https://doi.org/10.3311/PPci.44556

Periodica Polytechnica Civil Engineering

Optimized Physics-informed Machine Learning Modeling with GAIN Data Imputation for Sustainable Pozzolanic Concrete Strength Prediction

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: 04 May 2026, Accepted: 12 August 2026, Published online: 21 September 2026

Abstract

The reliable design of sustainable concrete demands predictive models integrating data-driven insights with fundamental physical laws. This paper presents a new framework to predict the 28-day compressive and tensile strengths of pozzolanic concrete with metakaolin, silica fume, and zeolite. This study collected 440 experimental samples from published studies, using 18 input variables related to mix composition, materials, and specimen geometry. Unlike previous research, this study included several influential factors, such as cement grade, the active solid mass of superplasticizer, pozzolanic particle size, and specimen dimensions for developing models. To address missing data, a Generative Adversarial Imputation Network (GAIN) was developed and tested its reliability by comparing it to XGBoost-based Multivariate Imputation by Chained Equations (MICE) with missing rates up to 90%. This paper proposes a new Physics-Informed Machine Learning (PIML) framework that combines an XGBoost predictor with fundamental concrete mechanics constraints, such as porosity–strength relationships, size-effect laws, and tensile–compressive strength correlations, which, to the best of our knowledge, have not been studied together before. Model hyperparameters and coefficients were optimized using the Improved Hybrid Growth Optimizer (IHGO) algorithm. The proposed PIML model achieved reliable predictive performance, with coefficients of determination (R^2) of 0.9612 and 0.8671 on an independent testing set for compressive and tensile strengths, respectively. Analysis using PDP and SHAP confirmed that pozzolanic fineness, cement quality, water content, and superplasticizer activity are the most important factors. This framework offers an accurate, understandable, and physically based tool for designing sustainable pozzolanic concrete.

Keywords

sustainable pozzolanic concrete, physics-informed machine learning, generative adversarial imputation network (GAIN), XGBoost, metaheuristic algorithm, hyperparameter optimization

1 Introduction

Societies globally face a major threat from climate change, a crisis directly linked to the construction industry. This industry accounts for 37% of global CO_2 emissions, with about a quarter of that coming from making cement [1]. Replacing some of the high-emission Ordinary Portland Cement (OPC) with supplementary cementitious materials (SCMs) is a known way to make concrete last longer and lessen its environmental harm. SCMs, including industrial by-products and natural pozzolans such as metakaolin [2], silica fume [3], and zeolite, contribute to these improvements through their pozzolanic properties [4, 5].

In chemical terms, pozzolanic supplementary cementitious materials react with portlandite, which is calcium hydroxide, to create a secondary calcium-silicate-hydrate (C–S–H) gel [6]. This reaction improves the pore structure of the material, making it less permeable. This reduction in permeability restricts the entry of harmful substances, such as sulfates. As a result, these materials prolong the lifespan of concrete and lessen the environmental and financial burdens linked to cement creation, marking them as a versatile approach for sustainable concrete [7].

Traditional experimental methods for testing the mechanical properties of pozzolanic concrete require laboratory facilities, take considerable time (28 days), are expensive, and can be affected by experimental differences [8]. Because of this, it is important to find a more affordable and efficient way as a reliable tool to measure the strength of pozzolanic concrete using previous experimental data [9].

Cite this article as: Kaveh, A., Eskandari, A., Piri, M. "Optimized Physics-informed Machine Learning Modeling with GAIN Data Imputation for Sustainable Pozzolanic Concrete Strength Prediction", Periodica Polytechnica Civil Engineering, 2026. https://doi.org/10.3311/PPci.44556

Machine learning (ML) provides a promising option by analyzing complex data to predict performance [10, 11]. Many studies have shown the reliable and accurate applicability of ML in different areas, especially in civil engineering [12]. Recent studies have used different supervised machine learning methods to look at large datasets and get better at predicting how strong pozzolanic concrete is. A summary of relevant investigations utilizing soft computing approaches, including details of their respective models and datasets, is provided in Table 1 [8, 9, 13–18]. While these methods predict well, they do not explain much about how concrete actually works physically. Conventional machine learning approaches are further restricted by their dependence on supervised learning frameworks and the demand for interpretability in physical terms [19]. This research introduces Physics-Informed Machine Learning as a framework designed to integrate empirical knowledge with data-driven modeling approaches.

This study expands on earlier work by presenting a new structure that improves how well we predict the strength of pozzolanic concrete. It makes these predictions more accurate, physically sound, and widely applicable. The novelty of this work presents three main areas: input feature engineering, data imputation (GAIN) and handling, and the design of the predictive model (PIML). To address deficiencies in current input parameters, our model incorporates four important, previously overlooked factors that are physically deterministic into the predictive feature set. These additions include:

1. The Standard 28-day Compressive Strength of the Cement Matrix (cement grade), which sets the fundamental strength of the paste.
2. The Active Solid Mass of Superplasticizer (calculated as Solid Content (%) * Dosage (kg)), providing a standardized and precise measure of the chemically active component across different commercial products.
3. The Pozzolanic Material Particle Size, which largely controls how dense it packs and how much water it needs.
4. The Sample Size and Geometry, which directly considers how sample size affects measured strength.

To avoid the bias that comes from just filling in missing data with mean imputation methods when combining datasets, the Generative Adversarial Imputation Network (GAIN) and Multivariate Imputation by Chained Equations (MICE) with XGBoost are employed. This makes sure our training data is more accurate.

Table 1 Application of different ML models to pozzolanic concrete

Year	Ref.	Datasets	Model used	Input variables	Output variables
2025	[9]	761	XGBoost; GBR; RFR; KNN; SVR; Stacking	12 inputs: Cement; silica fume; fly ash; slag; nano silica; limestone powder; steel fiber; water; aggregates; superplasticizer; curing time	Compressive strength
2025	[13]	97	ANN; GPR; SVM; RFR	8 inputs: Cement; pozzolanic materials; w/b ratio; sand; gravel; superplasticizer; curing period	Bond strength
2024	[8]	158	AdaBoost; XGBoost; GBRT; LightGBM	7 inputs: Cement; silica fume; w/c ratio; fine aggregate; coarse aggregate; superplasticizer; curing period	Tensile strength
2024	[14]	526	LR; SVM; CatBoost; GB; RF; DT	8 inputs: Cement; pozzolan; w/c ratio; water; fine aggregate; coarse aggregate; superplasticizer; curing period	Compressive strength
2024	[15]	130	Bagging; RF; AdaBoost; GB; XGBoost; CatBoost; LightGBM	7 inputs: cement type; cement content; pozzolan type; pozzolan content; w/b ratio; exposure condition; age of concrete	Aging factor
2023	[16]	1073	XGBoost; RF; SVM; LightGBM	12 inputs: Water content; cement content; slag content; fly ash content; silica fume content; fine aggregate content; coarse aggregate content; superplasticizer content; fresh density; compressive strength; age of compressive strength test; age of migration test	Chloride resistance
2021	[17]	283	DT; SVM	6 inputs: Cement; silica fume; water; fine aggregate; coarse aggregate; superplasticizer	Compressive strength
2021	[18]	367	DNN; GBR; DTR; SVR; RF; kNN	12 inputs: Cement; silica fume; metakaolin; limestone powder; slag; pulverised fuel ash; perlite powder; Ground pumice; w/b ratio; water; fine aggregate; coarse aggregate; superplasticizer	Compressive strength; Embodied carbon

The cornerstone of predictive models proposed and employed in this study is a novel Physics-Informed Machine Learning (PIML). It guides the data-based model using three key physical laws from concrete mechanics:

1. the porosity–strength relationship (Balshin's Law), which links material strength to effective pore volume fraction;
2. the size–effect law (Bažant's Law), which accounts for changes in measured strength due to specimen geometry;
3. the tensile–compressive strength relationship (ACI CODE 318-08 [20]), which enforces physical consistency between the two predicted mechanical properties.

This creates a strong, combined solution that balances prediction accuracy with known physical rules.

In this study, it is intended to develop accurate and reliable models based on an experimental data set (four overlooked inputs are included, and their missing points are imputed using GAIN and MICE models), which are also empowered by employing three physical laws to predict the strength of pozzolanic concrete.

2 Basic concepts and definitions

2.1 Performance evaluation matrices

Statistical measures function as quantitative gauges for evaluating how well machine learning models perform. They provide a scalable way to measure different parts of a model's effectiveness, such as predictive accuracy and precision. Interpreting these metrics facilitates a deeper understanding of a model's operational strengths [9].

2.1.1 Coefficient of determination

Coefficient of determination (R^2) measures the concordance of a regression model's predictions with the actual data. A correlation coefficient value of one indicates that the model fits the data perfectly, whereas $R^2 = 0$ signifies that the model does not explain any of the data's variation [21]. Equation (1) provides the formulation for R^2:

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

where a_i shows the actual observed value, e_i denotes the corresponding predicted value, $\bar{a}_i$ denotes the mean of observed values, and n signifies the total number of data points.

2.1.2 Mean Squared Error

Mean Squared Error (MSE) quantifies model accuracy by averaging the squared residuals, the differences between predicted and actual values. Compared to Mean Absolute Error (MAE), MSE is more susceptible to outliers because squaring the errors before averaging gives a much larger weight to bigger discrepancies. Superior model performance is indicated by a lower MSE. Equation (2) defines the Mean Squared Error:

$$\mathrm{MSE} = \frac{\sum_{i=1}^{n} (a_i - e_i)^2}{n}. \tag{2}$$

2.2 Normalization

Data normalization is a critical step when variables operate on different scales to ensure that a limited number of highmagnitude variables do not disproportionately influence the learning process [8]. In this study, the adopted Min-Max normalization technique was implemented within the range of 0 to 1 [21]. The formulation of the scaling function is provided in Eq. (3):

$$X_n = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \tag{3}$$

where X_n is the normalized variable, X is the raw value, and $X_{\min}$ and $X_{\max}$ are the observed minimum and maximum of X.

2.3 Missing data imputation

Effective machine learning relies on complete, high-quality data. In practice, though, data often has missing values that can reduce predictive accuracy. To address this, robust data preprocessing requires advanced imputation methods. In this study, we used several imputation techniques to handle missing data. Let represent the d-dimensional space. Let $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_d$ denote the d-dimensional space a complete data vector. Consider a random variable $X = (X_1, \ldots, X_d)$, which may be continuous or binary, defined on $\mathcal{X}$. The data vector is denoted by $\boldsymbol{X}$, and the corresponding mask vector by $\boldsymbol{M} = (M_1, \ldots, M_d)$, a random variable with a state space of $\{0,1\}^d$. For each part $i = \in \{1, \ldots, d\}$, the space is formed by $\tilde{X}_i = X_i \cup \{Nan\}$, where the Nan symbol denotes a missing value. A random variable $\tilde{X} = (\tilde{X}_1, \ldots, \tilde{X}_d)$, where $\tilde{X} \in \tilde{\boldsymbol{\mathcal{X}}}$ is defined on the product space $\tilde{X} = \tilde{X}_1 \times \ldots \times \tilde{X}_d$ [22], where $\tilde{\boldsymbol{\mathcal{X}}}$ denotes the space of observed data vectors that may contain missing values. The specific scaling transformation is defined by Eq. (4):

$$\tilde{X}_i = \begin{cases} X_i, & \text{if } M_i = 1 \\ Nan, & \text{if } M_i = 0 \end{cases}. \tag{4}$$

2.3.1 Generative adversarial imputation nets

A Generative Adversarial Network (GAN) is an unsupervised generative model, a class of algorithms designed to characterize and model the underlying probability distribution of a dataset [23]. In this study, we used the Generative Adversarial Imputation Nets (GAIN) framework for reliable data imputation. GAIN adapts the adversarial training approach from GANs directly to handle missing data. It does this by setting up a Minimax game between a Generator (G) and a Discriminator (D). The diagram of the imputation architecture based on GAN is depicted in Fig. 1.

Generator

The Generator is responsible for producing estimated values for the missing entries. It receives the data samples of $\tilde{X}$, M, and a random noise vector $\boldsymbol{Z}$. Random noise is introduced into the missing components to initialize the imputation process and produce a complete data matrix [23]. The imputed data and the complete data are provided in Eqs. (5) and (6):

$$\bar{\boldsymbol{X}} = G\left(\tilde{X}, M, (1-M) \odot \boldsymbol{Z}\right), \tag{5}$$

$$\hat{X} = M \odot \tilde{X} + (1-M) \odot \bar{\boldsymbol{X}}, \tag{6}$$

where $\bar{\boldsymbol{X}}$ denotes the imputed data vector, and G produces an output for each component, irrespective of whether the corresponding value was originally observed. $\hat{X}$ constitutes the final, complete dataset. The symbol $\odot$ represents element-wise multiplication.

Discriminator

The discriminator's goal is to tell apart real data from generated imputations. It receives $\hat{X}$ and the Hint vector ($\boldsymbol{H}$). The hint mechanism helps the Generator learn the true data distribution instead of converging to a trivial solution. The Hint vector is a sampled version of the actual mask. By giving the Discriminator some information about which data was missing, the mechanism encourages it to focus from the explicit mask to the more subtle differences between real and imputed data. This setup pushes the Generator to create more accurate imputations [22].

Objective function

The discriminator is trained to maximize the accuracy of its predictions for $\hat{M}$. Conversely, the generator is trained to minimize the discriminator's predictive accuracy of $\hat{M}$. The binary cross-entropy loss function is employed to quantify the divergence between the predicted probabilities and the true binary labels [19]. Given that the discriminator estimates the mask matrix utilizing the hint matrix, the objective function for the adversarial architecture is formulated as Eq. (7):

$$\min_G \max_D \mathbb{E}_{M,H,\hat{X}} \left[M^T \log \hat{\boldsymbol{M}} + (1-M)^T \log\left(1-\hat{\boldsymbol{M}}\right) \right], \tag{7}$$

where the log operator is applied element-wise, and $\hat{\boldsymbol{M}}$ denotes the predicted mask matrix, expressed as $\hat{\boldsymbol{M}} = D(\hat{X}, H)$. Furthermore, the fundamental binary cross-entropy loss function, denoted as $\mathcal{L}(x, y)$ is characterized as the negative expectation of the logarithm of the corrected predicted probabilities, as expressed in Eq. (8). Consequently, Eq. (9) can be reformulated and simplified.

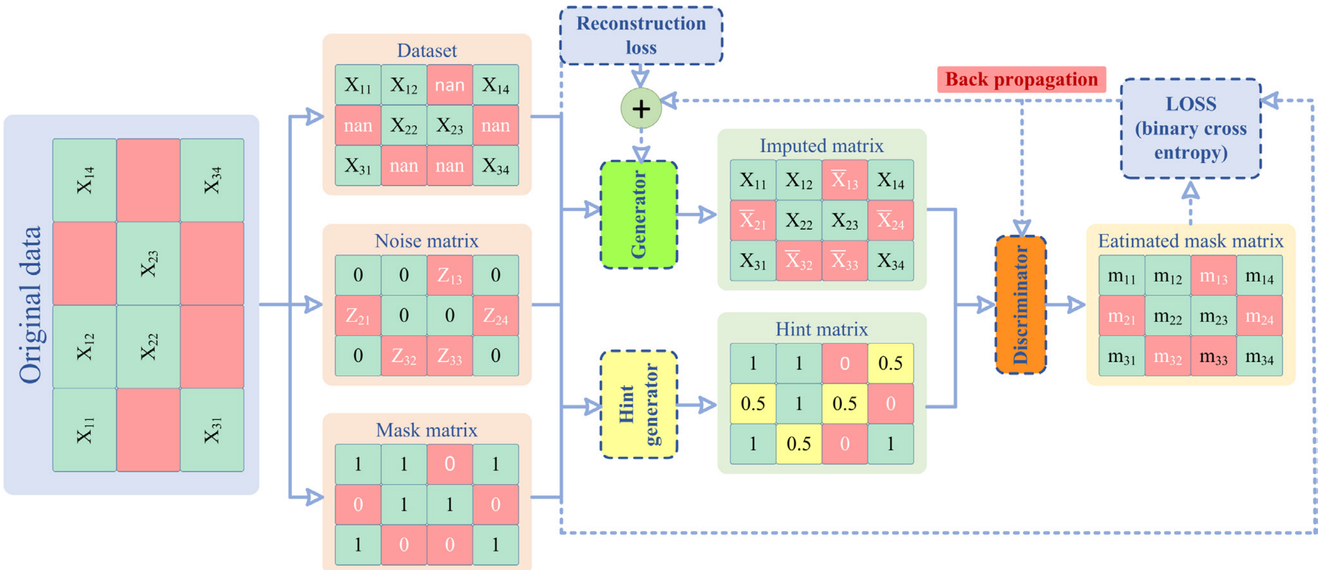

Fig. 1 The architecture of GAIN

$$\mathcal{L}\left(x,y\right)=\sum_{i=1}^{n}\left[x_i\log\left(y_i\right)+\left(1-x_i\right)\log\left(1-y_i\right)\right] \tag{8}$$

$$\min_{G}\max_{D}\mathbb{E}\left[\mathcal{L}\left(M,\boldsymbol{M}\right)\right] \tag{9}$$

2.3.2 Multivariate Imputation by Chained Equations with XGBoost

Multivariate Imputation by Chained Equations (MICE) is an iterative, model-based methodology that imputes missing values by minimizing a predefined error function across the dataset. It employs a sequential estimation procedure, wherein each variable containing missing data is imputed by conditioning on the remaining variables through an iterative cycle [24]. In this study, an iterative imputer with the Extreme Gradient Boosting (XGBoost) algorithm was used, which is a strong model that uses decision trees to make predictions, for every step of the imputation. In the first round, any missing data is filled with a simple average from the existing data for that feature. After that, features are handled one after another. For each selected feature with missing observations, the previously imputed values are discarded before re-estimation. The removed set is then temporarily considered the dependent variable, whereas the remaining features are designated as independent variables. An XGBoost regressor is subsequently trained using only the rows that have no missing data. The initially missing values are replaced with the new, better predictions from the XGBoost model. This iterative procedure, in which features with missing entries are successively updated, is repeated for a predetermined number of cycles until the imputation estimates converge.

2.4 Physics-Informed Machine Learning formulation

Physics-Informed Machine Learning (PIML) represents the incorporation of empirical principles into the machine learning training framework through the imposition of physics-based empirical constraints. This methodology guarantees that the model aligns with experimental observations while simultaneously conforming to established empirical and physical laws governing concrete strength prediction [21]. In contrast to conventional machine learning models, the proposed modelling framework integrates the data-driven Extreme Gradient Boosting (XGBoost) algorithm with physics-informed constraints as hints that encode fundamental principles of pozzolanic concrete behavior. The total loss function (L_{total}) is formulated as a weighted sum of the XGBoost loss and the physics-informed loss components, as presented in Eq. (10):

$$L_{\text{total}}=L_{\text{data}}+\lambda_1 L_{\text{material}}+\lambda_2 L_{\text{geometry}}+\lambda_3 L_{\text{CS}-\text{TS}}, \tag{10}$$

where λ represents adaptive weighting coefficients that balance the contribution of the respective constraint terms.

2.4.1 The data-driven XGBoost loss

The data loss component enforces consistency between the model's predicted outputs and the empirically observed values within the experimental dataset [21]. It is shown in Eq. (11):

$$L_{\text{data}}=\text{MSE}\left(f,f_{\text{actual}}\right), \tag{11}$$

where f is the strength predicted by the trained model, and f_{actual} is the corresponding experimentally measured value.

2.4.2 Material-based physics constraint

This regularization term establishes a basic connection between a material's mechanical strength and its internal structure, with porosity as the principal governing factor. It is formulated based on the Balshin Law [25], which relates material strength to the effective pore volume fraction (ϕ), as illustrated in Eq. (12):

$$L_{\text{material}}=\text{MSE}\left(f,\sigma_0\left(1-\phi\right)^n\right), \tag{12}$$

where f is the strength predicted by the trained model, and σ_0 (strength at zero porosity) and n (Balshin exponent) are treated as trainable. The PIML is set up to learn a sub-network on its own that predicts porosity, wherein porosity is expressed as a functional relationship of the mix design parameters, including cement grade, pozzolan weight, water-to-cement ratio, and superplasticizer dosage.

2.4.3 Geometry-based physics constraint

This term applies the Size Effect Law to account for changes in experimental results due to different testing procedures, which shows that the strength decreases as the sample gets bigger. This constraint is implemented through the well-known formulas, Bažant's Size Effect Law [26], as expressed in Eqs. (13) and (14):

$$L_{\text{geometry}}=\text{MSE}\left(\frac{f_i}{f_{ref}},f\left(D_i\right)\right), \tag{13}$$

$$f\left(D_i\right)=\frac{1}{\sqrt{1+\dfrac{D_i}{D_0}}}, \tag{14}$$

where f_i denotes the predicted strength for the i-th specimen, f_{ref} refers to the intrinsic strength, D_i is the characteristic structural dimension, $f(D_i)$ constitutes the fundamental component of Bažant's Size Effect Law, and D_0 is treated as trainable.

2.4.4 Inter-property constraint

This constraint integrates the codified empirical relationship, as defined in standard ACI CODE 318-08 [20], as illustrated in Eq. (15), linking the tensile and compressive strength of concrete into the model's learning objective:

$$L_{\mathrm{CS-TS}} = \mathrm{MSE}\left(f_{\mathrm{TS}}, a \cdot f_{\mathrm{CS}}^{b}\right), \tag{15}$$

where a and b are considered as hyperparameters to be tuned that enforce a physically consistent correlation between the predicted mechanical properties.

2.5 K-fold cross-validation

When sample sizes are small, changes in how the training data is divided can lead to quite different results. K-fold cross-validation is the most common way to create validation sets. This method divides the dataset into K folds and iteratively trains the model K times. Each time, a different fold is used for validation to help prevent overfitting [27]. This study applied a 5-fold cross-validation scheme, allocating 72% of the data for model training and 18% for validation.

2.6 Hyperparameter tuning

Hyperparameter tuning entails the systematic modification of model parameters to identify the configuration that yields optimal performance. This procedure increases predictive accuracy and strengthens the model's capacity to generalize effectively to previously unseen data. To find the most reliable and precise predictions, hyperparameter tuning requires setting value limits and evaluating model behavior within the limits [28]. To handle missing data, Grid Search was used for the GAIN and MICE models. For the physics-informed model, the Improved Hybrid Growth Optimizer (IHGO) was employed. Both methods systematically explored for the best parameter settings.

2.6.1 Grid search

Grid search, as a subset-based method, explores hyperparameters by setting a lower and upper limit, along with a certain number of steps. It checks every mix of hyperparameter values inside that defined space to find the one that works best [29].

2.6.2 Improved Hybrid Growth Optimizer

Improved Hybrid Growth Optimizer (IHGO) is a hybrid algorithm that combines the Growth Optimizer (GO) with the Improved Arithmetic Optimization Algorithm (IAOA) to overcome GO's sensitivity to parameters and limited exploitation. It incorporates enhanced exploratory dynamics, an elitism-based replacement strategy, structural improvements, and a refined reflection phase. These modifications enhance IHGO's convergence stability, solution quality, and adaptability to discrete design spaces, making it well-suited for predictive modeling in civil engineering [30].

2.7 Regression slope analysis

Model selection efficacy was assessed on the basis of regression slope parameters, derived from plots of experimentally observed values along the x-axis against predicted values along the y-axis. A regression slope that goes over 0.8 shows better alignment between observed data and what the model predicts [31]. In linear regression analysis, the intercept (also called the bias) shows the expected result when all predictors are zero. Along with the slope coefficients, it shapes the regression line. If the intercept is close to zero, there is little bias. A large intercept might mean there is a systematic error. Both the intercept and slope help determine how well the regression line fits the actual data. Linear regression can also be used to compare two different measurements or methods [31]. This relationship is formally expressed in Eq. (16):

$$\hat{Y} = \beta_1 X + \beta_0, \tag{16}$$

where $\hat{Y}$ represents the estimated features, β_1 denotes the slope parameter, X corresponds to the predictor variable, and β_0 signifies the intercept term.

2.8 Partial dependence plot

Partial Dependence Plot (PDP) depicts the functional relationship between a restricted subset of input variables and the predicted outcomes of a model. It illustrates how selected input variables influence predictive outputs. As a global method, it highlights the overall relationship between variables and outcomes, providing insight into each feature's relative contribution [32].

2.9 Shapley additive explanations analysis

Shapley Additive Explanations (SHAP) summary plot offers a visual representation of how individual features affect the output of a predictive model. In the summary plot, the feature being examined is shown on the vertical axis, and its corresponding Shapley value is displayed on the horizontal axis. Each dot on the plot represents the Shapley value of a particular feature for one observation [33].

3 Research methodology

3.1 Data collection

This study employed a dataset of 440 distinct samples, compiled from the existing literature and detailed in Table S1 (available in the Supplement), to evaluate a predictive model for the mechanical properties of environmentally sustainable pozzolanic concrete at 28 days. The dataset is constituted by 18 input variables: Metakaolin Size (MS, μm), Silica Fume Size (SFS, μm), Zeolite Size (ZS, μm), Metakaolin Weight (MW, kg/m^3), Silica Fume Weight (SFW, kg/m^3), Zeolite Weight (ZW, kg/m^3), Grade of Cement (GC, MPa), Cement Content (C, kg/m^3), Water Content (W, kg/m^3), Active Solid mass of Superplasticizer (ASP, kg/m^3), Coarse Aggregate (CA, kg/m^3) and Fine Aggregate (FA, kg/m^3), Specimen Depth for Compressive Strength (DCS, mm), Specimen Length for Compressive Strength (LCS, mm), Specimen Width for Compressive Strength (WCS, mm), Specimen Diameter for Tensile Strength (DTS, mm), Specimen Length for tensile strength (LTS, mm), Specimen Width for Tensile Strength (WTS, mm). Compressive Strength (CS, MPa) and Tensile Strength (TS, MPa) were independently considered as distinct output variables. The collected data were subjected to statistical analysis. The descriptive statistics for the training and testing sets, including maximum and minimum values, mean, mode, median, standard deviation, variance, skewness, kurtosis, and missing rate (%), are summarized in Table 2.

This research measured how much the data varied by calculating the standard deviation (SD). A lower SD means most values are close to the mean, while a higher SD shows the values are more spread out [34]. Skewness and kurtosis help describe the shape of the data compared to a normal curve. Positive skewness reflects the tail stretches to the right, and negative skewness means it stretches to the left. If skewness is zero, the data is perfectly symmetrical. Negative kurtosis indicates the curve is flatter than normal, while positive kurtosis shows it is more peaked. A kurtosis of zero suggests the curve is similar to a normal distribution, neither too flat nor too peaked [35]. The missing rate (%) provides how complete the data is for each variable by reporting the share of missing or unrecorded values. If a variable has zero percent missing values, it means all data for that variable is present.

3.2 Imputation results

A comprehensive analysis of the dataset indicated the presence of partially missing values across DTS, LTS, WTS, and TS. A progressive missingness stability test was conducted to evaluate and compare the performance of GAIN and XGBoost imputation. Hyperparameter optimization *via* grid search was conducted for each model to maximize

Table 2 Descriptive statistics of input and output variables

		Max	Min	Mean	Mode	Median	SD	Var	Skew	Kur	Missing rate (%)
Inputs	MS	19	0	0.8734	0	0	2.3484	5.5151	4.9723	31.8036	0
	SFS	12.35	0	0.3155	0	0	1.5356	2.3580	6.4024	42.2502	0
	ZS	50	0	2.2683	0	0	9.0811	82.4669	4.6660	20.9269	0
	MW	156.7	0	18.7076	0	0	33.2739	1107.1533	1.7715	2.3725	0
	SFW	116.25	0	14.8230	0	0	24.2221	586.7119	1.7348	2.3938	0
	ZW	190	0	9.3661	0	0	29.2416	855.0708	3.5406	12.8301	0
	GC	55	32.5	44.6125	42.5	43	4.3255	18.7100	1.1802	0.5296	0
	C	663.7	97.5	396.3235	500	392.98	90.6141	8210.9071	0.0261	−0.0375	0
	W	240	100	173.8038	140	175	29.6706	880.3461	0.0464	−0.2607	0
	ASP	8.955	0	2.5507	0	1.924	2.3180	5.3733	0.6565	−0.6117	0
	CA	1445	563.12	1024.1237	951	1050.08	165.8136	27494.1513	−0.2786	2.0169	0
	FA	1206	118	748.6732	634	720	166.5401	27735.6135	−0.4711	0.3875	0
	DCS	150	100	130.3409	150	150	24.4506	597.8334	−0.4389	−1.8157	0
	LCS	300	100	139.8864	150	150	32.9801	1087.6864	1.0090	3.7106	0
	WCS	150	0	120.7955	150	150	44.4460	1975.4478	−1.6989	2.1474	0
	DTS	150	60	129.3488	150	150	26.6431	709.8544	−0.7031	−0.9398	51.14
	LTS	300	100	256.0465	300	300	63.1028	3981.9604	−1.0607	−0.1467	51.14
	WTS	400	0	9.0698	0	0	51.8271	2686.0465	6.5418	44.9440	51.14
Outputs	CS	117.4	16.64	57.7640	48	55.635	19.2441	370.3335	0.3541	−0.2267	0
	TS	7.533	2.07	4.3994	2.9	4.02	1.3941	1.9435	0.5406	−0.6597	53.41

performance, as summarized in Table 3. Fig. 2 exhibits that missing data were applied at rates ranging from 10% to 90% to evaluate method robustness. The performance of each imputation method was assessed using the coefficient of determination (R^2). The R^2 values of the XGBoost model indicate strong accuracy at low levels of missingness, but decline sharply as the missing rate increases. As shown in Fig. 2, GAIN's R^2 values remain consistently high (0.9661 at 10% missing, 0.9552 at 20%, and 0.9438 at 90% missing), while the XGBoost-based MICE imputation method shows significant performance degradation (R^2 dropping from 0.9881 to 0.7499 across the same range).

The GAIN model shows considerably better robustness across all missing rate levels, making it a suitable choice for datasets with high levels of missing data. The R^2 score at the 90% missing rate remains close to its value at 10%.

Table 3 Configuration parameters of the proposed GAIN and XGBoost model

Model	Name	Range	Optimal
GAIN	Epochs	[50, 100, 200, 300, 400, 500, 800, 1000]	100
	Batch size	[100, 200, 300, 400, 500, 600]	400
	Hidden dimension	[10, 20, 40, 50, 100, 200, 300, 400, 500]	100
	Hint rate	[0.8, 0.9, 1]	0.9
	Steps to train G after training D	[1, 2, 3, 4, 5]	1
	Alpha	[4, 6, 8, 9, 10, 12, 14]	8
	Learning rate for G	[1e-7, 1e-4, 1e-3, 1e-2]	1e-3
	Learning rate for D	[1e-7, 1e-4, 1e-3, 1e-2]	1e-4
	Activation function	[ReLU, LeakyReLU, ELU]	LeakyReLU
	Optimizer	[Adam, RMSprop, Adagrad, SGD]	Adam
XGBoost	Number of estimators	[50, 100, 300, 500, 600, 800, 1000]	500
	Learning rate	[0.01, 0.03, 0.05]	0.01
	Maximum dept	[1, 3, 5, 7, 8, 10]	5
	Lambda	[0.25, 0.5, 0.75, 1]	0.5

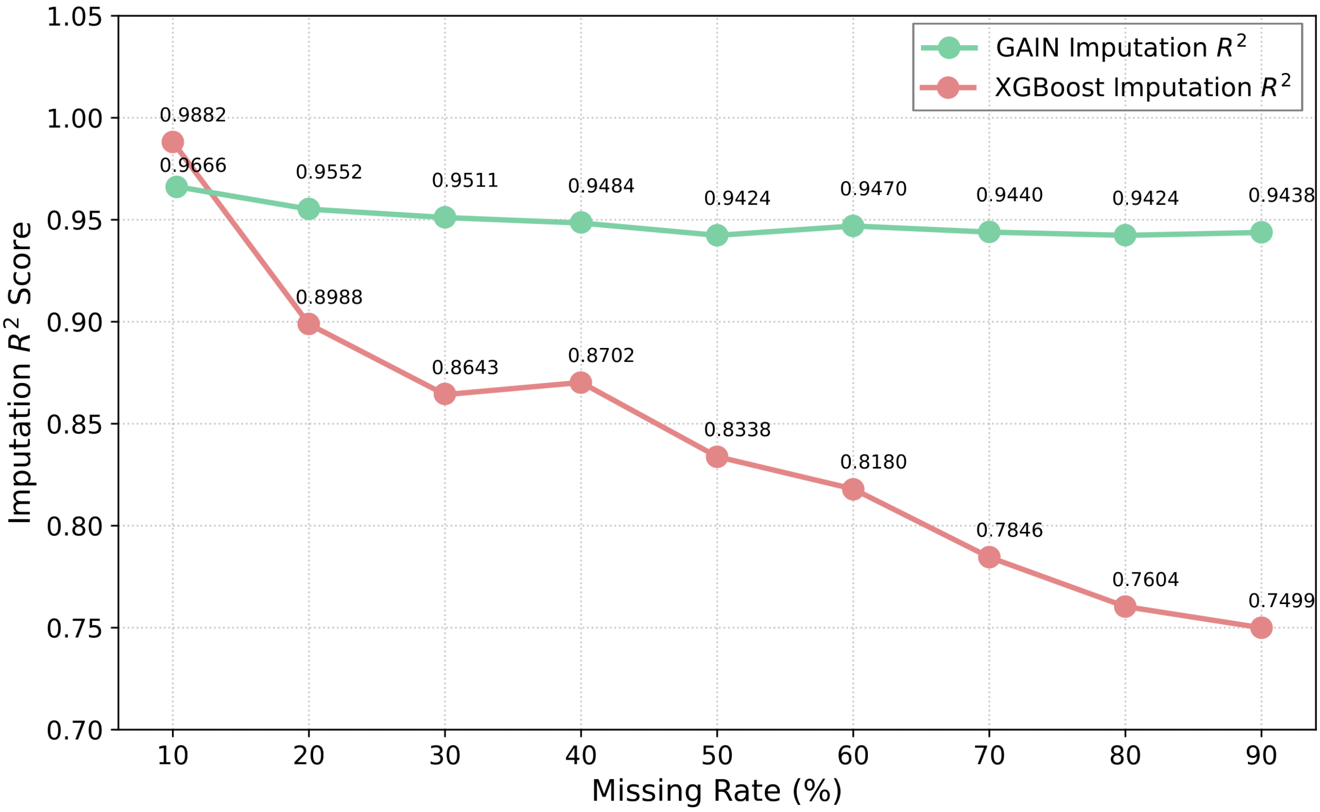

Fig. 2 Performance comparison of GAIN and XGBoost imputation methods

This stability reflects the strength of its generative adversarial architecture in capturing the intrinsic data distribution, not relying solely on predictive regression. While XGBoost performs slightly better when there is very little missing data, GAIN is much more stable and reliable when missingness is high. It is noticeable that GAIN does not become more accurate with less data; rather, it demonstrates remarkable robustness to high levels of missingness. For this reason, GAIN was chosen as the imputation method used in this study to keep the data as accurate as possible for pozzolanic concrete predictive models.

The superior robustness of GAIN can be attributed to its generative adversarial architecture. Unlike MICE, which relies on supervised regression using available features to predict missing values, a task that becomes increasingly difficult as more data is missing, GAIN learns the underlying joint probability distribution of the complete data. The generator produces imputations that are evaluated by the discriminator, which is provided with partial information about the mask vector through the hint mechanism. This adversarial process enables the generator to learn the true data distribution rather than simply performing feature-based regression. Consequently, even when a large proportion of values are missing, GAIN can generate plausible imputations that preserve the statistical properties of the original dataset.

An assessment of the Generator and Discriminator loss curves, as presented in Fig. 3, reveals the integrity of the model's training. In the initial stages of training, both networks display elevated loss values, which is normal as the Generator produces random outputs and the Discriminator can easily distinguish real from fake data. The Generator's loss starts around 1.8, drops quickly, and levels off near 1 by Epoch 40. This means the Generator is learning the data structure and improving at making realistic guesses. At the same time, the Discriminator loss steadily drops from about 0.7 to 0.2 by Epoch 100. When this happens, the GAN has converged, and the Discriminator can no longer easily tell real data from generated data. This result suggests the Generator has learned a data distribution very close to the real one.

3.3 Data visualization

Fig. 4 displays density histograms for all variables in both the original dataset and the dataset imputed with the GAIN algorithm. The features exhibit clear distribution patterns across their ranges. Pozzolanic material particle sizes are predominantly less than 10 μm, indicating that they are very fine and have a large surface area. They also have right-skewed tails that extend to larger sizes, demonstrating that while the majority of particles are ultrafine, there is a small amount of coarser material. FA, CA, and W have roughly normal distributions. The distributions of the real and GAIN-imputed data show a high degree of correspondence, with nearly perfect overlap. This result shows that the imputation model was able to learn and reproduce the original dataset's probability distributions. The imputation framework successfully assigned a zero value to WTS while producing valid estimates for DTS

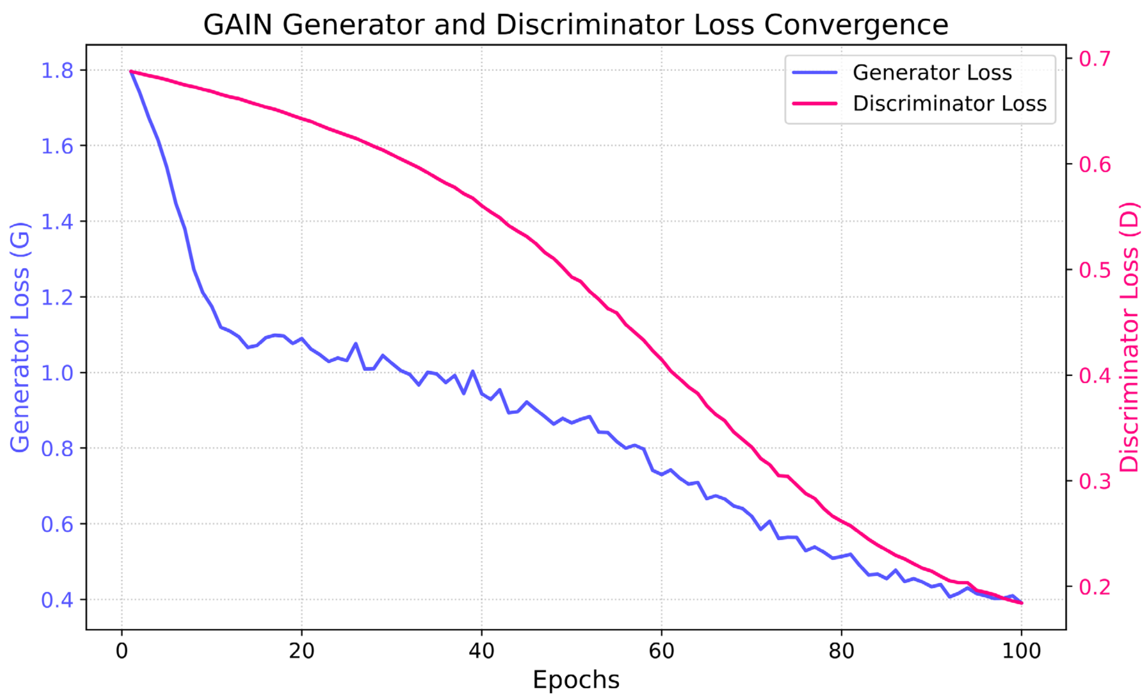

Fig. 3 GAIN Generator and Discriminator loss convergence

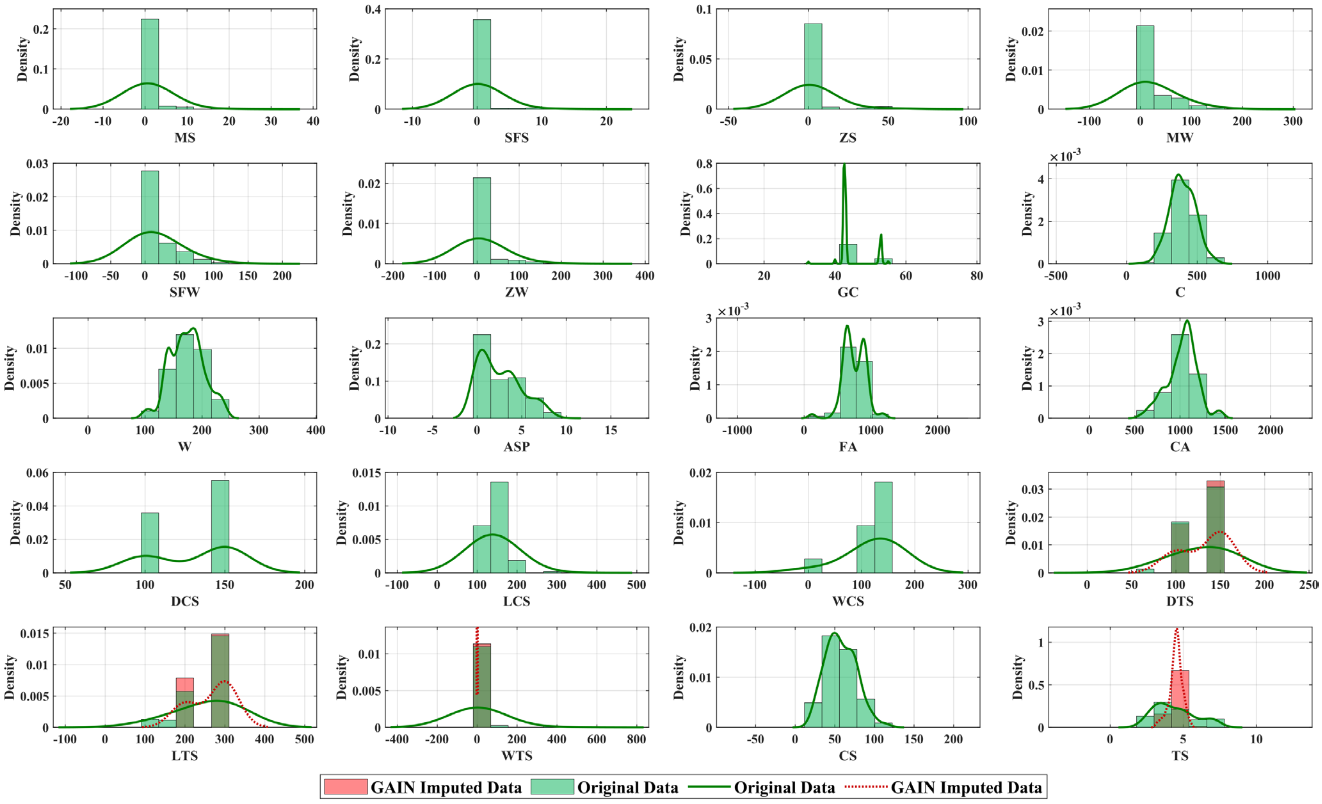

Fig. 4 Histograms of original and imputed data for all variables

and LTS. This accurately reflects the testing standard for cylindrical samples, where width is not defined and is correctly set to zero, whereas diameter and length are the essential measured parameters.

3.4 Pearson correlation analysis

Pearson's correlation coefficient, as shown in Fig. 5, measures linear connections between features. Values close to +1 indicate robust positive correlations, whereas values

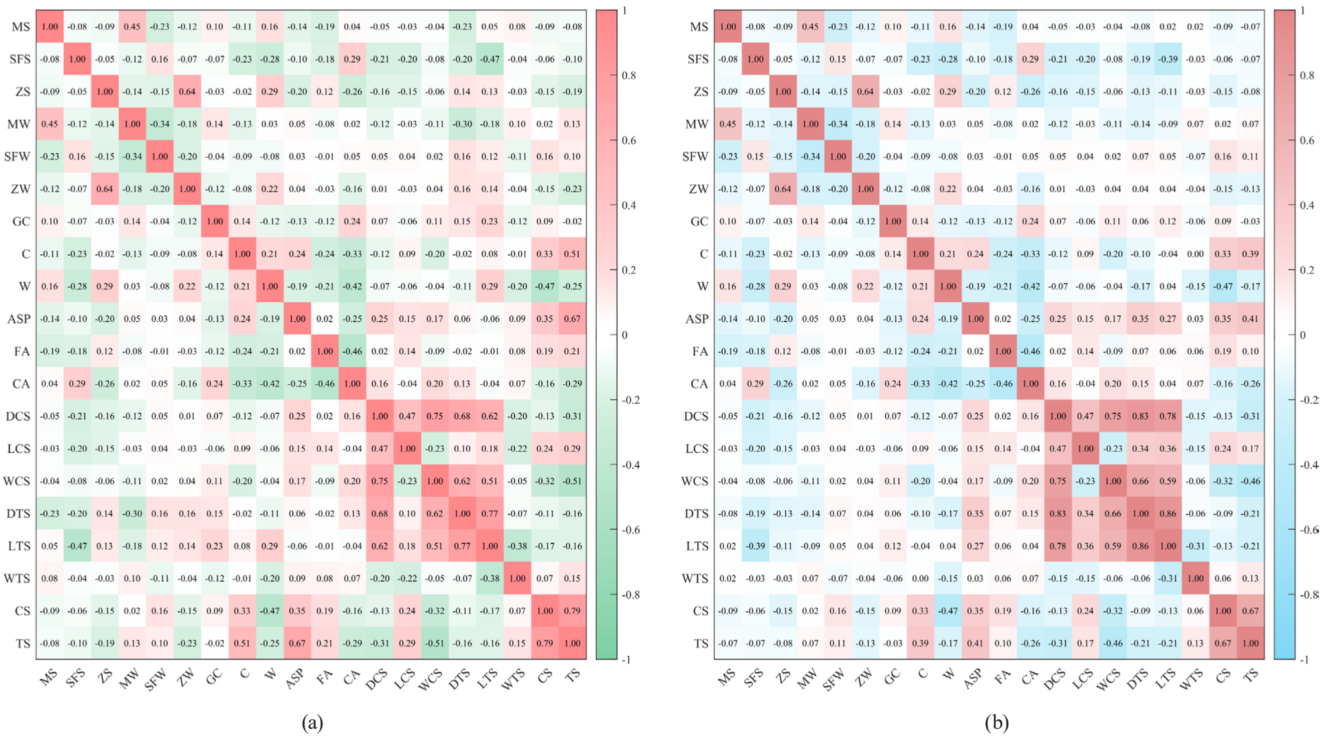

Fig. 5 Pearson correlation coefficients for all features of: (a) original data; (b) GAIN-imputed data

near −1 indicate strong negative correlations. On the other hand, numbers around zero suggest a weak or no linear association [35]. For variables without missing values (e.g., MS, SFS, ZS, MW, C, W), Pearson correlation coefficients are stable between the original and GAIN-imputed datasets. This consistency shows that the GAIN model kept the linear relationships among the complete features. ASP and C exhibit strong positive correlations with both compressive and tensile strength, suggesting that using more of these materials usually improves strength. On the other hand, larger sample sizes and higher water content are linked to lower strength, meaning bigger samples or more water tend to weaken mechanical performance. The heatmap illustrates interrelationships among the mix design variables. For instance, the sample dimensions exhibit strong positive intercorrelations, such that increases in one dimension correspond to increases in others. In addition, the positive effect of 0.24 between C and ASP indicates that high-cement mixes require more admixture to maintain workability. The inverse link of −0.19 between W and ASP indicates that mixes with less water often have more ASP to keep their fresh properties.

The correlation between DTS and LTS increased from 0.77 in the Original Data to 0.86 in the GAIN-imputed data, illustrating a stronger linear connection. The positive correlation between CS and TS went down from 0.79 in the real data to 0.67 in the GAIN-imputed dataset. This decrease suggests that the initial correlation of 0.79 may have been overestimated since it was based only on complete data. The imputed coefficient of 0.67 gives a more cautious estimate of the real relationship and confirms that CS is the main predictor of TS. Also, the association between ASP and TS dropped from 0.67 in the original dataset to 0.41 in the GAIN-imputed dataset, while the correlation between C and TS fell slightly from 0.51 to 0.39.

All features exhibit correlation coefficients below 0.95, which indicates there is no perfect linear relationship between them. This presents challenges for empirical fitting approaches, which typically struggle to account for intricate, non-linear factor interactions. AI-driven algorithms demonstrate strong predictive capacity by effectively capturing and analyzing nonlinear interactions among model variables [8].

3.5 Variance inflation factor

Multicollinearity inflates coefficient variance, reducing the stability and clarity of regression models. Variance Inflation Factor (VIF) is computed to assess multicollinearity among predictors and examine inter-variable correlations. It is noted that VIF values between 0 and 10 indicate the absence of problematic collinearity among variables [36]. Fig. 6 (a) and (b) illustrate that an initial VIF analysis was done separately on the original dataset and the GAIN-imputed dataset, using all input variables. The Initial VIF results showed severe and exact multicollinearity in both datasets. The VIF values for WCS, LCS, and DCS were especially high, reaching 50. In this context, a VIF of 50 serves as a computational placeholder representing a theoretically infinite level of multicollinearity. An iterative feature selection approach was adopted to address the critical issue, and WCS, DCS, and WTS were removed from the dataset. Fig. 6 (c) and (d) demonstrate a successful remediation. The elimination of these dimensions successfully reduced the VIF values to below the critical threshold of 10 across all remaining predictors in both datasets, satisfying the multicollinearity criterion. The evaluation of both the initial and revised VIF values within the GAIN imputation framework demonstrated a significant methodological advantage. Specifically, the VIF scores associated with the features were consistently reduced in the imputed dataset relative to those observed in the original dataset.

3.6 Physics-informed XGBoost

Section 3.6 describes the Physics-Informed XGBoost framework (PI-XGBoost) used to predict the 28-day compressive and tensile strengths of pozzolanic concrete. The key idea is to train an XGBoost regressor under a modified objective that combines data fidelity with soft physics-consistency penalties. The physics terms act as regularizers: they guide the learned mapping to remain consistent with established material trends (porosity–strength), specimen size effects, and tensile–compressive coupling, while still allowing the model to follow the data when the empirical evidence is strong, which is represented in Fig. 7.

Let x_i denote the input vector of the i-th specimen and y_i denote the measured strength. The XGBoost model output is given by Eq. (17), where $f(x)$ is an additive ensemble of regression trees. It is worth mentioning that the loss function for both compressive and tensile strength models is the same, which is used just for the tensile strength.

$$\hat{y}_i = f\left(x_i\right) \tag{17}$$

3.6.1 Objective function with physics regularization

Standard XGBoost regression minimizes the squared error between predictions and measurements. In PI-XGBoost,

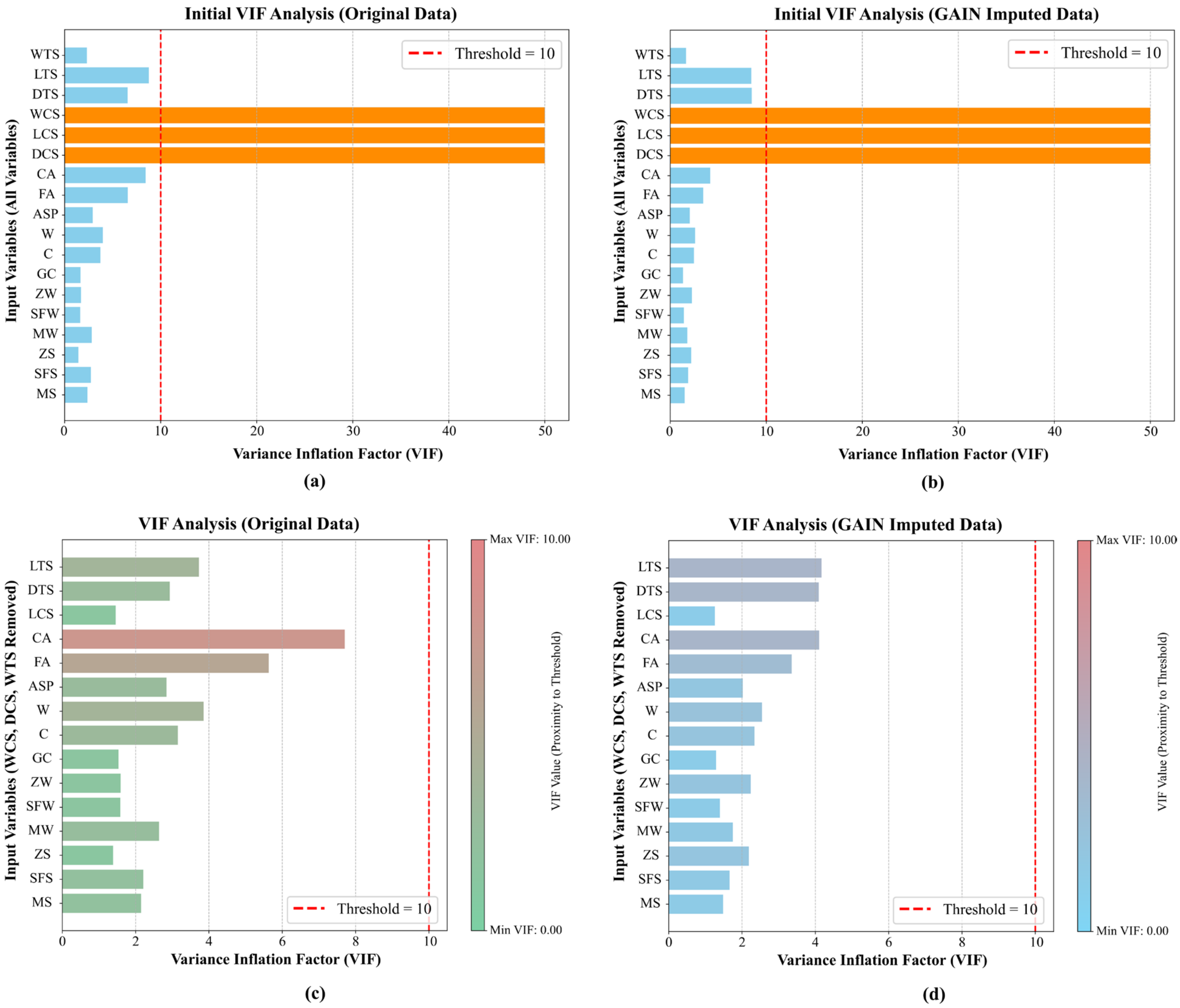

Fig. 6 VIF assessment of input variables: (a) initial variables (original dataset); (b) initial variables (GAIN imputed dataset); (c) revised variables (original dataset); (d) revised variables (GAIN imputed dataset)

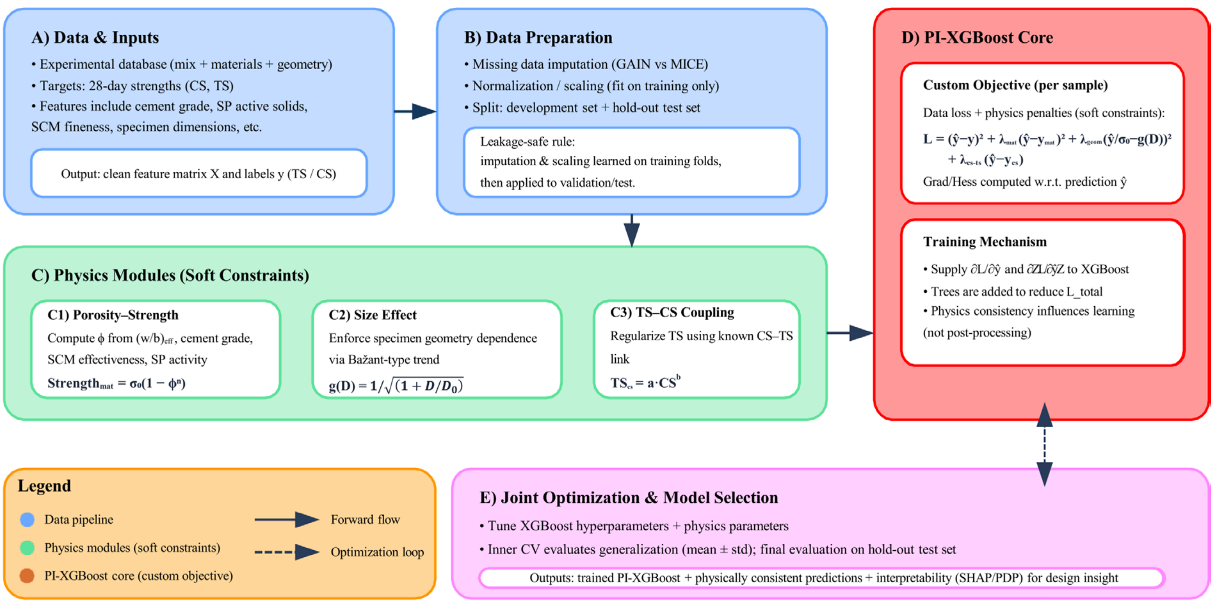

Fig. 7 Physics-Informed XGBoost training workflow

we augment this data loss with physics-based penalties. For each sample i, the total loss is defined as:

$$L_i(\hat{y}_i) = (\hat{y}_i - y_i)^2 + \lambda_{mat}(\hat{y}_i - y_i^{mat})^2 + \lambda_{geom}\left(\frac{\hat{y}_i}{\sigma_0} - g(D_i)\right)^2 + \lambda_{\text{CS-TS}}(\hat{y}_i - y_i^{\text{CS}})^2, \tag{18}$$

where λ_{mat}, λ_{geom}, $\lambda_{\text{CS-TS}}$ are non-negative weights controlling the influence of each constraint; y_i^{mat} is a porosity-based tensile estimate; $g(D_i)$ is a size-effect function based on specimen dimension D_i; and y_i^{CS} is a tensile estimate derived from a tensile–compressive correlation.

The total objective minimized by the booster is:

$$L = \sum_{i=1}^{N} L_i(y_i). \tag{19}$$

This formulation is physics-regularized: the constraints guide the learning but do not force perfect satisfaction; the model can deviate from the physics terms when data strongly supports it.

3.6.2 Physics components used in the objective

Material/porosity–strength consistency term

An effective binder mass and effective water content are computed to define an effective water-to-binder ratio. The binder effectiveness coefficients (k) account for metakaolin, silica fume, and zeolite contributions, while the superplasticizer term reflects the active solid mass contribution. An effective binder mass is defined as:

$$B_{eff} = C + k_{\text{MW}} \cdot \text{MW} + k_{\text{SFW}} \cdot \text{SFW} + k_{\text{ZW}} \cdot \text{ZW}, \tag{20}$$

and an effective water content is:

$$W_{eff} = W - c_{\text{SP}} \cdot \text{ASP}, \tag{21}$$

leading to an effective ratio:

$$\left(\frac{w}{b}\right)_{eff} = \frac{W_{eff}}{B_{eff}}. \tag{22}$$

A porosity proxy is then defined as a function of $(w/b)_{eff}$ and cement grade (GC). To avoid nonphysical values, ϕ is clipped to a plausible range.

$$\phi = \alpha_\phi \cdot \left(\frac{w}{b}\right)_{eff} + \beta_\phi + \gamma_{gc} \cdot \left(\frac{\text{GC}_{ref} - \text{GC}}{\text{GC}_{ref}}\right) \tag{23}$$

Using a Balshin-type form, a porosity-consistent tensile strength is computed and penalized through $(\hat{y}_i - y_i^{mat})^2$:

$$y_i^{mat} = \sigma_0 \cdot (1 - \phi^n), \tag{24}$$

where n and σ_0 are parameters calibrated during optimization.

Size-effect (geometry) consistency term

To incorporate specimen size dependence, we adopt a Bažant-type size-effect function. The geometry penalty is applied to the normalized prediction:

$$g(D) = \frac{1}{\sqrt{(1 + D/D_0)}}, \tag{25}$$

where D_i is the specimen characteristic dimension and D_0 is the size-effect transitional parameter. The constraint is imposed on the normalized prediction $\hat{y}_i / \sigma_0$:

$$\left(\frac{\hat{y}_i}{\sigma_0} - g(D_i)\right)^2. \tag{26}$$

This term discourages geometry-blind predictions when the dataset contains mixed specimen sizes.

Tensile–compressive coupling term

A power-law tensile–compressive relationship is included as a soft regularizer:

$$y_i^{\text{CS}} = a \cdot \text{CS}_i^b, \tag{27}$$

where CS_i is the measured 28-day compressive strength available in the dataset, and a, b are calibrated coefficients. Importantly, CS_i is not used as an input feature of $f(x)$. Instead, it is used only to regularize the tensile prediction during training so that $\widehat{\text{TS}}$ remains consistent with experimentally observed TS–CS relationships. During inference on new mixes, PI-XGBoost predicts TS from $\boldsymbol{x}$ alone and does not require compressive strength as an input.

3.6.3 Custom implementation of XGBoost objective with gradient and Hessian

XGBoost supports custom objectives by requiring, for each training sample, the first- and second-order derivatives of the loss with respect to the prediction $\hat{y}_i$. Since all penalty terms are quadratic in $\hat{y}_i$, closed-form expressions exist.

Define:

$$y_i^{geom} = \sigma_0 g(D_i). \tag{28}$$

Then the gradient is:

$$\frac{\partial L_i}{\partial \hat{y}_i} = 2(\hat{y}_i - y_i) + 2\lambda_{mat}(\hat{y}_i - y_i^{mat}) + \frac{2\lambda_{geom}}{\sigma_0^2}(\hat{y}_i - y_i^{geom}) + 2\lambda_{\text{CS-TS}}(\hat{y}_i - y_i^{\text{CS}}), \tag{29}$$

and the Hessian is:

$$\frac{\partial^2 L_i}{\partial \hat{y}_i^2} = 2\left(1\left|\lambda_{mat}\right|\lambda_{\text{CS-TS}}\right) + \frac{2\lambda_{geom}}{\sigma_0^2}. \tag{30}$$

These derivatives are supplied to the XGBoost training routine at each boosting iteration. Consequently, each tree is constructed to reduce the combined data-plus-physics loss, embedding physics consistency into the learned mapping.

3.6.4 Training workflow and parameterization

The PI-XGBoost model includes standard XGBoost hyperparameters and physics parameters as presented in Table 4. These parameters are optimized using the chosen metaheuristic optimizer (IHGO) under the predefined train/validation protocol. The final PI-XGBoost model is retrained using the selected parameters on the full development set and evaluated once on the hold-out test set.

By incorporating porosity-driven strength behavior, specimen size dependence, and tensile–compressive coupling directly into the boosting objective, PI-XGBoost reduces physically implausible predictions and improves interpretability. Unlike post-hoc filtering, physics consistency influences tree construction during training through the custom gradient/Hessian, enabling the model to exploit data patterns while respecting established mechanics trends under mixed materials and specimen geometries.

3.7 Bivariate probabilistic modeling

The correlation between pozzolanic concrete components and their mechanical behavior exhibits a fundamentally non-linear and probabilistic nature. To adequately represent this complexity, we applied Gaussian Kernel Density Estimation (KDE), a non-parametric approach for estimating the joint probability density function (PDF) associated with the input–output pairs [37]. Fig. 8 presents the bivariate probability distributions characterizing the relationships between input parameters and output measures. Analysis of the KDE surface topology reveals the dependence of compressive and tensile strengths on variations in input features.

The curvature of the KDE surfaces corresponding to MS, SFS, ZS, MW, SFW, and ZW inputs in relation to CS and TS reveals the threshold of pozzolanic efficiency.

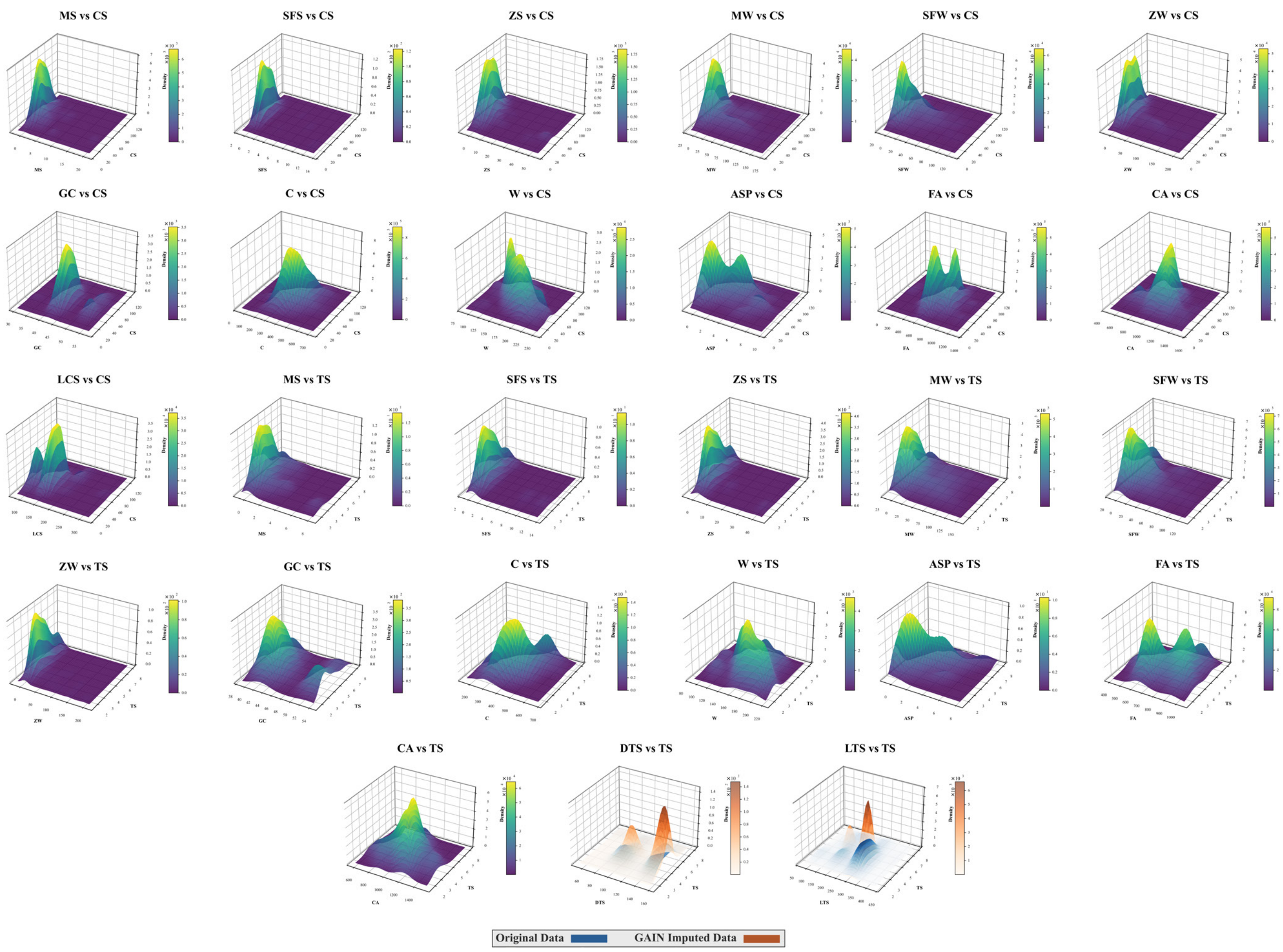

Fig. 8 Bivariate probability distributions of the input features against the outputs

Elevated peaks at lower dosage levels accelerated strength development, whereas surface flattening at higher dosages reflects diminishing returns. CS and TS both increase as C increases. When C is between about 200 and 400 kg/m^3, CS values are around 50 to 80, and TS values are 4 to 7 MPa. The KDE surface for FA shows two main clusters, with high-density regions at about 550 to 650 and 800 to 950 kg/m^3. Each of these clusters is linked to CS values of 60 to 90 MPa. The size of the specimen also matters; CS decreases as LCS grows from approximately 200 to 350 mm, and the KDE surface shows that the highest strength is found in smaller LCS. A dual-surface overlay was used to compare the original experimental data with the GAIN-imputed results. The relationship between DTS and LTS in relation to TS exhibits a comparably rigorous distribution, with standard diameters of 100 and 150 mm strongly associated with tensile strengths of 4 to 6 MPa.

4 Results and discussions

All the models were developed and trained in MATLAB-Python Integration (MATLAB R2024a [38] and Python 3.10.11 [39]) running at Operating System: Windows 10 Pro, version 22H2; Processor (CPU): 11$^{\text{th}}$ Gen Intel(R) Core (TM) i7-1165G7 @ 2.80GHz, 2.80 GHz; Memory (RAM): Total installed memory 16GB DDR4; Storage: 1TB NVMe; Graphics Card (GPU): NVIDIA GeForce MX450.

4.1 Hyperparameter optimization

The integration of machine learning and metaheuristics has been a practical and reliable solution in real-world problems [40, 41]. The research employed the IHGO metaheuristic algorithm to optimize the hyperparameters of the physics-informed machine learning framework. Optimization through IHGO markedly enhanced the predictive accuracy of the model for pozzolanic concrete, exhibiting distinct sensitivities to its principal parameters. Table 4 reports the search ranges, defined by lower and upper bounds, along with the optimal values of the hyperparameters.

4.2 Error assessment

Fig. 9 illustrates the convergence behavior of the IHGO-optimized models for CS and TS prediction. Fig. 9 depicts the progression of the best fitness value and MSE for training and validation datasets as a function of evaluation iterations. The best fitness curve for the CS model drops quickly at first, then levels off as the number of evaluations nears 1000. Since there are no sudden swings or divergence, the optimization process is numerically stable. The TS model's best fitness curve also improves sharply early on, but its convergence slows down more gradually later. This slower stabilization compared to the CS model suggests the TS model has higher optimization complexity.

Table 4 Optimized hyperparameters of physics-informed ML model

Model	Name	Compressive strength		Tensile strength	
		Range	Optimal	Range	Optimal
XGBoost	MaxDepth	[2, 6]	6	[2, 6]	6
	LearnRate	[0.02, 0.2]	0.1175	[0.02, 0.2]	0.2
	NumEstimator	[50, 200]	200	[50, 200]	200
	SubSample	[0.6, 0.9]	0.6	[0.6, 0.9]	0.6490
Physics-informed	$\lambda_{\text{material}}$	[0 ,3]	0	[0, 3]	0
	λ_{geom}	[0, 5]	0.8401	[0, 5]	5
	$\lambda_{\text{CS–TS}}$	–	–	[0, 3]	0
	σ_0	[93.92, 176.1]	112.5107	[6.0264, 15.0660]	6.0264
	n	[1, 6]	4.7402	[1, 6]	1.0039
	D_0	[27.7295, 693.2398]	107.2099	[26.0969, 652.4235]	145.8576
	α_ϕ	[0.1, 2]	2	[0.1, 2]	0.1000
	β_ϕ	[−0.2, 0.2]	0.2000	[−0.2, 0.2]	0.2000
	γ_{GC}	[−0.5, 0.5]	−0.4831	[−0.5, 0.5]	0.1693
	c_{SP}	[0, 10]	8.8833	[0, 10]	10
	k_{MW}	[0.3, 1.5]	1.2602	[0.3, 1.5]	0.7953
	k_{SFW}	[0.3, 1.5]	0.8783	[0.3, 1.5]	0.3000
	k_{ZW}	[0.3, 1.5]	1.5000	[0.3, 1.5]	0.3000
	a_{inter}	–	–	[0.1, 1]	0.1000
	b_{inter}	–	–	[0.3, 0.8]	0.3000

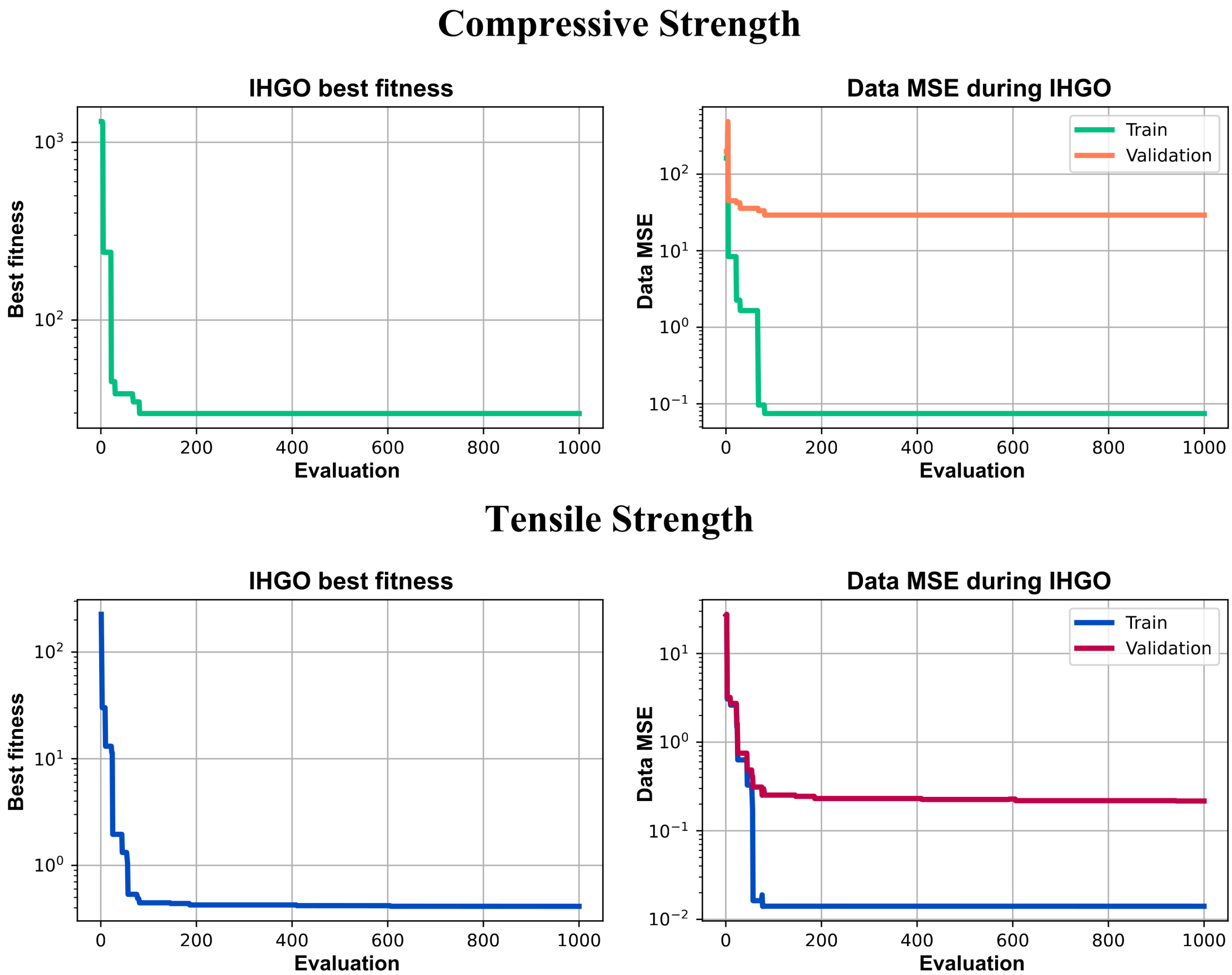

Fig. 9 IHGO Convergence Analysis for CS and TS

Regarding CS, the training set's MSE drops sharply from over 10^2 to under 10^{-1} within the first 100 evaluations. This demonstrates the algorithm's effectiveness in finding the best solution in the complex 15-D search space. The validation curve stabilizes at an MSE value of approximately 30, which is about halfway between 101 and 10^2 on the vertical axis. The training and validation MSE curves for the TS model decrease simultaneously. The training MSE reaches a high-precision plateau around 0.01, indicating that the model has accurately learned the complex relation between inputs and tensile strength. The validation curve stabilizes between 0.1 and 1.

Fig. 10 shows the error distributions for the training, validation, and testing phases of both CS and TS. This enables assessing model bias, dispersion, and robustness. For CS, the probability density of prediction errors is centered and symmetric around zero for both training and validation data. The mean errors (μ) for the training and validation sets are 0.021 MPa and −0.061 MPa, with low standard deviations (σ) of 0.358 and 0.40522. This indicates the model predicts accurately. The model does not consistently overestimate or underestimate compressive strength in the calibration datasets. The testing set has a wider error spread ($\sigma \approx 3.640$), but the mean error is still low at −1.016 MPa, staying close to zero.

In contrast, TS predictions show error distributions that are tightly clustered, with very small mean errors and narrow standard deviations for both training and validation. For the testing set, the mean error is −0.123 MPa and the standard deviation is 0.431. Most errors fall between −0.5 and 0.5, which shows that the model predicts the tensile strength of pozzolanic concrete accurately and with minimal variation.

4.3 Statistical analysis

Fig. 11 exhibits how the physics-informed ML model was evaluated with different metrics during training, validation, and testing. This demonstrates how well the model predicts the mechanical properties of pozzolanic concrete.

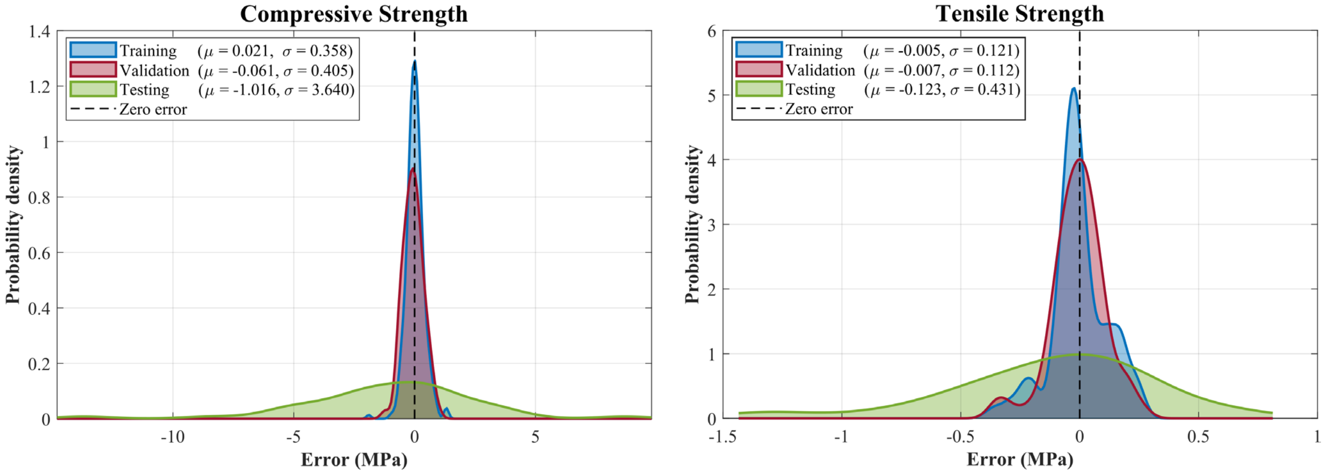

Fig. 10 Probability density functions of prediction errors for CS and TS

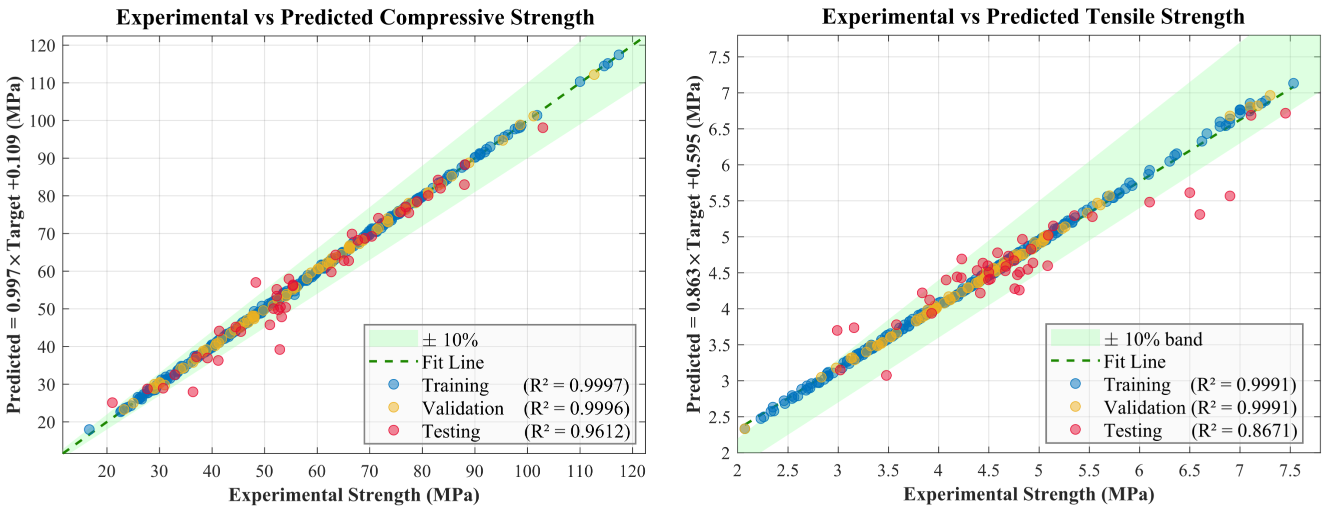

Fig. 11 Regression analysis of the model for CS and TS

This study used a 5-fold cross-validation, splitting 72% of the data for training and 18% for validation, while 10% of the entire dataset was kept as a separate testing set. This hold-out testing set was not used for training and hyperparameter tuning processes, so it provided an unbiased measure of the model's performance on entirely unseen data.

During training, the model predicted strong performance for compressive strength, with an R^2 value of 0.9997. In the validation and testing phases, the R^2 values were 0.9996 and 0.9612, which means the model fit the data well and had low prediction error. A slope of 0.997 and an intercept of 0.109 also demonstrate that the model's predictions were close to the actual values.

For tensile strength, the model had correlation coefficients of 0.9991 in both training and validation, showing strong and consistent accuracy. However, the testing phase yielded a comparatively lower correlation coefficient, measured at 0.8671. The regression analysis yielded a slope of 0.863 and an intercept of 0.595, demonstrating agreement with the actual measurements. Most observations lie within the confidence bounds, with prediction errors consistently constrained to within 10%.

4.4 Partial dependency analysis

Fig. 12 illustrates the model used to assess how input parameters affect compressive and tensile strength. The 95% confidence interval makes the analysis more reliable by showing the uncertainty in the estimated partial dependence.

The analysis found that smaller pozzolanic particle sizes are linked to higher predicted compressive strength in concrete. When MS, SFS, and ZS are smaller or stay fine, the predicted compressive strength rises gradually. If these values go above a certain point, the increase levels off or drops slightly. GC exhibits a nearly linear positive relationship with compressive strength, which points to clinker quality as an important factor. Adding more

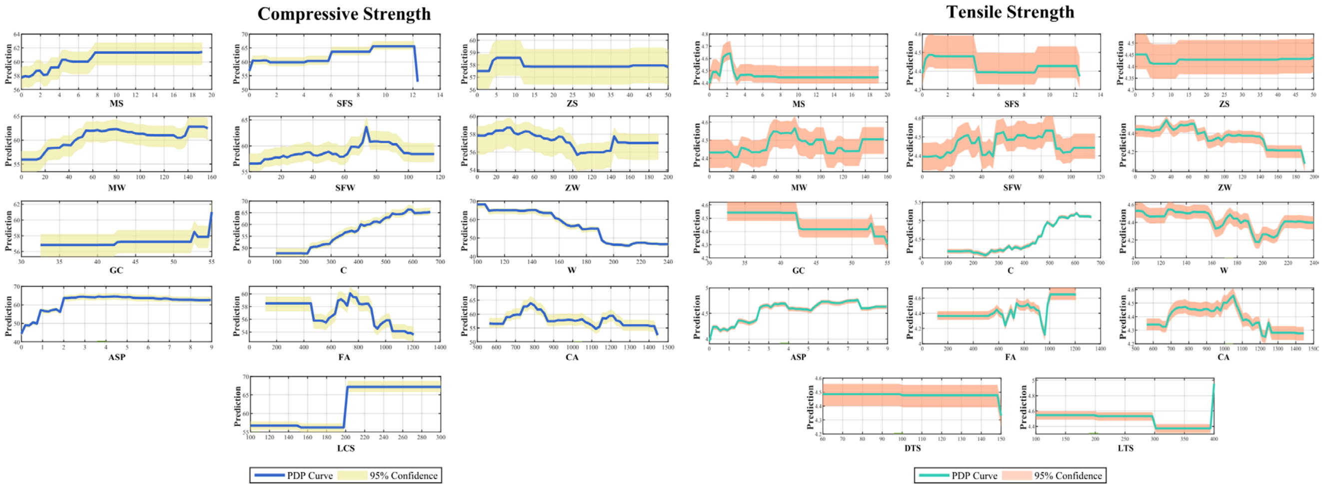

Fig. 12 PDP analysis of the variables for CS and TS

cement increases the cementitious matrix, which improves the bond between aggregates and paste and boosts the concrete's load-bearing performance [42].

The partial dependence plots for tensile strength show different patterns than those for compressive strength, due to the different ways these properties fail. Pozzolanic particle size has a stronger slope on TS than on CS. This greater sensitivity means TS depends more on microcracks, interfacial transition zone refinement, and how fine the pozzolanic particles are. W content has a bigger negative effect on TS. It reveals a steeper slope in TS compared with CS, showing that tensile strength is more affected by porosity. ASP has a stronger positive effect on TS than on CS; the superplasticizer helps by spreading out particles and reducing trapped air [43].

In contrast, water content has a negative relationship with strength, so higher water content leads to lower mechanical properties. The graphs show that the width of the 95% confidence intervals varies across the different subplots. Wider intervals appear where there is less data, which means more data is needed in those areas. Narrower intervals signify that the estimated relationship between the feature and the target variable is more reliable.

4.5 Variable importance of pozzolanic concrete

Fig. 13 presents the relative importance of the 13 and 14 input features in relation to CS and TS, respectively. The results indicate that sample size, ASP, and W have the strongest influence, ranking highest for both CS and TS. The model also gives high importance to Specimen Length, which means it recognizes how the size of the specimen affects strength results. The gain linked to ASP highlights the key role of chemical admixtures in reaching high concrete density. In addition, the model values the size of pozzolanic materials more than their weight, suggesting it focuses on the 'filler effect' and particle fineness rather than just the amount added.

4.6 SHAP analysis

Fig. 14 presents SHAP values, illustrating that each input variable affects the model's predictions across all features. The y-axis lists features by their importance, with red

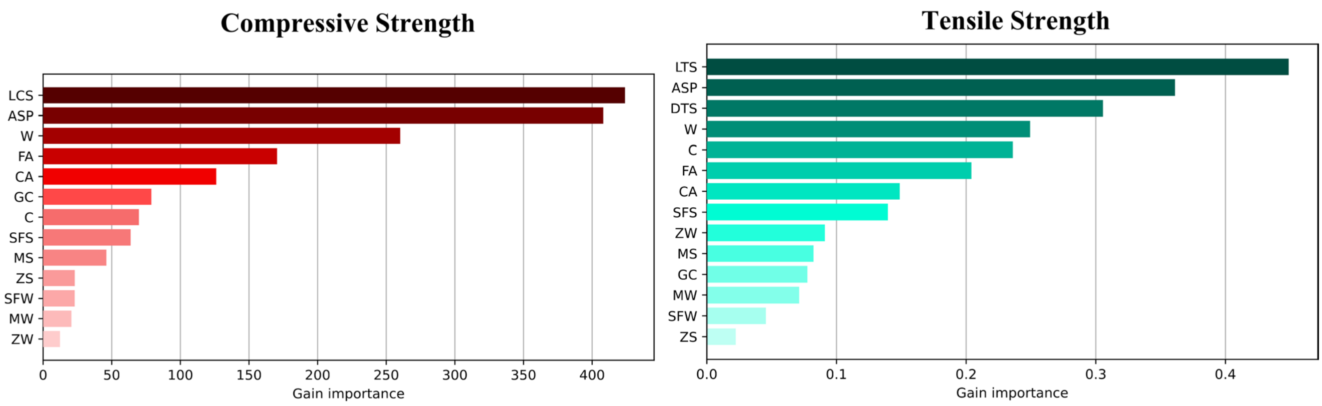

Fig. 13 Relative importance of input variables for CS and TS

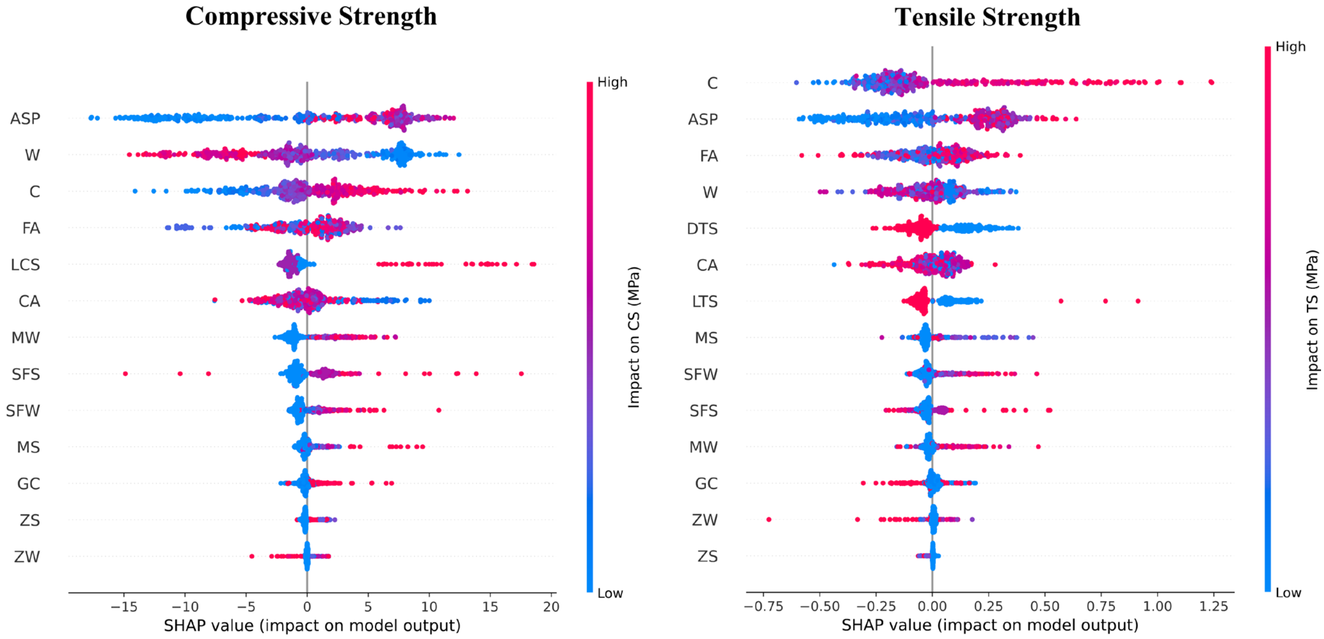

Fig. 14 SHAP value analysis of feature impacts on CS and TS

showing higher values and blue indicating lower values. The x-axis shows SHAP values, which measure each feature's positive or negative contribution to the prediction, compared to a baseline [44].

Fig. 14 illustrates the complex connections between inputs and predicted compressive strength. Metakaolin and silica fume, in terms of both weight and particle size, exert a beneficial influence on CS, enhance packing density, promote pozzolanic reactions, and improve resistance to cracking. Using finer pozzolanic particles creates a more refined pore structure and lowers porosity, which are important for increasing compressive strength. ASP and C also have a positive effect, with SHAP values around 12 and 13, respectively. On the other hand, water content is the main negative factor, since too much water increases porosity in the hardened matrix. FA and CA have both positive and negative SHAP values. Aggregates usually improve strength by increasing packing density, but too much aggregate can reduce workability and matrix uniformity, while too little aggregate lowers resistance to shrinkage [9].

The principal factors influencing tensile strength prediction are illustrated. MW, SFS, and SFW indicate that tensile strength exhibits greater sensitivity than compressive strength to microcracking and pore connectivity, which are governed by the pozzolanic particle size distribution and dosage. The pozzolanic reaction of SF and MK reduces voids and pores through extra hydration. This increase in calcium silicate (C–S–H) gel makes the concrete surface more compact and less porous [45].

5 Conclusions

This work developed novel physics-informed machine learning models whose hyperparameters and coefficients were tuned via a metaheuristic algorithm named IHGO to predict the 28-day mechanical properties of concrete that uses pozzolanic materials as a partial replacement for cement. The dataset was obtained from experiments in which cement was replaced with Metakaolin, Silica Fume, and Zeolite. The method combines physics-informed machine learning with metaheuristic optimization. The main findings are as follows:

- The Generative Adversarial Imputation Network (GAIN) showed strong accuracy, even with much missing data. When 10% of the data was missing, the R^2 value was 0.9661. This dropped to 0.9552 when 20% was missing, and was still 0.9438 even with 90% of the dataset gone.
- Removing redundant features solved the problem of severe multicollinearity and brought all VIF values to acceptable levels.
- Pearson's correlation coefficient indicates that the GAIN model preserved the original linear relationships and provided more realistic, careful correlation estimates. Since the correlations were not perfect ($r < 0.95$) and the factors interacted in complex ways, the data structure needs AI-driven algorithms for accurate, non-linear predictions.
- The IHGO algorithm could efficiently tune hyperparameters and coefficients of the PIML models, leading to accurate and reliable models.

- The symmetric error distributions across all phases demonstrate high predictive accuracy for both compressive and tensile strength models. Although the testing sets have more dispersion, the low average errors and small standard deviations show that the models are reliable for predicting pozzolanic concrete properties.
- The PIML-IHGO model showed strong predictive performance for both compressive and tensile strength. In training and validation, it reached R^2 values close to 0.999. During testing, the model stayed reliable, with R^2 values of 0.96 for CS and 0.87 for TS. These results show that the model is stable and highly reliable for characterizing pozzolanic concrete.
- In comparison with existing machine learning models reported in the literature (Table 1), the proposed PIML-IHGO framework demonstrates competitive or superior predictive performance. For compressive strength prediction, Khan et al. [9] evaluated six machine learning models and reported R^2 values ranging from approximately 0.893 to 0.953 in testing. Omotayo et al. [14] evaluated six algorithms and achieved a training and testing R^2 of 0.963 and 0.935, respectively, using CatBoost as the best of the predictions. Nafees et al. [17] employed Decision Tree and SVM with AdaBoost ensemble techniques and reported overall R^2 values of 0.94 and 0.89, respectively. These results demonstrate that the proposed framework offers a robust, interpretable, and physically sound approach for designing sustainable pozzolanic concrete.
- The Partial Dependence Plot (PDP) reveals a rise in mechanical properties as the pozzolanic material weight (MW, SFW, ZW) increases, and Smaller pozzolanic particles had a stronger impact than larger ones.
- SHAP-based feature importance analysis revealed that metakaolin and silica fume had a stronger positive effect in comparison to zeolite in both compressive strength and tensile strength.

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.

Abbreviations

ASP	Active Solid mass of Superplasticizer (kg/m^3)
C	Cement Content (kg/m^3)
CA	Coarse Aggregate (kg/m^3)
CS	Compressive Strength (MPa)
C–S–H	Calcium-Silicate-Hydrate
D	Discriminator
DCS	Specimen Depth for Compressive Strength (mm)
D_i	Characteristic Structural Dimension
DTS	Specimen Diameter for Tensile Strength (mm)
FA	Fine Aggregate (kg/m^3)
f_{CS}	Compressive Strength of Pozzolanic Concrete
f_{TS}	Tensile Strength of Pozzolanic Concrete
G	Generator
GAIN	Generative Adversarial Imputation Network
GC	Grade of Cement (MPa)
GO	Growth Optimizer
H	Hint vector
IAOA	Improved Arithmetic Optimization Algorithm
IHGO	Improved Hybrid Growth Optimizer
KDE	Gaussian Kernel Density Estimation
LCS	Specimen Length for Compressive Strength (mm)
$L_{\mathrm{CS–TS}}$	Inter-Property Constraint
L_{data}	Data-Driven XGBoost Loss
L_{geometry}	Geometry-Based Physics Constraint
L_{material}	Material-Based Physics Constraint
L_{total}	Total Loss Function
LTS	Specimen Length for tensile strength (mm)
MICE	Multivariate Imputation by Chained Equations
ML	Machine Learning
MS	Metakaolin Size (μm)
MSE	Mean Squared Error
MW	Metakaolin Weight (kg/m^3)
n	Balshin Exponent
ϕ	Effective Pore Volume Fraction
OPC	Ordinary Portland Cement
PDP	Partial Dependence Plot
PIML	Physics-Informed Machine Learning
R^2	Coefficient of determination
SCMs	Supplementary Cementitious Materials
SD (σ)	Standard Deviation
SFS	Silica Fume Size (μm)
SFW	Silica Fume Weight (kg/m^3)
SHAP	Shapley Additive Explanations
TS	Tensile Strength (MPa)
VIF	Variance Inflation Factor

W	Water Content (kg/m^3)
WCS	Specimen Width for Compressive Strength (mm)
WTS	Specimen Width for Tensile Strength (mm)
XGBoost	Extreme Gradient Boosting
ZS	Zeolite Size (µm)
ZW	Zeolite Weight (kg/m^3)
μ	Mean Errors
σ_0	Strength at Zero Porosity

References

[1] Singh, R. P., Vanapalli, K. R., Cheela, V. R. S., Peddireddy, S. R., Sharma, H. B., Mohanty, B. "Fly ash, GGBS, and silica fume based geopolymer concrete with recycled aggregates: Properties and environmental impacts", Construction and Building Materials, 378, 131168, 2023. https://doi.org/10.1016/j.conbuildmat.2023.131168

[2] Jing, H., Li, M., Zhang, Y., Gao, M. "Hydration kinetics, microstructure and physicochemical performance of metakaolin-blended cementitious composites", Construction and Building Materials, 408, 133756, 2023. https://doi.org/10.1016/j.conbuildmat.2023.133756

[3] Türk, E., Karataş, M., Dener, M. "Rheological, mechanical and durability properties of self-compacting mortars containing basalt powder and silica fume", Construction and Building Materials, 356, 129229, 2022. https://doi.org/10.1016/j.conbuildmat.2022.129229

[4] Nguyen, T.-T., Dang, V.-H. "Experimental and Probabilistic Investigations of the Effect of Fly Ash Dosage on Concrete Compressive Strength and Stress-strain Relationship", Periodica Polytechnica Civil Engineering, 66(4), pp. 1098–1113, 2022. https://doi.org/10.3311/PPci.20607

[5] 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, in press. (Accepted for publication: November 2023) https://doi.org/10.24200/sci.2023.60833.7013

[6] 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

[7] Elahi, M. M. A., Shearer, C. R., Naser Rashid Reza, A., Saha, A. K., Khan, M. N. N., Hossain, M. M., Sarker, P. K. "Improving the sulfate attack resistance of concrete by using supplementary cementitious materials (SCMs): A review", Construction and Building Materials, 281, 122628, 2021. https://doi.org/10.1016/j.conbuildmat.2021.122628

[8] Al-Abdaly, N. M., Seno, M. E., Thwaini, M. A., Imran, H., Ostrowski, K. A., Furtak, K. "Advanced Ensemble Machine-Learning Models for Predicting Splitting Tensile Strength in Silica Fume-Modified Concrete, Buildings, 14(12), 4054, 2024. https://doi.org/10.3390/buildings14124054

[9] Khan, A. Q., Muhammad, S. G., Raza, A., Chaimahawan, P., Pimanmas, A. "Advanced machine learning techniques for predicting compressive strength of ultra-high performance concrete", Frontiers of Structural and Civil Engineering, 19(4), pp. 503–523, 2025. https://doi.org/10.1007/s11709-025-1169-4

[10] 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

[11] Baldo, N., Miani, M., Rondinella, F., Manthos, E., Valentin, J. "Road Pavement Asphalt Concretes for Thin Wearing Layers: A Machine Learning Approach towards Stiffness Modulus and Volumetric Properties Prediction", Periodica Polytechnica Civil Engineering, 66(4), pp. 1087–1097, 2022. https://doi.org/10.3311/PPci.19996

[12] 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

[13] Hamid, F. L., Hassan, B. R., Askari, D. F. A., Ahmad, D. A., Shkwr, S. S., Ahmad, S. A. "Predicting Bond Strength Between Normal Concrete with Pozzolanic Materials and Steel Bars Using Machine Learning Techniques", Iranian Journal of Science and Technology, Transactions of Civil Engineering, 50(3), pp. 2561–2577, 2026. https://doi.org/10.1007/s40996-025-01964-6

[14] Omotayo, O. O., Arum, C., Ikumapayi, C. M. "Assessment of Machine Learning Methods for Concrete Compressive Strength Prediction", Journal of Soft Computing in Civil Engineering, 8(4), pp. 116–140, 2024. https://doi.org/10.22115/scce.2024.398515.1648

[15] Taffese, W. Z., Wally, G. B., Magalhães, F. C., Espinosa-Leal, L. "Concrete aging factor prediction using machine learning", Materials Today Communications, 40, 109527, 2024. https://doi.org/10.1016/j.mtcomm.2024.109527

[16] Hosseinzadeh, M., Mousavi, S. S., Hosseinzadeh, A., Dehestani, M. "An efficient machine learning approach for predicting concrete chloride resistance using a comprehensive dataset", Scientific Reports, 13(1), 15024, 2023. https://doi.org/10.1038/s41598-023-42270-3

[17] Nafees, A., Amin, M. N., Khan, K., Nazir, K., Ali, M., Javed, M. F., Aslam, F., Musarat, M. A., Vatin, N. I. "Modeling of Mechanical Properties of Silica Fume-Based Green Concrete Using Machine Learning Techniques", Polymers, 14(1), 30, 2022. https://doi.org/10.3390/polym14010030

[18] Lavercombe, A., Huang, X., Kaewunruen, S. "Machine Learning Application to Eco-Friendly Concrete Design for Decarbonisation", Sustainability, 13(24), 13663, 2021. https://doi.org/10.3390/su132413663

[19] Rahman, M. A., Lu, Y. "EcoBlendNet: A physics-informed neural network for optimizing supplementary material replacement to reduce the carbon footprint during cement hydration", Journal of Cleaner Production, 464, 142777, 2024. https://doi.org/10.1016/j.jclepro.2024.142777

[20] ACI Committee "ACI CODE 318-08 Building Code Requirements for Structural Concrete and Commentary", American Concrete Institute, Farmington Hills, MI, USA, 2008.

[21] Iqbal, N., Noureldin, M. "Empirical physics-informed neural networks for prediction of concrete strength using nondestructive testing", Asian Journal of Civil Engineering, 27(1), pp. 263–279, 2026. https://doi.org/10.1007/s42107-025-01502-9

[22] Yoon, J., Jordon, J., van der Schaar, M. "GAIN: Missing Data Imputation using Generative Adversarial Nets", In: 35th International Conference on Machine Learning (ICML 2018), Stockholm, Sweden, 2018, pp. 5689–5698. ISBN 9781510867963 [online] Available at: https://proceedings.mlr.press/v80/yoon18a/yoon18a.pdf [Accessed: 01 May 2026]

[23] Gao, S., Wan, C., Zhou, Z., Hou, J., Xie, L., Xue, S. "Enhanced data imputation framework for bridge health monitoring using Wasserstein generative adversarial networks with gradient penalty", Structures, 57, 105277, 2023. https://doi.org/10.1016/j.istruc.2023.105277

[24] Lyngdoh, G. A., Zaki, M., Krishnan, N. M. A., Das, S. "Prediction of concrete strengths enabled by missing data imputation and interpretable machine learning", Cement and Concrete Composites, 128, 104414, 2022. https://doi.org/10.1016/j.cemconcomp.2022.104414

[25] Bal'shin, M. Y. "Relation of mechanical properties of powder metals and their porosity and the ultimate properties of porous metal-ceramic materials", Doklady Akademii Nauk SSSR, 67, pp. 831–834, 1949.

[26] Bažant, Z. P., Yu, Q. "Universal Size Effect Law and Effect of Crack Depth on Quasi-Brittle Structure Strength", Journal of Engineering Mechanics, 135(2), pp. 78–84, 2009. https://doi.org/10.1061/(asce)0733-9399(2009)135:2(78)

[27] Zhao, N., Zhang, H., Xie, P., Chen, X., Wang, X. "Prediction of compressive strength of multiple types of fiber-reinforced concrete based on optimized machine learning models", Engineering Applications of Artificial Intelligence, 152, 110714, 2025. https://doi.org/10.1016/j.engappai.2025.110714

[28] Bypour, M., Mahmoudian, A., Yekrangnia, M., Kioumarsi, M. "Explainable tuned machine learning models for assessing the impact of corrosion on bond strength in concrete", Cleaner Engineering and Technology, 23, 100834, 2024. https://doi.org/10.1016/j.clet.2024.100834

[29] Bergstra, J., Bengio, Y. "Random Search for Hyper-Parameter Optimization", Journal of Machine Learning Research, 13, pp. 281–305, 2012. [online] Available at: https://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf [Accessed: 01 May 2026]

[30] 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

[31] Javed, M. F., Khan, M., Nehdi, M. L., Abuhussain, M. "Application of metaheuristic algorithms for compressive strength prediction of steel fiber reinforced concrete exposed to high temperatures", Materials Today Communications, 39, 108832, 2024. https://doi.org/10.1016/j.mtcomm.2024.108832

[32] Kashem, A., Karim, R., Malo, S. C., Das, P., Datta, S. D., Alharthai, M. "Hybrid data-driven approaches to predicting the compressive strength of ultra-high-performance concrete using SHAP and PDP analyses", Case Studies in Construction Materials, 20, e02991, 2024. https://doi.org/10.1016/j.cscm.2024.e02991

[33] Antonini, A. S., Tanzola, J., Asiain, L., Ferracutti, G. R., Castro, S. M., Bjerg, E. A., Ganuza, M. L. "Machine Learning model interpretability using SHAP values: Application to Igneous Rock Classification task", Applied Computing and Geosciences, 23, 100178, 2024. https://doi.org/10.1016/j.acags.2024.100178

[34] 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

[35] 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

[36] O'brien, R. M. "A Caution Regarding Rules of Thumb for Variance Inflation Factors", Quality & Quantity, 41(5), pp. 673–690, 2007. https://doi.org/10.1007/s11135-006-9018-6

[37] Fuentes, N. A., Flores, J. C., Egger, J. E., Vicencio, F. A., Aguilar, V., Yanez, S. J. "Multivariate joint probability distributions for seismic design parameters across multiple building codes", Bulletin of Earthquake Engineering, 21(14), pp. 6247–6271, 2023. https://doi.org/10.1007/s10518-023-01747-2

[38] The MathWorks, Inc. "MATLAB, (R2024a)", [computer program] Available at: https://www.mathworks.com/ [Accessed: 01 May 2026]

[39] Python Software Foundation "Python (3.10.11)", [computer program] Available at: https://www.python.org/ [Accessed: 01 May 2026]

[40] 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

[41] Guo, Y., Gao, D., Ma, W., Liu, J. "Multi-objective mix optimization of geopolymer recycled aggregate concrete using prior knowledge embedded neural network and evolutionary algorithm", Engineering Applications of Artificial Intelligence, 164, 113254, 2026. https://doi.org/10.1016/j.engappai.2025.113254

[42] Akbarpour, A., Mahdikhani, M. "Effects of natural zeolite and sulfate environment on mechanical properties and permeability of cement–bentonite cutoff wall", European Journal of Environmental and Civil Engineering, 27(3), pp. 1165–1178, 2023. https://doi.org/10.1080/19648189.2022.2075940

[43] Wang, X.-H., Fang, Z.-C., Zheng, L. "Effect of dose and types of the water reducing admixtures and superplasticizers on concrete strength and durability behaviour: a review", Journal of Civil Engineering and Management, 30(1), pp. 33–48, 2024. https://doi.org/10.3846/jcem.2024.20145

[44] Su, N., Guo, S., Shi, C., Zhu, D. "Predictions of mechanical properties of Fiber Reinforced Concrete using ensemble learning models", Journal of Building Engineering, 98, 110990, 2024. https://doi.org/10.1016/j.jobe.2024.110990

[45] Khan, S., Parveez, M., Tantray, M. A. "Metakaolin-infused concrete for sustainable and durable building rehabilitation", Journal of Building Pathology and Rehabilitation, 10(1), 66, 2025. https://doi.org/10.1007/s41024-025-00578-9