CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS
Volume: 07 Issue: 03 | July 2026 ISSN: 2660-5309

Article

An Interpretable Optimized Artificial Neural Network Framework for Concrete Compressive Strength Prediction

Taisir Mohsin*[1], Duaa Muhsin[2], Waffaa Mohammed[3]

1. Department of Medical Physics, University of Al Shatrah ,Iraq
2. Department of Civil Techniques, Southern Technical University , Iraq
3. Department of Anatomy and Histology, .University of Al Shatrah ,Iraq

* Correspondence: taisirmohsin@shu.edu.iq, duaa.dookhi@stu.edu.iq, waffaa@shu.edu.iq

Abstract: Predicting concrete compressive strength is paramount to the structural safety, quality control and sustainable mix design processes. Although artificial neural networks (ANNs) are powerful nonlinear modeling capabilities , their performance is sensitive to data quality, and hyper-parameter configuration. This paper proposes a strong and full framework for automated machine learning that combines rigorous data preprocessing with full Bayesian hyper-parameter optimization. The proposed framework employs interquartile range (IQR)-based outlier removal on all eight input features (cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate, and curing age) and the target compressive strength, before removing duplicate mixture designs. After cleaning, the dataset is normalized to [−1, 1] and split into training, validation and test sets. A feedforward neural network is then searched over a wide hyper-parameter space, including network depth, neurons per layer, learning rate, training algorithm and activation function, with validation mean squared error as the objective .The optimized model achieves strong predictive performance on an independent test set : R=0.939 ,RMSE=5.304 MPa ,MAE=4.167 MPa and MAPE=14.832. Critically, the optimization process selected logsig as the best activation function, which is appropriate due to the positive, saturating nature of concrete strength growth. The study simultaneously performs statistical data cleansing and tuning of the model hyper-parameters to provide a clear and robust AI-based system for predicting the concrete compressive strength in smart construction projects.

Keywords: Artificial Neural Network, Bayesian Optimization, Concrete Strength, Sensitivity Analysis.

Citation: Mohsin T., Muhsin D., Mohammed W. An Interpretable Optimized Artificial Neural Network Framework for Concrete Compressive Strength Prediction. Central Asian Journal of Mathematical Theory and Computer Sciences 2026, 7(3), 85-98.

Received: 10th Mar 2026
Revised: 11th Apr 2026
Accepted: 20th May 2026
Published: 19th Jun 2026

1. Introduction

Concrete is the most consumed construction material in the world, appreciated for its durability, versatility and cost efficiency. One of the most important mechanical properties that is directly related to structural safety, service life, and design specifications is its compressive strength, which is defined as the maximum pressure per unit area that a unit of concrete can sustain before complete failure. In practice, compressive strength is most often established by conducting destructive laboratory tests on concrete cylinders or cubes from the construction site after they have cured [1]. This process is slow, costly, and rarely practicable for on-site quality control during construction. consequently, interest is shifting towards data driven predictive models that can quickly and accurately predict compressive strength from both mixture proportions and curing conditions.

Concrete is generally characterized based on eight variables: cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate, and Age (days). These components interact in highly nonlinear and interdependent manners, which makes analytical or empirical modeling difficult.

In this context, artificial neural networks (ANNs) have been identified as an advantageous tool that can effectively learn complex (and often non-monotonic) relations directly from data without the need for explicit physical equations [2][3]. The standard benchmark dataset for this task consisting of 1,030 experimental records with 8 inputs (previously mentioned) and compressive strength (MPa) as the target was proposed by Yeh (1998) and has become the benchmark dataset of choice in machine learning-related studies on concrete [4].

Choosing a powerful machine learning algorithm alone cannot guarantee correct outcomes. The actual secret lies in correctly tuning the parameters of this algorithm (hyper-parameter tuning). Concrete stren- gth is influenced by complicated interactions between the different components of the mix, and any small errors in setting up the model will lead to incorrect predictions .

Traditional tuning methods (eg random or grid search) are computationally expensive and sometime miss best results so this paper utilizes the technique of 'Bayesian Optimization'[5]**.** This technique is characterized by its intelligent search process; it explores the results of previous searches to get the best settings with the fewest trials. In this way, we make sure that we build a predictive model that is not just highly accurate but also reliable, ready to be used as an application in engineering.

Despite this, the performance of ANNs remain extremely sensitive to data quality and hyper-parameter configuration. Concrete datasets from the real world often include outliers, due to measurement errors, variations in curing conditions or other physically implausible mix designs, in addition to duplicate entr- ies that lead to the over-representation of specific formulations. In this study, the raw dataset of 1,030 samples is thoroughly cleaned of such anomalous defects prior to modeling.

To overcome these limitations, a robust and integrated machine learning framework is proposed in this paper to predict concrete compressive strength which integrates:

• IQR method for Statistical data cleaning (outliers in input features and target output removal).

• Remove duplicates to keep only unique mixture designs.

• Using min-max normalization to ensure numerical stability in the process of training.

• Bayesian optimization is performed to automatically and efficiently tune a wide range of hyper-parameters (network depth, width, learning rate, training algorithm, Epochs and activation functions) while minimizing validation error.

The standard concrete compressive strength dataset was used to implement the methodology in MATLAB [4]. The optimized model shows good predictive performance on an independent test set in terms of RMSE, MAE and the Correlation coefficient (R).

This study helps in the wide usage of AI applications in civil engineering by developing a clear, automated, and statistically valid approach to predicting the concrete strength. The approach not only improves the prediction accuracy but also facilitates virtual mix design tests rapidly by avoiding laboratory mixes, thus promoting sustainable concrete development (saving materials, laboratory costs, and lessen the carbon emissions associated with traditional trial-and-error methods.

2. Related works

Machine learning is considered a powerful tool for modeling the properties of concrete, and compre- ssive strength still receives most of the focus because of its importance for structural design and quality control[3][6] . While early work using artificial neural networks (ANNs) was feasible, much recent research has focused on robustness, automation and reproducibility using advanced optimization and preprocessing methods.

Al-Ghamdi [7] developed an artificial neural network (ANN) model trained on generated synthetic data to predict concrete compressive strength. The model used a structure with 4 inputs, two hidden layers, one output layer and the Levenberg-Marquardt algorithm. The ANN model, trained on 500 computer-generated mixes, achieved acceptable prediction accuracy (R=0.80) on real data, while warning that overusing synthetic data could lead to overfitting.

Thaba [8] developed an optimized artificial neural network (ANN) model to predict concrete compre- ssive strength using 776 samples collected from previous studies. The researcher relied on the Random Search method to tune the parameters, reaching an optimal structure consisting of two hidden layers (352 and 480 neurons, respectively) with tanh and ReLU activation functions, and the Adam optimization algorithm. The model achieved a coefficient of determination R^2=0.87 .

Although Gabriel et al. [9] achieved high prediction accuracy (R^2=1.0) using hybrid models to predict high-strength concrete, their study relied on limited experimental data (326 samples) and manual parameter tuning, with performance evaluated on internal data.

Khan et al.[10] developed an artificial neural network (ANN) model using the Levenberg-Marquardt Backpropagation (LMBP) algorithm to predict the compressive strength of normal and high-strength concrete. The study relied on a large dataset comprising 1,637 samples and 8 input variables. The optimal architecture was determined through sensitivity analysis: a single hidden layer with 15 neurons, with five-fold cross-validation (5-fold CV) applied. The model achieved high accuracy (R = 0.963 (and training stopped at epoch 33 to prevent overfitting.

Naved et al. [11] developed a deep neural network (DNN) model to predict concrete strength, and their data processing was limited to normalization only without removing outliers. To tune the parameters, they used the ready-made Keras Tuner library for automated search for the best network architecture (5 hidden layers with a distribution of neurons(12, 16, 16, 40, 26), learning rate 0.001), where their final model achieved a coefficient of determination R^2 = 0.89 on the test data.

Xiong et al.[12] proposed a Bayesian-optimized fully connected neural network (BO-FCNN) to predict concrete compressive strength using the UCI dataset. Their approach involved preprocessing data and tuned key hyper parameters, such as the number of neurons and learning parameters, achieving an R^2 of approximately 0.90 MAE = 3.5028 MPa MSE = 21.0745 MPa². However, the optimization was limited to a subset of network parameters.

According to Amit Kumar Sah et al.[13] , four models—an artificial neural network (ANN), multiple linear regression, a support vector machine, and a regression tree were employed and compared for performance using evaluation metrics . The ANN model outperformed the others where MAE=4.443 , RMSE=6.092 , R =0.71 and MAPE=15.112 . Although the study included data preprocessing steps, it did not employ systematic hyper-parameter optimization.

Despite proving the effectiveness of hyper-parameter optimization in enhancing predictive perform- ance , some references ignoring data quality standards and overlook physically unacceptable outlier values (such as unrealistic water/cement ratios), focusing on secondary characteristics instead of funda- mental engine-ering parameters. To fill up these gaps, this study suggests a predictive framework which integrate data cleansing with concrete compressive strength probabilistic optimization. The specific objectives are:

1. Physical consistency& Ensure Data Integrity: Outlier detection based on interquartile range (IQR) was implemented on all eight of the mixture components and the target compressive strength respectively, and duplicate mixture designs were removed in order ensure no bias in training and also be physically reasonable.
2. Systematically Optimize Network Configuration : Use Bayesian optimization to explore a large hyper-parameter space (including the numbers of hidden layers, numbers of neurons, training algorithms, activation functions, learning rates, epoch limits) within realistic engineering limits.
3. Target Primary Engineering Performance: converge directly on compressive strength prediction rather than secondary properties, herewith increasing relevance to structural design, quality assurance, and performance based standards.
4. Ensure Unbiased Model Evaluation: Test predictive performance on a completely unseen test set to provide generalizable and statistically valid assessment of the performance of multiple models.

This work proposes a clear engineering framework that combines a delicate data preprocessing with an efficient probabilistic tuning to improve predictive accuracy and physical reliability for AI-based concrete mix design.

3. Methodology

This section describes the end-to-end pipeline for predicting concrete compressive strength, comprising data acquisition , preprocessing [14][15] , Bayesian hyper-parameter optimization, and model evaluation [16][17] . All steps were implemented in MATLAB R2023a using the Neural Network Toolbox . The complete workflow detailed in Figure.1.

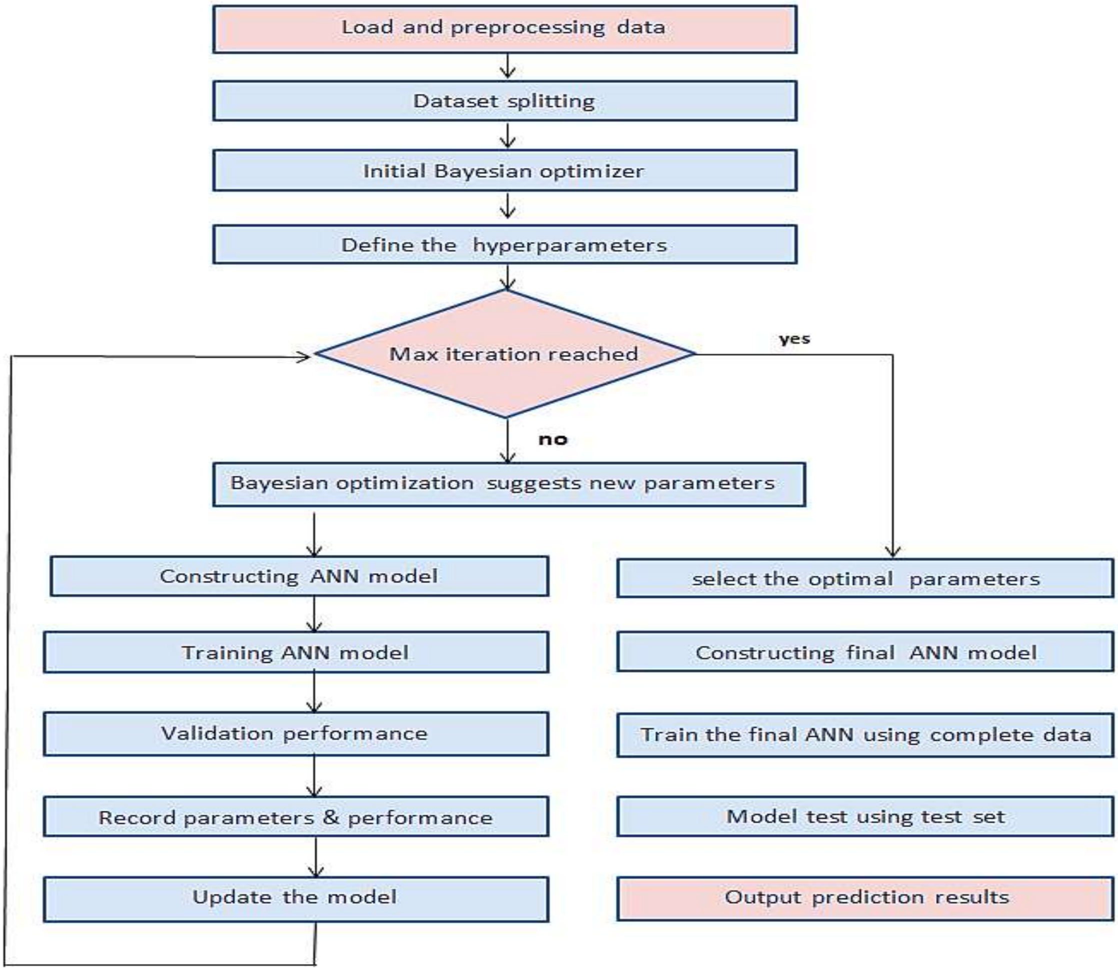


Figure 1. Schematic workflow of the proposed predictive framework.

3.1. Dataset Description

The study uses the standard concrete compressive strength dataset originally introduced by Yeh [4], which contains 1,030 experimental records. Each record includes eight input features:

Table 1 demonstrates the statistical details of selected dataset. Table 1 breaks down these parameters into their different values such as mean, standard deviation, minimum value, maximum value, and occu- rrence of different percentile like 25%, 50%, and 75%. This Table shows the dimensions and scale of selected input parameters and target variable. In this research, the concrete compressive strength (CS) was selected as target variable which depends on different ingredients such as Cement (kg/m³), Blast furnace slag (kg/m³), Fly ash (kg/m³), Water (kg/m³), Superplasticizer (kg/m³), Coarse aggregate (kg/m³), Fine aggregate (kg/m³) and Age (days) ,which were selected as input features. The dataset was download- ded from Kaggle [8] and corresponds to the UCI Machine Learning Repository benchmark.

Table 1. Statistics of the features.

parameters	Mean	Std	Min	25%	50%	75%	Max
cement	281.17	104.51	102	192	272.9	350	540
Blast furnace slag	73.896	86.279	0	0	22	143	359.4
Fly ash	54.188	63.997	0	0	0	118.3	200.1
water	181.57	21.354	121.8	164.9	185	192	247

superplasticizer	6.2047	5.9738	0	0	6.4	10.2	32.2
Coarse aggregate	972.92	77.754	801	932	968	1029.4	1145
Fine aggregate	773.58	80.176	594	730.4	779.5	824	992.6
age	45.662	63.17	1	7	28	56	365
Concrete compressive strength	35.818	16.706	2.33	23.7	34.445	46.2	82.6

3.2. Data Preprocessing

To ensure data integrity, a two-stage cleaning procedure was applied:

1. Outlier Removal: The interquartile range (IQR) method was used to detect and remove outliers in all eight input features and the output variable. For each variable, outliers were defined as values outside the interval [18] [19] :

$$[Q1-1.5\cdot IQR,\ Q3+1.5\cdot IQR]$$

where Q1 and Q3 are the 25th and 75th percentiles, respectively. Any sample containing at least one outlier in the inputs or outputs was excluded.

2. Duplicate Elimination: Identical mixture designs (i.e., duplicate rows in the input space) were identified , only the first occurrence was retained to avoid overrepresentation. After cleaning, the dataset size was reduced from 1,030 to 903 .

3.3. Data Normalization and Partitioning

The cleaned data was normalized to the range [−1, 1] .The normalized dataset was partitioned into 70% training (for model learning), 15% validation (for Bayesian optimization) and 15% testing (for final unbiased evaluation).

3.4. Bayesian Hyper-parameter Optimization

A feedforward neural network was optimized using Bayesian optimization to minimize the mean squared error (MSE) on the validation set. The search space encompassed six key hyper-parameters shown in Table.2 :

Table. 2 Hyper-parameters search space and ranges.

Hyper-parameters	**value**
Neurons per layer	integer values from 1 to 12
Learning rate (lr)	real values from 10−4 to 10−1 , sampled on a logarithmic scale
Number of hidden layers	integer values from 1 or 2
Training algorithm	selection from trainlm (Levenberg-Marquardt), trainbr (Bayesian Regularization), traingdx (Gradient Descent with Momentum and Adaptive Learning Rate) and trainscg (Scaled Conjugate Gradient).
Activation function	selection from tansig (hyperbolic tangent sigmoid), logsig (logistic sigmoid)
Epochs	integer values from 20 to 100

The objective function trained a network with the proposed hyper-parameters and returned the validation MSE. A total of 70 evaluations were performed. The hyper-parameter search space included the learning rate to maintain compatibility with first-order training algorithms. However, this parameter is conditionally relevant: when the Levenberg–Marquardt (trainlm) algorithm is selected, the learning rate is ignored, whereas it remains active and influential for other training algorithm.

3.5. Final Model Training and Evaluation

The best hyper-parameter configuration (lowest validation MSE) was used to retrain the network on the full training set (training and validation data). The final model was evaluated on the held-out test set . In this study, equations (1) to (4) were used to evaluate the performance of artificial neural network (ANN) model :

- Root Mean Squared Error (RMSE) = $\sqrt{\frac{1}{n} \sum_{i=1}^{n}\left(y_i^a - y_i^b\right)^2}$ …………… (1)
- Mean Absolute Error (MAE) = $\frac{1}{n} \sum_{i=1}^{n} | y_i^a - y_i^b |$ ……………(2)
- Mean Squared Error (MAPE) = $\frac{1}{n} \sum_{i=1}^{n} \left|\frac{y_i^a - y_i^b}{y_i^a}\right| * 100\%$ ………… (3)
- Correlation Coefficient (R) = $\frac{\sum (y_i^a - \bar{y}^a)\ (y_i^b - \bar{y}^b)}{\sqrt{\sum(y_i^a - \bar{y}^a)^2\ \sum(y_i^b - \bar{y}^b)^2}}$ ....(4)

In this equations, y_i^a indicates actual values of selected parameters, y_i^aindicates the model predicted values, and ‘n’ denotes the total number of data used in training or testing.

4. Results and Discussion

4.1. Optimization Outcome

Figure (2) shows the development of the parameter optimization using the Bayesian optimization algorithm. A rapid decrease in the objective function is observed during the first ten evaluations from 0.057 to 0.022, followed by relative stability until evaluation 47. At evaluation 48, an additional decrease to around 0.021 occurred, indicating the algorithm's ability to escape local minima and continue searching for better solutions. The gap between the observed values (blue line) and the predicted values (green line) indicates the algorithm's success in exploring regions that yielded better results than expected. The algorithm successfully reduced the error (MSE) to an approximate final value of 0.021 after 70 evaluations.

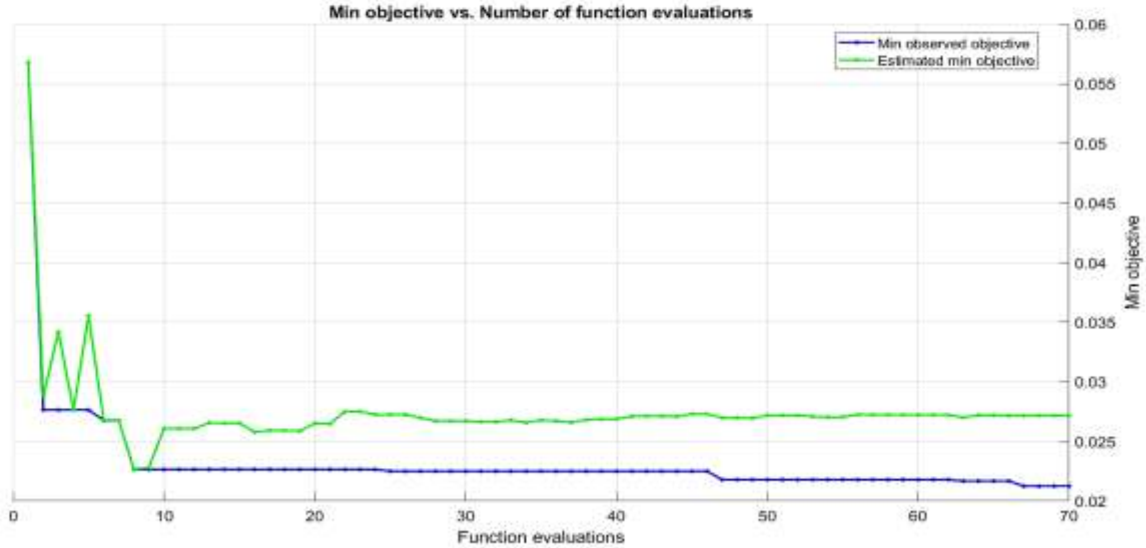


Figure 2. Convergence behavior of Bayesian hyper-parameter optimization.

Bayesian optimization converged within 70 trials to the optimal configuration shown in Table .3. Based on these optimal values, one hidden layer was established between input and output layers with optimal parameters, which are presented in Figure 3.

Table 3. Optimal hyper-parameters for ANN.

Hyper-parameters	value
Neurons per layer	12 neurons
Learning rate (lr)	0.001642
Number of hidden layers	1 hidden layer
Training algorithm	trainlm
Activation function	logsig
Epochs	21
validation MSE	0.021249

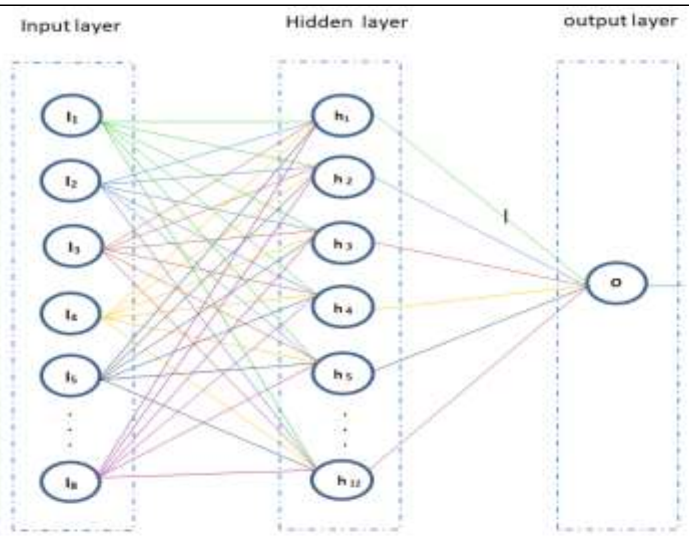


Figure 3. optimal neural network architecture.

4.2. Test Performance

The final model was evaluated on the held-out test set (15% of the cleaned data). Figure 4 illustrates the evolution of the mean squared error (MSE) across the training, validation, and testing phases. The curve shows a rapid and effective decline in error values during the early epochs, where the model reaching its lowest validation performance at the sixth epoch with a value of 0.018639. The stability of the test and validation curves after this point reflects the success of the early stopping mechanism in preventing overfitting and ensuring high generalization ability. These results confirm the efficiency of the raining process and the model's reliability in providing accurate and stable predictions.

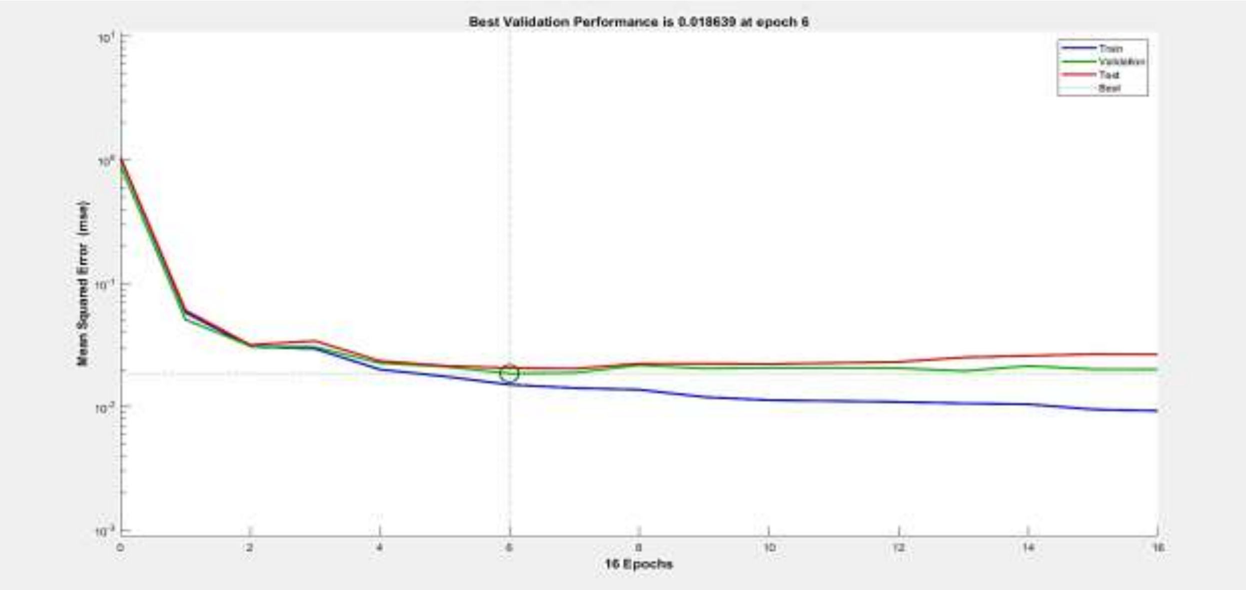
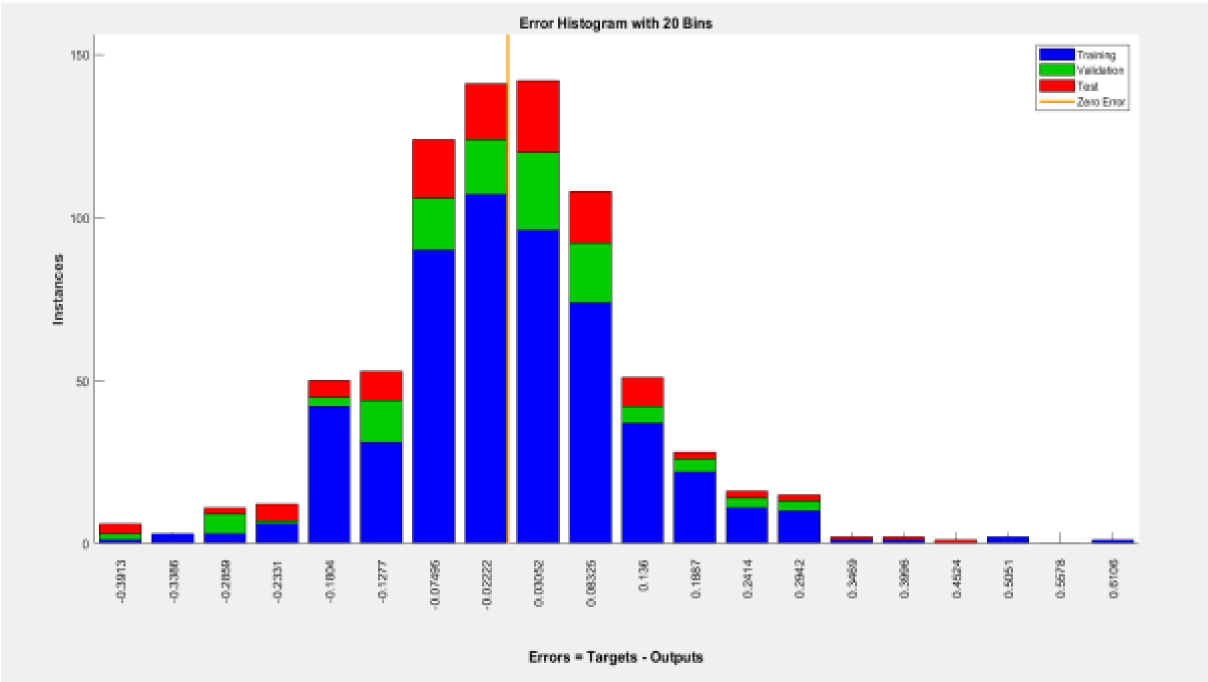
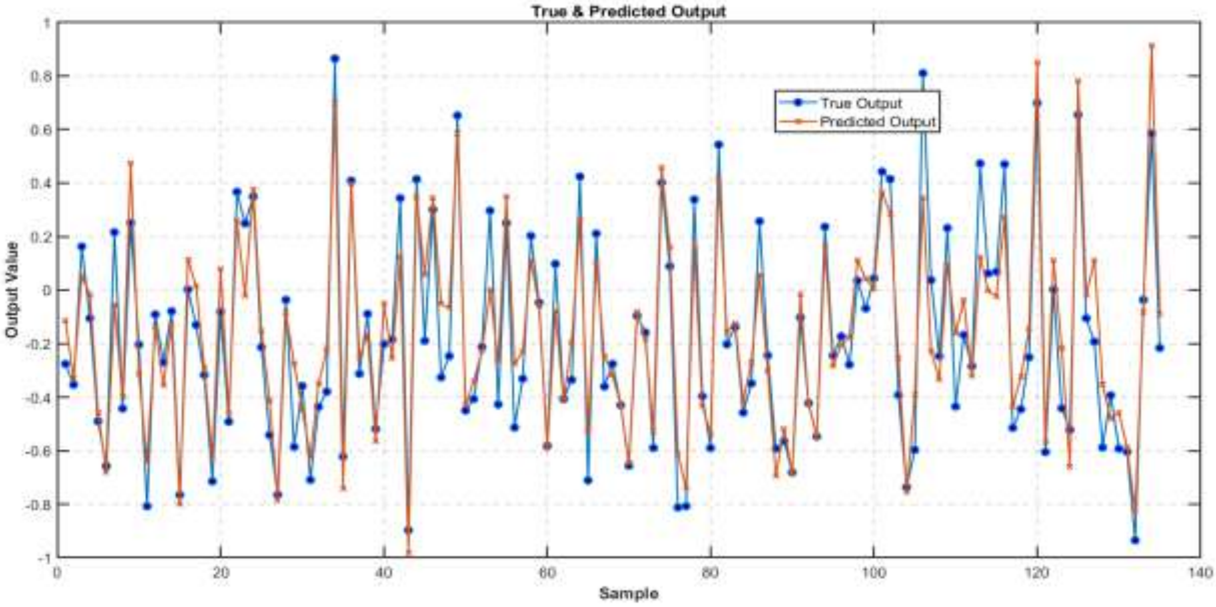


Figure 4. MSE evolution for training, validation and test sets across epochs.



(a) Error histogram for all the steps of ANN



(b) Distribution of actual and predicted concrete compressive strength with errors

Figure 5. Comprehensive model performance assessment.

Figure 5. presents a detailed performance assessment of the model. (a) shows the histogram of the model error distribution divided into 20 bins, where the horizontal axis represents the error values calculated by the equation (Errors = Targets - Outputs), while the vertical axis represents the number of instances. The vertical orange line indicates the zero error point, which represents a perfect match between the predicted and actual values. The distribution follows a pattern close to a normal (Gaussian) distribution, where most values are concentrated around the center and gradually decrease toward the extremes, indicating the random nature of the errors and the absence of systematic bias. Additionally, the vast majority of predictions are concentrated in the bins near the zero line (at values -0.02222 and 0.03052), which indicates high model accuracy as most errors are very small. The distribution also exhibits a relative symmetry between the negative and positive sides, meaning there is no systematic bias towards overestimation or underestimation. The plot shows the distribution of three data sets (Training set in blue, Validation set in green and Test set in red).It is observed that the three groups have same distribution pattern with about the same statistical behavior, which indicates good generalization capacity, no overfitting and good performance for different datasets. Analysis shows that the model is decent for use in practical applications (high accuracy, normal distribution, and generalization ability).

(b) shows the actual values compared to the values expected by the model. The model is able to follow real data trend very well and predicts the values closely .Both of the curves match at the majority of points and show the same pattern of variation. This indicates that the model is valid for predicting data not provided when training.

Figure 6. depicts the accuracy of the model since it compares how close the predicted values were to the actual values. There is a dense distribution of points very close to the ideal match line (Y = T), indic- ating high predictive accuracy. The value of correlation coefficients (R) were 0.956 for training, 0.935 for validation and 0.944 for test. The fact that the values are so close to each other at different steps shows that the model is stable, and it is not overfitting since it is working well with new data. In general, the model predicts with high accuracy, thus giving it usability in practice.

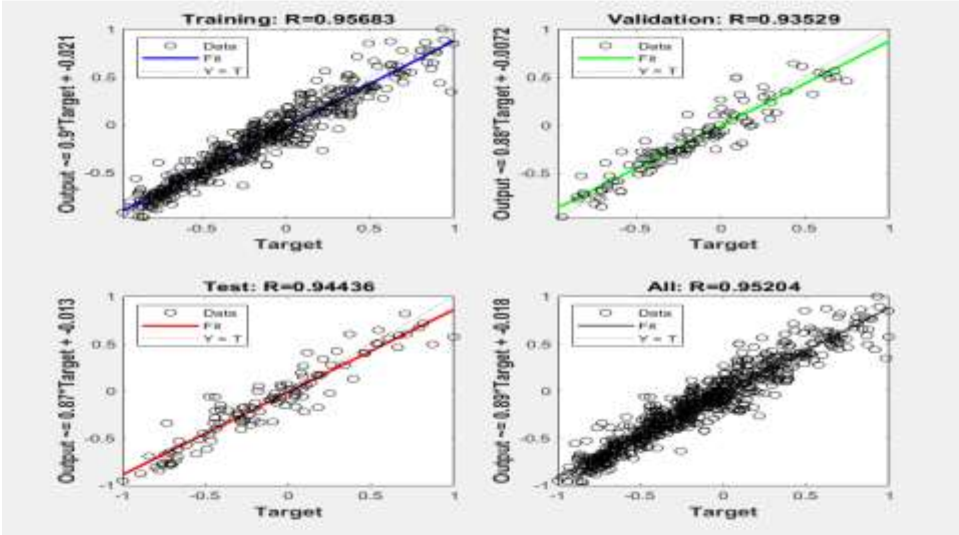


Figure 6. Correlation between actual and predicted values.

Table 4 summarizes the performance metrics in a de-normalized format. The predictive results exhibit good accuracy in the range of compressive strength (10–60 MPa) suitable for most actual engineering applications. The high correlation indicates that the model captures the mixture strength relationship very well. It shows a direct comparison of the proposed model with recent studies. Remarkably, our framew- ork starts with the same complete dataset as in similar studies and applies reliable preprocessing steps such as IQR-based outlier elimination, deduplication, normalization. while this reduces the effective sample size and makes the modeling harder , our model still gives better error metrics and competitive correlation performance as detailed in the table.

Table 4. Comparative performances of proposed model with recent studies.

Study / Model	MAE (MPa)	RMSE (MPa)	R (Pearson)	MAPE (%)
recent studies	4.443	6.092	0.71	15.112
proposed model	3.703	5.304	0.939	14.83

The main reasons for the enhancement noticed in the study are : extensive data preprocessing and systematic tuning of hyper-parameter. Preprocessing, involving removal of outliers with IQR and normalization of all inputs and output, stabilized the training process and reduced the impact of noise or inconsistencies in records. Also hyper-parameter optimization helped train the model efficiently and avoiding overfitting. These methodological choices together justify the results in this work, which demonstrate that to get a more accurate and definitive prediction of compressive strengths, it is important to integrate data preparation with tuning of hyper-parameter.

4.3. Importance of Input Variables

Here we will be working with Permutation Feature Importance type, which is a model-agnostic statistical method that attempts to measure the effect of each variable in a ML model on the prediction accuracy [20]. The basic idea is based on a simple principle: an important variable will cause a marked decline in model performance when its values are randomly shuffled, while a variable that is not important will not significantly affect accuracy when its values are shuffled.The process involves first calculating the original model error, then randomly shuffling the values of each variable while keeping the other variables intact and re-measuring the error. The difference between the new error and the original error (ΔRMSE) represents the importance of that variable [21] . Figure 7. displaying 'Feature Importance from Permutation Analysis,' we notice that the cement variable is the most important with a value of 0.33, indicating that the amount of cement plays the biggest role in determining concrete strength, followed by the fly ash variable (English (United States) US) with an importance value of 0.25, then high furnace slag. Then comes the blast furnace slag with a value of 0.23, while fly ash ranks fourth with a value of 0.10. Other variables such as water and coarse aggregate showed very low importance with a value of 0.03 each, as well as superplasticizer and fine aggregate with a value of 0.04 each, indicating that cement components and cement replacement materials are the decisive factors in determining concrete strength, while the influence of other variables is reduced. This may be due to the limited range of their variation in the data or because their effect is nonlinear. Overall, we notice that only three variables (cement, blast furnace slag, and fly ash) account for about 81% of the total importance, reflecting an unequal distribution of importance among the variables.

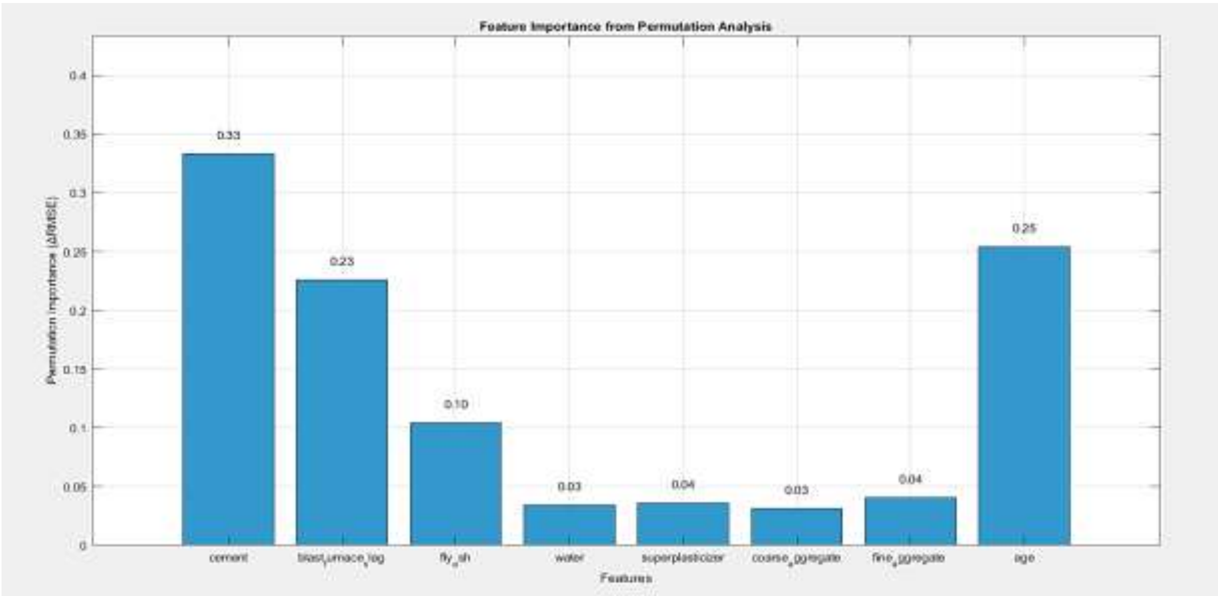


Figure 7. Relative Importance of Input Variables in the Neural Network Model.

4.4. Sensitivity Analysis

Sensitivity analysis is a statistical tool used to measure the impact of changes in inputs on the outputs of predictive models, which helps in identifying the most influential variables and improving the accuracy of the model [22]**.** Figure 8. shows that the variable 'age' is the most influential with a sensitivity value of 0.854, followed by 'cement' with a value of 0.780, then 'blast furnace slag' with a value of 0.506, while the effects of other variables decrease to range between 0.208 and 0.132 for 'fly ash', 'water', 'aggregate', and 'superplasticizer'. Thus, it can be concluded that the model outputs mainly depend on the factors of time and the content of binding materials, while the influence of other components plays a secondary role in determining the results.

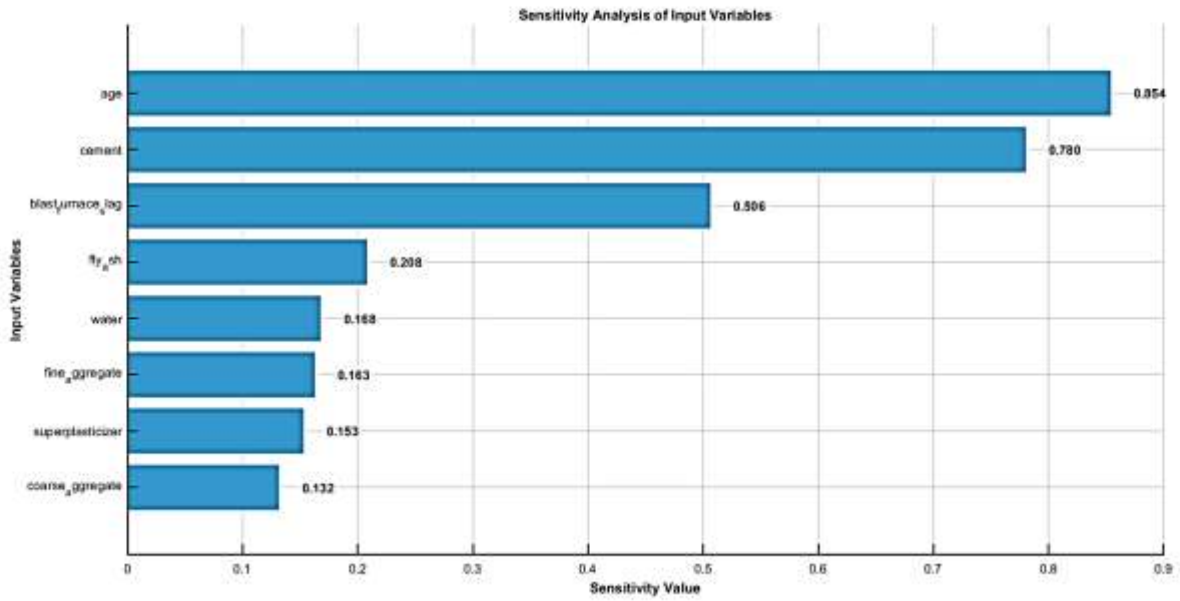


Figure 8. Sensitivity Analysis of Input Variables.

5. Conclusion

This study aimed to develop a simple and accurate predictive model for calculating the compressive strength of concrete based on eight basic components, with the goal of facilitating the engineering design process and reducing reliance on costly and time-consuming laboratory tests. Three practical steps were carried out:

1. Processing and cleaning data to handle duplicate and outlier values that may affect the results.

2. Normalizing the data to make all inputs treated equally such as a component with larger values dominates the component with smaller values.

3. Tuning the model hyper-parameters automatically and intelligently through Bayesian optimization saved manual tuning time and improve predictions accuracy significantly. The result was a model with (8) inputs, a single output, and a single hidden layer with (12) neurons.

Results indicated that compared to the previous model, the enhanced model had greater predictive power (R=0.939) on test data and low error rates (MAE=3.703, RMSE=5.304, RMAE=14.83). These values vindicate the fact that combining structured data with smart optimization gives models accuracy and stability.

the importance analysis indicates that (cement) is the most important variable affecting the compressive strength, which means it is the critical factor in controlling the regulated concrete mix. Model predictions remained stable under normal variation in component proportions, which was confirmed with the sensitivity analysis. however, it also showed strong sensitivity to [age] which requires closer attention in the field.

However, the present work focuses on laboratory-prepared data under fixed conditions, neglecting the influence of dynamic field factors including climate variations, casting modes or raw material quality. the study recommends to validate the model with real field datasets in order to augment its practical reliability on varying implementation conditions.

REFERENCES

[1] B. Matthews, D. Allaix, S. Wijte, and M. Vullings, "Advancing non-destructive concrete compressive strength estimation: Large-Scale datasets and machine learning framework," *NDT E Int.*, p. 103549, 2025.

[2] J.-S. Chou and A.-D. Pham, "Enhanced artificial intelligence for ensemble approach to predicting high performance concrete compressive strength," *Constr. Build. Mater.*, vol. 49, pp. 554–563, 2013.

[3] B. Ni, M. Z. Rahman, S. Guo, and D. Zhu, "A review on properties and multi-objective performance predictions of concrete based on machine learning models," *Mater. Today Commun.*, vol. 44, p. 112017, 2025.

[4] I.-C. Yeh, "Modeling of strength of high-performance concrete using artificial neural networks," *Cem. Concr. Res.*, vol. 28, no. 12, pp. 1797–1808, 1998.

[5] G. Onorato, "Bayesian optimization for hyperparameters tuning in neural networks," *arXiv Prepr. arXiv2410.21886*, 2024.

[6] D. Luo, K. Wang, D. Wang, A. Sharma, W. Li, and I. H. Choi, "Artificial intelligence in the design, optimization, and performance prediction of concrete materials: a comprehensive review," *Npj Mater. Sustain.*, vol. 3, no. 1, p. 14, 2025.

[7] S. J. Alghamdi, "Prediction of Concrete's Compressive Strength via Artificial Neural Network Trained on Synthetic Data," *Eng. Technol. Appl. Sci. Res.*, vol. 13, no. 6, pp. 12404–12408, 2023.

[8] J. Thapa, "Concrete compressive strength prediction by artificial neural network approach," *J. Eng. Issues Solut.*, vol. 3, no. 1, pp. 76–90, 2024.

[9] M. M. Jibril *et al.*, "High strength concrete compressive strength prediction using an evolutionary computational intelligence algorithm," *Asian J. Civ. Eng.*, vol. 24, no. 8, pp. 3727–3741, 2023.

[10] A. Q. Khan, H. A. Awan, M. Rasul, Z. A. Siddiqi, and A. Pimanmas, "Optimized artificial neural network model for accurate prediction of compressive strength of normal and high strength concrete," *Clean. Mater.*, vol. 10, p. 100211, 2023.

[11] M. Naved, M. Asim, and T. Ahmad, "Prediction of concrete compressive strength using deep neural networks based on hyperparameter optimization," *Cogent Eng.*, vol. 11, no. 1, p. 2297491, 2024.

[12] J. Xiong, Z. Ma, H. Yang, C. Yang, and J. Chen, "Bayesian-Optimized Fully Connected Neural Network For Enhanced Prediction Accuracy In Concrete Compressive Strength Estimation," *Civ. Environ. Eng.*, vol. 21, no. 2, pp. 1292–1303, 2025.

[13] A. K. Sah and Y.-M. Hong, "Performance comparison of machine learning models for concrete compressive strength prediction," *Materials (Basel).*, vol. 17, no. 9, p. 2075, 2024.

[14] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput.*, vol. 97, p. 105524, 2020.

[15] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artif. Intell. Rev.*, vol. 22, no. 2, pp. 85–126, 2004.

[16] T. O. Hodson, "Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not," *Geosci. Model Dev. Discuss.*, vol. 2022, pp. 1–10, 2022.

[17] H. Khoshvaght, R. R. Permala, A. Razmjou, and M. Khiadani, "A critical review on selecting performance evaluation metrics for supervised machine learning models in wastewater quality prediction," *J. Environ. Chem. Eng.*, vol. 13, no. 6, p. 119675, 2025.

[18] J. W. Tukey, *Exploratory data analysis*, vol. 2. Springer, 1977.

[19] D. C. Hoaglin, F. Mosteller, and J. W. Tukey, *Understanding robust and exploratory data analysis*. John Wiley & Sons, 2000.

[20] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[21] A. Fisher, C. Rudin, and F. Dominici, "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously," *J. Mach. Learn. Res.*, vol. 20, no. 177, pp. 1–81, 2019.

[22] T. Mohsin and R. Shakir, "Development of an artificial neural network model for predicting standard penetration test N-values from cone penetration test data," *Int. J. Comput. Methods Exp. Meas*, vol. 13, no. 3, pp. 680–696, 2025.