Sustainable Structures
ISSN: 2789-3111 (Print); ISSN: 2789-312X (Online)
http://www.sustain-dpl.com/journal/sust
DOI: 10.54113/j.sust.2024.000056

ORIGINAL ARTICLE

Evaluating suitability of regression models in small data regimes using concrete with recycled copper tailings as a case study

Ahed Habib[a,*], Samer Barakat[b], Samir Dirar[c], Salah Al-Toubat[b], Zaid A. Al-Sadoon[b]

[a] Research Institute of Sciences and Engineering, University of Sharjah, Sharjah, United Arab Emirates.
[b] Department of Civil and Environmental Engineering, University of Sharjah, Sharjah, United Arab Emirates.
[c] Department of Architectural Engineering, University of Sharjah, Sharjah, United Arab Emirates.
[*]Corresponding Author: Ahed Habib. Email: ahabib@sharjah.ac.ae.

Abstract: The utilization of regression models for the prediction of construction material properties is well-established, yet their performance when applied to small datasets is still unclear. This study investigates the performance of different regression models combined with various data preprocessing techniques in contexts where data is limited. Specifically, the research focuses on evaluating the suitability of five regression models across nine different data processing scenarios using concrete with recycled copper tailing as a case study. This study aims to determine which combinations of regression models and preprocessing methods yield the most accurate predictions in small data regimes. This research is motivated by the necessity to enhance prediction reliability in the field of construction materials, where experimental data can often be scarce or costly to obtain. Within the study context, a dataset comprising 21 experimental specimens is used to evaluate the performance of the models on various concrete properties, including fresh density, compressive strength, flexural strength, pull-off strength, abrasion resistance, water penetration, rapid chloride ion permeability, and air permeability. Through rigorous evaluation involving a 10-fold cross-validation process to verify accuracy, the research demonstrates that selecting the optimal regression model and data preprocessing technique selection substantially improves prediction outcomes, even with limited data. The findings highlight the importance of this research, suggesting that even small datasets, when handled correctly, can provide robust insights.

Keywords: Regression models; small data regimes; copper tailing concrete; multivariable regression; data preprocessing

1 Introduction

The introduction of regression models into the field of construction materials has significantly advanced the predictive capabilities concerning the properties of various materials [1-4]. Historically, the application of these models has been extensively explored and refined, especially with conventional datasets of considerable size [5-9]. However, the robustness of these models in scenarios where data is scarce remains a significant challenge. Existing literature on regression-based modeling of construction material properties primarily explored the capabilities and effectiveness of machine learning algorithms [10-15]. They often demonstrated the application of regression models across various scenarios without specifically addressing the influence of dataset size. Researchers like Steyerberg et al. [16], Habib et al. [17], and Koya et al. [18] investigated the technical performance and refinement of these models and

Received: 24 May 2024; Received in revised form: 24 July 2024; Accepted: 8 October 2024

showed their potential to accurately predict outcomes based on the features and relationships within the datasets used. However, these discussions generally do not differentiate the performance impacts related to the size of the dataset, leaving a gap in understanding how these models perform with smaller, less comprehensive datasets. Previous studies have discussed various methods for expanding small datasets, such as data augmentation [19-21] and synthetic data generation [22-24]. Nevertheless, their use induces a new degree of uncertainty, which is generally unfavorable [25, 26]. On the other hand, given the cost and limitations in obtaining experimental data, there is a need for research on the applicability and reliability of regression models when the experimental data availability is limited. Accordingly, this study aims to answer the following questions: How well do various regression models perform in small data regimes?

Recently, the use of copper tailings in concrete has attracted attention as part of the industry's aim to contribute to recycling solid wastes [27-31]. Copper tailings are a by-product of copper mining. They can be recycled as a substitute for traditional sand or as an additive in concrete [32-34]. Studies have shown varied results in terms of strength and durability when copper tailings are used [35-39]. In this regard, the use of copper tailings has been proven experimentally to affect the concrete properties at low replacement ratios while also presenting challenges depending on their amount [40-43]. On the other hand, the literature highlights a significant gap regarding the development of regression models for estimating the properties of concrete mixtures that incorporate copper tailings. This study is set apart by addressing several key aspects that are underexplored in existing research. Firstly, it evaluates the impact of regression model selection in small data regimes with multicollinearity (as typically expected in recycled aggregated concrete datasets) on the estimation performance. Secondly, it explores the combination of various data preprocessing techniques with regression models to enhance the predictability of concrete properties with limited data availability. Thirdly, this research focuses on copper recycled concrete and develops models for predicting a variety of mechanical and durability properties where the literature lacks a similar one. Within the study context, the investigations seek to identify the optimal combinations of models and techniques that enhance prediction accuracy in small data scenarios, focusing on a range of concrete properties such as strength, durability, and permeability. The use of a small dataset of 21 experimental specimens for this investigation provides a challenging yet realistic case. This research is crucial for advancing the field of construction materials, particularly in developing estimation models where experimental data may be limited or expensive to obtain.

	Cement (kg/m^3)	Water (kg/m^3)	Coarse Aggregate (kg/m^3)	Fine Aggregate (kg/m^3)	Copper Tailing (kg/m^3)	Admixture (kg/m^3)	Compaction Factor	Fresh Concrete Density (kg/m^3)	Compressive Strength (MPa)	Flexural Strength (MPa)	Pull-off Strength (MPa)	Depth of Abrasion (mm)	Depth of Water Penetration (mm)	Charge Passed in Coulombs	Air Permeability Index (Bar/min)	
Cement (kg/m^3)	1.00	0.22	-0.46	-0.02	-0.02	-0.38	-0.61	-0.06	0.17	0.18	-0.03	0.12	0.19	-0.32	0.15	High Positive Correlation
Water (kg/m^3)	0.22	1.00	-0.97	-0.12	-0.04	-0.53	0.24	-0.77	-0.63	0.34	-0.19	0.13	0.22	0.65	0.56	
Coarse Aggregate (kg/m^3)	-0.46	-0.97	1.00	0.11	0.04	0.58	-0.05	0.71	0.53	-0.35	0.18	-0.15	-0.25	-0.50	-0.54	
Fine Aggregate (kg/m^3)	-0.02	-0.12	0.11	1.00	-0.98	-0.60	-0.19	-0.46	0.29	0.45	0.02	-0.75	-0.92	-0.42	0.71	
Copper Tailing (kg/m^3)	-0.02	-0.04	0.04	-0.98	1.00	0.71	0.16	0.61	-0.21	-0.52	0.00	0.75	0.90	0.33	-0.81	
Admixture (kg/m^3)	-0.38	-0.53	0.58	-0.60	0.71	1.00	0.15	0.83	0.14	-0.58	0.16	0.34	0.49	0.00	-0.80	
Fresh Concrete Density (kg/m^3)	-0.06	-0.77	0.71	-0.46	0.61	0.83	-0.15	1.00	0.36	-0.57	0.14	0.39	0.39	-0.34	-0.88	
Compressive Strength (MPa)	0.17	-0.63	0.53	0.29	-0.21	0.14	-0.56	0.36	1.00	0.04	0.54	-0.52	-0.28	-0.79	-0.10	
Flexural Strength (MPa)	0.18	0.34	-0.35	0.45	-0.52	-0.58	-0.14	-0.57	0.04	1.00	0.08	-0.38	-0.38	-0.11	0.53	
Pull-off Strength (MPa)	-0.03	-0.19	0.18	0.02	0.00	0.16	-0.14	0.14	0.54	0.08	1.00	-0.32	0.08	-0.54	-0.06	
Depth of Abrasion (mm)	0.12	0.13	-0.15	-0.75	0.75	0.34	0.19	0.39	-0.52	-0.38	-0.32	1.00	0.69	0.37	-0.62	
Depth of Water Penetration (mm)	0.19	0.22	-0.25	-0.92	0.90	0.49	0.11	0.39	-0.28	-0.38	0.08	0.69	1.00	0.36	-0.57	
Charge Passed in Coulombs	-0.32	0.65	-0.50	-0.42	0.33	0.00	0.60	-0.34	-0.79	-0.11	-0.54	0.37	0.36	1.00	0.10	
Air Permeability Index (Bar/min)	0.15	0.56	-0.54	0.71	-0.81	-0.80	0.00	-0.88	-0.10	0.53	-0.06	-0.62	-0.57	0.10	1.00	High Negative Correlation

Fig. 1. Correlation analysis between the inputs and the outputs.

2 Materials and Method

2.1 Investigated Parameters and Developed Database

This study explores the effectiveness of employing regression models alongside various data preprocessing techniques to estimate the properties of concrete that incorporate recycled copper tailings using a small dataset. Specifically, the experimental results from Thomas et al. [27] were collected and used as the basis for the numerical analyses. The dataset includes 21 distinct mixtures, each tested for eight different properties, such as strength and durability. The descriptive statistics for the selected data are detailed in 错误!未找到引用源。. Additionally, a correlation analysis between the input variables and the outputs is depicted in **Fig. 1**. In general, the concrete experimental results used herein were all tested at 28 days except for the fresh concrete density. The specimen shapes and sizes utilized in the reference study for the compressive strength test is a cube of 100×100×100 mm, the pull-off and flexural strength test is a beam of 100×100×500 mm, the abrasion test is a cube of 100×100×100 mm, the water permeability test is a cube of 100×100×100 mm, the rapid chloride permeability test is a cylinder of 102 mm diameter and 51 mm in height, and air permeability is a cube of 150×150×150 mm. All the details regarding the experimental program, the data measurement procedure, and the findings are provided in detail in the reference study by Thomas et al. [27].

Table 1. Descriptive statistics for the dataset used in this study

	Number of Observations	Mean	Standard Deviation	Minimum	First Quartile	Median	Third Quartile	Maximum
Cement (kg/m^3)	21	393.44	14.71	380.00	380.00	387.00	413.33	413.33
Water (kg/m^3)	21	176.93	16.12	154.80	154.80	186.00	190.00	190.00
Coarse Aggregate (kg/m^3)	21	1159.59	30.03	1133.43	1133.43	1144.84	1200.51	1200.51
Fine Aggregate (kg/m^3)	21	452.39	116.45	272.02	353.41	454.10	555.88	640.75
Copper Tailing (kg/m^3)	21	202.24	142.80	0.00	65.40	194.67	333.79	432.02
Admixture (kg/m^3)	21	0.60	0.42	0.00	0.25	0.57	0.95	1.20
Fresh Concrete Density (kg/m^3)	21	2296.13	47.36	2199.96	2268.73	2294.85	2319.81	2384.95
Compressive Strength (MPa)	21	35.94	2.98	30.98	34.18	36.15	38.61	41.07
Flexural Strength (MPa)	21	4.47	0.22	4.10	4.32	4.53	4.65	4.93
Pull-off Strength (MPa)	21	2.07	0.22	1.74	1.88	2.05	2.17	2.50
Depth of Abrasion (mm)	21	1.42	0.22	1.11	1.24	1.39	1.53	1.96
Depth of Water Penetration (mm)	21	6.20	0.92	4.57	5.51	6.50	7.03	7.35
Charge Passed in Coulombs	21	535.07	58.73	431.51	498.63	533.15	586.85	648.22
Air Permeability Index (Bar/min)	21	0.19	0.03	0.13	0.16	0.18	0.22	0.25

2.2 Regression Models

Multivariate linear regression models are crucial in data analysis and predictive modeling. Common regression techniques include multiple linear regression, ridge regression, lasso regression, ElasticNet regression, and Bayesian ridge regression. Each of these techniques has its unique characteristics, robustness, and practical utility. The rationale for selecting these specific cases compared to advanced machine learning models comes from the need to maintain simplicity and robustness in the face of a small dataset. The primary goal herein is to avoid complex models with numerous hyperparameters and coefficients, which can compromise the robustness of the training process given the limited data. Linear regression variants were chosen for their straightforward nature and their ability to improve training scenarios and core assumptions while using the simplicity of their base model. Additionally, to account for potential nonlinearity, the study incorporated data

preprocessing techniques based on polynomials and other nonlinear forms, addressing nonlinearity indirectly without resorting to more complex models. This approach ensures a balanced and methodologically sound analysis, focusing on enhancing prediction reliability within the constraints of a small dataset.

Multiple linear regression (MLR) provides a core methodology for correlating a single dependent variable with several independent variables. This model is especially important in interdisciplinary studies where the interdependence among variables is significant. Tranmer and Elliot [44] highlight its applicability, which is mathematically expressed in (1.

$$Y = \beta X + \varepsilon \tag{1}$$

where $Y = [y_1, \dots, y_n]^T$ is the output; $X = \begin{bmatrix} x_{1,1} & \dots & x_{1,k} \\ \vdots & \ddots & \vdots \\ x_{n,k} & \dots & x_{n,k} \end{bmatrix}$ is the input matrix for n observations and k inputs; $\beta = [\beta_1, \dots, \beta_k]^T$ represents the coefficients to be estimated; $\varepsilon = [\varepsilon_1, \dots, \varepsilon_k]^T$ represents the random errors.

The ordinary least squares estimator, in this case, is given in (2).

$$\beta = (X^T X)^{-1} X^T Y \tag{2}$$

Ridge Regression modifies MLR by incorporating a regularization term into the loss function, aiming to control model complexity. As McDonald [45] points out, this strategy is effective in tackling multicollinearity and improving predictive precision by contracting the regression coefficients. This adaptation is crucial when independent variables exhibit correlation, with β coefficients modified accordingly, as shown in (3).

$$\hat{\beta}^* = \left(X^T X + \alpha I_p\right)^{-1} X^T Y \tag{3}$$

where $\hat{\beta}^*$ is the ridge estimator; $\alpha > 0$ is the complexity parameter that controls the amount of shrinkage and ensure that $E\left[(\hat{\beta}^* - \beta)^T(\hat{\beta}^* - \beta)\right] < E\left[(\hat{\beta} - \beta)^T(\hat{\beta} - \beta)\right]$; I_p is the identity matrix.

The ridge regression also solve issues inherent in ordinary least squares by penalizing coefficient magnitude to optimize a penalized residual sum squared through the ℓ_2 regularization norm as follows:

$$\min_{\beta} = \|\beta X - y\|_2^2 + \alpha\|\beta\|_2^2 \tag{4}$$

Expanding on ridge regression, lasso regression introduces an ability to reduce certain coefficients to zero, thereby streamlining variable selection, as described by Ranstam and Cook [46]. This feature is exceptionally useful in scenarios involving high-dimensional datasets, where it simplifies the model to avert overfitting and enhance interpretability. The optimization criterion involves minimizing the least-squares penalty augmented by $\alpha\|\beta\|_1$, where α is a fixed scalar and $\|\beta\|_1$ is the absolute norm ℓ_1 of the coefficient vector.

$$\min_{\beta} = \frac{1}{2n_{samples}}\|\beta X - y\|_2^2 + \alpha\|\beta\|_1 \tag{5}$$

ElasticNet Regression considers the penalties of both ridge and lasso regression into a unified penalty framework. This amalgamation captures the strengths of each approach, facilitating a balance between variable selection and the correction of multicollinearity. It is particularly effective in datasets where predictor correlations are high or when predictors outnumber observations, addressing the constraints of applying lasso or ridge regression singly. The objective function herein can be defined as follows:

$$\min_{\beta} = \frac{1}{2n_{samples}}\|\beta X - y\|_2^2 + \alpha\rho\|\beta\|_1 + \frac{\alpha(1-\rho)}{2}\|\beta\|_2^2 \tag{6}$$

where ρ is a parameter that is utilized to control the convex combination of ℓ_1 and ℓ_2.

Bayesian Ridge Regression adopts a probabilistic approach by imposing a prior distribution on the coefficients, (7), as elaborated by Zhao et al. [47]. This Bayesian framework enables coefficient estimation under conditions of uncertainty, proving invaluable in cases affected by time lags or irregular data signals.

$$P(\beta, \Sigma_\varepsilon | Y, X) \propto \ P(Y|X, \beta, \Sigma_\varepsilon) P(\beta, \Sigma_\varepsilon) \tag{7}$$

Bedoui and Lazar [48] utilized this model by integrating an empirical Bayesian approach with a ridge penalty as follows:

$$\min(\|\beta X - Y\|_2{}^2 + \alpha \|\beta\|_2{}^2) \tag{8}$$

2.3 Data Preprocessing Models

Nowadays, a variety of data preprocessing techniques are available to enhance both the predictive accuracy and interpretability of regression models. This section explores several widely employed methods, including standardization, normalization, discretization, polynomial feature transformation, principal component analysis (PCA), kernel PCA, backward elimination, and forward selection. Each method provides distinct benefits and is suited for specific scenarios within data analysis.

Standardization adjusts data to have a mean of zero and a standard deviation of one. The standardization formula is as follows:

$$Z = \frac{x - \mu}{\sigma} \tag{9}$$

where Z is the standardized value; x is the original value; μ is the mean; σ is the standard deviation. This approach is particularly advantageous when data features differ in units or scale, as it mitigates these discrepancies and facilitates the optimal performance of algorithms that presume normally distributed data, such as logistic regression and support vector machines. Standardization aims to enhance model training by aligning features to a uniform scale, addressing the issue of multicollinearity, which is often present in small datasets and can hinder model performance.

Normalization modifies the data dimensions to ensure the range is between 0 and 1, scaling data based on the minimum and maximum values of the features. The formula generally used for normalization is as follows:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{10}$$

where x' is the normalized value; x is the original value; $\min(x)$ is the minimum value; $\max(x)$ is the maximum value. This technique is useful for models sensitive to data magnitude, including neural networks and k-nearest neighbors. Normalization ensures that each feature contributes proportionately to the final prediction, preventing any single feature from dominating due to its larger range. This is particularly important in small datasets where the presence of outliers or extreme values can disproportionately influence model outcomes.

Discretization converts continuous variables into discrete ones by establishing a series of contiguous intervals within the variables' range. This technique is valuable for transforming numerical data into categorical variables, aiding in the modeling of complex relationships between variables, and enhancing model interpretability and robustness. In small datasets, discretization can help simplify the model and reduce the risk of overfitting by limiting the number of unique values a variable can take.

Polynomial feature transformation generates features derived from existing variables, which are either powers or interaction terms of the original set. This method is particularly useful when a nonlinear relationship between predictors and the outcome is anticipated. For example, given a feature x, polynomial features could include x^2 and x^3. By incorporating squared, cubic terms, and interaction terms into the dataset, models can identify more intricate patterns, potentially boosting accuracy. However, this increase in complexity heightens the risk of overfitting, necessitating management through techniques like regularization. For small datasets, careful application of polynomial transformations can help uncover hidden relationships without overly complicating the model.

Principal component analysis (PCA) reduces dimensionality by transforming a large set of variables into a smaller one while retaining most of the original information. The transformation is achieved through the following formula:

$$Z = XW \tag{11}$$

where Z is the matrix of principal components; X is the matrix of original variables; W is the matrix of eigenvectors. PCA uses models, enhances performance, and reduces the risk of overfitting by ensuring that the first principal component exhibits the highest variance, with each subsequent component having the maximum variance possible under orthogonality constraints with the preceding components. PCA is extensively utilized in exploratory data analysis and in enhancing the efficiency of predictive models. It can be used in small datasets where dimensionality reduction can simplify the model and improve generalization.

Kernel PCA extends traditional PCA by incorporating techniques from kernel methods to facilitate nonlinear dimensionality reduction. The kernel PCA transformation can be expressed as follows:

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_i) \tag{12}$$

where K is the kernel function; ϕ is the mapping function. By applying a nonlinear kernel function to the data and then conducting linear PCA on the transformed data, kernel PCA is adept at uncovering structures in data that are not linearly separable, thus providing superior input features for machine learning models. This is particularly beneficial in small datasets where capturing complex, nonlinear relationships can significantly enhance model performance.

Backward elimination is a feature selection strategy used in model development that starts with all predictors and progressively removes the least significant predictor until the optimal predictor set is determined. This method emphasizes model simplification without compromising predictive accuracy and is particularly effective when handling multiple collinear variables, aiming to enhance model interpretability by eliminating redundant predictors. In small datasets, backward elimination helps in reducing the risk of overfitting by selecting the most relevant features.

In contrast, forward selection begins with no variables in the model and keeps adding the most significant predictor at each step until a new variable does not significantly improve model performance. This technique is advantageous when dealing with numerous predictors, making it computationally impractical to fit models with all possible combinations. Forward selection offers a practical approach to identifying an appropriate subset of features, balancing model performance and complexity, which is critical in small datasets where overfitting is a common concern.

In summary, each data preprocessing technique employed in this study addresses specific challenges associated with small datasets, such as multicollinearity, overfitting, and disproportionate influence of outliers. This study aims to enhance the performance and robustness of regression models in predicting concrete properties with limited data availability by carefully selecting and applying these techniques.

2.4 Model Development and Performance Assessment Strategy

This study implemented a comprehensive methodology to develop and refine predictive models through a robust, data-driven approach. In general, the regression model selection and evaluation process started with the input dataset containing pairs of features and target values. This dataset was split into an 80% training set and a 20% testing set. Thereafter, the preprocessing methods and regression models were then defined. For each preprocessing method, the training and testing sets were transformed accordingly. Each regression model was initialized with default parameters, and a grid search with cross-validation was utilized to tune the model's hyperparameters. The best parameters were identified by minimizing the error between the predicted and measured values. The model with the optimized parameters was then trained on the preprocessed training set. Predictions were made on both the training and testing sets using the optimized model. Performance metrics such as the coefficient of correlation (R), normalized root mean squared error (NRMSE), and normalized mean absolute error (NMAE) were computed to evaluate the model's performance. The results, including performance metrics, optimized parameters, and model outputs, are saved. This process was repeated for all combinations of models and preprocessing methods to determine the best-performing model-preprocessing pair. **Fig. 2** illustrates the pseudo-code developed in this study, where the scikit-learn library was utilized to introduce the data preprocessing techniques and the regression models. On the other hand, it is worth noting that, unlike advanced machine learning models, the selected data

preprocessing techniques and regression models are simple in nature, robust in terms of training time, and low in computational complexity, which makes them suitable for fast handling by researchers as well as practitioners in resource-constrained environments.

3 Results and Discussions

This section provides a detailed analysis to illustrate the critical importance of model and preprocessing technique selection in small data regimes. In this regard, it reports the performance of each of the developed models with respect to the investigated copper tailing concrete properties.

Input: Dataset $D = \{(X_i, y_i)\}$, where $X_i \in \mathbb{R}^n$ and $y_i \in \mathbb{R}$, $i = 1$ to N

Split: Training dataset (80%) $D_{Training} = \{(X_{Train,i}, y_{Train,i})\}$ and Testing dataset (20%) $D_{Testing} = \{(X_{Test,i}, y_{Test,i})\}$

Initialize: Define preprocessing methods P and regression models M

P = {Original, Standardized, Normalized, Discretized, Polynomial Features, PCA, Kernel PCA, Back Elimination, Forward Selection}

M = {MLR, Ridge, Lasso, ElasticNet, Bayesian Ridge}

For each preprocessing method $p \in P$ do:

Apply p to obtain $X_{Train-p}$ and X_{Test-p} from X_{Train} and X_{Test}

For each model $m \in M$ do:

Initialize m with default parameters

Define grid parameters range for hyperparameter tuning

Set grid search CV with parameters θ_m, loss function L, on $(X_{Train-p}, y_{Train})$

Train model and find best parameters

$\theta_{Best} = argmin_\theta \; L(m(X_{Train-p}, \theta), y_{Train})$

Evaluate model with best found parameters

m_{Best}= m trained with θ_{Best}

$y_{Train-Pred} = m_{Best}(X_{Train-p})$

$y_{Test-Pred} = m_{Best}(X_{Test-p})$

Compute performance metrics

$$R = 1 - \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}$$

$$NRMSE = \sqrt{\frac{1}{(y_{true}^{max} - y_{true}^{min}) \cdot N} \sum_{i=1}^{N} (y_{true,i} - y_{pred,i})^2}$$

$$NMAE = \frac{1}{(y_{true}^{max} - y_{true}^{min}) \cdot N} \sum_{i=1}^{N} |y_{true,i} - y_{pred,i}|$$

Output: Save performance metrics, model-optimized parameters, and model outputs

Repeat until all models and preprocessing combinations are evaluated

Fig. 2. A summary of the Python code used for developing the regression models in this study.

3.1 Fresh Mixture Density

The performance analysis of regression models combined with various preprocessing techniques on the prediction of fresh mixture density of concrete with recycled copper tailings is shown in **Fig. 3**. In general, all regression models, irrespective of the preprocessing technique, achieved very high R values ranging from 0.96 to 1.00 during both the training and testing phases for the original data format. On the other hand, the NRMSE and NMAE values ranged from 0 to 0.13 and 0 to 0.12, respectively, showing acceptable results. This suggests an almost good agreement between the measured and estimated values. The influence of preprocessing on model performance was significant in certain cases. Standardization and normalization generally showed a slight deterioration in metrics such as R, NRMSE, and NMAE for models like Ridge and ElasticNet, particularly in the testing phase, where a slight drop was observed compared to the other cases.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': False}
	Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.1}
	Lasso	1.00	1.00	0.00	0.01	0.00	0.00	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	1.00	1.00	0.01	0.01	0.00	0.01	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Standardized	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': True}
	Ridge	0.99	0.99	0.03	0.05	0.02	0.04	{'alpha': 0.1}
	Lasso	1.00	1.00	0.01	0.01	0.01	0.01	{'alpha': 0.1, 'selection': 'cyclic'}
	ElasticNet	1.00	1.00	0.01	0.01	0.01	0.01	{'alpha': 0.001, 'l1_ratio': 0.2}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 1e-05}
Normalized	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': True}
	Ridge	0.97	0.96	0.07	0.13	0.04	0.09	{'alpha': 1}
	Lasso	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.001, 'selection': 'random'}
	ElasticNet	1.00	1.00	0.00	0.01	0.00	0.00	{'alpha': 0.01, 'l1_ratio': 1.0}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Discretized	MLR	0.98	0.99	0.06	0.11	0.04	0.10	{'fit_intercept': True}
	Ridge	0.97	0.99	0.06	0.12	0.04	0.11	{'alpha': 10}
	Lasso	0.98	0.99	0.06	0.11	0.04	0.10	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.97	0.99	0.07	0.13	0.04	0.12	{'alpha': 1, 'l1_ratio': 0.2}
	Bayesian Ridge	0.98	0.99	0.06	0.10	0.04	0.09	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Polynomial Features	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': False}
	Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 100}
	Lasso	1.00	1.00	0.00	0.02	0.00	0.01	{'alpha': 0.001, 'selection': 'random'}
	ElasticNet	1.00	1.00	0.00	0.01	0.00	0.00	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 1e-06, 'alpha_2': 1e-05, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': True}
	Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.1}
	Lasso	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.001, 'selection': 'cyclic'}
	ElasticNet	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.001, 'l1_ratio': 0.0}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Kernel PCA*	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': True}
	Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.1}
	Lasso	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 10, 'selection': 'cyclic'}
	ElasticNet	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 10, 'l1_ratio': 0.6}
	Bayesian Ridge*	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Back Elimination	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': False}
	Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.1}
	Lasso	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	1.00	1.00	0.01	0.01	0.00	0.01	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Forward Selection	MLR	1.00	1.00	0.00	0.00	0.00	0.00	{'fit_intercept': False}
	Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha': 0.1}
	Lasso	1.00	1.00	0.00	0.01	0.00	0.00	{'alpha': 0.001, 'selection': 'random'}
	ElasticNet	1.00	1.00	0.01	0.01	0.00	0.01	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	1.00	1.00	0.00	0.00	0.00	0.00	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}

*Optimal combination of pre-processing technique and linear regression approach.

High Performance ↑ ↓ Low Performance

Fig. 3. Performance assessment and optimized parameters for the models developed to predict the density of fresh concrete containing copper tailing.

This suggests that while these techniques often help in improving model generalization, they may not always be necessary or effective for all types of data or models. In contrast, the application of kernel PCA before applying the Bayesian Ridge model emerged as the most effective combination, maintaining an R-value of 1.00 with the lowest error metrics in both the training and testing phases. This indicates that for complex datasets, even when small in size, advanced preprocessing techniques combined with non-traditional linear regression models can enhance predictive performance considerably. The worst preprocessing cases were observed in the discretized case, where almost all regression models yielded bad performance. Despite the limited size of the dataset, the robust cross-

validation scheme (10 folds) and the data division (80% training and 20% testing) ensured that the models were tested against unforeseen data effectively. The consistently high performance across most models and scenarios confirms the capability of the regression techniques to handle small datasets without overfitting. This is crucial for practical applications in construction material testing, where obtaining large datasets can be challenging or expensive.

3.2 Compressive Strength

The evaluation of regression models and preprocessing techniques in predicting the compressive strength of concrete containing copper tailings is provided in **Fig. 4**. The performance of models herein varied significantly depending on the data preprocessing applied. The models showed R values between 0.41 and 1.00 during training, with testing phases exhibiting a range of -0.73 to 0.94. The ranges of NRMSE and NMAE values are 0.01 to 0.6 and 0.00 to 0.53, respectively, across different preprocessing techniques except for a case where the error was almost 4, which indicates unstable model results. This highlights a broader variation in model performance compared to the fresh mixture density case.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.75	0.94	0.19	0.25	0.16	0.24	{'fit_intercept': True}
	Ridge	0.71	0.80	0.21	0.29	0.18	0.28	{'alpha': 1000}
	Lasso	0.72	0.77	0.20	0.28	0.18	0.27	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.70	0.81	0.21	0.29	0.19	0.28	{'alpha': 10, 'l1_ratio': 0.4}
	Bayesian Ridge	0.64	0.74	0.23	0.32	0.20	0.30	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Standardized	MLR	0.41	-0.73	0.43	0.60	0.37	0.53	{'fit_intercept': False}
	Ridge	0.70	0.82	0.22	0.31	0.19	0.29	{'alpha': 10}
	Lasso	0.61	0.79	0.26	0.33	0.21	0.30	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.69	0.83	0.23	0.32	0.20	0.30	{'alpha': 1, 'l1_ratio': 0.2}
	Bayesian Ridge	0.70	0.82	0.22	0.30	0.19	0.29	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Normalized	MLR	0.75	0.94	0.19	0.25	0.16	0.24	{'fit_intercept': False}
	Ridge	0.70	0.82	0.21	0.29	0.19	0.28	{'alpha': 1}
	Lasso	0.70	0.83	0.21	0.28	0.19	0.27	{'alpha': 0.1, 'selection': 'random'}
	ElasticNet	0.70	0.83	0.21	0.28	0.19	0.27	{'alpha': 0.1, 'l1_ratio': 1.0}
	Bayesian Ridge	0.70	0.82	0.22	0.30	0.19	0.29	{'alpha_1': 0.0001, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Discretized	MLR	0.74	0.29	0.20	0.36	0.17	0.35	{'fit_intercept': True}
	Ridge	0.67	0.75	0.24	0.33	0.20	0.31	{'alpha': 100}
	Lasso	0.68	0.86	0.23	0.30	0.20	0.28	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.70	0.85	0.22	0.29	0.19	0.28	{'alpha': 1, 'l1_ratio': 0.6}
	Bayesian Ridge	0.70	0.80	0.22	0.30	0.19	0.29	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Polynomial Features	MLR	1.00	0.86	0.01	0.24	0.00	0.22	{'fit_intercept': False}
	Ridge	0.98	0.88	0.05	0.24	0.04	0.20	{'alpha': 1000}
	Lasso	0.97	0.87	0.07	0.26	0.05	0.23	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	0.96	0.88	0.08	0.23	0.06	0.21	{'alpha': 100, 'l1_ratio': 0.2}
	Bayesian Ridge	0.96	0.88	0.08	0.23	0.06	0.21	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	0.75	0.94	0.19	0.25	0.16	0.24	{'fit_intercept': True}
	Ridge	0.71	0.80	0.21	0.29	0.18	0.28	{'alpha': 1000}
	Lasso	0.73	0.76	0.20	0.28	0.17	0.27	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.70	0.81	0.21	0.29	0.19	0.28	{'alpha': 100, 'l1_ratio': 0.0}
	Bayesian Ridge	0.64	0.74	0.23	0.32	0.20	0.30	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Kernel PCA*	MLR	1.00	0.73	3.95	3.86	3.95	3.86	{'fit_intercept': False}
	Ridge*	0.98	0.91	0.06	0.19	0.05	0.16	{'alpha': 500}
	Lasso	0.96	0.88	0.08	0.23	0.07	0.21	{'alpha': 100, 'selection': 'cyclic'}
	ElasticNet	0.96	0.88	0.08	0.23	0.06	0.21	{'alpha': 100, 'l1_ratio': 0.6}
	Bayesian Ridge	0.96	0.88	0.08	0.23	0.06	0.21	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Back Elimination	MLR	0.61	0.79	0.23	0.28	0.20	0.25	{'fit_intercept': True}
	Ridge	0.61	0.79	0.23	0.29	0.20	0.26	{'alpha': 500}
	Lasso	0.61	0.79	0.23	0.28	0.20	0.25	{'alpha': 0.001, 'selection': 'cyclic'}
	ElasticNet	0.61	0.79	0.23	0.28	0.20	0.25	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	0.61	0.79	0.23	0.29	0.20	0.26	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Forward Selection	MLR	0.75	0.94	0.19	0.25	0.16	0.24	{'fit_intercept': False}
	Ridge	0.71	0.80	0.21	0.29	0.18	0.28	{'alpha': 1000}
	Lasso	0.72	0.77	0.20	0.28	0.18	0.27	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.70	0.81	0.21	0.29	0.19	0.28	{'alpha': 10, 'l1_ratio': 0.4}
	Bayesian Ridge	0.64	0.74	0.23	0.32	0.20	0.30	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}

*Optimal combination of pre-processing technique and linear regression approach.

High Performance

Low Performance

Fig. 4. Performance assessment and optimized parameters for the models developed to predict the compressive strength of concrete containing copper tailing.

The preprocessing techniques had a notable impact on the performance metrics. Standardization, for instance, generally resulted in a significant degradation in R, NRMSE, and NMAE for models like the MLR, particularly evident during the testing phase, where performance dropped substantially. This

suggests that while standardization can aid in model generalization, it might not be uniformly beneficial across all data types or scenarios, possibly due to the sensitivity of compressive strength to changes in scale or distribution of input variables. Conversely, the use of kernel PCA with Ridge regression emerged as the best preprocessing-model combination, achieving the highest overall performance in both the training and testing phases. This combination's success highlights the utility of dimensionality reduction techniques in enhancing model performance, especially in contexts involving small datasets where conventional models might struggle with feature overfitting or underrepresentation. Such results underline the complexities of modeling compressive strength in cases with small data. Despite the challenges posed by the limited dataset size, the robustness of the cross-validation process (10 folds) and the data splitting strategy (80% training, 20% testing) ensured effective validation of the models against unseen data. The variation in performance across different models and preprocessing techniques underscores the necessity of careful selection of both the model and the preprocessing method to optimize prediction accuracy in small data regimes.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.64	0.82	0.26	0.40	0.18	0.32	{'fit_intercept': False}
	Ridge	0.63	0.85	0.26	0.39	0.18	0.31	{'alpha': 1000}
	Lasso	0.58	0.98	0.28	0.36	0.21	0.28	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.59	0.94	0.27	0.35	0.20	0.27	{'alpha': 1, 'l1_ratio': 0.6}
	Bayesian Ridge	0.61	0.97	0.27	0.36	0.20	0.29	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 1e-06}
Standardized	MLR	0.64	0.82	0.26	0.40	0.18	0.32	{'fit_intercept': True}
	Ridge	0.62	0.85	0.30	0.41	0.25	0.31	{'alpha': 100}
	Lasso	Model Failed to Converge						
	ElasticNet	0.61	0.85	0.31	0.42	0.27	0.32	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	0.62	0.84	0.27	0.39	0.20	0.30	{'alpha_1': 0.0001, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Normalized	MLR	0.64	0.82	0.26	0.40	0.18	0.32	{'fit_intercept': True}
	Ridge	0.61	0.55	0.29	0.41	0.24	0.32	{'alpha': 10}
	Lasso	0.61	0.43	0.27	0.40	0.20	0.31	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	0.61	0.52	0.30	0.42	0.25	0.33	{'alpha': 1, 'l1_ratio': 0.0}
	Bayesian Ridge	0.62	0.66	0.27	0.40	0.20	0.31	{'alpha_1': 0.0001, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Discretized	MLR	0.66	0.95	0.25	0.33	0.17	0.28	{'fit_intercept': True}
	Ridge	0.62	0.74	0.28	0.39	0.22	0.31	{'alpha': 100}
	Lasso	Model Failed to Converge						
	ElasticNet	0.62	0.71	0.29	0.40	0.23	0.31	{'alpha': 10, 'l1_ratio': 0.0}
	Bayesian Ridge	0.63	0.80	0.27	0.38	0.20	0.30	{'alpha_1': 0.0001, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Polynomial Features	MLR	1.00	-0.09	0.01	0.74	0.01	0.63	{'fit_intercept': False}
	Ridge	0.99	0.33	0.06	0.46	0.04	0.35	{'alpha': 100}
	Lasso	0.84	0.52	0.19	0.40	0.13	0.27	{'alpha': 100, 'selection': 'cyclic'}
	ElasticNet	0.85	0.35	0.18	0.41	0.12	0.28	{'alpha': 100, 'l1_ratio': 0.8}
	Bayesian Ridge	0.99	0.34	0.06	0.46	0.04	0.35	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 0.0001}
PCA	MLR	0.64	0.82	0.26	0.40	0.18	0.32	{'fit_intercept': True}
	Ridge	0.63	0.85	0.26	0.39	0.18	0.31	{'alpha': 1000}
	Lasso	0.60	0.95	0.27	0.36	0.20	0.28	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.60	0.92	0.27	0.36	0.20	0.28	{'alpha': 1, 'l1_ratio': 0.8}
	Bayesian Ridge	0.61	0.97	0.27	0.36	0.20	0.29	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 1e-06}
Kernel PCA*	MLR	1.00	0.04	7.57	7.76	7.57	7.74	{'fit_intercept': False}
	Ridge	0.92	-0.13	0.13	0.46	0.09	0.34	{'alpha': 1000}
	Lasso*	0.69	0.90	0.25	0.39	0.17	0.31	{'alpha': 100, 'selection': 'random'}
	ElasticNet*	0.69	0.90	0.25	0.39	0.17	0.31	{'alpha': 100, 'l1_ratio': 1.0}
	Bayesian Ridge	0.89	-0.07	0.16	0.48	0.12	0.37	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Back Elimination	MLR	0.51	-0.49	0.29	0.49	0.23	0.41	{'fit_intercept': False}
	Ridge	0.50	-0.30	0.29	0.46	0.23	0.38	{'alpha': 1000}
	Lasso	0.48	-0.22	0.30	0.44	0.25	0.36	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.49	-0.29	0.30	0.45	0.23	0.38	{'alpha': 100, 'l1_ratio': 0.0}
	Bayesian Ridge	0.49	-0.29	0.29	0.46	0.23	0.38	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-05}
Forward Selection	MLR	0.64	0.82	0.26	0.40	0.18	0.32	{'fit_intercept': False}
	Ridge	0.63	0.85	0.26	0.39	0.18	0.31	{'alpha': 1000}
	Lasso	0.58	0.98	0.28	0.36	0.21	0.28	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.59	0.94	0.27	0.35	0.20	0.27	{'alpha': 1, 'l1_ratio': 0.6}
	Bayesian Ridge	0.61	0.97	0.27	0.36	0.20	0.29	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 1e-06}

High Performance ↑ Low Performance ↓

*Optimal combination of pre-processing technique and linear regression approach.

Fig. 5. Performance assessment and optimized parameters for the models developed to predict the flexural strength of concrete containing copper tailing.

3.3 Flexural Strength

The performance of models in predicting the flexural strength of concrete containing copper tailing is illustrated in **Fig. 5**. This property proved challenging, with generally lower accuracy levels observed across different preprocessing techniques. However, the application of kernel PCA preprocessing

improved model performance, especially when coupled with an ℓ_1 regularized case such as the lasso and ElasticNet models, highlighting its effectiveness in dealing with complex property predictions. These two cases were selected as the best ones because they have the highest results and the closest training and testing metrics to each other. The best models under this configuration demonstrated average predictive accuracy for training and high for testing, meaning that although other properties of concrete were all accurately predicted, it is possible that some cases would yield an average performance in small datasets. The reliability and consistency of the results of both training and testing and the performance in predicting unseen data upon 10-fold cross-validation indicate that these models are practically viable for predicting flexural strength but should be used with caution given the considerably lower performance in the training case. On the other hand, the huge change in the performance of the various models highlights the importance of optimizing the model type and pre-processing case to achieve suitable results.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.35	-0.38	0.27	0.50	0.22	0.44	{'fit_intercept': False}
	Ridge	0.24	-0.53	0.28	0.52	0.23	0.41	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.23	-0.54	0.28	0.52	0.23	0.41	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Standardized	MLR	0.22	-0.51	0.42	1.08	0.33	0.81	{'fit_intercept': False}
	Ridge	0.26	-0.57	0.29	0.46	0.23	0.40	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.26	-0.57	0.29	0.46	0.23	0.40	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Normalized	MLR	0.35	-0.38	0.27	0.50	0.22	0.44	{'fit_intercept': False}
	Ridge	0.25	-0.47	0.29	0.46	0.23	0.40	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.25	-0.48	0.29	0.46	0.23	0.40	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Discretized	MLR	0.30	-0.41	0.28	0.55	0.22	0.40	{'fit_intercept': True}
	Ridge	0.24	-0.47	0.29	0.46	0.23	0.40	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.24	-0.46	0.29	0.46	0.23	0.40	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Polynomial Features*	MLR	0.99	-0.08	0.05	0.81	0.02	0.79	{'fit_intercept': False}
	Ridge	0.96	0.94	0.08	0.28	0.05	0.25	{'alpha': 10}
	Lasso*	0.94	0.95	0.10	0.14	0.08	0.11	{'alpha': 100, 'selection': 'random'}
	ElasticNet*	0.94	0.95	0.10	0.14	0.08	0.11	{'alpha': 100, 'l1_ratio': 1.0}
	Bayesian Ridge	0.95	0.93	0.09	0.17	0.07	0.15	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	0.35	-0.38	0.27	0.50	0.22	0.44	{'fit_intercept': True}
	Ridge	0.24	-0.53	0.28	0.52	0.23	0.41	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.23	-0.54	0.28	0.52	0.23	0.41	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Kernel PCA	MLR	0.99	-0.31	2.81	3.72	2.81	3.66	{'fit_intercept': False}
	Ridge	0.95	0.90	0.09	0.21	0.06	0.17	{'alpha': 1000}
	Lasso	0.94	0.94	0.10	0.16	0.07	0.13	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.94	0.94	0.10	0.16	0.07	0.13	{'alpha': 100, 'l1_ratio': 0.2}
	Bayesian Ridge	0.95	0.93	0.09	0.17	0.07	0.15	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Back Elimination	MLR	0.15	0.21	0.29	0.44	0.23	0.40	{'fit_intercept': True}
	Ridge	0.15	0.21	0.29	0.44	0.23	0.40	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.15	0.21	0.29	0.45	0.23	0.40	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Forward Selection	MLR	0.35	-0.38	0.27	0.50	0.22	0.44	{'fit_intercept': False}
	Ridge	0.24	-0.53	0.28	0.52	0.23	0.41	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.23	-0.54	0.28	0.52	0.23	0.41	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}

High Performance

Low Performance

*Optimal combination of pre-processing technique and linear regression approach.

Fig. 6. Performance assessment and optimized parameters for the models developed to predict the pull-off strength of concrete containing copper tailing.

3.4 Pull-off Strength

Fig. 6 demonstrates the challenges involved in predicting the pull-off strength of concrete with copper tailings. Among the various models and preprocessing methods, the combination of polynomial features preprocessing with Lasso and ElasticNet models was the most successful, significantly outperforming others with R values of 0.94 and 0.95 in the training and testing phases and low testing

errors, NRMSE and NMAE, at approximately 0.14 and 0.11, respectively. On the other hand, multiple models failed to converge, indicating that estimating the pull-out is a challenging task under small data regimes This particular combination's effectiveness in capturing complex, nonlinear relationships is evident from its robust performance across both training and testing phases, supported by a rigorous 10-fold cross-validation process. Such results not only demonstrate the models' reliability in practical applications but also emphasize the importance of selecting advanced preprocessing techniques and sophisticated models to manage the complexities of predicting construction material properties, especially when dealing with limited data.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.81	0.95	0.16	0.16	0.12	0.13	{'fit_intercept': False}
	Ridge	0.71	0.89	0.19	0.22	0.13	0.17	{'alpha': 1000}
	Lasso	0.65	0.92	0.20	0.24	0.13	0.18	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.65	0.92	0.20	0.24	0.13	0.18	{'alpha': 1, 'l1_ratio': 1.0}
	Bayesian Ridge	0.67	0.91	0.20	0.22	0.14	0.18	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Standardized	MLR	0.81	0.95	0.16	0.16	0.12	0.13	{'fit_intercept': True}
	Ridge	0.66	0.90	0.21	0.25	0.14	0.20	{'alpha': 10}
	Lasso	0.67	0.90	0.20	0.23	0.13	0.18	{'alpha': 0.01, 'selection': 'cyclic'}
	ElasticNet	0.65	0.92	0.21	0.28	0.14	0.23	{'alpha': 0.1, 'l1_ratio': 0.4}
	Bayesian Ridge	0.65	0.90	0.21	0.26	0.14	0.21	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Normalized	MLR	0.81	0.95	0.16	0.16	0.12	0.13	{'fit_intercept': True}
	Ridge	0.65	0.89	0.21	0.26	0.14	0.20	{'alpha': 1}
	Lasso	0.65	0.92	0.21	0.27	0.14	0.22	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	0.65	0.92	0.21	0.27	0.14	0.22	{'alpha': 0.01, 'l1_ratio': 1.0}
	Bayesian Ridge	0.65	0.89	0.21	0.28	0.14	0.22	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Discretized	MLR	0.69	0.87	0.19	0.21	0.14	0.17	{'fit_intercept': True}
	Ridge	0.67	0.86	0.20	0.24	0.14	0.20	{'alpha': 10}
	Lasso	0.66	0.86	0.23	0.33	0.16	0.27	{'alpha': 0.1, 'selection': 'cyclic'}
	ElasticNet	0.66	0.86	0.21	0.30	0.14	0.24	{'alpha': 0.1, 'l1_ratio': 0.6}
	Bayesian Ridge	0.67	0.87	0.21	0.27	0.14	0.21	{'alpha_1': 0.0001, 'alpha_2': 0.0001, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Polynomial Features	MLR	0.99	0.88	0.04	0.29	0.02	0.20	{'fit_intercept': False}
	Ridge	0.97	0.92	0.07	0.15	0.05	0.14	{'alpha': 1000}
	Lasso	0.93	0.97	0.10	0.08	0.08	0.07	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.94	0.98	0.09	0.08	0.08	0.06	{'alpha': 10, 'l1_ratio': 0.6}
	Bayesian Ridge	0.95	0.95	0.08	0.11	0.07	0.10	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	0.81	0.95	0.16	0.16	0.12	0.13	{'fit_intercept': True}
	Ridge	0.71	0.89	0.19	0.22	0.13	0.17	{'alpha': 1000}
	Lasso	0.65	0.92	0.20	0.24	0.13	0.18	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.65	0.92	0.20	0.24	0.13	0.19	{'alpha': 10, 'l1_ratio': 0.2}
	Bayesian Ridge	0.67	0.91	0.20	0.22	0.14	0.18	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Kernel PCA*	MLR	0.98	0.86	2.15	1.70	2.15	1.69	{'fit_intercept': False}
	Ridge	0.97	0.94	0.06	0.14	0.05	0.11	{'alpha': 1000}
	Lasso*	0.94	0.98	0.09	0.07	0.08	0.06	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.96	0.96	0.08	0.10	0.07	0.08	{'alpha': 1, 'l1_ratio': 0.4}
	Bayesian Ridge	0.95	0.97	0.08	0.09	0.07	0.07	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Back Elimination	MLR	0.01	-0.39	0.27	0.38	0.20	0.33	{'fit_intercept': True}
	Ridge	0.01	-0.39	0.27	0.38	0.20	0.33	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.01	-0.39	0.27	0.38	0.20	0.33	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Forward Selection	MLR	0.81	0.95	0.16	0.16	0.12	0.13	{'fit_intercept': False}
	Ridge	0.71	0.89	0.19	0.22	0.13	0.17	{'alpha': 1000}
	Lasso	0.65	0.92	0.20	0.24	0.13	0.18	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.65	0.92	0.20	0.24	0.13	0.18	{'alpha': 1, 'l1_ratio': 1.0}
	Bayesian Ridge	0.67	0.91	0.20	0.22	0.14	0.18	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}

*Optimal combination of pre-processing technique and linear regression approach.

High Performance

Low Performance

Fig. 7. Performance assessment and optimized parameters for the models developed to predict the depth of abrasion of concrete containing copper tailing.

3.5 Abrasion Resistance

Fig. 7 delineates the results for predicting the abrasion resistance of concrete containing copper tailing. The evaluation of the models across various preprocessing techniques shows a notable variance in performance. The polynomial features preprocessing with lasso regression demonstrated exceptional performance, achieving perfect accuracy in both the training and testing phases. This optimal combination significantly outperformed other methods, underscoring the potential of incorporating higher-degree polynomial transformations to capture complex interactions within the data effectively. The original dataset, without any preprocessing, also yielded strong results with MLR, achieving an R value of 0.97 in the testing phase. This suggests that the basic characteristics of the dataset are well-

suited for regression analyses without necessitating transformations. The consistency in performance across both the training and testing phases, validated through a rigorous 10-fold cross-validation process, indicates a robust model capable of reliably predicting abrasion resistance in unseen data, even within a small dataset context.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.95	0.97	0.10	0.15	0.08	0.14	{'fit_intercept': False}
	Ridge	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 1000}
	Lasso	0.95	0.96	0.11	0.15	0.09	0.13	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 100, 'l1_ratio': 0.0}
	Bayesian Ridge	0.95	0.96	0.11	0.15	0.09	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Standardized	MLR	0.95	0.97	0.10	0.15	0.08	0.14	{'fit_intercept': True}
	Ridge	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 1}
	Lasso	0.93	0.95	0.14	0.15	0.10	0.12	{'alpha': 0.1, 'selection': 'random'}
	ElasticNet	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 0.1, 'l1_ratio': 0.0}
	Bayesian Ridge	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Normalized	MLR	0.95	0.97	0.10	0.15	0.08	0.14	{'fit_intercept': True}
	Ridge	0.95	0.97	0.11	0.15	0.09	0.14	{'alpha': 0.1}
	Lasso	0.95	0.97	0.11	0.14	0.09	0.13	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 0.01, 'l1_ratio': 0.0}
	Bayesian Ridge	0.95	0.97	0.11	0.15	0.09	0.14	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Discretized	MLR	0.96	0.99	0.10	0.13	0.08	0.11	{'fit_intercept': True}
	Ridge	0.95	0.99	0.11	0.09	0.09	0.08	{'alpha': 10}
	Lasso	0.95	0.99	0.11	0.09	0.09	0.07	{'alpha': 0.1, 'selection': 'random'}
	ElasticNet	0.96	0.99	0.10	0.09	0.09	0.07	{'alpha': 0.1, 'l1_ratio': 0.6}
	Bayesian Ridge	0.95	0.99	0.10	0.09	0.09	0.08	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Polynomial Features*	MLR	0.99	0.83	0.04	0.29	0.02	0.27	{'fit_intercept': False}
	Ridge*	0.97	1.00	0.08	0.10	0.05	0.08	{'alpha': 1000}
	Lasso	0.96	0.98	0.10	0.12	0.07	0.11	{'alpha': 100, 'selection': 'random'}
	ElasticNet	0.96	0.98	0.10	0.12	0.07	0.11	{'alpha': 100, 'l1_ratio': 1.0}
	Bayesian Ridge	0.96	0.98	0.10	0.13	0.07	0.12	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	0.95	0.97	0.10	0.15	0.08	0.14	{'fit_intercept': True}
	Ridge	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 1000}
	Lasso	0.94	0.96	0.12	0.14	0.09	0.13	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.93	0.95	0.12	0.15	0.09	0.12	{'alpha': 10, 'l1_ratio': 0.2}
	Bayesian Ridge	0.95	0.96	0.11	0.15	0.09	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Kernel PCA	MLR	0.99	0.70	2.53	2.45	2.53	2.43	{'fit_intercept': False}
	Ridge	0.99	0.94	0.06	0.16	0.04	0.13	{'alpha': 10}
	Lasso	0.95	0.98	0.10	0.12	0.08	0.11	{'alpha': 100, 'selection': 'random'}
	ElasticNet	0.95	0.98	0.10	0.12	0.08	0.11	{'alpha': 100, 'l1_ratio': 1.0}
	Bayesian Ridge	0.96	0.98	0.10	0.13	0.07	0.11	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Back Elimination	MLR	0.02	0.83	0.34	0.39	0.31	0.37	{'fit_intercept': True}
	Ridge	0.02	0.83	0.34	0.39	0.31	0.37	{'alpha': 1000}
	Lasso	Model Failed to Converge						
	ElasticNet	Model Failed to Converge						
	Bayesian Ridge	0.02	0.83	0.34	0.39	0.31	0.37	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Forward Selection	MLR	0.95	0.97	0.10	0.15	0.08	0.14	{'fit_intercept': True}
	Ridge	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 1000}
	Lasso	0.95	0.97	0.10	0.15	0.08	0.13	{'alpha': 0.01, 'selection': 'random'}
	ElasticNet	0.95	0.97	0.11	0.15	0.09	0.13	{'alpha': 100, 'l1_ratio': 0.0}
	Bayesian Ridge	0.95	0.96	0.11	0.15	0.09	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}

High Performance ↑ Low Performance ↓

*Optimal combination of pre-processing technique and linear regression approach.

Fig. 8. Performance assessment and optimized parameters for the models developed to predict the depth of water penetration of concrete containing copper tailing.

3.6 Water Penetration

Fig. 8 shows the models developed to predict water penetration depth in concrete containing copper tailing performed uniformly well across various preprocessing techniques. In general, the polynomial features preprocessing with ridge regression achieved a perfect R value of 1.00 in the testing phase and 0.1 and 0.08 NRMSE and NMAE, respectively, marking it as the standout preprocessing-model combination. This high level of accuracy indicates an excellent fit of the model to the data, capable of capturing the intricate relationships that affect water penetration in concrete. The original and standardized data cases both supported high model accuracy, with MLR and Ridge models consistently showing R values of 0.95 or higher in the testing phase. The effective prediction across diverse preprocessing techniques reinforces the robustness of the regression models, ensuring reliable predictions of water penetration depth in practical scenarios, particularly when data availability is limited.

3.7 Rapid Chloride Ion Permeability

The results for predicting rapid chloride ion permeability are shown in **Fig. 9**. The models herein displayed generally high performance, but the polynomial features preprocessing with Bayesian ridge regression again emerged as the most effective, achieving an R value of 0.97, an NRMSE of 0.1, and NMAE of 0.07 in the testing phase. This model's ability to handle complex non-linear relationships within the data highlights its suitability for predicting properties related to the durability of construction materials. While the original dataset provided a solid baseline for model performance (R=0.89 in testing for MLR), the enhanced preprocessing techniques, particularly polynomial features, facilitated a significant improvement in model accuracy. This improvement, evidenced by the rigorous 10-fold cross-validation, suggests that the selected models are very capable of accurately predicting chloride ion permeability in new, unseen data samples.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.88	0.89	0.13	0.19	0.11	0.18	{'fit_intercept': True}
	Ridge	0.87	0.90	0.14	0.20	0.12	0.18	{'alpha': 500}
	Lasso	0.87	0.90	0.14	0.19	0.11	0.17	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.87	0.90	0.14	0.19	0.11	0.17	{'alpha': 10, 'l1_ratio': 1.0}
	Bayesian Ridge	0.87	0.90	0.14	0.20	0.12	0.18	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Standardized	MLR	0.88	0.90	0.13	0.19	0.11	0.18	{'fit_intercept': True}
	Ridge	0.87	0.90	0.14	0.20	0.11	0.18	{'alpha': 1}
	Lasso	0.87	0.90	0.14	0.20	0.11	0.18	{'alpha': 1, 'selection': 'cyclic'}
	ElasticNet	0.87	0.90	0.14	0.20	0.11	0.18	{'alpha': 0.1, 'l1_ratio': 0.0}
	Bayesian Ridge	0.87	0.90	0.14	0.20	0.11	0.18	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Normalized	MLR	0.88	0.90	0.13	0.19	0.11	0.18	{'fit_intercept': False}
	Ridge	0.87	0.90	0.14	0.20	0.11	0.18	{'alpha': 0.1}
	Lasso	0.87	0.90	0.14	0.21	0.11	0.19	{'alpha': 1, 'selection': 'random'}
	ElasticNet	0.87	0.90	0.14	0.21	0.11	0.19	{'alpha': 1, 'l1_ratio': 1.0}
	Bayesian Ridge	0.87	0.90	0.14	0.20	0.11	0.18	{'alpha_1': 1e-06, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Discretized	MLR	0.87	0.87	0.14	0.21	0.12	0.18	{'fit_intercept': True}
	Ridge	0.87	0.87	0.14	0.22	0.11	0.19	{'alpha': 10}
	Lasso	0.86	0.88	0.15	0.25	0.12	0.22	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.87	0.88	0.14	0.22	0.11	0.19	{'alpha': 1, 'l1_ratio': 0.6}
	Bayesian Ridge	0.87	0.88	0.14	0.21	0.11	0.19	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Polynomial Features	MLR	0.99	0.71	0.04	0.31	0.02	0.26	{'fit_intercept': False}
	Ridge	0.98	0.91	0.06	0.16	0.04	0.12	{'alpha': 1000}
	Lasso	0.97	0.96	0.06	0.10	0.05	0.08	{'alpha': 100, 'selection': 'cyclic'}
	ElasticNet	0.97	0.96	0.06	0.11	0.05	0.08	{'alpha': 100, 'l1_ratio': 0.8}
	Bayesian Ridge	0.97	0.96	0.07	0.10	0.05	0.08	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	0.88	0.90	0.13	0.19	0.11	0.18	{'fit_intercept': True}
	Ridge	0.87	0.90	0.14	0.20	0.12	0.18	{'alpha': 500}
	Lasso	0.87	0.90	0.14	0.19	0.11	0.17	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.87	0.90	0.14	0.19	0.11	0.17	{'alpha': 10, 'l1_ratio': 0.8}
	Bayesian Ridge	0.87	0.90	0.14	0.20	0.12	0.18	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Kernel PCA*	MLR	0.99	0.69	2.81	2.97	2.81	2.95	{'fit_intercept': False}
	Ridge	0.99	0.86	0.05	0.21	0.03	0.15	{'alpha': 100}
	Lasso	0.97	0.95	0.06	0.12	0.05	0.09	{'alpha': 100, 'selection': 'random'}
	ElasticNet	0.97	0.95	0.06	0.12	0.05	0.09	{'alpha': 100, 'l1_ratio': 1.0}
	Bayesian Ridge*	0.97	0.97	0.07	0.10	0.05	0.07	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-05}
Back Elimination	MLR	0.67	0.64	0.22	0.29	0.18	0.25	{'fit_intercept': False}
	Ridge	0.67	0.64	0.20	0.31	0.18	0.28	{'alpha': 0.1}
	Lasso	0.67	0.64	0.20	0.31	0.18	0.28	{'alpha': 0.001, 'selection': 'cyclic'}
	ElasticNet	0.67	0.64	0.20	0.31	0.18	0.28	{'alpha': 0.001, 'l1_ratio': 0.0}
	Bayesian Ridge	0.67	0.64	0.20	0.31	0.18	0.28	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}
Forward Selection	MLR	0.88	0.89	0.13	0.19	0.11	0.18	{'fit_intercept': True}
	Ridge	0.87	0.90	0.14	0.20	0.12	0.18	{'alpha': 500}
	Lasso	0.87	0.90	0.14	0.19	0.11	0.17	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.87	0.90	0.14	0.19	0.11	0.17	{'alpha': 10, 'l1_ratio': 1.0}
	Bayesian Ridge	0.87	0.90	0.14	0.20	0.12	0.18	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 1e-06, 'lambda_2': 0.0001}

*Optimal combination of pre-processing technique and linear regression approach.

High Performance ↑ Low Performance

Fig. 9. Performance assessment and optimized parameters for the models developed to predict the charge passed in coulombs of concrete containing copper tailing.

3.8 Air Permeability

Fig. 10 presents the performance assessment for predicting air permeability in concrete containing copper tailing. The polynomial features preprocessing, combined with ridge regression, achieved the highest accuracy, with an R value of 0.98, an NRMSE of 0.08, and an NMAE of 0.07 in the testing phase. This optimal configuration underscores the effectiveness of advanced preprocessing in

enhancing the predictive accuracy of regression models concerning air permeability. The uniform high performance across the original, standardized, and normalized data setups with minimal preprocessing adjustments indicates that the fundamental data characteristics are well-suited for modeling using regression techniques. The strong performance across both training and testing phases confirms the models' capability to provide reliable predictions, which is crucial for practical applications where air permeability plays a critical role in material quality assessments.

Preprocessing	Model	R		NRMSE		NMAE		Best Parameters
		Training	Testing	Training	Testing	Training	Testing	
Original	MLR	0.97	0.97	0.06	0.12	0.05	0.11	{'fit_intercept': False}
	Ridge	0.97	0.94	0.07	0.14	0.05	0.13	{'alpha': 1000}
	Lasso	0.97	0.92	0.08	0.15	0.06	0.15	{'alpha': 0.1, 'selection': 'random'}
	ElasticNet	0.97	0.93	0.07	0.16	0.06	0.14	{'alpha': 0.1, 'l1_ratio': 0.4}
	Bayesian Ridge	0.97	0.95	0.07	0.14	0.05	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Standardized	MLR	0.97	0.97	0.06	0.12	0.05	0.11	{'fit_intercept': True}
	Ridge	0.97	0.96	0.07	0.13	0.05	0.12	{'alpha': 0.1}
	Lasso	0.97	0.95	0.07	0.14	0.05	0.13	{'alpha': 0.001, 'selection': 'cyclic'}
	ElasticNet	0.97	0.95	0.07	0.14	0.05	0.13	{'alpha': 0.001, 'l1_ratio': 0.8}
	Bayesian Ridge	0.97	0.96	0.07	0.14	0.05	0.13	{'alpha_1': 1e-06, 'alpha_2': 1e-06, 'lambda_1': 0.0001, 'lambda_2': 0.0001}
Normalized	MLR	0.97	0.97	0.06	0.12	0.05	0.11	{'fit_intercept': True}
	Ridge	0.97	0.95	0.07	0.15	0.05	0.13	{'alpha': 0.1}
	Lasso	0.97	0.96	0.07	0.16	0.06	0.14	{'alpha': 0.001, 'selection': 'cyclic'}
	ElasticNet	0.97	0.95	0.07	0.15	0.05	0.13	{'alpha': 0.001, 'l1_ratio': 0.4}
	Bayesian Ridge	0.97	0.95	0.07	0.15	0.05	0.13	{'alpha_1': 0.0001, 'alpha_2': 1e-06, 'lambda_1': 0.0001, 'lambda_2': 0.0001}
Discretized	MLR	0.98	0.95	0.06	0.24	0.05	0.23	{'fit_intercept': False}
	Ridge	0.97	0.96	0.06	0.20	0.05	0.19	{'alpha': 1}
	Lasso	0.97	0.96	0.07	0.20	0.05	0.19	{'alpha': 0.001, 'selection': 'cyclic'}
	ElasticNet	0.97	0.95	0.07	0.21	0.05	0.19	{'alpha': 0.01, 'l1_ratio': 0.2}
	Bayesian Ridge	0.98	0.96	0.06	0.21	0.05	0.20	{'alpha_1': 1e-06, 'alpha_2': 1e-06, 'lambda_1': 0.0001, 'lambda_2': 0.0001}
Polynomial Features	MLR	1.00	0.92	0.01	0.25	0.01	0.22	{'fit_intercept': False}
	Ridge	0.99	0.97	0.03	0.15	0.02	0.12	{'alpha': 100}
	Lasso	0.98	0.97	0.06	0.11	0.05	0.10	{'alpha': 10, 'selection': 'random'}
	ElasticNet	0.98	0.98	0.05	0.08	0.04	0.08	{'alpha': 10, 'l1_ratio': 0.4}
	Bayesian Ridge	0.99	0.97	0.03	0.15	0.02	0.12	{'alpha_1': 1e-06, 'alpha_2': 1e-05, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
PCA	MLR	0.97	0.97	0.06	0.12	0.05	0.11	{'fit_intercept': True}
	Ridge	0.97	0.94	0.07	0.14	0.05	0.13	{'alpha': 1000}
	Lasso	0.97	0.92	0.07	0.15	0.06	0.15	{'alpha': 0.1, 'selection': 'cyclic'}
	ElasticNet	0.97	0.93	0.07	0.15	0.05	0.14	{'alpha': 0.1, 'l1_ratio': 0.4}
	Bayesian Ridge	0.97	0.95	0.07	0.14	0.05	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Kernel PCA*	MLR	1.00	0.97	1.62	2.07	1.62	2.07	{'fit_intercept': False}
	Ridge*	0.99	0.98	0.04	0.08	0.03	0.07	{'alpha': 1000}
	Lasso	0.97	0.94	0.07	0.15	0.05	0.14	{'alpha': 10, 'selection': 'cyclic'}
	ElasticNet	0.97	0.94	0.07	0.15	0.05	0.14	{'alpha': 10, 'l1_ratio': 1.0}
	Bayesian Ridge	0.99	0.98	0.04	0.10	0.03	0.08	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Back Elimination	MLR	0.97	0.97	0.06	0.12	0.05	0.11	{'fit_intercept': False}
	Ridge	0.97	0.94	0.07	0.14	0.05	0.13	{'alpha': 1000}
	Lasso	0.97	0.92	0.08	0.15	0.06	0.15	{'alpha': 0.1, 'selection': 'random'}
	ElasticNet	0.97	0.93	0.07	0.16	0.06	0.14	{'alpha': 0.1, 'l1_ratio': 0.4}
	Bayesian Ridge	0.97	0.95	0.07	0.14	0.05	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
Forward Selection	MLR	0.97	0.97	0.06	0.12	0.05	0.11	{'fit_intercept': False}
	Ridge	0.97	0.94	0.07	0.14	0.05	0.13	{'alpha': 1000}
	Lasso	0.97	0.92	0.08	0.15	0.06	0.15	{'alpha': 0.1, 'selection': 'cyclic'}
	ElasticNet	0.97	0.93	0.07	0.16	0.06	0.14	{'alpha': 0.1, 'l1_ratio': 0.4}
	Bayesian Ridge	0.97	0.95	0.07	0.14	0.05	0.13	{'alpha_1': 1e-06, 'alpha_2': 0.0001, 'lambda_1': 0.0001, 'lambda_2': 1e-06}
***Optimal combination of pre-processing technique and linear regression approach.**								

High Performance ↑ ↓ Low Performance

Fig. 10. Performance assessment and optimized parameters for the models developed to predict the air permeability index of concrete containing copper tailing.

4 Conclusion

The aim of this study is to evaluate the performance of various regression models combined with different data preprocessing techniques in predicting the properties of concrete containing recycled copper tailings, particularly in scenarios where data is limited. This study addresses a significant gap in the literature concerning the robustness and reliability of regression models when applied to small datasets, which is crucial given the high cost and difficulty of obtaining experimental data in construction material research. Based on the aforementioned statements, the following conclusions are drawn:

(1) The performance of regression models varied significantly across different preprocessing techniques, emphasizing the importance of selecting appropriate preprocessing methods to enhance prediction accuracy in small data regimes.

(2) The findings highlight that even small datasets, when appropriately processed and analyzed, can yield reliable and robust predictions, which is vital for advancing construction material research where data limitations are common due to practicality or cost reasons.

(3) Overall, it is possible to develop a predictive regression model when using a small data regime, where most of the concrete parameters were accurately estimated in both training and testing cases and when 10-fold cross-validation was used.

(4) Polynomial feature transformation and kernel PCA improved model performance across a variety of cases compared to other regression techniques, indicating the utility of polynomial-based data preprocessing techniques in capturing nonlinear relationships within the data.

(5) Specific combinations of regression models and preprocessing techniques, such as kernel PCA with ridge regression for compressive strength and polynomial features with lasso regression for pull-off strength, proved to be most effective in optimizing prediction accuracy.

(6) The study underscored the critical role of data preprocessing in handling small datasets effectively. Techniques like polynomial feature transformation and kernel PCA were particularly beneficial in modeling complex relationships within the data.

While the study provides significant insights, it acknowledges limitations, such as the reliance on a specific type of concrete and numerical models. Future research should explore a broader array of materials and more varied data conditions to validate and possibly enhance the generalizability of the findings. Moreover, they could investigate the applicability of other numerical techniques for such a prediction. Finally, the detailed interpretation of these results highlights how the findings can be applied in real-world scenarios, such as the design and quality control of concrete materials. The study emphasizes that with the correct preprocessing and model selection, even limited data can yield accurate predictions, which is essential for practical decision-making in construction material design. Furthermore, the observed trade-offs between model complexity and interpretability are crucial; while advanced preprocessing techniques like kernel PCA offer improved performance, they also introduce some degree of complexity. Balancing this complexity with the need for interpretable models is vital for practical applications, where understanding the underlying mechanisms is as important as prediction accuracy.

Funding Statement

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

CRediT authorship contribution statement

Habib: Conceptualization, Investigation, Formal analysis, Writing – original draft. **Barakat**: Conceptualization, Investigation, Formal analysis, Writing – original draft. **Dirar**: Conceptualization, Investigation, Formal analysis, Writing – original draft. **Al-Toubat**: Conceptualization, Investigation, Formal analysis, Writing –review & editing. **Al-Sadoon**: Conceptualization, Investigation, Formal analysis, Writing –review & editing.

Conflicts of Interest

All authors declare that they have no conflicts of interest.

Data Availability Statement

Some or all data, models, or codes that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] Tam VW, Butera A, Le KN, Da Silva LC, Evangelista AC. A prediction model for compressive strength of CO2 concrete using regression analysis and artificial neural networks. Construction and Building Materials, 2022; 324: 126689. https://doi.org/10.1016/j.conbuildmat.2022.126689.
[2] Mostafa O, Alotaibi E, Al-Ateyat A, Nassif N, Barakat S. Prediction of punching shear capacity for fiber-reinforced polymer concrete slabs using machine learning. In 2022 Advances in Science and Engineering Technology International Conferences (ASET), 2022; 1-6. IEEE.
[3] Moein MM, Saradar A, Rahmati K, Mousavinejad SHG, Bristow J, Aramali V, Karakouzian M. Predictive models for concrete properties using machine learning and deep learning approaches: A review. Journal of Building Engineering, 2023; 63: 105444. https://doi.org/10.1016/j.jobe.2022.105444.
[4] Alotaibi E, Nassif N, Barakat S. Data-driven reliability and cost-based design optimization of steel fiber reinforced concrete suspended slabs. Structural Concrete, 2023; 24(2): 1856-1867. https://doi.org/10.1002/suco.202200282.
[5] Alam MS, Sultana N, Hossain SZ. Bayesian optimization algorithm based support vector regression analysis for estimation of shear capacity of FRP reinforced concrete members. Applied Soft Computing, 2021; 105: 107281. https://doi.org/10.1016/j.asoc.2021.107281.
[6] Kate GK, Nayak CB, Thakare SB. Optimization of sustainable high-strength–high-volume fly ash concrete with and without steel fiber using Taguchi method and multi-regression analysis. Innovative Infrastructure Solutions, 2021; 6(2): 102. https://doi.org/10.1007/s41062-021-00472-6.
[7] Habib A, Yildirim U. Estimating mechanical and dynamic properties of rubberized concrete using machine learning techniques: a comprehensive study. Engineering Computations, 2022; 39(8): 3129-3178. https://doi.org/10.1108/EC-09-2021-0527.
[8] Habib A, Yildirim U. Simplified modeling of rubberized concrete properties using multivariable regression analysis. Materiales de Construcción, 2022; 72(347): e289. https://doi.org/10.3989/mc.2022.13621.
[9] Moaf FO, Kazemi F, Abdelgader HS, Kurpińska M. Machine learning-based prediction of preplaced aggregate concrete characteristics. Engineering Applications of Artificial Intelligence, 2023; 123: 106387. https://doi.org/10.1016/j.engappai.2023.106387.
[10] Yu Y, Li W, Li J, Nguyen TN. A novel optimised self-learning method for compressive strength prediction of high performance concrete. Construction and Building Materials, 2018; 184: 229-247. https://doi.org/10.1016/j.conbuildmat.2018.06.219.
[11] Yu Y, Zhang C, Xie X, Yousefi AM, Zhang G, Li J, Samali B. Compressive strength evaluation of cement-based materials in sulphate environment using optimized deep learning technology. Developments in the Built Environment, 2023; 16: 100298. https://doi.org/10.1016/j.dibe.2023.100298.
[12] Sandeep MS, Tiprak K, Kaewunruen S, Pheinsusom P, Pansuk W. Shear strength prediction of reinforced concrete beams using machine learning. Structures, 2023; 47: 1196-1211. https://doi.org/10.1016/j.istruc.2022.11.140.
[13] Paudel S, Pudasaini A, Shrestha RK, Kharel E. Compressive strength of concrete material using machine learning techniques. Cleaner Engineering and Technology, 2023; 15: 100661. https://doi.org/10.1016/j.clet.2023.100661.
[14] Sami BHZ, Sami BFZ, Kumar P, Ahmed AN, Amieghemen GE, Sherif MM, El-Shafie A. Feasibility analysis for predicting the compressive and tensile strength of concrete using machine learning algorithms. Case Studies in Construction Materials, 2023; 18: e01893. https://doi.org/10.1016/j.cscm.2023.e01893.
[15] Habib M, Okayli M. Evaluating the Sensitivity of Machine Learning Models to Data Preprocessing Technique in Concrete Compressive Strength Estimation. Arabian Journal for Science and Engineering, 2024; 1-19. https://doi.org/10.1007/s13369-024-08776-2.
[16] Steyerberg EW, Eijkemans MJ, Harrell FE Jr, Habbema JDF. Prognostic modelling with logistic regression analysis: a comparison of selection and estimation methods in small data sets. Statistics in Medicine, 2000; 19(8): 1059-1079. https://doi.org/10.1002/(SICI)1097-0258(20000430)19:8<1059:AID-SIM412>3.0.CO;2-0.
[17] Habib A, Yildirim U, Habib M. Applying Kernel principal component analysis for enhanced multivariable regression modeling of rubberized concrete properties. Arabian Journal for Science and Engineering, 2023; 48(4): 5383-5396. https://doi.org/10.1007/s13369-022-07435-8.
[18] Koya BP, Aneja S, Gupta R, Valeo C. Comparative analysis of different machine learning algorithms to predict mechanical properties of concrete. Mechanics of Advanced Materials and Structures, 2022; 29(25): 4032-4043. https://doi.org/10.1080/15376494.2021.1917021.
[19] Islam MM, Hossain MB, Akhtar MN, Moni MA, Hasan KF. CNN based on transfer learning models using data augmentation and transformation for detection of concrete crack. Algorithms, 2022; 15(8): 287. https://doi.org/10.3390/a15080287.
[20] Chen N, Zhao S, Gao Z, Wang D, Liu P, Oeser M, et al. Virtual mix design: Prediction of compressive strength of concrete with industrial wastes using deep data augmentation. Construction and Building Materials, 2022; 323: 126580. https://doi.org/10.1016/j.conbuildmat.2022.126580.

[21] Ford E, Maneparambil K, Kumar A, Sant G, Neithalath N. Transfer (machine) learning approaches coupled with target data augmentation to predict the mechanical properties of concrete. Machine Learning with Applications, 2022; 8: 100271. https://doi.org/10.1016/j.mlwa.2022.100271.
[22] Marani A, Nehdi ML. Predicting shear strength of FRP-reinforced concrete beams using novel synthetic data driven deep learning. Engineering Structures, 2022; 257: 114083. https://doi.org/10.1016/j.engstruct.2022.114083.
[23] Hong Y, Park S, Kim H, Kim H. Synthetic data generation using building information models. Automation in Construction, 2021; 130: 103871. https://doi.org/10.1016/j.autcon.2021.103871.
[24] Zeng S, Wang X, Hua L, Altayeb M, Wu Z, Niu F. Prediction of compressive strength of FRP-confined concrete using machine learning: A novel synthetic data driven framework. Journal of Building Engineering, 2024; 94: 109918. https://doi.org/10.1016/j.jobe.2024.109918.
[25] Aguilar E, Nagarajan B, Khantun R, Bolaños M, Radeva P. Uncertainty-aware data augmentation for food recognition. In 2020 25th International Conference on Pattern Recognition (ICPR), 2021; 4017-4024. IEEE. https://doi.org/10.1109/ICPR48806.2021.9412706.
[26] Cheraghzade M, Roohi M. Incorporating uncertainty in mechanics-based synthetic data generation for deep learning–based structural monitoring. In Society for Experimental Mechanics Annual Conference and Exposition, 2023; 57-65. Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-37003-8_9.
[27] Thomas BS, Damare A, Gupta RC. Strength and durability characteristics of copper tailing concrete. Construction and Building Materials, 2013; 48: 894-900. https://doi.org/10.1016/j.conbuildmat.2013.07.075.
[28] Gupta RC, Mehra P, Thomas BS. Utilization of copper tailing in developing sustainable and durable concrete. Journal of Materials in Civil Engineering, 2017; 29(5): 04016274. https://doi.org/10.1061/(ASCE)MT.1943-5533.000181.
[29] Dandautiya R, Singh AP. Utilization potential of fly ash and copper tailings in concrete as partial replacement of cement along with life cycle assessment. Waste Management, 2019; 99: 90-101. https://doi.org/10.1016/j.wasman.2019.08.036.
[30] Muleya F, Mulenga B, Zulu SL, Nwaubani S, Tembo CK, Mushota H. Investigating the suitability and cost-benefit of copper tailings as partial replacement of sand in concrete in Zambia: an exploratory study. Journal of Engineering, Design and Technology, 2021; 19(4): 828-849. https://doi.org/10.1108/JEDT-05-2020-0186.
[31] Pei C, Chen P, Tan W, Zhou T, Li J. Effect of wet copper tailings on the performance of high-performance concrete. Journal of Building Engineering, 2023; 74: 106931. https://doi.org/10.1016/j.jobe.2023.106931.
[32] Kundu S, Aggarwal A, Mazumdar S, Dutt KB. Stabilization characteristics of copper mine tailings through its utilization as a partial substitute for cement in concrete: preliminary investigations. Environmental Earth Sciences, 2016; 75: 1-9. https://doi.org/10.1007/s12665-015-5089-9.
[33] Zhang Y, Shen W, Wu M, Shen B, Li M, Xu G, Chen X. Experimental study on the utilization of copper tailing as micronized sand to prepare high performance concrete. Construction and Building Materials, 2020; 244: 118312. https://doi.org/10.1016/j.conbuildmat.2020.118312.
[34] Esmaeili J, Aslani H, Onuaguluchi O. Reuse potentials of copper mine tailings in mortar and concrete composites. Journal of Materials in Civil Engineering, 2020; 32(5): 04020084. https://doi.org/10.1061/(ASCE)MT.1943-5533.0003145.
[35] Onuaguluchi O, Eren Ö. Copper tailings as a potential additive in concrete: consistency, strength and toxic metal immobilization properties. Magazine of Concrete Research, 2012; 64(11): 1015-1023.
[36] Onuaguluchi O, Eren Ö. Durability-related properties of mortar and concrete containing copper tailings as a cement replacement material. Magazine of Concrete Research, 2012; 64(11): 1015-1023. https://doi.org/10.1680/macr.11.00170.
[37] Ghazi AB, Jamshidi-Zanjani A, Nejati H. Clinkerisation of copper tailings to replace Portland cement in concrete construction. Journal of Building Engineering, 2022; 51: 104275. https://doi.org/10.1016/j.jobe.2022.104275.
[38] Ghazi AB, Jamshidi-Zanjani A, Nejati H. Utilization of copper mine tailings as a partial substitute for cement in concrete construction. Construction and Building Materials, 2022; 317: 125921. https://doi.org/10.1016/j.conbuildmat.2021.125921.
[39] Esmaeili J, Aslani H. Use of copper mine tailing in concrete: strength characteristics and durability performance. Journal of Material Cycles and Waste Management, 2019; 21(3): 729-741. https://doi.org/10.1007/s10163-019-00831-7.
[40] Onuaguluchi O, Eren Ö. Recycling of copper tailings as an additive in cement mortars. Construction and Building Materials, 2012; 37: 723-727. https://doi.org/10.1016/j.conbuildmat.2012.08.009.
[41] Huang XY, Ni W, Cui WH, Wang ZJ, Zhu LP. Preparation of autoclaved aerated concrete using copper tailings and blast furnace slag. Construction and Building Materials, 2012; 27(1): 1-5. https://doi.org/10.1016/j.conbuildmat.2011.08.034.

[42] Prahallada C, Shanthappa BC. Use of copper ore tailings-as an excellent pozzolana in the preparation of concrete. International Journal of Advanced Research in Engineering and Applied Sciences, 2014; 3(3): 1-10.
[43] Vargas F, Alsina MA, Gaillard JF, Pasten P, Lopez M. Copper entrapment and immobilization during cement hydration in concrete mixtures containing copper tailings. Journal of Cleaner Production, 2021; 312: 127547. https://doi.org/10.1016/j.jclepro.2021.127547.
[44] Tranmer M, Elliot M. Multiple linear regression. The Cathie Marsh Centre for Census and Survey Research (CCSR), 2008; 5(5): 1-5.
[45] McDonald GC. Ridge regression. Wiley Interdisciplinary Reviews: Computational Statistics, 2009; 1(1): 93-100.
[46] Ranstam J, Cook JA. LASSO regression. Journal of British Surgery, 2018; 105(10): 1348.
[47] Zhao H, Ding Y, Meng L, Qin Z, Yang F, Li A. Bayesian multiple linear regression and new modeling paradigm for structural deflection robust to data time lag and abnormal signal. IEEE Sensors Journal, 2023. https://doi.org/10.1109/JSEN.2023.3294912
[48] Bedoui A, Lazar NA. Bayesian empirical likelihood for ridge and lasso regressions. Computational Statistics & Data Analysis, 2020; 145: 106917. https://doi.org/10.1016/j.csda.2020.106917.