

[1] Nafees, A., Khan, S., Javed, M. F., Alrowais, R., Mohamed, A. M., Mohamed, A., Vatin, N. I. "Forecasting the mechanical properties of plastic concrete employing experimental data using machine learning algorithms: DT, MLPNN, SVM, and RF", Polymers, 14(8), 1583, 2022. https://doi.org/10.3390/polym14081583

[2] Ofuyatan, O. M., Muhit, I. B., Babafemi, A. J., Osibanjo, I. "Evaluation of self-compacting rubberized concrete properties: Experimental and machine learning approach", Structures, 58, 105423, 2023. https://doi.org/10.1016/j.istruc.2023.105423

[3] Pal, A., Ahmed, K. S., Mangalathu, S. "Data-driven machine learning approaches for predicting slump of fiber-reinforced concrete containing waste rubber and recycled aggregate", Construction and Building Materials, 417, 135369, 2024. https://doi.org/10.1016/j.conbuildmat.2024.135369

[4] Piri, M., Shirzadi Javid, A. A., Momen, R. "Investigation of mechanical and durability properties of recycled aggregate concrete containing crumb rubber considering a new model of elastic modulus", Scientia Iranica, 2023. https://doi.org/10.24200/sci.2023.60833.7013

[5] Ul Islam, M. M., Li, J., Roychand, R., Saberian, M., Chen, F. "A comprehensive review on the application of renewable waste tire rubbers and fibers in sustainable concrete", Journal of Cleaner Production, 374, 133998, 2022. https://doi.org/10.1016/j.jclepro.2022.133998

[6] Thomas, B. S., Kumar, S., Mehra, P., Gupta, R. C., Joseph, M., Csetenyi, L. J. "Abrasion resistance of sustainable green concrete containing waste tire rubber particles", Construction and Building Materials, 124, pp. 906–909, 2016. https://doi.org/10.1016/j.conbuildmat.2016.07.110

[7] Eisa, A. S., Elshazli, M. T., Nawar, M. T. "Experimental investigation on the effect of using crumb rubber and steel fibers on the structural behavior of reinforced concrete beams", Construction and Building Materials, 252, 119078, 2020. https://doi.org/10.1016/j.conbuildmat.2020.119078

[8] Eldin, N. N., Senouci, A. B. "Measurement and prediction of the strength of rubberized concrete", Cement and Concrete Composites, 16(4), pp. 287–298, 1994. https://doi.org/10.1016/0958-9465(94)90041-8

[9] Khatib, Z. K., Bayomy, F. M. "Rubberized Portland cement concrete", Journal of Materials in Civil Engineering, 11(3), pp. 206–213, 1999. https://doi.org/10.1061/(ASCE)0899-1561(1999)11:3(206)

[10] Kaveh, A. "Applications of artificial neural networks and machine learning in civil engineering", Springer, 2024. ISBN 978-3-031-66050-4 https://doi.org/10.1007/978-3-031-66051-1

[11] Lu, S., Wang, J., Jing, G., Qiang, W., Rad, M. M. "Rail defect classification with deep learning method", Acta Polytechnica Hungarica, 19(6), pp. 225–241, 2022. https://doi.org/10.12700/aph.19.6.2022.6.16

[12] Abdollahzadeh, A., Masoudnia, R., Aghababaei, S. "Predict strength of rubberized concrete using artificial neural network", [pdf] WSEAS Transactions on Computers, 10(2), pp. 31–40, 2011. Available at: https://www.wseas.us/e-library/transactions/computers/2011/52-190.pdf [Accessed: 27 August 2025]

[13] Topçu, İ. B., Sarıdemir, M. "Prediction of rubberized concrete properties using artificial neural network and fuzzy logic", Construction and Building Materials, 22(4), pp. 532–540, 2008. https://doi.org/10.1016/j.conbuildmat.2006.11.007

[14] Dat, L. T. M., Dmitrieva, T. L., Duong, V. N., Canh, D. T. N. "An Artificial intelligence approach for predicting compressive strength of eco-friendly concrete containing waste tire rubber", IOP Conference Series: Earth and Environmental Science, 612(1), 012029, 2020. https://doi.org/10.1088/1755-1315/612/1/012029

[15] Gao, X., Yang, J., Zhu, H., Xu, J. "Estimation of rubberized concrete frost resistance using machine learning techniques", Construction and Building Materials, 371, 130778, 2023. https://doi.org/10.1016/j.conbuildmat.2023.130778

[16] Habib, A., Yildirim, U. "Estimating mechanical and dynamic properties of rubberized concrete using machine learning techniques: a comprehensive study", Engineering Computations, 39(8), pp. 3129–3178, 2022. https://doi.org/10.1108/ec-09-2021-0527

[17] Ismael Jaf, D. K., Abdalla, A., Mohammed, A. S., Abdulrahman, P. I., Kurda, R., Mohammed, A. A. "Hybrid nonlinear regression model versus MARS, MEP, and ANN to evaluate the effect of the size and content of waste tire rubber on the compressive strength of concrete", Heliyon, 10(4), e25997, 2024. https://doi.org/10.1016/j.heliyon.2024.e25997

[18] Ly, H.-B., Nguyen, T.-A. "Machine learning-driven innovations in green eco-environmental rubberized concrete design towards sustainability", Materials Today Communications, 39, 108551, 2024. https://doi.org/10.1016/j.mtcomm.2024.108551

[19] Gregori, A., Castoro, C., Venkiteela, G. "Predicting the compressive strength of rubberized concrete using artificial intelligence methods", Sustainability, 13(14), 7729, 2021. https://doi.org/10.3390/su13147729

[20] Hadzima-Nyarko, M., Nyarko, E. K., Ademović, N., Miličević, I., Kalman Šipoš, T. "Modelling the influence of waste rubber on compressive strength of concrete by artificial neural networks", Materials, 12(4), 561, 2019. https://doi.org/10.3390/ma12040561

[21] Sun, Y., Li, G., Zhang, J., Qian, D. "Prediction of the strength of rubberized concrete by an evolved random forest model", Advances in Civil Engineering, 2019(1), 5198583, 2019. https://doi.org/10.1155/2019/5198583

[22] El-Khoja, A. M. N., Ashour, A. F., Abdalhmid, J., Dai, X., Khan, A. "Prediction of rubberised concrete strength by using artificial neural networks", International Journal of Structural and Construction Engineering, 12(11), pp. 1068–1073, 2018.

[23] Diaconescu, R.-M., Barbuta, M., Harja, M. "Prediction of properties of polymer concrete composite with tire rubber using neural networks", Materials Science and Engineering: B, 178(17), pp. 1259–1267, 2013. https://doi.org/10.1016/j.mseb.2013.01.014

[24] Gesoğlu, M., Güneyisi, E., Özturan, T., Özbay, E. "Modeling the mechanical properties of rubberized concretes by neural network and genetic programming", Materials and Structures, 43(1), pp. 31–45, 2010. https://doi.org/10.1617/s11527-009-9468-0

[25] Gunawan, H. C. F., Thedy, J., Setiadji, B. H., Han, A. L., Ottele, M. "A novel Pareto Front Symbiotic Organism Search (PF-SOS) combined with metaheuristic-optimized machine learning for optimal recycled aggregate concrete mixtures", Journal of Building Engineering, 108, 112991, 2025. https://doi.org/10.1016/j.jobe.2025.112991

[26] Javed, M. F., Fawad, M., Lodhi, R., Najeh, T., Gamil, Y. "Forecasting the strength of preplaced aggregate concrete using interpretable machine learning approaches", Scientific Reports, 14(1), 8381, 2024. https://doi.org/10.1038/s41598-024-57896-0

[27] Hasanipanah, M., Abdullah, R. A., Iqbal, M., Ly, H.-B. "Predicting rubberized concrete compressive strength using machine learning: a feature importance and partial dependence analysis", Journal of Science and Transport Technology, 3(1), pp. 26–43, 2023. https://doi.org/10.58845/jstt.utt.2023.en.3.1.27-44

[28] Kaveh, A., Eskandari, A., Movasat, M. "Buckling resistance prediction of high-strength steel columns using Metaheuristic-trained Artificial Neural Networks", Structures, 56, 104853, 2023. https://doi.org/10.1016/j.istruc.2023.07.043

[29] Shayanfar, M., Zaherbin, P., Jelokhani Niaraki, P., Eskandari, A. "Force-displacement relation for lumped plasticity model of compact square concrete-filled steel tube columns", Results in Engineering, 23, 102619, 2024. https://doi.org/10.1016/j.rineng.2024.102619

[30] Kaveh, A., Eskandari, A. "Tuned African Vultures Optimization Algorithm for Optimal Design of Skeletal Structures Employing Multi-Stage Parameter Adjustment", Iranian Journal of Science and Technology, Transactions of Civil Engineering, 49(2), pp. 1211–1232, 2025. https://doi.org/10.1007/s40996-024-01662-9

[31] Vadivel, T. S., Suseelan, A., Karthick, K., Safran, M., Alfarhood, S. "Experimental Investigation and Machine Learning Prediction of Mechanical Properties of Rubberized Concrete for Sustainable Construction", Scientific Reports, 14(1), 22725, 2024. https://doi.org/10.1038/s41598-024-73504-7

[32] Yang, L., Lai, B., Xu, R., Hu, X., Su, H., Cusatis, G., Shi, C. "Prediction of alkali-silica reaction expansion of concrete using artificial neural networks", Cement and Concrete Composites, 140, 105073, 2023. https://doi.org/10.1016/j.cemconcomp.2023.105073

[33] Jia, J.-F., Chen, X.-Z., Bai, Y.-L., Li, Y.-L., Wang, Z.-H. "An interpretable ensemble learning method to predict the compressive strength of concrete", Structures, 46, pp. 201–213, 2022. https://doi.org/10.1016/j.istruc.2022.10.056

[34] Breiman, L. "Random forests", Machine Learning, 45(1), pp. 5–32, 2001. https://doi.org/10.1023/A:1010933404324

[35] Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., Scholkopf, B. "Support vector machines", IEEE Intelligent Systems and Their Applications, 13(4), pp. 18–28, 1998. https://doi.org/10.1109/5254.708428

[36] Friedman, J. H. "Stochastic gradient boosting", Computational Statistics & Data Analysis, 38(4), pp. 367–378, 2002. https://doi.org/10.1016/S0167-9473(01)00065-2

[37] Yang, X.-S. "Engineering optimization: An introduction with metaheuristic applications", John Wiley & Sons, 2010. ISBN 978-0-470-58246-6

[38] Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., …, Lindauer, M. "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges", Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 13(2), e1484, 2023. https://doi.org/10.1002/widm.1484

[39] Kaveh, A., Biabani Hamedani, K. "A hybridization of growth optimizer and improved arithmetic optimization algorithm and its application to discrete structural optimization", Computers & Structures, 303, 107496, 2024. https://doi.org/10.1016/j.compstruc.2024.107496

[40] Kaveh, A., Ghazaan, M. I., Bakhshpoori, T. "An improved ray optimization algorithm for design of truss structures", Periodica Polytechnica Civil Engineering, 57(2), pp. 97–112, 2013. https://doi.org/10.3311/ppci.7166

[41] Jia, H., Zhang, J., Rao, H., Abualigah, L. "Improved sandcat swarm optimization algorithm for solving global optimum problems", Artificial Intelligence Review, 58(1), 5, 2024. https://doi.org/10.1007/s10462-024-10986-x

[42] Shafaie, V., Movahedi Rad, M. "Dem-driven investigation and AutoML-Enhanced prediction of Macroscopic behavior in cementitious composites with Variable frictional parameters", Materials & Design, 254, 114069, 2025. https://doi.org/10.1016/j.matdes.2025.114069

[43] Khan, K., Iqbal, M., Jalal, F. E., Nasir Amin, M., Waqas Alam, M., Bardhan, A. "Hybrid ANN models for durability of GFRP rebars in alkaline concrete environment using three swarm-based optimization algorithms", Construction and Building Materials, 352, 128862, 2022. https://doi.org/10.1016/j.conbuildmat.2022.128862

[44] Moosbauer, J., Herbinger, J., Casalicchio, G., Lindauer, M., Bischl, B. "Explaining hyperparameter optimization via partial dependence plots", In: Advances in Neural Information Processing Systems 34 (NeurIPS 2021), online, 2021, pp. 2280–2291. ISBN 9781713845393 [online] Available at: https://proceedings.neurips.cc/paper/2021/hash/12ced2db6f0193dda91ba86224ea1cd8-Abstract.html [Accessed: 27 August 2025]

[45] Cheng, B., Mei, L., Long, W.-J., Kou, S., Li, L., Geng, S. "Ai-guided proportioning and evaluating of self-compacting concrete based on rheological approach", Construction and Building Materials, 399, 132522, 2023. https://doi.org/10.1016/j.conbuildmat.2023.132522

[46] MathWorks "MATLAB, (R2024a)", [computer program] Available at: https://www.mathworks.com [Accessed: 27 August 2025]

[47] Karunarathna, S., Linforth, S., Kashani, A., Liu, X., Ngo, T. "Effect of recycled rubber aggregate size on fracture and other mechanical properties of structural concrete", Journal of Cleaner Production, 314, 128230, 2021. https://doi.org/10.1016/j.jclepro.2021.128230

[48] Shafaie, V., Movahedi Rad, M. "Multi-objective genetic algorithm calibration of colored self-compacting concrete using DEM: An integrated parallel approach", Scientific Reports, 14(1), 4126, 2024. https://doi.org/10.1038/s41598-024-54715-4

[49] Raghavan, D., Huynh, H., Ferraris, C. F. "Workability, mechanical properties, and chemical stability of a recycled tyre rubber-filled cementitious composite", Journal of Materials Science, 33(7), pp. 1745–1752, 1998. https://doi.org/10.1023/a:1004372414475