

Machine Learning Prediction of Concrete Compressive Strength: Model Comparison, CatBoost Optimization, and SHAP Interpretation

Musthafa 'Abduh Fakhruddin^{1*}, Sri Winarno², Acun Kardanawati³

^{1,2}Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, Indonesia

³Sistem Informasi, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, Indonesia

Email: ¹*111202013125@mhs.dinus.ac.id, ²sri.winarno@dsn.dinus.ac.id, ³acun.kardanawati@dsn.dinus.ac.id

(*: corresponding author)

Abstract - Accurate prediction of concrete compressive strength is vital for structural design, yet conventional testing is constrained by lengthy curing requirements. Machine learning offers an alternative by modeling non-linear mix-performance interactions. This study presents a comparative framework evaluating nine regression algorithms using the UCI Concrete Compressive Strength dataset (1,005 samples). Performance was assessed via 10x5 repeated cross-validation with 95% confidence intervals, and statistical significance was evaluated using a Linear Mixed-Effects Model with Holm-Bonferroni corrected pairwise t-tests. Tree-based ensembles outperformed linear approaches, with CatBoost yielding the highest baseline cross-validation R^2 of 0.931 (95% CI: 0.927 to 0.935). Subsequent Bayesian hyperparameter optimization via Optuna's Tree-structured Parzen Estimator (400 trials) improved the final CatBoost model's performance to a test R^2 of 0.943, RMSE of 4.142 MPa, and MAE of 2.616 MPa. SHAP analysis indicated that curing age, the water-to-binder ratio, and cement are the dominant predictors, while the model exhibited physically consistent behavior aligned with concrete hydration kinetics and Abrams' law. This work's key contribution is jointly integrating correlation-corrected statistical validation, multi-model Bayesian optimization, and domain-informed feature engineering with SHAP interpretation, rarely combined in prior concrete-strength studies. The framework offers an accurate, interpretable tool for preliminary concrete mix design.

Keywords: Bayesian optimization, CatBoost, Concrete compressive strength, SHAP interpretability, Machine Learning.

1. INTRODUCTION

Concrete compressive strength is a critical parameter governing structural safety, quality, and performance across the life cycle of infrastructure [1]. However, traditional laboratory testing methods are associated with low efficiency, high costs, and energy consumption [2]. Predicting this property is inherently complex, as it results from nonlinear interactions between various mixture components, such as cement, water-binder ratios, and aggregates, as well as the curing duration [3]. As a result, traditional empirical equations like Abrams' law are insufficient to model these nonlinear relationships, leading to the adoption of data-driven machine learning methods for more efficient strength prediction [4].

To overcome these limitations, machine learning has emerged as an alternative, demonstrating higher predictive accuracy than conventional approaches by learning patterns from large datasets, such as the benchmark UCI dataset established by Yeh [5]. However, the comparative methodologies in current concrete strength studies are often restricted in scope. A substantial share of the literature evaluates only a small set of candidate models, typically two to four algorithms, identifying the optimal model primarily through point estimates such as R^2 , RMSE, and MAE e.g., [6], [7]. Even when comparative studies encompass a broader range of algorithms, the dominant practice relies on metric-based ranking without employing formal nonparametric frameworks, such as the Friedman or Wilcoxon tests, to verify whether observed performance differences are statistically significant e.g., [8], [9].

Although formal statistical validation is a standard practice in general machine learning methodologies to assess model variance and prevent evaluation bias from single-test-set predictions [10], it remains uncommon in concrete materials research. General ML literature frequently utilizes nonparametric multiple comparison tests, such as the Friedman test and its variants [11] or agnostic statistical regressions [12], to verify that performance differences are statistically significant rather than variations caused by specific data splits. While these validation methods are applied in other regression domains e.g., [13], [14], [15], their limited use in concrete strength prediction indicates a methodological gap.

Furthermore, two methodological gaps persist regarding model interpretability and hyperparameter tuning. First, many traditional machine learning models remain 'black-box,' and although Explainable AI methods such as SHAP have been introduced, their application remains inconsistent e.g. [6], [8], [9]. Second, hyperparameter configurations require dataset-specific tuning, yet optimization procedures and exact parameter settings are frequently underreported, limiting reproducibility e.g. [8], [9].

As compactly summarized in Table 1. Comparison of Recent Machine Learning Studies on Concrete Compressive Strength, while benchmark datasets like UCI are widely used, the integration of advanced Bayesian hyperparameter optimization coupled with explainable AI and statistical tests remains underexplored in tandem. Rather than simply proposing another incremental predictive model, this study addresses these intertwined

deficiencies by establishing a comprehensive, reproducible, and interpretable framework. The primary objectives are to: (1) compare nine distinct ML regression algorithms and validate their performance differences statistically via a Linear Mixed-Effects Model (Likelihood-Ratio Test) followed by Nadeau-Bengio corrected + Holm-adjusted t-tests, (2) optimize the top-performing models using Bayesian search via Optuna, and (3) interpret the specific influence of compositional features on predictions using SHAP. By integrating model comparison, statistical validation, systematic optimization, and SHAP-based interpretability, this research provides a reproducible and interpretable methodology for concrete strength prediction in civil engineering.

Table 1. Comparison of Recent Machine Learning Studies on Concrete Compressive Strength

Reference	Dataset	Algorithms	Tuning Method	Statistical Test	XAI*	Limitation / Gap
Fathy et al., 2025 [6]	Custom (324 samples)	XGBoost, RF, M5P	Grid Search	None	SHAP, ICE, PDP, SHAP	Lacks non-parametric statistical validation for performance.
Elhishi et al., 2024 [9]	UCI (1030 samples)	XGBoost, RF, SVM, etc.	Underreported	None	SHAP	Underreports Hyperparameter tuning and lacks formal statistical tests.
Chen et al., 2024 [7]	Commercial (189 samples)	BPNN, SVM, RF	Particle Swarm Optimization	None	None	Omits XAI interpretability and relies solely on basic error metrics.
Javed et al., 2024 [8]	Literature (261 samples)	XGBoost, Gradient Boosting, CatBoost, etc.	Underreported	None	SHAP	Underreports tuning strategies and lacks formal statistical validation.
This Study	UCI (1005)	9 Algorithms	Optuna (Bayesian)	LMM (LRT) + Nadeau-Bengio corrected, Holm-adjusted t-test	SHAP	-

*Explainable AI (XAI)

2. RESEARCH METHODS

This section outlines the methodology for developing, optimizing, and interpreting machine learning models to predict concrete compressive strength. The research framework divided into interconnected stages, as illustrated in the flowchart in Figure 1.

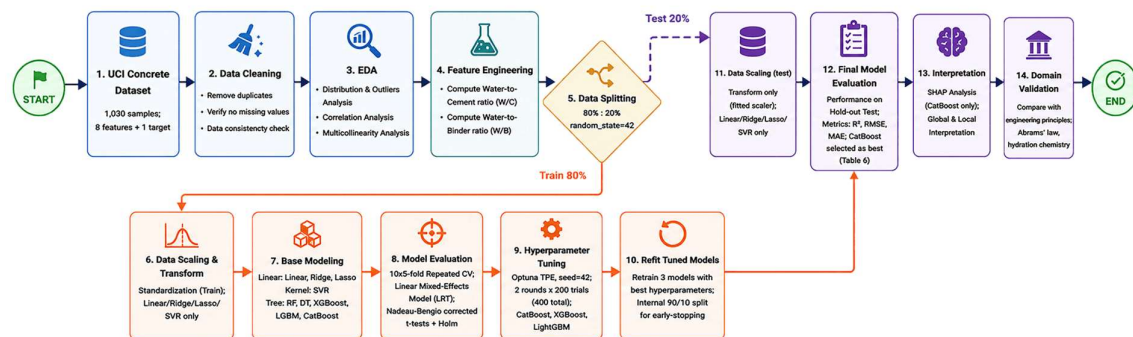


Figure 1. Research Flowchart.

2.1. Dataset Collection and Data Cleaning

This study utilizes the Concrete Compressive Strength dataset [5] from the UCI Machine Learning Repository. It comprises 1,030 samples with eight input variables and one target variable: Compressive Strength (MPa). Because a machine learning model's performance heavily depends on the quality of its training data [16], a cleaning process was conducted prior to modeling. This included standardizing column names, removing duplicate rows, and verifying the absence of missing values.

2.2. Exploratory Data Analysis (EDA)

Exploratory data analysis (EDA) was conducted to uncover underlying data characteristics and variable relationships [17]. First, data distributions and outliers were evaluated using histograms, Kernel Density Estimation (KDE), and the Interquartile Range (IQR). Feature independence was then analyzed via a Pearson correlation heatmap. Finally, multicollinearity was assessed using the Variance Inflation Factor (VIF) to detect redundancies that inflate regression variance.

2.3. Feature Engineering

To incorporate established concrete mix design principles into the predictive models, two domain-informed features were engineered from the original input variables: the water-to-cement ratio (W/C) and the water-to-binder ratio (W/B), where all constituent quantities are expressed in kg/m³. These ratios provide physically interpretable descriptors of the balance between water and cementitious materials, which is widely recognized as a key factor governing concrete compressive strength [18], [19]. Both engineered features were incorporated into the predictor set before data preprocessing, model training, and hyperparameter optimization.

2.4. Data preprocessing

Because raw datasets require careful preprocessing to ensure reliable machine learning outcomes [20], the dataset was first partitioned into training and testing sets (80:20 ratio, random state 42). Because linear and distance-based algorithms require consistent feature scales [21], standardization was applied specifically to the Linear, Ridge, Lasso, and SVR models to enhance performance. Crucially, to avoid preprocessing leakage [22], the scaler was fitted exclusively on the training set and subsequently used to transform the test set.

2.5. Base Modeling

The base modeling phase establishes an initial benchmark by evaluating nine distinct regression algorithms using their library-default hyperparameters, with the exception that the three gradient-boosting models (XGBoost, LightGBM, CatBoost) were trained using the same internal-validation early-stopping strategy (10% held-out split, patience = 100 rounds) as the final tuned model, to keep the baseline-to-tuned comparison consistent. This step assesses the models' baseline predictive performance prior to tuning [23] and ensures transparent baseline configurations, a critical factor often omitted in algorithmic research [24]. Furthermore, because uncontrolled pseudo-randomness during model generation can hinder replication [25], a fixed random state of 42 was applied across all stochastic components. This practice mitigates sensitivity to training conditions and aligns with broader efforts to improve reproducibility standards in machine learning [26]. As illustrated in the methodological flowchart, these algorithms are grouped into three primary paradigms: Linear Models, a Kernel-Based Model, and Tree-Based Models.

2.6. Model Evaluation

Evaluating regression models requires more than a single metric to fully capture prediction quality, which is particularly crucial for concrete strength prediction where accuracy directly impacts structural safety [27], [28]. This study therefore employs a multi-metric approach (R^2 , RMSE, and MAE) where R^2 quantifies explained variance, RMSE (in MPa) emphasizes large deviations, and MAE quantifies average error uniformly, with the RMSE-MAE discrepancy monitored to flag localized extreme errors [29]. Actual-versus-predicted and residual scatter plots further complement these metrics by revealing systematic deviations, nonlinearity, and heteroscedasticity [30], [31].

The complete evaluation pipeline follows a data partitioning protocol to prevent information leakage. First, the full dataset is split into a training partition (80%) and a held-out test partition (20%) using random sampling. All subsequent model training, cross-validation, and hyperparameter optimization are performed exclusively on the training partition, the test partition is reserved solely for the final, unbiased performance assessment. Preprocessing steps, including feature scaling for distance-based algorithms, are fitted only on the training data and subsequently applied to the test data to avoid data leakage.

To evaluate performance stability, all models underwent a 10x5-fold repeated cross-validation on the training partition, a technique used to reduce the variance associated with a standard single train-test split [32]. By repeating a 5-fold cross-validation ten times with different random splits, this process generates 50 evaluation scores per model, providing a broader assessment of model generalization across various data permutations. For each algorithm, the mean and standard deviation of the performance metrics (R^2 , RMSE, and MAE) were recorded. Additionally, a 95% Confidence Interval (CI) was computed using the student's t-distribution with 49 degrees of freedom ($n_{scores} - 1$) to estimate the uncertainty of the out-of-sample performance. The utilization of 50

evaluation scores yields substantially narrower confidence intervals compared to a single 5-fold run, serving as a more reliable indicator of performance consistency [33].

The final step in the evaluation pipeline assesses whether the observed performance differences among models are statistically significant. Because the 50 scores generated from repeated cross-validation are computed on overlapping training sets within each repeat, they exhibit inherent statistical dependence [33]. To appropriately model this dependency structure, a Linear Mixed-Effects Model (LMM) was employed as the omnibus test [34], [35]. In this framework, the machine learning algorithm was specified as a fixed effect representing the systematic performance difference of interest, while the cross-validation repeat was specified as a random effect to account for variability arising from repeated data partitioning. The significance of the fixed algorithm effect was evaluated using a Likelihood-Ratio Test (LRT) comparing the full model against a nested null model excluding the algorithm effect.

Upon rejection of the omnibus null hypothesis ($p - value < 0.05$), post-hoc pairwise comparisons were performed to identify significant differences between algorithms. For each pair, a Nadeau-Bengio corrected resampled paired t-test was applied to the 50-fold-level score difference ($k = 10 \text{ repeats} \times 5 \text{ folds}$), which inflates the variance estimate to account for the statistical dependence introduced by overlapping training sets across folds and repeats [36], [37], [38]. Holm-Bonferroni correction was subsequently applied across all pairwise comparisons to control the family-wise error rate. Finally, Cohen's d_z was reported for each comparison to quantify effect size and distinguish statistical significance from practical significance.

2.7. Hyperparameter Tuning

This phase employs a two-stage hyperparameter tuning strategy using Optuna (v.4.9.0), a framework for Bayesian optimization [39]. The search is driven by the Tree-Structured Parzen Estimator (TPE) with a fixed random seed (seed=42) for reproducibility, which models conditional probability distributions of hyperparameter configurations to maximize the 5-fold cross-validation R^2 score computed on the training partition.

The optimization consists of two sequential 200-trial rounds (400 total). This identical two-round procedure (400 trials) was applied independently to each of the three tree-based candidates, CatBoost, XGBoost, and LightGBM. The first round explores a broad parameter space, while the second refines the boundaries around the top-performing region to achieve convergence. Trial histories are logged via SQLite. To counteract potential optimism from this adaptive narrowing, unbiased generalization performance is evaluated on a held-out test set.

Using the optimal configuration (detailed in Table 5), the final model is trained for up to 3,000 boosting iterations with an early stopping patience of 100 rounds. Model fit and convergence are evaluated via learning curves, followed by a range-specific error analysis across low, medium, and high compressive strength categories to identify any prediction biases.

2.8. Interpretation Model

As the final step in this research pipeline, an Interpretation Model is deployed to interpret the black-box nature of the optimized algorithm and ensure transparent decision-making [40]. This phase utilizes SHapley Additive exPlanations (SHAP), a feature attribution method grounded in cooperative game theory [41], to compute the marginal contribution of each feature via the TreeExplainer algorithm [42]. The SHAP framework systematically decomposes predictions into additive feature effects, dividing the evaluation into Global Analysis and Local Interaction. For the Global Analysis, Bar and Beeswarm plots are aggregated across the testing dataset to rank overall feature importance, determine impact directions, and uncover multi-modal trends. Conversely, Local Interaction evaluations employ Dependence plots alongside Local Bar charts to dissect complex feature synergies and identify the top contributing features, by magnitude and direction, driving the prediction for specific individual samples. Finally, these SHAP outputs are subjected to Engineering Domain Validation to ensure that the computationally derived feature attributions logically align with established civil engineering principles, such as evaluating the specific impacts and correlations of raw ingredients like cement and superplasticizer on compressive strength [43].

3. RESULTS AND DISCUSSION

3.1. Data Collection and Cleaning Result

The original dataset comprised 1,030 samples. An initial exploratory analysis identified 25 duplicate records, where all input features and the target variable were identical. Given that the target values in these duplicate rows matched precisely up to six decimal places, they likely represent repeated reporting or aggregation artifacts rather than true experimental replicates. Regardless of their origin, retaining exact duplicates can introduce data leakage during cross-validation, where identical samples appearing in both training and validation folds artificially inflate evaluation metrics. Consequently, these 25 exact duplicates were removed to ensure unbiased model evaluation,

yielding a final dataset of 1,005 samples. An inspection identified 9 groups (20 rows) of identical mix composition with differing compressive strength; unlike the 25 exact duplicates removed above, these were retained since a differing target rules out simple data-entry duplication and most plausibly reflects normal specimen-to-specimen variability in replicate strength testing. To confirm that this deduplication rule did not bias the modeling process, a sensitivity analysis was performed using the baseline CatBoost model. The performance difference was negligible; the model achieved an R^2 of 0.942 on the raw dataset (1,030 samples) compared to an R^2 of 0.943 on the deduplicated dataset (1,005 samples). This minimal variance ($\Delta R^2 = 0.001$) demonstrates that the removal of duplicates did not alter the fundamental modeling outcomes but effectively prevented potential validation leakage. No missing values were found in any column. The descriptive statistics of the cleaned dataset are presented in Table 2.

Table 2. Descriptive Statistics of The Concrete Dataset (N = 1,005)

Variable	Count	Mean	Std	Min	25%	50%	75%	Max
Cement	1005	278.6	104.3	102.0	190.7	265.0	349.0	540.0
Blast Furnace Slag	1005	72.0	86.2	0.0	0.0	20.0	142.5	359.4
Fly Ash	1005	55.5	64.2	0.0	0.0	0.0	118.3	200.1
Water	1005	182.1	21.3	121.8	166.6	185.7	192.9	247.0
Superplasticizer	1005	6.0	5.9	0.0	0.0	6.1	10.0	32.2
Coarse Aggregate	1005	974.4	77.6	801.0	932.0	968.0	1031.0	1145.0
Fine Aggregate	1005	772.7	80.3	594.0	724.3	780.0	822.2	992.6
Age	1005	45.9	63.7	1.0	7.0	28.0	56.0	365.0
Compressive Strength	1005	35.3	16.3	2.3	23.5	33.8	44.9	82.6

3.2. Exploratory Data Analysis Result

Through the Exploratory Data Analysis (EDA) framework, the distributional characteristics and the presence of statistical outliers were thoroughly evaluated, as illustrated in Figure 2. Histogram and Kernel Density Estimation (KDE) analyses revealed a highly right-skewed distribution for the concrete Age variable, characterized by a mean of 45.9 exceeding the median of 28.0. This skewness is physically justified since the dataset predominantly comprises early-age concrete samples within the standard 28-day curing period. Furthermore, visual inspection using a box plot identified several potential outliers in the Compressive Strength variable. These observations were retained because they are considered plausible measurements representing the natural variability of concrete mixtures rather than experimental errors, thereby preserving the dataset's representativeness for machine learning model development.

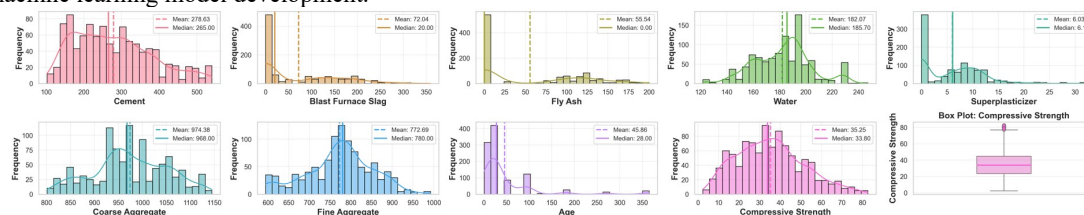


Figure 2. Combined histogram with KDE curves for all variables and box plot for the target variable

To investigate the linear relationships between the concrete constituents and the target variable, as well as to quantify multicollinearity, a statistical analysis was conducted with all results unified in Figure 3. The evaluation of the Pearson correlation heatmap highlighted a notable negative inter-feature correlation between Superplasticizer and Water. This statistical interaction aligns with concrete engineering principles, as water-reducing admixtures allow for significant water content reduction without sacrificing mixture workability. Regarding relationships with the target variable, the correlation bar chart demonstrates that Cement, Superplasticizer, and Age exhibited the strongest positive correlations with Compressive Strength, reflecting the fundamental chemistry of Calcium Silicate Hydrate (C-S-H) gel formation and the time-dependent nature of progressive cement hydration. Conversely, Water demonstrated a pronounced negative correlation with strength, a pattern qualitatively consistent with Abrams' law regarding the inverse relationship between the water-to-cement ratio and mechanical performance. This bivariate correlation is examined more rigorously via the engineered W/C and W/B ratio features in Section 3.5. Finally, building upon these correlation findings, the Variance Inflation Factor (VIF) analysis showed that although several predictors had VIF values greater than 5, all remained below the commonly accepted threshold of 10, indicating no severe multicollinearity. This level of multicollinearity is generally considered acceptable for predictive modeling. Moreover, the complex and inherently non-linear

relationships among concrete mixture constituents further support the use of advanced tree-based machine learning algorithms, which are less sensitive to multicollinearity than conventional linear regression models.

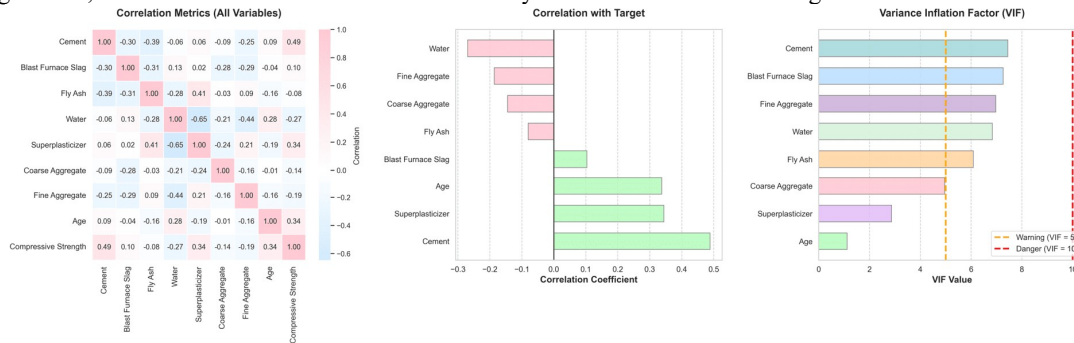


Figure 3. Pearson Correlation Heatmap, Feature-Target Correlation Bar Chart, and VIF Bar Chart

3.3. Model Evaluation and Statistical Analysis

All nine baseline models, using default hyperparameters, were initially evaluated with a train-test split to assess their predictive performance. As presented in Table 3, performance differences emerged across the three algorithmic categories. The tree-based ensemble models consistently outperformed the other approaches, with CatBoost achieving the best overall performance, followed closely by XGBoost and LightGBM. Random Forest also demonstrated strong predictive capability, whereas the standalone Decision Tree showed weaker generalization, indicating a greater tendency toward overfitting. Linear Regression and Ridge Regression produced nearly identical results, consistent with the earlier VIF analysis indicating only moderate multicollinearity among the predictors. Lasso Regression performed slightly worse, suggesting that most input variables contribute meaningful predictive information and that stronger coefficient shrinkage provides little benefit. The kernel-based SVR outperformed the linear models in the training set but remained less competitive than the tree-based ensembles on the test set.

Table 3. Performance of Nine Baseline Models

Performance Metric	Cat Boost	XG Boost	Light GBM	Random Forest	Decision Tree	SVR	Linear R.	Ridge R.	Lasso R.
R^2 (train)	0.988	0.992	0.978	0.985	0.997	0.659	0.615	0.615	0.589
RMSE (train)	1.789	1.469	2.392	1.958	0.844	9.350	9.937	9.937	10.260
MAE (train)	1.146	0.517	1.569	1.271	0.056	7.195	7.836	7.840	8.226
R^2 (test)	0.946	0.935	0.928	0.918	0.906	0.607	0.592	0.592	0.566
RMSE (test)	4.017	4.399	4.628	4.932	5.283	10.833	11.034	11.032	11.385
MAE (test)	2.429	2.816	3.096	3.437	3.102	8.210	8.749	8.749	9.005

To visually validate these findings, actual-versus-predicted scatter plots and residual analyses for four representative models are presented in Figure 4. The selected models represent the linear, kernel-based, and tree-based paradigms, with CatBoost and Random Forest included to compare two tree-based approaches. CatBoost and Random Forest exhibited predictions tightly clustered around the perfect prediction line ($y = x$) and randomly distributed residuals centered near zero, indicating well-specified models. In contrast, Linear Regression and SVR showed greater prediction dispersion, systematic underestimation of high-strength concrete, and heteroscedastic funnel-shaped residuals. This violation of the homoscedasticity assumption required for Ordinary Least Squares (OLS) regression further demonstrates the suitability of non-linear tree-based models for this structural engineering dataset.

To evaluate model stability, all models were assessed using 10-repeats of 5-fold cross-validation on the training partition. The cross-validation metrics, alongside their 95% confidence intervals (95% CI) computed from the 50 total folds, are summarized in Table 4. The overall ranking remained consistent with the initial evaluation, with CatBoost maintaining the highest average R^2 score and lowest error metrics. LightGBM and XGBoost followed closely, while Random Forest achieved slightly lower results. The confidence intervals for the tree-based ensembles indicate consistent predictive performance across different data partitions.

To determine whether the observed performance disparities reflected systematic differences rather than random variation, a Linear Mixed-Effects Model (LMM) was fitted to the repeat-level mean R^2 scores obtained from the 10×5 repeated cross-validation. The machine learning algorithm was specified as a fixed effect, whereas the cross-validation repeat was modeled as a random effect to account for between-repeat variability. The significance of the fixed algorithm effect was assessed using a Likelihood-Ratio Test (LRT). The analysis revealed

a highly significant effect of algorithm (LR = 607.14, $p < 0.0001$), rejecting the null hypothesis of equal predictive performance.

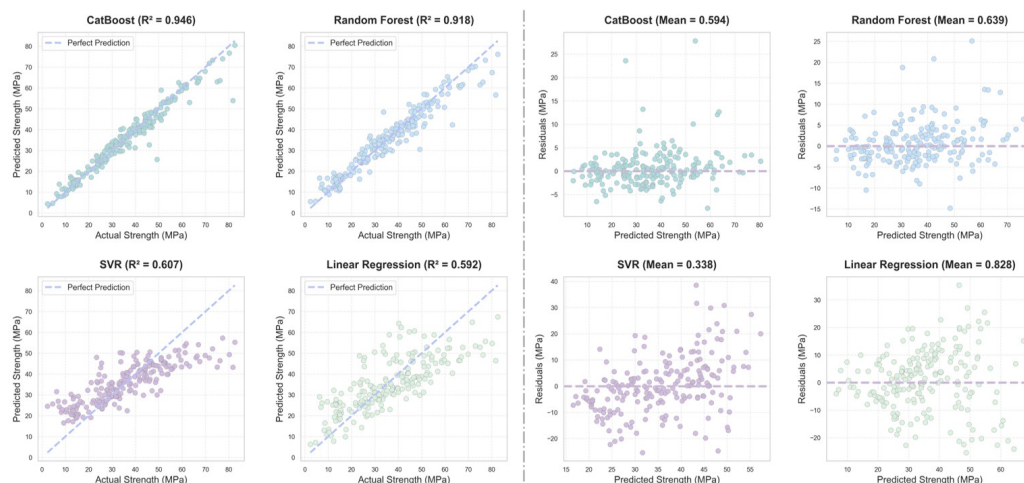


Figure 4. Actual vs. Predicted Scatter Plots and Residual Plots

Table 4 10-repeats of 5-fold Cross Validation Results

Model	R ² (95% CI)	RMSE (95% CI)	MAE (95% CI)
CatBoost	0.931 (0.927 - 0.935)	4.173 (4.053 - 4.293)	2.784 (2.713 - 2.855)
LightGBM	0.916 (0.912 - 0.920)	4.606 (4.496 - 4.717)	3.219 (3.159 - 3.280)
XGBoost	0.910 (0.906 - 0.915)	4.762 (4.640 - 4.883)	3.182 (3.117 - 3.248)
Random Forest	0.898 (0.893 - 0.902)	5.087 (4.964 - 5.211)	3.586 (3.510 - 3.663)
Decision Tree	0.820 (0.811 - 0.830)	6.726 (6.544 - 6.908)	4.496 (4.386 - 4.606)
SVR	0.600 (0.593 - 0.608)	10.089 (9.911 - 10.266)	7.914 (7.769 - 8.060)
Ridge Regression	0.596 (0.584 - 0.609)	10.113 (9.969 - 10.256)	7.970 (7.844 - 8.096)
Linear Regression	0.596 (0.583 - 0.609)	10.115 (9.971 - 10.259)	7.967 (7.841 - 8.094)
Lasso Regression	0.578 (0.569 - 0.587)	10.357 (10.221 - 10.494)	8.310 (8.179 - 8.440)

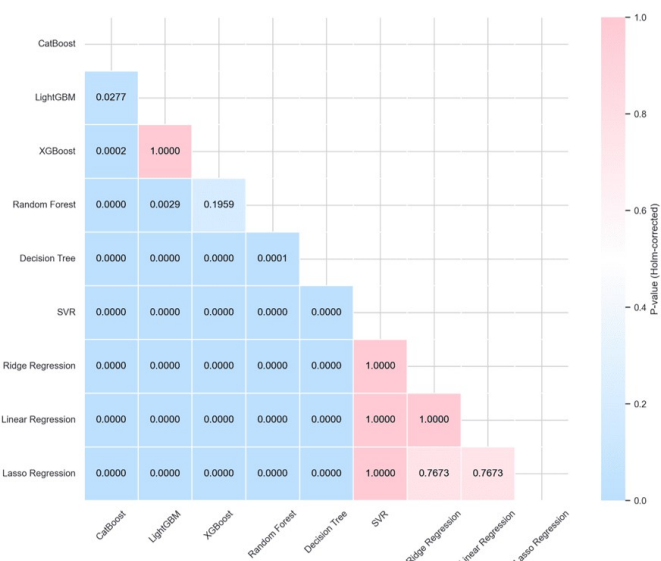


Figure 5. p – values Heatmap of Pairwise Post-Hoc Nadeau-Bengio Corrected t-Tests (Holm-Adjusted)

Following the significant omnibus test, pairwise comparisons were performed using Nadeau-Bengio corrected paired t-tests applied to the 50 fold-level score differences (10 repeats $\times$ 5 folds), with Holm-Bonferroni adjustment to control the family-wise error rate. The post-hoc results, summarized in Figure 5, indicated that most

pairwise differences were statistically significant ($p_{ho} < 0.05$). CatBoost significantly outperformed all linear and kernel-based models, as well as XGBoost ($p_{holm} = 0.0002$, *Cohen's* $dz = 2.45$) and demonstrated a smaller advantage over LightGBM ($p_{holm} = 0.0277$, $dz = 1.62$). No significant differences were observed between LightGBM and XGBoost ($p_{holm} = 1.00$), nor between Random Forest and XGBoost ($p_{ho} = 0.196$). Likewise, no statistically significant differences were found among the linear-family models (Linear Regression, Ridge Regression, SVR, Lasso Regression).

3.4. Hyperparameter Tuning and Final Model Performance

To maximize the predictive capability of the top-performing tree-based ensembles identified in the baseline evaluation, hyperparameter optimization was executed for CatBoost, LightGBM, and XGBoost. The optimization was conducted using the Optuna framework (v.4.9.0, TPE sampler with seed=42) across two sequential Bayesian optimization rounds (broad search followed by refined search) comprising 400 total trials for each model. All trials were evaluated exclusively on the training partition via an inner 5-fold cross-validation to prevent data leakage. The finalized optimal hyperparameter configurations for all three models are summarized in Table 5.

Table 5. Hyperparameter Search Space and Final Optimized Hyperparameter Configuration

Model	Hyperparameter	V1	V2	Optimized
CatBoost	depth	3-8	3-6	4
	learning_rate	1e-3-0.1	0.04621-0.13863	0.11336
	l2_leaf_reg	1.0-10.0	1.12646-3.37937	3.004822
	random_strength	1e-3-10.0	2.55942-7.67827	4.295638
	border_count	32-255	106-197	147
	bootstrap_type	Bayesian/Bernoulli	Bernoulli	Bernoulli
XGBoost	subsample	0.5-1.0	0.8216-1.0	0.946154
	max_depth	3-10	7-12	10
	learning_rate	1e-3-0.3	0.01553-0.04660	0.02428
	subsample	0.5-1.0	0.43321-0.80454	0.583407
	colsample_bytree	0.4-1.0	0.3-0.54419	0.491697
	reg_alpha	1e-4-10.0	0.3789-1.1367	0.887763
	reg_lambda	1e-4-10.0	2.24239-6.72717	5.967654
	min_child_weight	1-10	5-12	10
	gamma	0.0-5.0	0.43225-1.29674	0.521435
	num_leaves	20-300	60-111	66
LightGBM	max_depth	3-12	6-11	11
	learning_rate	1e-3-0.3	0.08482-0.25446	0.09542
	subsample	0.5-1.0	0.56675-1.0	0.848586
	colsample_bytree	0.4-1.0	0.69123-1.0	0.981649
	reg_alpha	1e-4-10.0	0.000919-0.002757	0.002151
	reg_lambda	1e-4-10.0	0.041091-0.123272	0.047798
	min_child_samples	5-100	28-67	46
	subsample_freq	1-10	1-2	1

It is acknowledged that this adaptive narrowing may produce mildly optimistic cross-validation scores. However, the final model evaluation is conducted on a strictly held-out test set that was never involved in any tuning trial. Rather than claiming the discovery of a global optimum, it is concluded that the search stabilized within the evaluated sub-space. The optimization-history plot, which tracks the progression of the objective function across the 400 trials and visually demonstrates this stabilization, is presented in Figure 6.

Configured with the optimal hyperparameters (3,000 maximum iterations and an early stopping patience of 100), the optimized models were evaluated on the unseen test set. As presented in Table 6, CatBoost remained the best-performing model among the three gradient boosting algorithms. Hyperparameter optimization improved the repeated cross-validation performance from a baseline mean R^2 of 0.931 (Table 4) to 0.940 (+0.97%), indicating enhanced predictive capability during model development. Compared with the baseline test performance, the optimized model exhibited a slight reduction, R^2 $0.946 \rightarrow 0.943$ (−0.36%), RMSE $4.017 \rightarrow 4.142$ MPa (+3.10%), and MAE $2.429 \rightarrow 2.616$ MPa (+7.67%), reflecting modest trade-off consistent with the regularizing effect of early stopping. The optimized model maintained a small training-testing performance gap (training $R^2 = 0.987$, testing $R^2 = 0.943$, $\Delta R^2 = 0.045$), indicating that the optimization produced a stable model without evidence of overfitting.

To evaluate operational reliability and understand the model's physical limitations, prediction errors were segmented into three equal-frequency strength categories (tertiles), detailed in Table 7 and visually supported by the combined diagnostic plots in Figure 7. This segmented analysis reveals that prediction accuracy gradually degrades as concrete strength increases. This performance trend is physically expected within the engineering domain, as high-strength concrete formulations are significantly more sensitive to minute variations in mix proportions, coupled with the fact that the training dataset inherently contains fewer samples in this extreme range. Finally, the learning curve analysis depicted alongside the error distribution in Figure 7 demonstrates a persistent upward trajectory in the validation score without reaching an absolute plateau. The positive gradient of this curve at the maximum available sample size implies that the final model has not yet reached its theoretical performance ceiling, suggesting that incorporating additional physical training data in future iterations is projected to yield even further improvements in predictive accuracy.

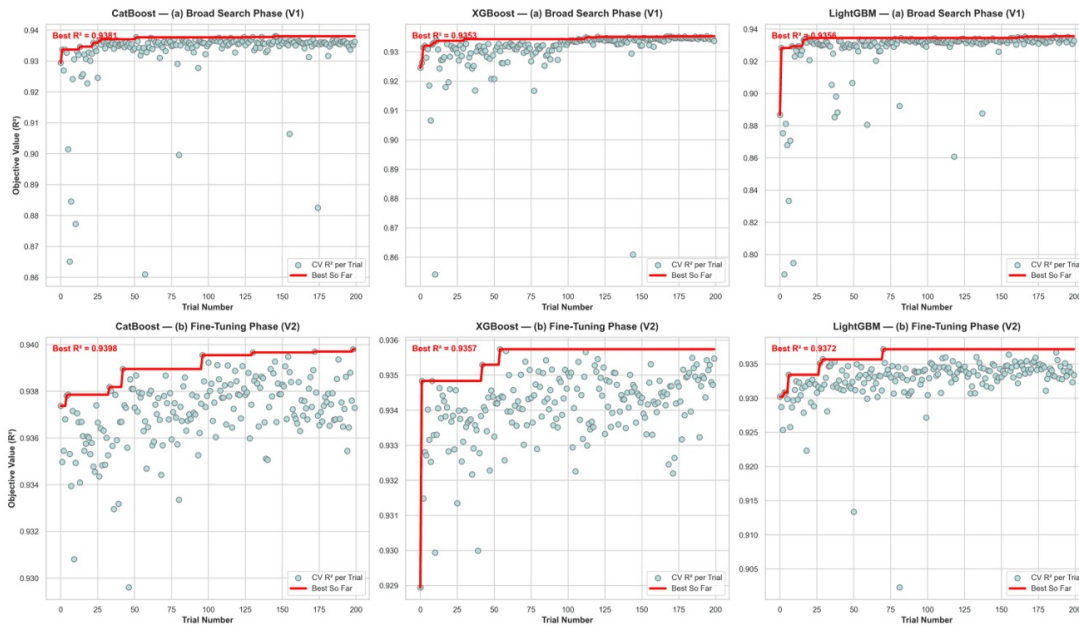


Figure 6. Optimization History of the Model

Table 6. Performance of The Optimized Model

Model	Test R ²	Test RMSE (MPa)	Test MAE (MPa)	CV R ² (mean)
CatBoost	0.943	4.142	2.616	0.94
XGBoost	0.934	4.427	2.827	0.937
LightGBM	0.932	4.507	2.868	0.934

Table 7. Error Analysis Per Strength Tertile Category for CatBoost Model

Category	Range Threshold (MPa)	MAE (MPa)	RMSE (MPa)	Count
Low	≤ 27.009	1.791	2.368	67
Medium	27.009-43.786	2.436	3.173	67
High	> 43.786	3.620	5.982	67

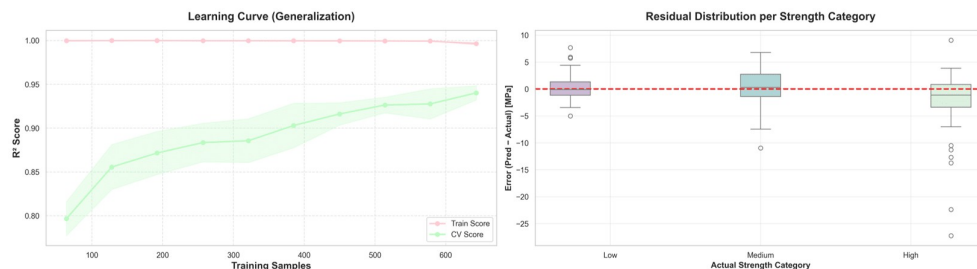


Figure 7. The Model Learning Curve and Error Distribution by Strength Category

3.5. SHAP-Based Model Interpretation

As the final stage of the research pipeline, SHAP (SHapley Additive exPlanations) analysis was applied to the optimized CatBoost model to enhance model interpretability by quantifying both global and local feature contributions. The global feature importance results, presented in Figure 8, identify Age as the dominant predictor of concrete compressive strength, followed by the engineered Water-to-Binder Ratio (W/B), Cement, Water, and the Water-to-Cement Ratio (W/C). The emergence of W/B as the second most influential feature indicates that incorporating domain-informed mix-design ratios enables the model to capture relationships not explicitly represented by the original constituent variables. This ranking is broadly consistent with previous interpretable machine learning studies, in which curing age and water-related ratio features were identified as dominant predictors of concrete compressive strength [18], while also agreeing with established concrete hydration theory [19].

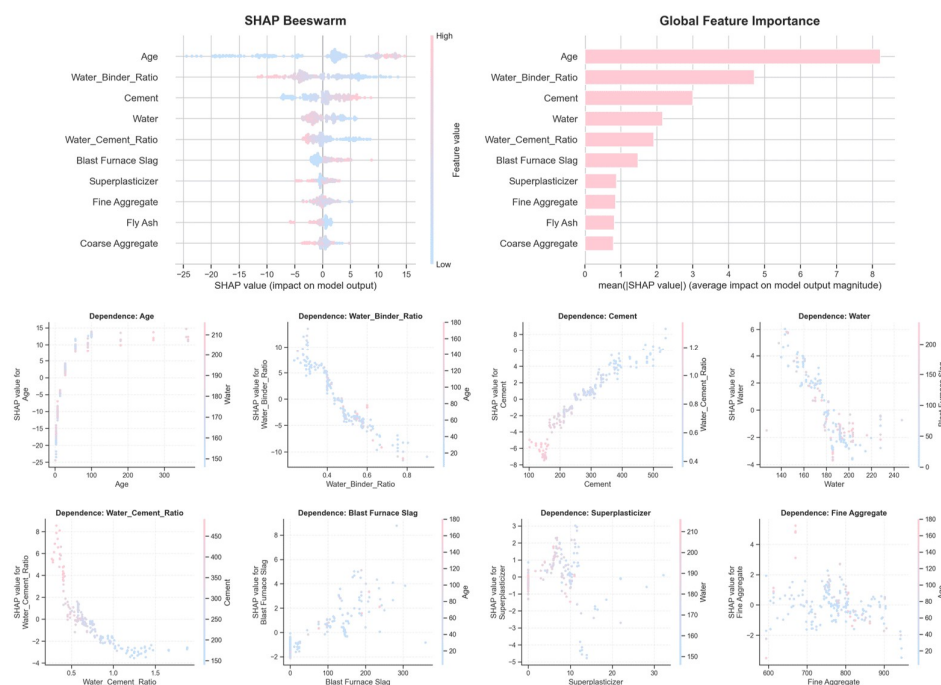


Figure 8. Global SHAP Analysis of the Optimized CatBoost Model

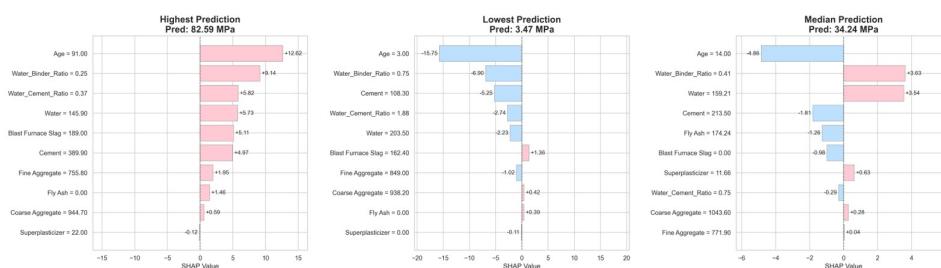


Figure 9. Local SHAP Plots for Representative Sample

The SHAP beeswarm and dependence plots further illustrate the direction and magnitude of these relationships. Increasing curing age is associated with higher predicted compressive strength as hydration advances. In contrast, both the engineered Water-to-Binder Ratio and Water-to-Cement Ratio exhibit strong monotonic negative relationships, indicating that increasing water relative to the cementitious binder is associated with lower predicted compressive strength, consistent with Abrams' law [19]. Cement content contributes positively to the predictions, whereas increasing water content generally produces negative SHAP contributions. Blast Furnace Slag exhibits a moderate positive influence, reflecting its supplementary cementitious role. The remaining variables exhibit comparatively smaller contributions, indicating that their effects become secondary once binder composition and curing conditions are considered.

Local interpretability was subsequently investigated using representative samples corresponding to the highest-, lowest-, and median-strength predictions (Figure 9). For the highest-strength concrete, long curing age together with low W/B and W/C ratios, high cement content, and appropriate supplementary cementitious materials collectively contributed positively to the prediction. Conversely, the weakest prediction was primarily associated with a high Water-to-Binder Ratio, low cement content, short curing age, and reduced binder content, all of which contributed negatively to the prediction. The median-strength sample exhibited balanced positive and negative SHAP contributions, reflecting intermediate mix characteristics. Although SHAP quantifies learned statistical associations rather than causal mechanisms, the observed feature contributions are consistent with established concrete hydration theory and mix-design principles. Notably, the engineered Water-to-Binder Ratio emerged as the second most influential predictor in the optimized CatBoost model, surpassing all individual raw constituent variables except curing age. This finding suggests that domain-informed feature engineering improved the model's representation of concrete mix-design characteristics.

3.6. Discussion and Research Limitations

Although all three top-performing tree-based ensemble models were independently optimized and compared (Table 6), the performance margin between CatBoost and its closest competitors ($\Delta Test R^2 \approx 0.009 - 0.011$) was not directly evaluated using pairwise statistical tests on the optimized models. Consequently, while CatBoost achieved the highest predictive performance, the practical significance of its advantage after hyperparameter optimization warrants further investigation. Second, the analysis was constrained to a single benchmark dataset, limiting the assessment of model generalizability across different geographic regions, material sources, and concrete production conditions. Furthermore, although Optuna was employed for Bayesian hyperparameter optimization, the progressive refinement of the search space across optimization stages required manual definition of parameter boundaries, introducing a limited degree of researcher judgment regarding the explored hyperparameter ranges [44]. Finally, the relatively modest test-set size restricts the statistical precision of performance estimates within specific concrete strength ranges [10], [11].

To address these limitations, future studies should validate the proposed framework using diverse external datasets representing different raw materials, environmental conditions, and construction practices to better assess model robustness and transferability [45]. Beyond tree-based ensemble methods, investigating deep learning architectures such as TabNet and Multilayer Perceptrons (MLPs) may further improve the modeling of complex nonlinear relationships. In addition, deploying the optimized prediction model as a web-based decision-support system or interactive graphical user interface (GUI) would facilitate practical implementation for real-time concrete mix design and construction-site applications [1].

4. CONCLUSION

This study established a comprehensive framework for predicting concrete compressive strength by integrating comparative machine learning, Bayesian hyperparameter optimization, domain-informed feature engineering, and SHAP-based model interpretation. The evaluation of nine regression algorithms demonstrated that tree-based ensemble methods consistently outperformed linear and kernel-based approaches, with CatBoost achieving the best overall predictive performance. Statistical analysis using a Linear Mixed-Effects Model with Likelihood-Ratio Test (LRT) and Holm-adjusted Nadeau-Bengio corrected pairwise comparisons confirmed that the observed performance differences among the evaluated models were statistically significant.

Bayesian optimization using Optuna further refined the CatBoost model, yielding a final test R^2 of 0.943, RMSE of 4.142 MPa, and MAE of 2.616 MPa. Although the optimized model exhibited a slight reduction in held-out test performance compared with the baseline configuration, this behavior is consistent with the regularizing effect of early stopping and indicates a more stable model with good generalization characteristics. Because the final evaluation was conducted on a single held-out dataset, these performance metrics should be interpreted as evidence of local predictive capability rather than definitive proof of broad generalizability.

The SHAP analysis further enhanced model transparency by revealing that Age and the engineered Water-to-Binder Ratio (W/B) were the two most influential predictors, followed by Cement, Water, and the Water-to-Cement Ratio (W/C). These findings indicate that incorporating domain-informed mix-design ratios enabled the model to capture relationships consistent with established concrete hydration theory and Abrams' law, thereby improving both interpretability and engineering relevance.

Overall, the proposed framework provides an accurate and interpretable decision-support tool for preliminary concrete mix design and strength estimation, while complementing rather than replacing standardized laboratory testing. Future research should focus on validating the proposed framework using independent external datasets representing diverse material sources, environmental conditions, and production practices to further evaluate its robustness and generalizability before practical deployment.

REFERENCES

- [1] M. Shaaban, M. Amin, S. Selim, and I. M. Riad, "Machine learning approaches for forecasting compressive strength of high-strength concrete," *Sci. Rep.*, vol. 15, no. 1, p. 25567, 2025, doi: 10.1038/s41598-025-10342-1.
- [2] J. Huang, M. M. S. Sabri, D. V. Ulrikh, M. Ahmad, and K. A. M. Alsaffar, "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, p. 4193, 2022, doi: 10.3390/ma15124193.
- [3] H. Fu, X. Zhou, P. Xu, and D. Sun, "Prediction of Compressive Strength of Concrete Using Explainable Machine Learning Models," *Materials*, vol. 18, no. 21, p. 5009, 2025, doi: 10.3390/ma18215009.
- [4] A. K. Sah and Y.-M. Hong, "Performance Comparison of Machine Learning Models for Concrete Compressive Strength Prediction," *Materials*, vol. 17, no. 9, p. 2075, 2024, doi: 10.3390/ma17092075.
- [5] 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, doi: 10.1016/S0008-8846(98)00165-3.
- [6] I. N. Fathy, H. A. Dahish, M. K. Alkharisi, A. A. Mahmoud, and H. E. E. Fouad, "Predicting the compressive strength of concrete incorporating waste powders exposed to elevated temperatures utilizing machine learning," *Sci. Rep.*, vol. 15, no. 1, p. 25275, 2025, doi: 10.1038/s41598-025-11239-9.
- [7] X. Chen, X. Zhang, and W.-Z. Chen, "Advanced Predictive Modeling of Concrete Compressive Strength and Slump Characteristics: A Comparative Evaluation of BPNN, SVM, and RF Models Optimized via PSO," *Materials*, vol. 17, no. 19, p. 4791, 2024, doi: 10.3390/ma17194791.
- [8] M. F. Javed, M. Fawad, R. Lodhi, T. Najeh, and Y. Gamil, "Forecasting the strength of preplaced aggregate concrete using interpretable machine learning approaches," *Sci. Rep.*, vol. 14, no. 1, p. 8381, 2024, doi: 10.1038/s41598-024-57896-0.
- [9] S. Elhishi, A. M. Elashry, and S. El-Metwally, "Unboxing machine learning models for concrete strength prediction using XAI," *Sci. Rep.*, vol. 13, no. 1, p. 19892, 2023, doi: 10.1038/s41598-023-47169-7.
- [10] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Sci. Rep.*, vol. 14, no. 1, p. 6086, 2024, doi: 10.1038/s41598-024-56706-x.
- [11] J. Liu and Y. Xu, "T-Friedman Test: A New Statistical Test for Multiple Comparison with an Adjustable Conservativeness Measure," *International Journal of Computational Intelligence Systems*, vol. 15, no. 1, p. 29, 2022, doi: 10.1007/s44196-022-00083-8.
- [12] J. M. Gorriz, J. Ramirez, F. Segovia, C. Jimenez-Mesa, F. J. Martinez-Murcia, and J. Suckling, "Statistical agnostic regression: A machine learning method to validate regression models," *J. Adv. Res.*, vol. 80, pp. 503–533, 2026, doi: 10.1016/j.jare.2025.04.026.
- [13] L. W. Rizkallah, "Enhancing the performance of gradient boosting trees on regression problems," *J. Big Data*, vol. 12, no. 1, p. 35, 2025, doi: 10.1186/s40537-025-01071-3.
- [14] J. Ma *et al.*, "A comprehensive comparison among metaheuristics (MHs) for geohazard modeling using machine learning: Insights from a case study of landslide displacement prediction," *Eng. Appl. Artif. Intell.*, vol. 114, p. 105150, 2022, doi: 10.1016/j.engappai.2022.105150.
- [15] R. S. Ajin, S. Segoni, and R. Fanti, "Optimization of SVR and CatBoost models using metaheuristic algorithms to assess landslide susceptibility," *Sci. Rep.*, vol. 14, no. 1, p. 24851, 2024, doi: 10.1038/s41598-024-72663-x.
- [16] P.-O. Côté, A. Nikanjam, N. Ahmed, D. Humeniuk, and F. Khomh, "Data cleaning and machine learning: a systematic literature review," *Automated Software Engineering*, vol. 31, no. 2, p. 54, 2024, doi: 10.1007/s10515-024-00453-w.
- [17] F. C. Oettl, J. F. Oeding, R. Feldt, C. Ley, M. T. Hirschmann, and K. Samuelsson, "The artificial intelligence advantage: Supercharging exploratory data analysis," *Knee Surgery, Sports Traumatology, Arthroscopy*, vol. 32, no. 11, pp. 3039–3042, 2024, doi: 10.1002/ksa.12389.
- [18] Y. Zhang, W. Ren, Y. Chen, Y. Mi, J. Lei, and L. Sun, "Predicting the compressive strength of high-performance concrete using an interpretable machine learning model," *Sci. Rep.*, vol. 14, no. 1, p. 28346, 2024, doi: 10.1038/s41598-024-79502-z.
- [19] A. M. Neville, *Properties of Concrete*, 5th ed. Harlow, UK: Prentice Hall, 2012.
- [20] D. Sofia and P. Sekarpuji, "Penerapan Metode Stacking Ensemble Untuk Analisis Sentimen Pada Ulasan Aplikasi Ruang Guru," *IDEALIS: InDonEsiA Journal Information System*, vol. 8, no. 2, pp. 248–257, 2025, doi: 10.36080/ideal.v8i2.3559.
- [21] P. Li, Z. Chen, X. Chu, and K. Rong, "DiffPrep: Differentiable Data Preprocessing Pipeline Search for Learning over Tabular Data," *Proceedings of the ACM on Management of Data*, vol. 1, no. 2, pp. 1–26, 2023, doi: 10.1145/3589328.
- [22] M. A. Bouke and A. Abdullah, "An empirical study of pattern leakage impact during data preprocessing on machine learning-based intrusion detection models reliability," *Expert Syst. Appl.*, vol. 230, p. 120715, 2023, doi: 10.1016/j.eswa.2023.120715.
- [23] P. Probst, A.-L. Boulesteix, and B. Bischl, "Tunability: importance of hyperparameters of machine learning algorithms," *Journal of Machine Learning Research*, vol. 20, no. 53, pp. 1–32, 2019, Accessed: Jul. 14, 2026. [Online]. Available: <https://jmlr.org/papers/volume20/18-444/18-444.pdf>
- [24] F. Shehzad, T. Breuer, M. Maistro, and D. Jannach, "'We Share Our Code Online': Why This Is Not Enough to Ensure Reproducibility and Progress in Recommender Systems Research," in *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, Prague, Czech Republic: Association for Computing Machinery, 2025, pp. 884–893. doi: 10.1145/3705328.3748157.

- [25] H. Ahmed and J. Lofstead, "Managing Randomness to Enable Reproducible Machine Learning," in *Proceedings of the 5th International Workshop on Practical Reproducible Evaluation of Computer Systems*, Minneapolis, MN, USA, 2022, pp. 15–20. doi: 10.1145/3526062.3536353.
- [26] H. Semmelrock *et al.*, "Reproducibility in machine-learning-based research: Overview, barriers, and drivers," *AI Mag.*, vol. 46, no. 2, p. e70002, 2025, doi: 10.1002/aaai.70002.
- [27] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Comput. Sci.*, vol. 7, p. e623, 2021, doi: 10.7717/peerj-cs.623.
- [28] W. G. Mokodaser, H. Koapaha, and S. I. Adam, "Model Random Forest Data Historis Multivariat Untuk Prediksi Pendapatan Asuransi," *IDEALIS: InDonEsiA journal Information System*, vol. 8, no. 2, pp. 267–276, 2025, doi: 10.36080/idealis.v8i2.3512.
- [29] Y. M. Jaradat, M. A. Alia, M. Z. Masoud, A. A. Manasrah, I. A. Jannoud, and O. Alheyasat, "Beyond One-Size-Fits-All: Comparing and Selecting Regression Metrics for Robust Model Assessment," in *2025 12th International Conference on Information Technology (ICIT)*, Amman, Jordan, 2025, pp. 416–422. doi: 10.1109/ICIT64950.2025.11049268.
- [30] I. Hussain, K. B. Ching, C. Utraphan, K. G. Tay, and A. Noor, "Evaluating machine learning algorithms for energy consumption prediction in electric vehicles: A comparative study," *Sci. Rep.*, vol. 15, no. 1, p. 16124, 2025, doi: 10.1038/s41598-025-94946-7.
- [31] W. Li, D. Cook, E. Tanaka, and S. VanderPlas, "A Plot is Worth a Thousand Tests: Assessing Residual Diagnostics with the Lineup Protocol," *Journal of Computational and Graphical Statistics*, vol. 33, no. 4, pp. 1497–1511, 2024, doi: 10.1080/10618600.2024.2344612.
- [32] R. T. Nakatsu, "Validation of machine learning ridge regression models using Monte Carlo, bootstrap, and variations in cross-validation," *Journal of Intelligent Systems*, vol. 32, no. 1, p. 20220224, 2023, doi: 10.1515/jisys-2022-0224.
- [33] S. Bates, T. Hastie, and R. Tibshirani, "Cross-Validation: What Does It Estimate and How Well Does It Do It?," *J. Am. Stat. Assoc.*, vol. 119, no. 546, pp. 1434–1445, 2024, doi: 10.1080/01621459.2023.2197686.
- [34] V. A. Brown, "An Introduction to Linear Mixed-Effects Modeling in R," *Adv. Methods Pract. Psychol. Sci.*, vol. 4, no. 1, p. 2515245920960351, 2021, doi: 10.1177/2515245920960351.
- [35] Z. Yu, M. Guindani, S. F. Grieco, L. Chen, T. C. Holmes, and X. Xu, "Beyond t test and ANOVA: applications of mixed-effects models for more rigorous statistical analysis in neuroscience research," *Neuron*, vol. 110, no. 1, pp. 21–35, 2022, doi: 10.1016/j.neuron.2021.10.030.
- [36] C. Nadeau and Y. Bengio, "Inference for the Generalization Error," *Mach. Learn.*, vol. 52, no. 3, pp. 239–281, 2003, doi: 10.1023/A:1024068626366.
- [37] T. G. Dietterich, "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms," *Neural Comput.*, vol. 10, no. 7, pp. 1895–1923, 1998, doi: 10.1162/089976698300017197.
- [38] R. R. Bouckaert and E. Frank, "Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms," in *Advances in Knowledge Discovery and Data Mining*, Sydney, Australia, 2004, pp. 3–12. doi: https://doi.org/10.1007/978-3-540-24775-3_3.
- [39] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, AK, USA, 2019, pp. 2623–2631. doi: 10.1145/3292500.3330701.
- [40] S. Ali, F. Akhlaq, A. S. Imran, Z. Kastrati, S. M. Daudpota, and M. Moosa, "The enlightening role of explainable artificial intelligence in medical & healthcare domains: A systematic literature review," *Comput. Biol. Med.*, vol. 166, p. 107555, 2023, doi: 10.1016/j.compbiomed.2023.107555.
- [41] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, in NIPS'17. Long Beach, California, USA, 2017, pp. 4768–4777.
- [42] S. M. Lundberg *et al.*, "From local explanations to global understanding with explainable AI for trees," *Nat. Mach. Intell.*, vol. 2, no. 1, pp. 56–67, 2020, doi: 10.1038/s42256-019-0138-9.
- [43] Z. Wang, H. Liu, M. N. Amin, K. Khan, M. T. Qadir, and S. A. Khan, "Optimizing machine learning techniques and SHapley Additive exPlanations (SHAP) analysis for the compressive property of self-compacting concrete," *Mater. Today Commun.*, vol. 39, p. 108804, 2024, doi: 10.1016/j.mtcomm.2024.108804.
- [44] P. Novello, G. Poëtto, D. Lugato, and P. M. Congedo, "Goal-Oriented Sensitivity Analysis of Hyperparameters in Deep Learning," *J. Sci. Comput.*, vol. 94, no. 3, p. 45, 2023, doi: 10.1007/s10915-022-02083-4.
- [45] Y. T. Altuncı, "A Comprehensive Study on the Estimation of Concrete Compressive Strength Using Machine Learning Models," *Buildings*, vol. 14, no. 12, p. 3851, 2024, doi: 10.3390/buildings14123851.