SSRG International Journal of Civil Engineering
ISSN: 2348-8352/ https://doi.org/10.14445/23488352/IJCE-V12I10P111

Volume 12 Issue 10, 139-146, October 2025

Original Article

Predicting Concrete Strength Using Regression-Based Machine Learning Techniques

S.Selvi[1], S.E. Murthy[2], Balaji Govindan[3], D. Leela Rani[4], Mutyala Suresh[5]

[1]*Department of Electrical and Electronics Engineering, Panimalar Engineering College, Chennai, Tamil Nadu, India.*
[2]*Department of Electrical and Electronics Engineering, Knowledge Institute of Technology, Salem, Tamil Nadu, India.*
[3]*Department of Civil Engineering, M. Kumarasamy College of Engineering, Karur, Tamil Nadu, India.*
[4]*Department of Electronics and Communication Engineering, School of Engineering, Mohan Babu University (erstwhile Sree Vidyanikethan Engineering College), Tirupati, Andhra Pradesh, India.*
[5]*Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India.*

[1]*Corresponding Author : selviselvaraj@gmail.com*

Received: 11 August 2025 Revised: 13 September 2025 Accepted: 12 October 2025 Published: 31 October 2025

Abstract *- Accurate prediction of concrete Compressive Strength (CS) is crucial for optimizing mix design and ensuring structural reliability. The prediction of concrete Strength remains challenging owing to the complex relationship between the components of the concrete mixture. Although several traditional methods are available, they are mostly experiment-based and are expensive and often inaccurate. This research employs Machine Learning (ML) approaches to estimate the Compressive Strength of concrete (CS) and to analyse how the input parameters influence the output response. Five tree-based regression algorithms, such as Random Forest and other boosting variants, were assessed to determine their predictive capability. The dataset contains 1030 samples with features such as water, fly ash, cement, age, coarse and fine aggregates, superplasticizer, and slag. Serve as inputs for developing the ML models. The model's accuracy is evaluated by metrics such as R^2, RMSE, and MAE. Based on the evaluation metrics, CatBoost is the best-performing model for predicting concrete compression strength. SHAP analysis further revealed that the age of the concrete and the amount of cement used are the most influential factors.*

Keywords *- Artificial Intelligence, Catboost, Compressive Strength, Machine Learning, Shap.*

1. Introduction

Concrete is a fundamental material used in the construction industry, and it plays a crucial role in modern infrastructure development. It is widely utilized in buildings, bridges, and infrastructure projects due to its Strength, durability, and cost-effectiveness [1]. The main composition of concrete is sand, Cement, and aggregates, which are mixed with water. The concrete's durability depends on each component's mechanical and physical properties. One of its most critical properties is Compressive Strength (CS), which directly influences load-bearing capacity and structural stability [2]. Traditionally, destructive testing methods, such as compression tests, determine the Compressive Strength. While these techniques are highly accurate, they are time-consuming, expensive, and labour-intensive, making them impractical for all applications [3]. The rapid development of Artificial Intelligence (AI) and Machine Learning (ML) has led researchers to apply these techniques for designing practical, cost-effective, and non-destructive approaches [4]. Several machine learning algorithms have been applied to estimate the compressive Strength of concrete, including kernel-based, instance-based, and tree-based models [5]. Comparative analyses have shown that ML-based methods generally provide more accurate predictions than traditional regression approaches.

In addition, ensemble learning algorithms, including XGBoost, LightGBM, and CatBoost, have demonstrated superior performance by effectively modeling the complex and nonlinear interactions among the components of concrete mixtures [6]. Several studies have supported these findings. Rahman et al. [7] assessed the effectiveness of Gradient boosting machines such as GBM, LightGBM, and CatBoost, demonstrating that boosting algorithms consistently yielded lower prediction errors than standalone ML models [8, 9]. Additionally, instead of a standalone model, some researchers also proposed hybrid ensemble models, which this combines one or two ML models to improve the predictive performance. These hybrid models leverage the strengths of each model and compensate for their weaknesses by strategically blending them. Mousavi et al. [10] proposed a hybrid ensemble model, a fusion of Random Forest and SVR, which further optimized predictive performance through feature fusion and hyperparameter tuning.

Table 1. Statistical Summary of Input and Output

Attribute	Average	Std. Deviation	Min	Lower Quartile	Median	Upper Quartile	Max
Cement (kg/m³)	281.17	104.51	102.00	192.38	272.90	350.00	540.00
Slag (kg/m³)	73.90	86.28	0.00	0.00	22.00	142.95	359.40
Fly Ash (kg/m³)	54.19	64.00	0.00	0.00	0.00	118.30	200.10
Water (kg/m³)	181.57	21.35	121.80	164.90	185.00	192.00	247.00
Superplastic	6.20	5.97	0.00	0.00	6.40	10.20	32.20
Coarse Aggregate (kg/m³)	972.92	77.75	801.00	932.00	968.00	1029.40	1145.0
Fine Aggregate (kg/m³)	773.58	80.18	594.00	730.95	779.50	824.00	992.60
Age (days)	45.66	63.17	1.00	7.00	28.00	56.00	365.00
Compressive Strength (MPa)	35.82	16.71	2.33	23.71	34.45	46.14	82.60

One main limitation of ML-based prediction is the lack of model interpretability. Nasir et al. [11] addressed this issue by implementing an explainable AI (XAI) technique like Shapley Additive Explanations (SHAP). It is a method from XAI that enables understanding how each input feature influences the prediction. Similarly, Huang et al. [12] applied the Local Interpretable Model Agnostic (LIME) technique to increase the computational efficiency. Li et al. [13] introduced a Bayesian Neural Network (BNN)-based uncertainty quantification model, which reduced overfitting issues while maintaining high prediction accuracy. In another study, Singh et al. [14] evaluated multiple feature selection methods, concluding that genetic algorithm-based feature selection significantly improved ML model accuracy by removing redundant variables.

Despite the growing number of studies, there is limited work that systematically compares multiple advanced tree-based ML models on the same dataset while also considering explainability and overfitting analysis, highlighting a clear research gap in conducting comprehensive evaluations.

Machine learning offers a variety of algorithms for predicting compressive Strength, each with advantages and limitations. In this study, five tree-based models are employed to predict the CS. These models are known for their ability to work with complex datasets, capture non-linear relationships, and provide accurate predictions. It also gives insights into feature importance, helping engineers understand which mix components most influence compressive Strength.

Moreover, the ensemble and tree-based models enhance prediction accuracy, offering practical guidance for material design and quality control. While ML is increasingly applied in the construction sector due to its ability to manage large datasets and uncover underlying patterns, certain challenges remain, including the need for large datasets and the difficulty in selecting the most appropriate models and input features, especially for specialized concrete types like fiber-reinforced composites [15, 16]. This work compares these five ML models to understand better their performance in predicting CS.

2. Dataset Description

In this study, the dataset contains 1030 samples with 8 input variables collected from previous research papers. The eight input variables, the target variable, and the statistical values of the variables are listed in Table 1. The feature names shown in the figures correspond to the original dataset variable. The CS of concrete mainly depends on the type and amount of materials used in the mixture. Cement serves as the primary binding agent in concrete, enabling it to harden and gain strength through hydration; higher cement content generally results in a stronger and denser mix, Water is essential for the chemical reaction with cement, but too much water can weaken the strength and workability. Superplasticizer enhances the concrete Strength with a lower water-cement ratio. Coarse aggregate contributes to the mechanical Strength by bearing the load, while fine aggregate fills the voids between particles, making the mix denser. The age is also a critical factor, as the concrete gains strength over time due to ongoing hydration.

Blast Furnace Slag, a waste product from the steel industry, can be used as a supplementary cementitious material. It helps improve concrete Strength over time and enhances durability. However, concrete with slag tends to develop early Strength more slowly. Fly Ash, a by-product of burning coal, strengthens the concrete matrix through pozzolanic reactions, refining the pore structure and contributing to Strength at later stages. However, it reacts slowly but improves the structure of the concrete in the long run. Both materials help reduce environmental impact and improve long-term performance. Still, they may significantly delay early strength gain if used in large amounts. Since these materials influence Strength differently, AI models can help predict their combined effect more accurately. The relationship between inputs and output features was visualized using histogram plots. The histogram plot is shown in Figure 1.

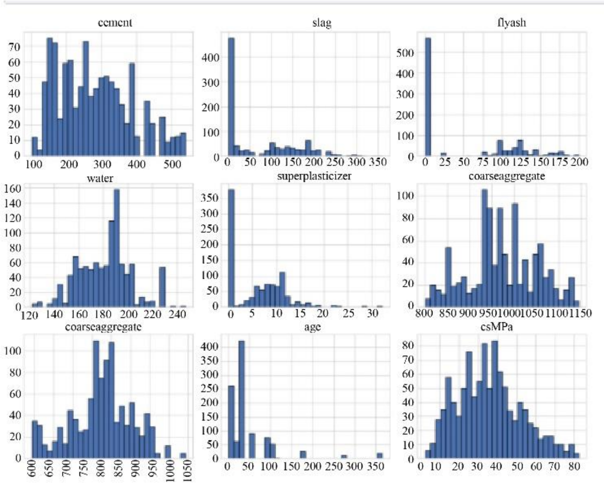

Fig. 1 Histogram of Input Variables

The histogram shows that the input variables, Cement**,** water**,** coarse aggregate**,** and fine aggregate, are used over a broad and balanced range, and the values are spread smoothly. Wate**r** values are mostly between 160 and 200, typical in concrete mixes. The other materials, such as slag**,** fly ash**,** and superplasticizer, have values close to zero, indicating they are used in small amounts or not in many samples. The concrete age is also mostly low, with most samples tested before 50 days and a few going up to 365 days. The compressive Strength is mostly between 20 and 60 MPa, appears to follow a normal distribution, showing the dataset is well balanced and suitable for machine learning models training. The histogram helps us to understand the dataset distribution and supports training the ML models. The linear relationship between the input and output parameters is represented in Figure 2. Cement shows the highest correlation with the compression strength, with a correlation coefficient of 0.5, and the correlation matrix is given in Figure 2.

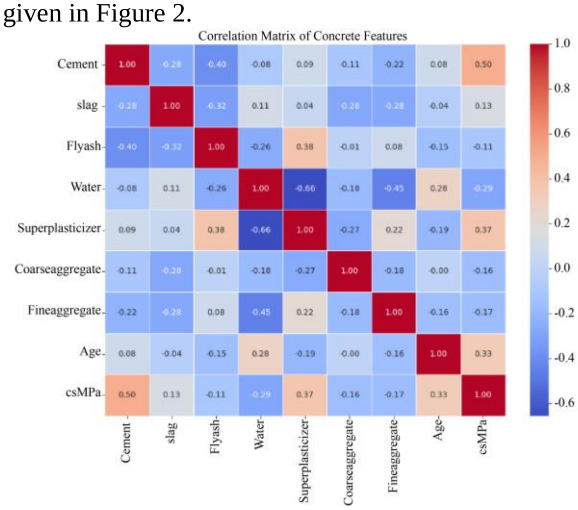

Fig. 2 Correlation Matrix of Input Variables

To complete the correlation analysis, the Variation Inflation Factor (VIF) was calculated and listed in Table 2.

Table 2. VIF value

Feature	VIF Value
Cement content	7.49
Ground granulated slag	7.28
Flyash content	6.17
Mixing Water	7.00
Superplasticizer	2.96
Coarse Aggregate weight	5.07
Fine Aggregate weight	7.01
Curing Age	1.12

The VIF indicates how much a feature is correlated with other features. The VIF value between 5 and 10 indicates moderate multicollinearity Table 2. It can be observed that all VIF values are less than 10. Hence, the dataset is suitable for regression analysis without any adjustments for multicollinearity.

3. Methodology

3.1. Random Forest Regression Model

Random Forest is a machine learning technique that constructs an ensemble of decision trees, each trained on randomly selected subsets of the dataset and features.. The predictions from all trees are then combined, usually by averaging, to produce a final output that is more reliable and accurate than that of an individual tree [9]. This approach helps the model work well even when the data has complex relationships. Due to its robustness [10], Random Forest is widely used in regression tasks, such as predicting concrete compressive Strength, mainly when the data contains many variables and is not simple [11].

3.2. Gradient Boosting Regressor

Gradient Boosting is a Sequential Machine Learning technique in which each model is trained to correct the errors of the preceding models. By combining these successive learners, the algorithm gradually improves prediction accuracy and reduces overall error [14] It uses gradient descent to reduce the errors step by step, thereby increasing the model's accuracy with each iteration [15]. This method makes predictions for both regression and classification problems [18]. Because of its flexibility and strong predictive power, it is widely used for real-world datasets [18].

3.3. XGBoost

It is a faster and powerful version of Gradient Boosting that introduces system and algorithmic enhancements for improved performance and scalability [16]. It uses regularization techniques to prevent overfitting and efficiently handles missing values and large datasets [17]. XGBoost also employs parallel processing, which significantly speeds up training. Because of these advantages [20], this model is commonly used in data science competitions and real-world applications that require high model performance and interpretability [20].

3.4. LightGBM

It is a fast and high-performance boosting algorithm [17]. It uses a leaf-wise tree growth strategy, which allows it to grow deeper trees and focus on parts of the data with the most errors [17]. This results in faster training and improved accuracy. This model is particularly suited for large volumes of data with many features and has become a popular choice for its speed, accuracy, and memory efficiency in predictive modeling tasks [18].

3.5. CatBoost Regressor

The CatBoost algorithm extends Gradient boosting by natively supporting categorical data, allowing it to process these features without additional preprocessing steps [16]. By employing methods like target encoding, CatBoost effectively captures the relationships in mixed-type datasets, enhancing predictive accuracy [15]. It also includes mechanisms to reduce overfitting and improve prediction accuracy. Its ability to work efficiently with categorical data makes it highly suitable for structured datasets in real-world applications [19].

3.6. Performance Metrics

The predictive accuracy of the machine learning models was evaluated using various performance metrics, which indicate how well the predicted outputs correspond to the observed values derived from the input features. The coefficient of determination (R^2) represents the proportion of variance in the dependent variable explained by the independent variables [22]. Values approaching 1 signify a higher level of agreement between the predicted and actual results. Equation (1) gives the formula for calculating the R^2 value

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i-(y_i\hat{}))}{\sum_{i=1}^{n}(y_{i-}\ y_i^{-})} \quad (1)$$

n = total number of data points

y_i =actual (true) value (also called the target variable)

$y_i\hat{}$= predicted value (from the model)

y_i^{-} = mean of actual values

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left[y_i - y_i\hat{}\right] \quad (2)$$

The MAE evaluates the average deviation of predictions from actual values. The formula for MAE is given by Equation 2.

The next metric, RMSE, is given in Equation 3. It squares the differences before averaging, which more severely penalizes higher errors [23]. Because of this, RMSE is more sensitive to outliers.

$$RMSE = \frac{1}{n}\sqrt{\sum_{i}^{n}(y_i - y_i\hat{})2} \quad (3)$$

A smaller RMSE reflects better prediction performance [23]. Collectively, these metrics offer a detailed assessment of model effectiveness and allow for meaningful comparisons across different algorithms and datasets [23].

4. Evaluation of the Model

The methodology flowchart in Figure 3. Show the steps involved in the prediction of compression strength. To make the data suitable for training, several pre-processing steps are applied. These include cleaning the data and normalizing or standardizing the features using the Scikit-learn (Sklearn) library. The pre-processed dataset was partitioned into training and test subsets, with a 70:30 ratio for model training and testing, respectively. The training data was used to build and adjust the machine learning algorithms, while the test set was used to check how well they work on new data. K-fold cross-validation was also applied to ensure the results were consistent across different parts of the dataset.

Table 3. Evaluation results

Algorithm	Training dataset			Validation dataset		
	MAE	RMSE	R^2	MAE	RMSE	R^2
Random Forest	1.3021	1.9841	0.9861	3.7385	5.4719	0.8838
Gradient Boosting	2.9049	3.8717	0.9472	4.1350	5.4934	0.8829
XGBoost	0.9848	1.6060	0.9909	3.2579	4.9698	0.9041
LightGBM	1.4231	2.1998	0.9830	3.2005	4.7066	0.9140
CatBoost	1.2397	1.8459	0.9880	2.6358	4.0395	0.9367

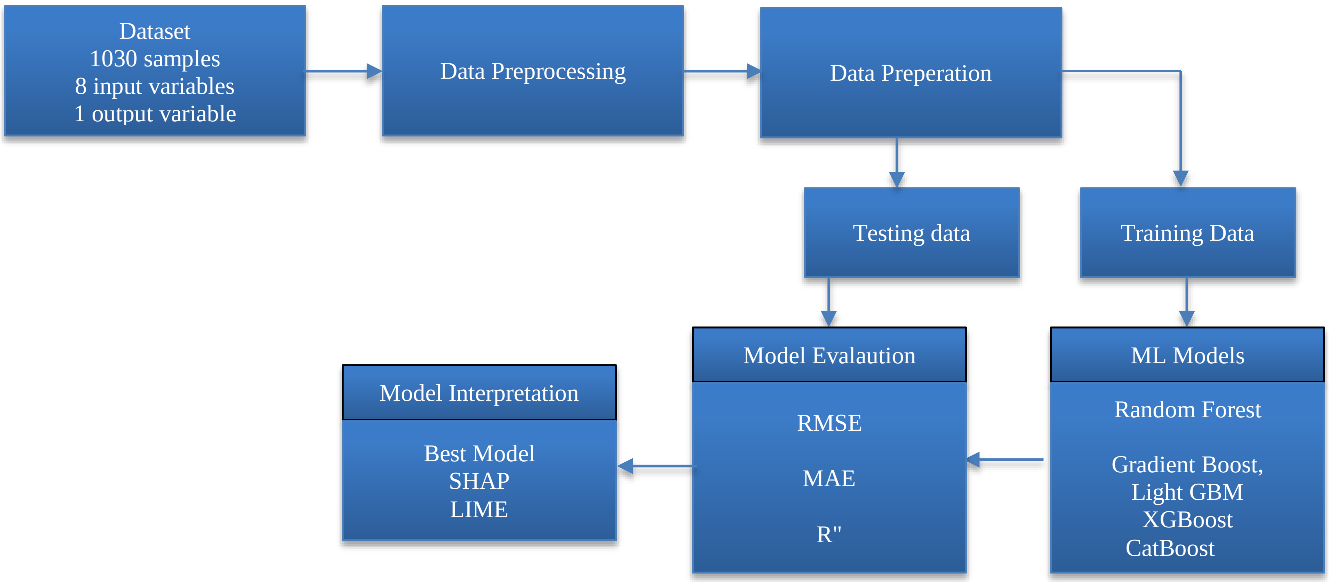

Fig. 3 Methodology flow chart

In this case, a 10-fold cross-validation is used. Ten equal sections make up the dataset, which includes 1,030 samples and eight input variables. Nine parts are used for training. And one part is reserved for validation in each cycle. Since this procedure is repeated ten times, each component was tested once. This process reduces overfitting, and the performance is assessed using the average results across the folds. The best-performing model configurations were retrained using the whole training dataset for the final evaluation on the test set. After the model is trained and fine-tuned, it undergoes a detailed evaluation to assess predictive performance. Key metrics like R^2 measure how well the prediction matches the actual value, while MAE and RMSE evaluate the accuracy and magnitude of error, with RMSE giving more weight to larger errors. Visual tools such as scatter plots and histograms illustrate prediction accuracy, error distribution, and potential bias. Following the evaluation metrics shown in the Table.3, CatBoost demonstrated superior performance across all models. It achieves the lowest test error values (MAE = 2.6358, RMSE = 4.0395) and the highest R^2 score (R^2 = 0.9367) on the testing dataset, indicating the model predicts accurately and works well on new data. While XGBoost and LightGBM perform well in training, they show relatively larger gaps between training and test performance, showing some over-fitting. For instance, XGBoost has an excellent training R^2 of 0.9909 but drops to 0.9041 in testing, highlighting that the model captures patterns in the training data that might not generalize to unseen input. Though not severe, this performance gap suggests overfitting as the testing results remain relatively strong. CatBoost, on the other hand, maintains a strong balance between training (R^2 = 0.9880) and testing (R^2 = 0.9367) metrics, with minimum overfitting. Its ability to natively handle categorical data, internal regularization, and a robust Gradient boosting framework helps perform well. In conclusion, CatBoost is the most reliable model, offering both high accuracy and robustness on unseen data, making it a reliable option for predicting concrete compressive Strength in this study.

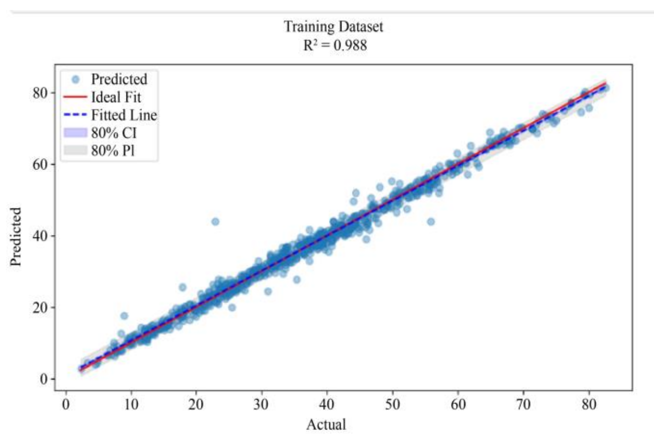

Fig. 4 (a) Actual vs. CatBoost predictions for the training set

Figures 4(a) and 4(b) illustrate the CatBoost model's predicted Compressive Strength (CS) values plotted against the actual values for the training and test datasets. The red line marks the ideal fit, while the shaded areas show the 80% Prediction Interval and 80% Confidence Interval. In the training set, 5.70% and 88.47% of predictions fall inside the confidence and prediction intervals. For the test set, 12.14% and 90.29% of predictions fall inside the respective intervals. This suggests that the model's point predictions are generally reliable, although the narrow confidence intervals may be sensitive to variations in data.

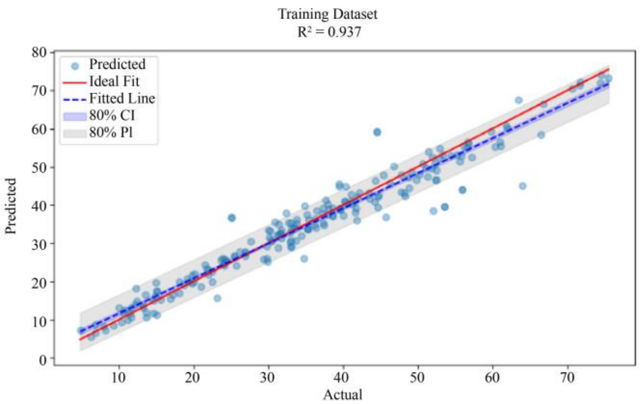

Fig. 4 (b) Actual vs. CatBoost predictions for the testing set.

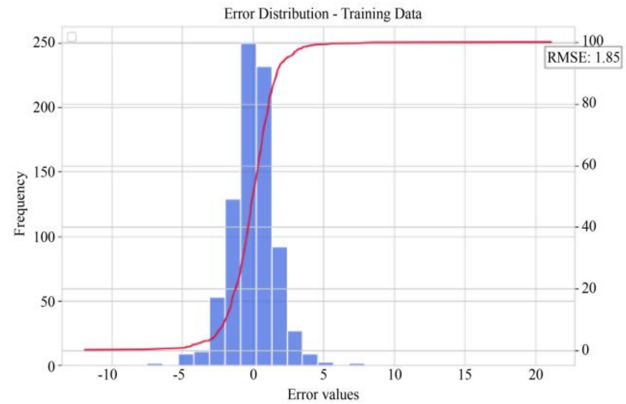

Fig. 5 (a) RMSE values for the training set

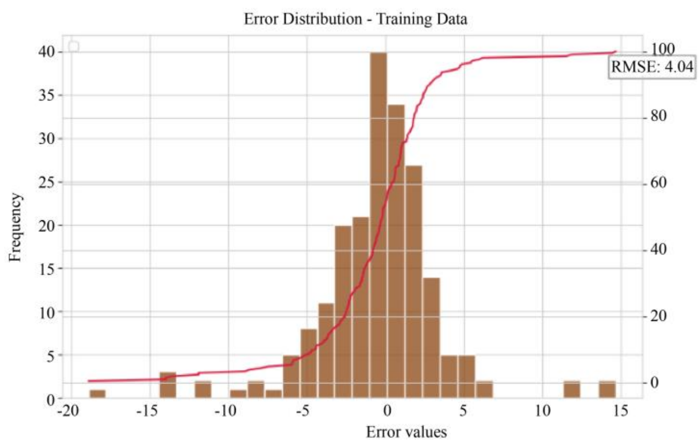

Fig. 5 (b) RMSE values for the test set

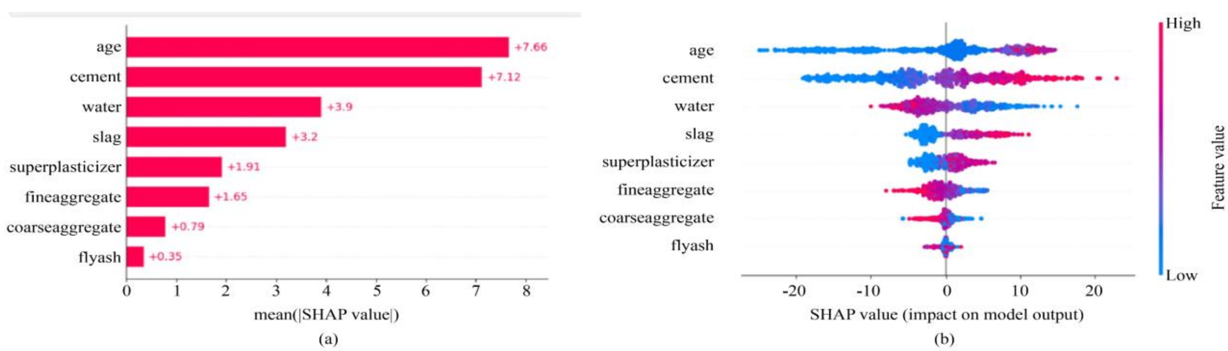

Fig. 6 SHAP global feature importance (a) mean absolute SHAP value, and (b) global SHAP value.

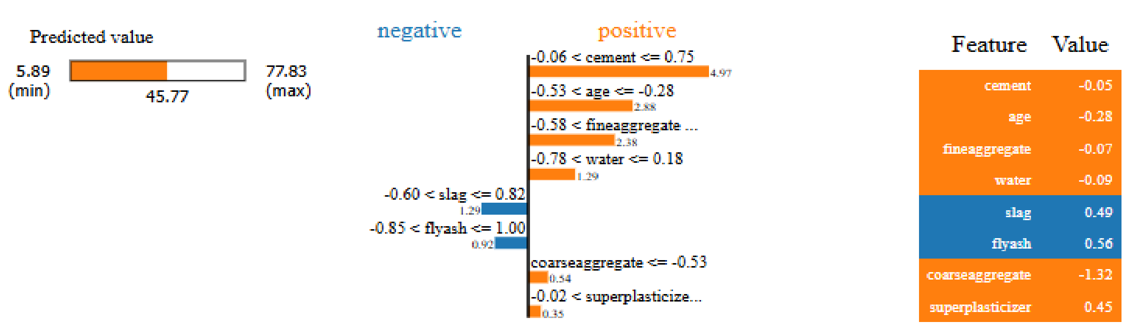

Fig. 7 local interpretation of one data point with actual value 44.2 MPa compared with predicted value 45.77MPa

Figure 5(a) and (b) shows the error distribution between the actual and the predicted values. Most prediction errors fall within ±5 MPa for the training dataset and ±10 MPa for the testing dataset, with RMSE values of 1.85 and 4.04, respectively. This indicates good accuracy and reliable performance of the CatBoost model

5. SHAP-Based Evaluation

Figure 6 presents the SHAP analysis illustrating how each input feature influences predictions. Each data point is represented by a dot in the plot, and its position indicates how well it reflects the magnitude of the feature's effect on the prediction. The dot's color reflects the feature value, with red representing high values and blue representing low values. From the plot, it is clear that the age of the concrete and the amount of Cement used have the most substantial positive impact on the compressive Strength; higher values of both tend to increase the predicted Strength. In contrast, a high water content generally lowers the prediction. Other features like slag, superplasticizer, and fine aggregate influence the prediction, but to a lesser extent, while coarse aggregate and flyash have the least impact..

The bar chart highlights how much, on average, each feature contributes across all data points. Age and Cement again emerge as the most important, with the highest mean SHAP values, followed by water and slag. These four features are the key drivers in predicting compressive Strength, while the remaining variables contribute less significantly.

The LIME plot is shown in Figure 7, showing how the model makes its prediction for one sample. The model predicted compressive Strength is 45.77 MPa, close to the actual value of 44.28 MPa. LIME breaks down the prediction into positive and negative contributions from each feature. The orange bars show features that increase the prediction strength, termed Positive values. In this case, features like cement, age, fine aggregate, and water positively contributed to increasing the predicted strength. The blue bars show the features that decrease the prediction strength, hence termed as Negative values. Slag, fly ash, coarse aggregate, and superplasticizer reduced the predicted value. This analysis highlights the influence of each input attribute on the model's output at every data point.

6. Conclusion

Five machine learning models were evaluated in this study for predicting concrete Strength using input features such as composition and curing age. Among the models evaluated, CatBoost outperformed the other models, demonstrating high R^2 scores for both training and testing data, with fewer prediction errors. SHAP analysis helped identify which input features influence concrete strength.

While the CatBoost model performed well on the current dataset, the study has some limitations. The dataset used was relatively small and tailored to one particular type of concrete mix, which may reduce the model's effectiveness when applied to varied compositions and conditions. To improve real-world applications, future research should focus on using larger and more diverse datasets that cover different environmental conditions and concrete compositions. This would enhance the model's performance and generalization.

In conclusion, machine learning models offer a fast, cost-effective, and non-destructive way, a scalable alternative to estimate concrete Strength. The methods can help engineers design better mixes and improve quality control.

References

[1] M. Ahmadi, H. Naderpour, and A. Kheyroddin, "Utilization of Artificial Neural Networks to Prediction of the Capacity of CCFT Short Columns Subject to Short Term Axial Load," *Archives of Civil and Mechanical Engineering*, vol. 14, no. 3, pp. 510-517, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[2] Faeze Khademi et al., "Predicting Strength of Recycled Aggregate Concrete Using Artificial Neural Network, Adaptive Neuro-Fuzzy Inference System and Multiple Linear Regression," *International Journal of Sustainable Built Environment*, vol. 5, no. 2, pp. 355-369, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[3] Zhibin He et al., "A Comparative Study of Artificial Neural Network, Adaptive Neuro Fuzzy Inference System and Support Vector Machine for Forecasting River Flow in the Semiarid Mountain Region," *Journal of Hydrology*, vol. 509, pp. 379-386, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[4] Anh-Duc Pham, Nhat-Duc Hoang, and Quang-Trung Nguyen, "Predicting Compressive Strength of High-Performance Concrete Using Metaheuristic-Optimized Least Squares Support Vector Regression," *Journal of Computing in Civil Engineering*, vol. 30, no. 3, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[5] Doddy Prayogo, Foek Tjong Wong, and Daniel Tjandra, "Prediction of High-Performance Concrete Strength Using a Hybrid Artificial Intelligence Approach," *MATEC Web of Conferences*, vol. 203, pp. 1-12, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Jui-Sheng Chou et al., "Machine Learning in Concrete Strength Simulations: Multi-Nation Data Analytics," *Construction and Building Materials*, vol. 73, pp. 771-780, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[7] Aman Kumar et al., "Prediction of FRCM–Concrete Bond Strength with Machine Learning Approach," *Sustainability*, vol. 14, no. 2, pp. 1-25, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Shi-Zhi Chen et al., "Ensemble Learning Based Approach for FRP-Concrete Bond Strength Prediction," *Construction and Building Materials*, vol. 302, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] Wang Zeyu et al., "Random Forest Based Hourly Building Energy Prediction," *Energy and Buildings*, vol. 171, pp. 11-25, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[10] Seyyed Mohammad Mousavi et al., "A New Predictive Model for Compressive Strength of HPC Using Gene Expression Programming," *Advances in Engineering Software*, vol. 45, no. 1, pp. 105-114, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[11] Jiandong Huang et al., "Predicting the Compressive Strength of the Cement-Fly Ash–Slag Ternary Concrete Using the Firefly Algorithm (FA) and Random Forest (RF) Hybrid Machine-Learning Method," *Materials*, vol. 15, no. 12, pp. 1-15, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Hongwei Song et al., "Predicting the Compressive Strength of Concrete with Fly Ash Admixture Using Machine Learning Algorithms," *Construction and Building Materials*, vol. 308, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Furqan Farooq et al., "A Comparative Study for the Prediction of the Compressive Strength of Self-Compacting Concrete Modified with Fly Ash," *Materials*, vol. 14, no. 17, pp. 1-27, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[14] Jesús de-Prado-Gil et al., "To Predict the Compressive Strength of Self Compacting Concrete with Recycled Aggregates Utilizing Ensemble Machine Learning Models," *Case Studies in Construction Materials*, vol. 16, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Tianyu Xie et al., "A Unified Model for Predicting the Compressive Strength of Recycled Aggregate Concrete Containing Supplementary Cementitious Materials," *Journal of Cleaner Production*, vol. 251, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] Kabiru O. Akande et al., "Performance Comparison of SVM and ANN in Predicting Compressive Strength of Concrete," *IOSR Journal of Computer Engineering*, vol. 16, no. 5, pp. 88-94, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[17] Ayaz Ahmad et al., "Comparative Study of Supervised Machine Learning Algorithms for Predicting the Compressive Strength of Concrete at High Temperature," *Materials*, vol. 14, no. 15, pp. 1-19, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Wenbin Lan et al., "Accurate Compressive Strength Prediction Using Machine Learning Algorithms and Optimization Techniques," *Journal of Engineering Applied Science*, vol. 71, pp. 1-20, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] De-Cheng Feng et al., "Machine Learning-Based Compressive Strength Prediction for Concrete: An Adaptive Boosting Approach," *Construction and Building Materials*, vol. 230, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Vivian W.Y. Tam et al., "A Prediction Model for Compressive Strength of CO2 Concrete Using Regression Analysis and Artificial Neural Networks," *Construction and Building Materials*, vol. 324, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] Zaher Mundher Yaseen et al., "Predicting Compressive Strength of Lightweight Foamed Concrete Using Extreme Learning Machine Model," *Advances in Engineering Software*, vol. 115, pp. 112-125, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[22] Ayaz Ahmad et al., "Application of Novel Machine Learning Techniques for Predicting the Surface Chloride Concentration in Concrete Containing Waste Material," *Materials*, vol. 14, no. 9, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[23] Ayaz Ahmad et al., "Compressive Strength Prediction of Fly Ash-Based Geopolymer Concrete via Advanced Machine Learning Techniques," *Case Studies in Construction Materials*, vol. 16, pp. 1-16, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Zhiyuan Li et al., "High Temporal Resolution Prediction of Street-Level PM2.5 and NOx Concentrations Using Machine Learning Approach," *Journal of Cleaner Production*, vol. 268, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[25] Mostafa Jalal, and Hamid Jalal, "RETRACTED: Behavior Assessment, Regression Analysis and Support Vector Machine (SVM) Modeling of Waste Tire Rubberized Concrete," *Journal of Cleaner Production*, vol. 273, pp. 1-15, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[26] Fangming Deng et al., "Compressive Strength Prediction of Recycled Concrete Based on Deep Learning," *Construction and Building Materials*, vol. 175, pp. 562-569, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[27] S. Selvi et al., "Enhancing Short-Term PV Power Forecasting Using Deep Learning Models: A Comparative Study of DNN and CNN Approaches," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 11, no. 8, pp. 73-80, 2024. [CrossRef] [Google Scholar] [Publisher Link]