Journal of Soft Computing in Civil Engineering 6-3 (2022) 39-62

Contents lists available at SCCE

Journal of Soft Computing in Civil Engineering

Journal homepage: www.jsoftcivil.com

Parametric Assessment of Concrete Constituent Materials Using Machine Learning Techniques

Yazan Alzubi[1]*, Jamal Al Adwan[1], Ahmed Khatatbeh[2], Buthainah Al-kharabsheh[2]

1. Civil Engineering Department, Faculty of Engineering Technology Al-Balqa Applied University, 11134 Amman, Jordan

2. Civil Engineering Department, Faculty of Engineering, Al Albayt University, 25113 Mafraq, Jordan

Corresponding author: *yazan.alzubi@bau.edu.jo*

https://doi.org/10.22115/SCCE.2022.334664.1417

ARTICLE INFO

Article history:
Received: 18 March 2022
Revised: 01 July 2022
Accepted: 14 July 2022

Keywords:
Concrete materials management;
Compressive strength;
Partial dependence analysis;
Machine learning;
Feature importance analysis.

ABSTRACT

Nowadays, technology has advanced, particularly in machine learning which is vital for minimizing the amount of human work required. Using machine learning approaches to estimate concrete properties has unquestionably triggered the interest of many researchers across the globe. Currently, an assessment method is widely adopted to calculate the impact of each input parameter on the output of a machine learning model. This paper evaluates the capability of various machine learning methodologies in conducting parametric assessments to understand the influence of each concrete constituent material on its compressive strength. It is accomplished by conducting a partial dependence analysis to quantify the effect of input features on the prediction results. As a part of the study, the effects of machine learning method selection for such analysis are also investigated by employing a concrete compressive strength algorithm developed using a decision tree, random forest, adaptive boosting, stochastic gradient boosting, and extreme gradient boosting. Additionally, the significance of the input features to the accuracy of the constructed estimation models is ranked through drop-out loss and MSE reduction. This investigation shows that the machine learning techniques could accurately predict the concrete's compressive strength with very high performance. Further, most analyzed algorithms yielded similar estimations regarding the strength of concrete constituent materials. In general, the study's results have shown that the drop-out loss and MSE reduction outputs were misleading, whereas the partial dependence plots provide a clear idea about the influence of the value of each feature on the prediction outcomes.

How to cite this article: Alzubi Y, Al Adwan J, Khatatbeh A, Al-kharabsheh B. Parametric assessment of concrete constituent materials using machine learning techniques. J Soft Comput Civ Eng 2022;6(3):39–62. https://doi.org/10.22115/scce.2022.334664.1417

1. Introduction

In the construction sector, there is no question that concrete is the most widely used material in the world [1,2]. In practice, the compressive strength of concrete is one of the most important factors to consider when it comes to mixture design and optimization, but getting it at mature age involves a lengthy experimental procedure [3,4]. However, mathematical modeling of this property using traditional approaches is deemed a significantly challenging task due to the nonlinear nature of the relation between the concrete's constituents and its characteristics. Recently, several studies have handled estimating the compressive strength of concrete to prove that machine learning-based modeling can substitute conventional methods [5–10]. Topcu & Saridemir [11] illustrated the capability of fuzzy logic for predicting the strength capacity of concrete containing fly ash. Chopra et al. [12] utilized a genetic programming approach for the same purpose. Moreover, Lee [13] and Habib & Yildirim [14] used artificial neural networks to estimate concrete characteristics. Barkhordari et al. [15] adopted ensemble machine learning models to estimate flyrock due to quarry blasting. The applications of bagging and boosting machine learning algorithms in simulating the properties of concrete were investigated by [16–18].

Indeed, machine learning techniques are very effective in estimating concrete properties at various maturity ages, and they can be employed to conduct detailed parametric assessments by evaluating the interaction and correlation between each input variable and the outcome. Nevertheless, their abilities to quantify the percentage of the impact that each constituent material has on the compressive strength of a concrete mixture, as well as to provide a detailed understanding of the influence of each constituent material's content on the output of the estimation model by using feature importance and partial dependence analyses have rarely been discussed in the literature. For instance, Jha et al. [19] evaluated the factors affecting concrete strength using feature importance analysis for different machine learning models. Anysz et al. [20] discussed the capabilities of explicable artificial intelligence methods to assess factors influencing the compressive strength of cement stabilized rammed earth. Ly et al. [21] developed partial dependence plots for rubberized concrete's compressive strength using deep neural network models. Su et al. [22] performed the partial dependence analysis for the compressive strength of slag-metakaolin geopolymer pastes using multivariate polynomial regression models. Dao et al. [23] conducted a sensitivity analysis for the compressive strength of foamed concrete utilizing multiple partial dependence plots based on conventional artificial neural networks. Mane et al. [24] developed an artificial neural network model to predict the flexural strength of concrete mixtures with pozzolanic materials. Nayak et al. [25] utilized an extreme learning machine to predict the compressive strength of concrete mixtures. Pandey et al. [26] adopted various machine learning approaches to design concrete mixtures with and without plasticizer. Naderpour et al. [27] developed artificial neural network models to predict the shear strength of reinforced concrete shear walls.

Thus, it can be seen that multiple studies have adopted feature importance analysis to conduct a parametric assessment for concrete constituent materials, although partial dependence analysis is more sensitive for such applications. Moreover, it can be seen that the literature is still lacking a study that evaluates and compares the capability of various machine learning models in reporting the effects of concrete's mixture proportioning on its compressive strength through partial dependence analysis. Accordingly, the primary objective of this contribution is to evaluate the influence of machine learning model selection on the results of the feature significance and partial dependency analyses. A key component is developing and evaluating several prediction techniques involving a decision tree, random forest, adaptive boosting, stochastic gradient boosting, and extreme gradient boosting. Hence, the investigations of feature importance and partial dependence analyses are carried out, and their findings are compared against each other. On the other hand, the limitation of the scope of the study is in the types of machine learning algorithms adopted for the investigations since it will mainly focus on commonly utilized models and will not go into detail about the current state-of-the-art approaches, thus will leave this point for future studies. Another limitation of the study is conducting a sensitivity analysis for which a method such as the cheap-to-evaluate uncertainty-aware global sensitivity analysis discussed by Amini et al. [28] can be utilized.

2. Materials and methods

Cement-based materials have traditionally been designed and characterized by experiments for decades [29]. As a result, significant discoveries were achieved through emerging modern computational approaches in construction materials science [30]. Currently, there is rising attention to scaling up the commonly utilized machine learning algorithms for classification, regression, clustering, or dimensionality to decrease tasks of massive datasets [31]. Indeed, machine learning is a powerful tool in artificial intelligence that combines statistics and computer science to develop more effective models that rely on training data to achieve a specific activity [32,33]. It is a fast-developing field that enables computer systems to gain knowledge directly from data and experience without programming [34]. The key target is to build a model that estimates the required parameter value by learning machine techniques inferred from the data characteristics and then using these models to perform feature importance and partial dependence analyses. Figure 1 represents the general methodology used in this study.

2.1. Machine learning algorithms

The decision tree (DT) regressor may be described according to the principle of divide and conquer to identify characteristics and model the relationships between them in big data [35]. The term comes from a tree structure in which the dataset (root node) is repeatedly divided into smaller subsets based on specific values of this property, and the initial group of tree branches is generated [36–38]. In practice, a training dataset is utilized to create a decision tree, and if all objects have the same decision class, the tree will only get a single node (leaf node) with the suitable decision, as described in Figure 2. On the other side, a separate feature is chosen whose value belongs to at least two distinct decision classes, and a collection of objects is partitioned into categories based on the value of the picked feature in order to construct a test node in a growing decision tree [39]. Every branch from the test node is followed by a series of prompting approaches that are performed on the remaining objects in terms of split until a leaf is attained depicting the decision. There are many types of decision tree algorithms; however, the classification and regression tree (CART) will be used in this research owing to its capabilities.

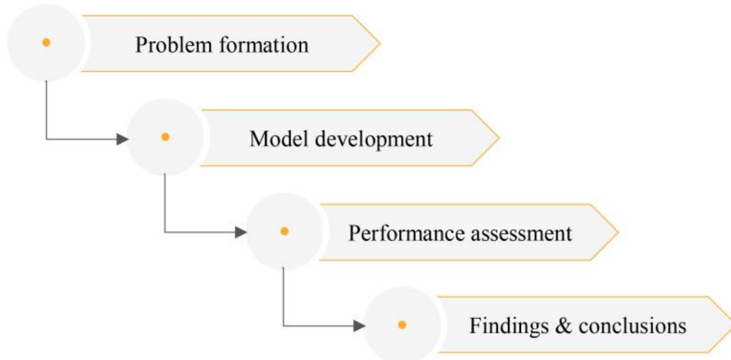

Fig. 1. General descriptive illustration of the research methodology.

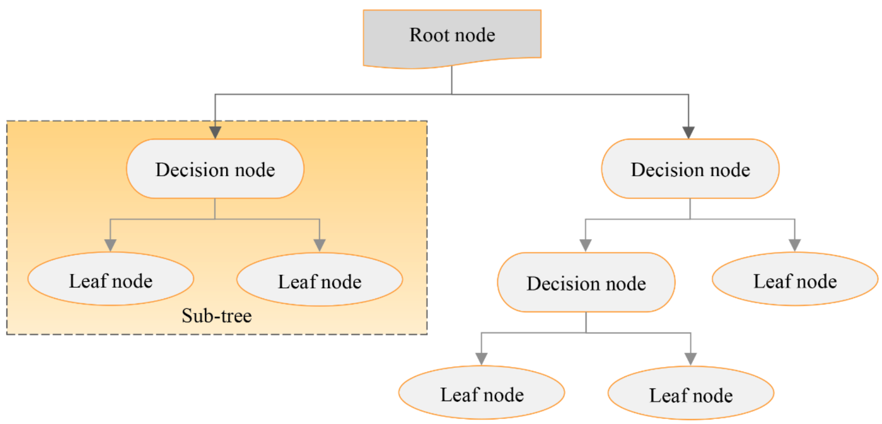

Fig. 2. Illustrative concept of decision tree (reproduced from Janikow [40]).

With an n-dimensional sample of training data $\{(x_1, y_1), \ldots, (x_n, y_n)\} \subset \chi \times R$ and the space of input patterns χ, the decision tree technique splits the attributes in a recursive procedure resulting in the output being assembled together based on their similarity. At a particular node m, the data is defined by Q_m with N_m samples, divided into two subsets $Q_m^{left}(\theta)$, and $Q_m^{rightt}(\theta)$ in Eq. 1 and 2, where each candidate split $\theta = (j, t_m)$ includes a j feature and t_m threshold.

$$Q_m^{left}(\theta) = \{(x, y) \mid x_i \le t_m\} \tag{1}$$

$$Q_m^{right}(\theta) = Q_m / Q_m^{left}(\theta) \tag{2}$$

Using the impurity function H() in Eq. 3, it is possible to determine the quality of a candidate split of node m.

$$G(Q_m, \theta) = \frac{N_m^{left}}{N_m} H\left(Q_m^{left}(\theta)\right) + \frac{N_m^{right}}{N_m} H\left(Q_m^{right}(\theta)\right) \tag{3}$$

Hence, from Eq. 4, the parameters that reduce the impurity are picked, and the procedure is repeated for $Q_m^{left}(\theta)$ and $Q_m^{rightt}(\theta)$ until the tolerance depth $N_m < min_{samples}$ or $N_m = 1$.

$$\theta^* = \arg\min_{\theta} G(Q_m, \theta) \tag{4}$$

A random forest (RF) is a massive number of single decision trees that serve as an ensemble where each unique tree votes to predict the most common class as the output of the random forest [41,42]. As the ensemble decreases the instability related to the building of individual decision trees, the random forest can deal with high-dimensional datasets and complicated relationships and frequently produces precise models [43,44]. The decision tree model has a single tree, but a random forest is made up of several trees, which is the most significant conceptual difference between them. The training dataset is randomly sampled from the original data to develop the algorithm, and any ignored data is specified as out-of-bag. The random forest has become more prevalent in the civil engineering field during the past several years, mainly to deliver practical models. As a result, it is implemented by creating many CARTs and integrating bootstrap and aggregation concepts [45].

The extremely randomized trees (ERT) technique randomizes choosing splits in a tree's nodes [46]. The ERT approach differs from the random forest algorithm in that the thresholds for each candidate's feature are picked randomly, and the best of them is used as the splitting criterion rather than searching for the most distinctive thresholds. The significance of this fact lies in its capacity to reduce model variance while slightly raising the bias.

Freund and Schapire [47] introduced the concept of adaptive boosting (AdaBoost), which was designed to fit the original dataset utilizing the defined regressor. The method serves by creating new copies of the regression algorithm employing the same training dataset while modifying the model weights based on the results of the previous trial. The implantation of this algorithm is conducted in relevance to Drucker [48]. An n-dimensional training dataset (specified before) is used in AdaBoost's formula, and the error e_i for the whole dataset is produced by training a base estimator (weak learner) f(x). Hence, a series of weak learners is built and integrated to construct a strong model H(x) using the technique described in Eq. 5.

$$H(x) = \nu \sum_{k-1}^{N} \left(\ln \frac{1}{\alpha_k} \right) g(x) \tag{5}$$

In this equation, ν is the learning rate, α_k denotes the weight of the base estimators calculated from Eq. 6, and g(x) represents the median of all $\alpha_k f_k(x)$.

$$\alpha_k = \frac{e_i}{1 - e_i} \tag{6}$$

Like adaptive boosting, stochastic gradient boosting (SGBoost) creates a new model ensemble to correct the residual error of the existing one but with a substantial variation that relies on reducing an objective function. In contrast to the regressor in the AdaBoost algorithm, the fragile predictor in the gradient boosting method is a bigger decision tree with numerous layers. For the training dataset (defined above), the gradient boosting estimator $\hat{y}_i$ with x_i input variable is specified by Eq. 7. Gradient boosting is generally a greedy procedure, as stated in Eq. 8.

$$\hat{y}_i = F_M(x_i) = \sum_{m=1}^{M} h_m(x_i) \tag{7}$$

$$F_m(x) = F_{m-1}(x) + \underset{h \in H}{\arg\min} \sum_{i=1}^{n} L\left[y_i, F_{m-1}(x_i) + h(x_i)\right] \tag{8}$$

M is the total number of estimators delivered in the algorithm, h_m refers to a weak learner, h(x) is the base predictor, and L (·) is the loss function with the negative gradient identified in Eq. 9.

$$g_m = -\frac{\partial L\left[y_i, F_{m-1}(x)\right]}{\partial F_{m-1}(x)} \tag{9}$$

eXtreme gradient boosting (XGBoost) is a machine learning approach that is both efficient and scalable, and it is used for tree boosting [49]. In general, both gradients boosting and XGBoost adhere to the gradient boosting concept, except that XGBoost utilizes a more regularized model to govern over-fitting cases to obtain better outcomes. The XGBoost method employs the exact greedy tree approach, and the expression in Eq. 10 denotes the goal function of this method.

$$obj = \sum_{i=1}^{n} L\left[\hat{y}_i, y_i\right] + \sum_{i=1}^{n} \omega(f_t) \tag{10}$$

Where L (·) denotes the loss function of the model's bias, and ω is a regular term applied for suppressing the algorithm's complexity.

2.2. Feature importance and partial dependence analyses

Indeed, model interpretation is a significant step that guides the development of various models and decision-making strategies. It can also construct trust between the trained algorithm and users [50]. As mentioned previously, developed machine learning models are used in performing parametric assessments and reporting the influence of each input feature on the compressive strength of concrete. Feature importance and partial dependence analyses were adopted for multiple studies within this context. However, the latter is more sensitive for such applications.

Feature importance analysis is an approach commonly adopted for the direct ranking of each feature on the final prediction and has three methods: drop-out loss, MSE reduction, and accumulated dependency. The Drop-out loss evaluates the feature importance of a specific set of input variables by first computing the loss in the model with and without the input parameter and then finding the difference between these errors [51]. In reality, a high value of drop-out loss implies that the variable has a considerable influence on the model's behavior, which afterward evaluates the variable's true implications on the target, indicating that the model is well fitted. The negative value of drop-out loss reveals that the existing variable within the dataset reduces the model accuracy and should be eliminated. More to the point, whenever there are variables with a large correlation size, the feature importance propagates asymmetrically, accordingly causing the measure to be unreliable.

The mean squared error (MSE) reduction is a procedure to determine the most critical characteristic of the dataset [52]. In general, this approach is performed by the tree traversed. When the outcomes of a particular node are dependent on the specified input variable, its error drop is multiplied by the number of samples routed to that node to its feature importance is added. The error decrease is computed by subtracting the MSE of data routed to that node from the MSE of its child nodes. In a multi-tree model, the resulting feature importance is an average of the feature importance of the developed trees. Generally, the feature importance obtained using the MSE reduction approach is sometimes misleading, especially in high cardinality features (many unique values) [53].

The cumulative effect, commonly called cumulative dependency, shows how much a given variable influences the model's mean expectation [54]. It is possible to emulate what the algorithm would estimate for a specific dataset if the value of the assigned feature varies (this process is called Ceteris Paribus). If the samples and their simulations are averaged, partial dependency for the selected feature is detected. Indeed, partial dependency is inconsistent and susceptible to linear correlation in some instances; hence a weighted mean should be used with weights corresponding to the inverse of the distances between the original samples. The accumulated dependency outcomes are similar to the partial dependence analysis. In this investigation, two methods of feature importance assessment are utilized and compared: the drop-out loss with 100 permutations and the MSE reduction to evaluate the significance of the predictors.

Typically, feature importance analysis determines whether or not the features are necessary for producing the model to be successful. On the other hand, it cannot indicate how the feature's value impacts the prediction results. Hence, partial dependency analysis is usually performed to evaluate this point. The partial dependency diagram depicts the feature's marginal influence on the estimated result of machine learning techniques and reveals whether the interaction between the target and a feature is linear, monotonic, or more complicated. The partial function shows the average limit impact on estimating a particular feature value. It may provide answers to concerns about how the model's prediction results differ for a feature, allowing the influence of the predictor to be clearly expressed. In the beginning, the partial dependency of a specific input x is derived by first substituting its values with a constant one, such as x_1. Then it is required to train the model and evaluate how much the prediction results have changed. After that, progressively modifying x_1 shows how the estimation results change depending on the investigated feature. Notably, one of the fundamental assumptions behind this form of analysis is that features are uncorrelated with one another. If this presumption is broken, the averages computed for the partial dependency graph will contain data points that are improbable or even impossible.

2.3. Dataset acquisition

This study intends to conduct a sensitivity assessment of concrete compressive strength using machine learning techniques to extract the model's cause and effect relationship between the inputs and outputs. In order to carry out the research, a dataset with a large sample size is utilized, and the descriptive statistics of the developed dataset are shown in Figure 3. This dataset has been used previously in various research studies [55–58]. It consists of 1005 mixtures with their corresponding compressive strengths. The constituent materials of each concrete specimen were adopted as an input feature of the machine learning techniques, while the compressive strength capacity was selected as the outcome of the prediction model.

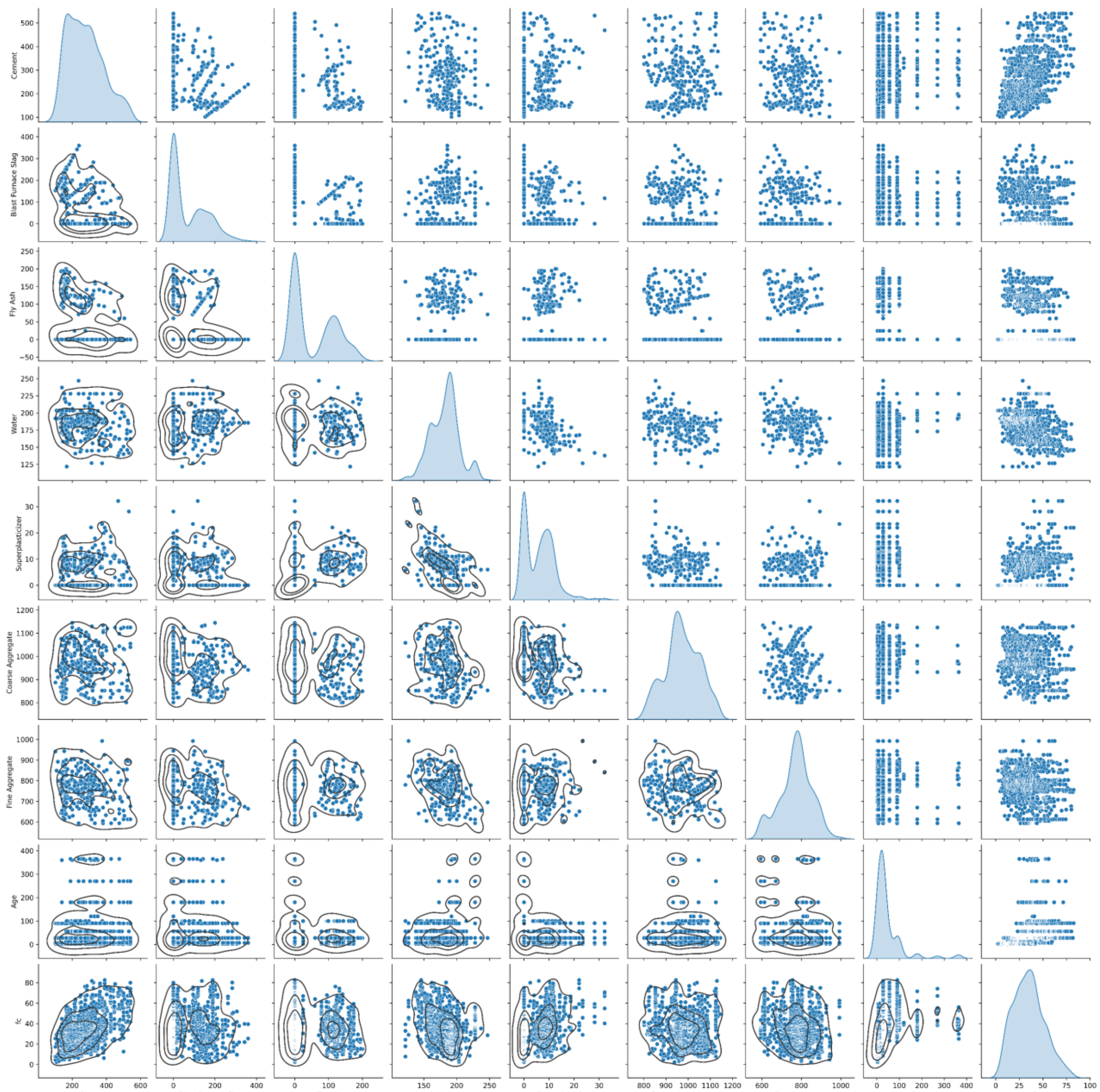

Fig. 3. Visual descriptive statistics of the utilized dataset.

2.4. Model development and hyperparameters tunning

The behavior of the machine learning technique is highly impacted by the selected hyperparameter values that should be tuned. In this study, the grid search technique with k-fold cross-validation was applied in the training stage to optimize the hyperparameters of the methods. As a result, the suggested methodology for building machine learning algorithms (Figure 4) initially begins by splitting the dataset into two groups: 70% training and 30% testing data. A cross-validation approach with 10-folds repetitions is conducted to ensure a suitable parameter choice. Once the hyperparameters of each approach are determined, the final tuned model performance is assessed by comparing the results of various scoring parameters on the test dataset.

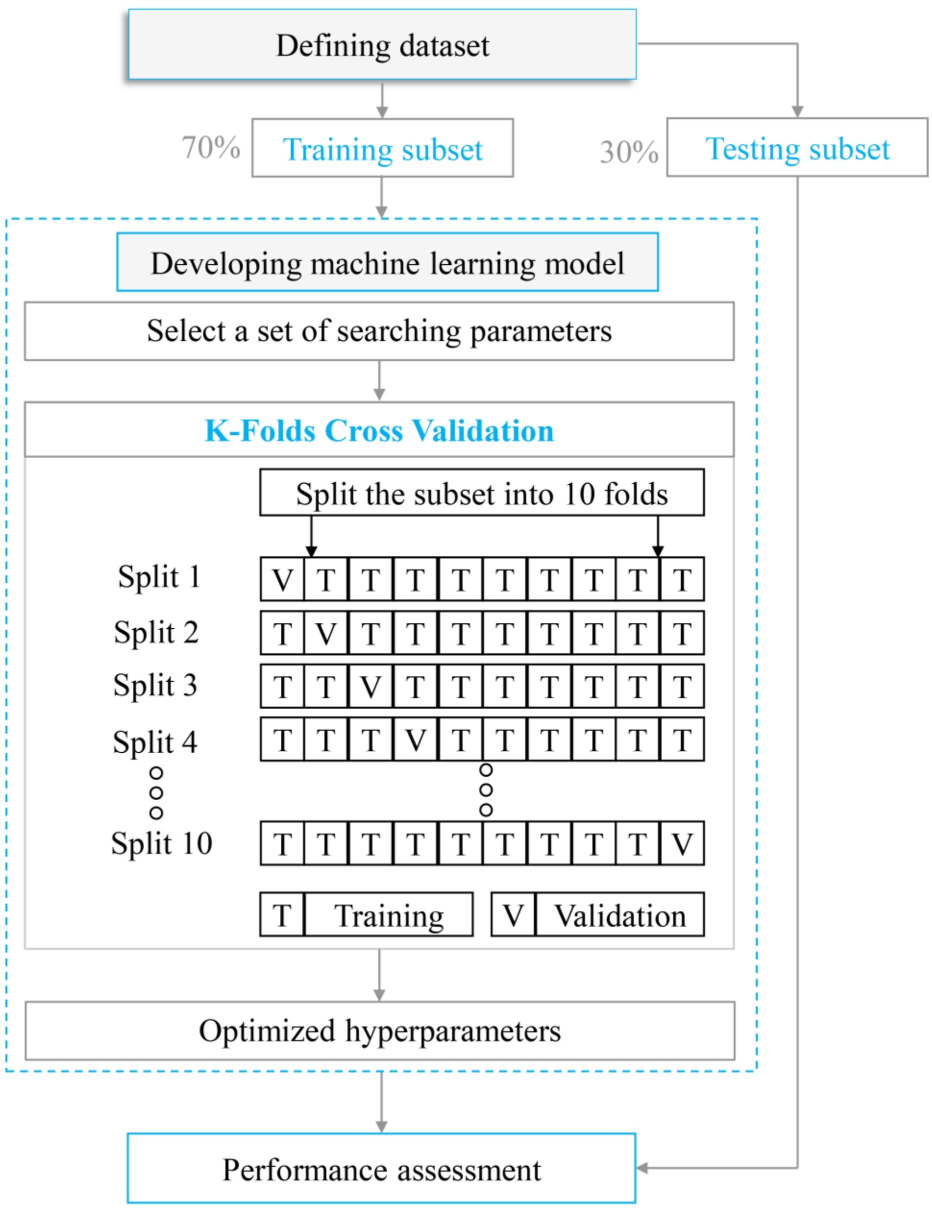

Fig. 4. Illustrative description for constructing machine learning model.

2.5. Quality assessment

Statistical measures and visual representations are adopted to analyze the performance of machine learning techniques. The goodness-of-fit is checked using the coefficient of determination, Eq. 11. The root mean square error (RMSE), Eq. 12, and mean absolute error (MAE), Eq. 13, were used for the error analysis.

$$R^2 = 1 - \frac{\sum (x_i - y_i)^2}{\sum (x_i - \overline{x}_i)^2} \tag{11}$$

$$RMSE = \left[\frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2 \right]^{1/2} \tag{12}$$

$$MAE = \frac{1}{n} \sum_{i=1}^{n} \left| y_i - \overline{y}_i \right| \tag{13}$$

x_i is the measured value, $\overline{x}_i$ is the mean of the measured values, y_i is the predicted value $\overline{y}_i$ is the mean of the predicted values, n is the number of observations, and m_{20} is the number of samples with a measured to the predicted ratio between 0.8 and 1.20.

3. Results and discussions

3.1. Estimation performance of the ınvestigated models

Indeed, the reliability of the decision on which input is more significant on the outcome of the machine learning technique is highly affected by the accuracy of the developed estimation model. Accordingly, measuring the performance of the investigated methods in predicting the compressive strength of concrete is considered essential. In general, the results of the estimations are shown in Figure 5 for both the training and testing stages. As seen there, the DT model was capable of reaching a significantly high accuracy in the training case. Whereas its performance dropped significantly for the testing dataset, which represents the existence of an overfitting issue. Similar trends can be observed for the cases of the ERT, AdaBoost, and SGBoost but with more controlled overfitting. On the other hand, the RF and XGBoost models resulted in lower training capabilities, while their testing results represented comparable performance to the ERT and SGBoost. This means that the RF and XGBoost techniques are more sensitive to overfitting issues. The residual plots for the considered models are provided in Figure 6. In general, it can be seen that most of the developed estimation models yielded similar results, with the exception that the DT model had the highest distortions in the residuals as compared to others.

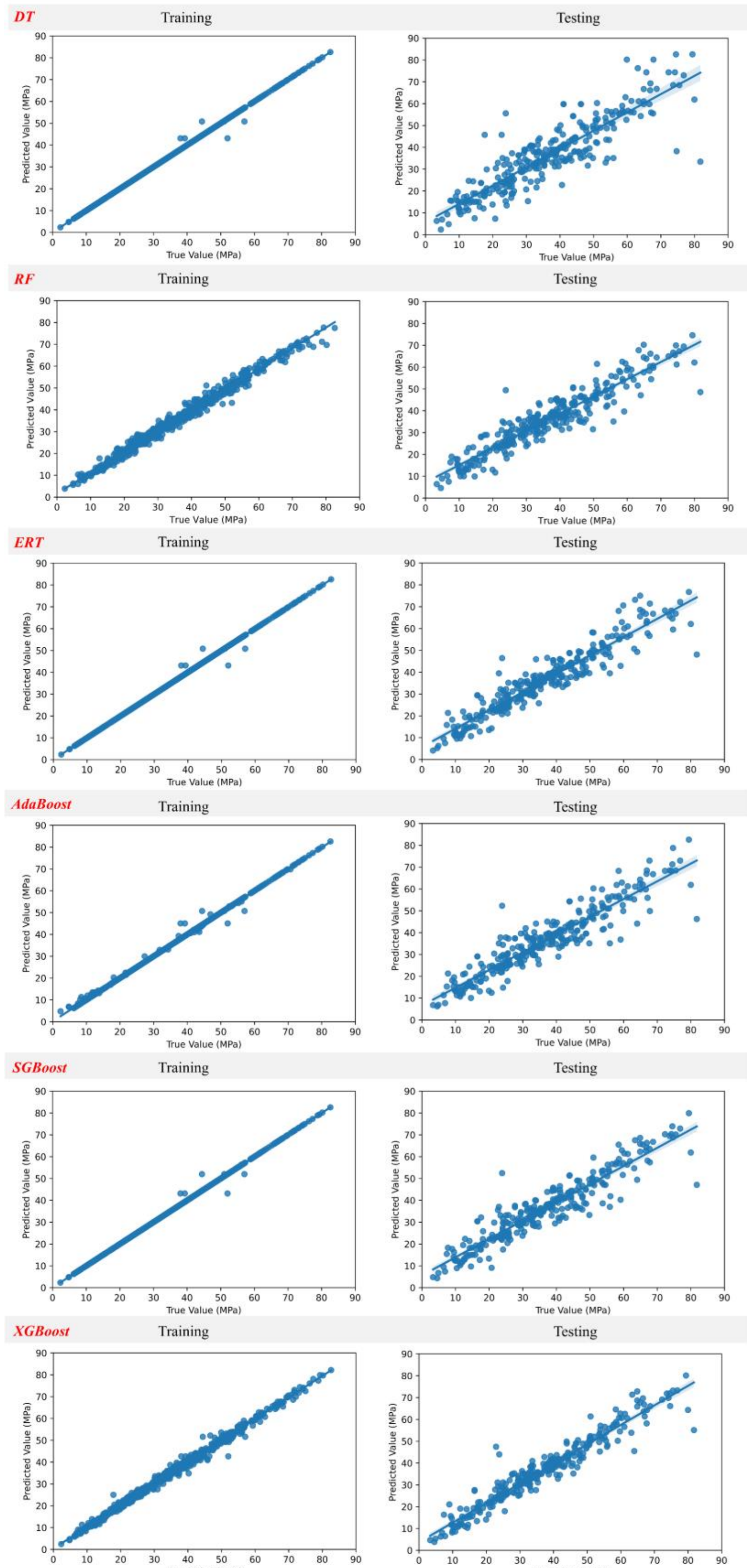

Fig. 5. Actual versus predicted plots for the investigated models.

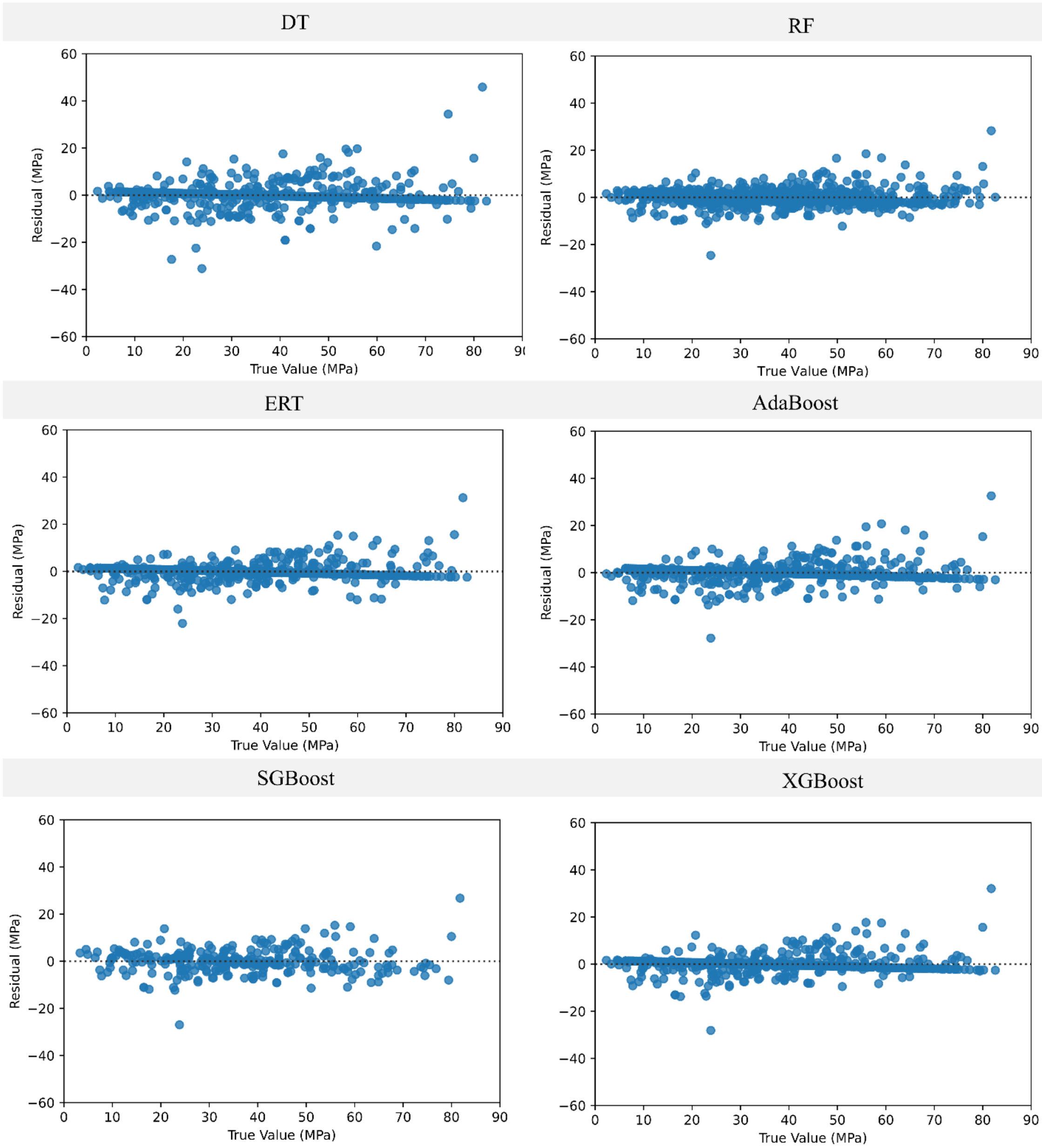

Fig. 6. Residual plots for the estimation results of the investigated models.

The error assessment of the fitting rates of the developed prediction techniques is shown in Table 1. Generally, it can be seen that there are drops in the performances of all models when comparing the training to the testing results, which is usually expected but should be controlled to avoid overfitting issues. Indeed, the fitting rate of the DT algorithm for the testing subset of data was the lowest among others, while the XGBoost reached the highest one. On the other hand, the maximum error achieved in the DT model was about 5% lower than the XGBoost model. Using the XGBoost model reduced the RMSE value by almost 43%, 22%, and 23% compared to the DT, ERT, and SGBoost models. Moreover, the ERT approach had higher capabilities than the RF model, which can be attributed to the enhanced randomization abilities of the ERT method.

Table 1
Performance of the investigated models in predicting concrete's compressive strength.

Training						
	DT	RF	ERT	AdaBoost	SGBoost	XGBoost
R^2	1.00	0.99	1.00	1.00	1.00	0.99
RMSE (MPa)	0.53	1.84	0.53	0.64	0.54	1.26
MAE (MPa)	0.04	1.30	0.04	0.11	0.05	0.84
Maximum Error (MPa)	8.90	10.49	8.90	7.04	8.85	9.34
Testing						
	DT	RF	ERT	AdaBoost	SGBoost	XGBoost
R^2	0.77	0.86	0.89	0.86	0.88	0.93
RMSE (MPa)	7.91	6.12	5.58	6.29	5.86	4.54
MAE (MPa)	5.27	4.40	3.78	4.25	3.84	2.94
Maximum Error (MPa)	48.33	33.20	33.66	35.51	34.64	26.64

Worst ***Best***

3.2. Feature ımportance

As highlighted previously, feature importance analysis can be used to rank input parameters based on their impact on the performance of the machine learning models and identify whether or not a specific input variable is necessary for producing the model to be successful. This study utilized two approaches to feature importance assessment: drop-out loss and MSE reduction. The results of these methods and their comparison are shown in Figure 7 and Table 2, respectively. In general, it can be seen that there are some fluctuations in the ranks obtained using each of the approaches to feature importance assessment in a particular machine learning model. For instance, the drop-out loss in the case of the DT model ranked the water as the fourth while the MSE reduction ranked it as the third among the input parameters. Additionally, by comparing the results of the machine learning models for the drop-out loss case, each one of the estimation techniques has its own ranking. In contrast, the ranks obtained using the DT and RF cases for the MSE reduction approach were identical, and the AdaBoost and SGBoost outcomes in that method were the same even though their coefficients of determination are different. This observation proves the known fact that it has a lower sensitivity as compared to the drop-out loss approach.

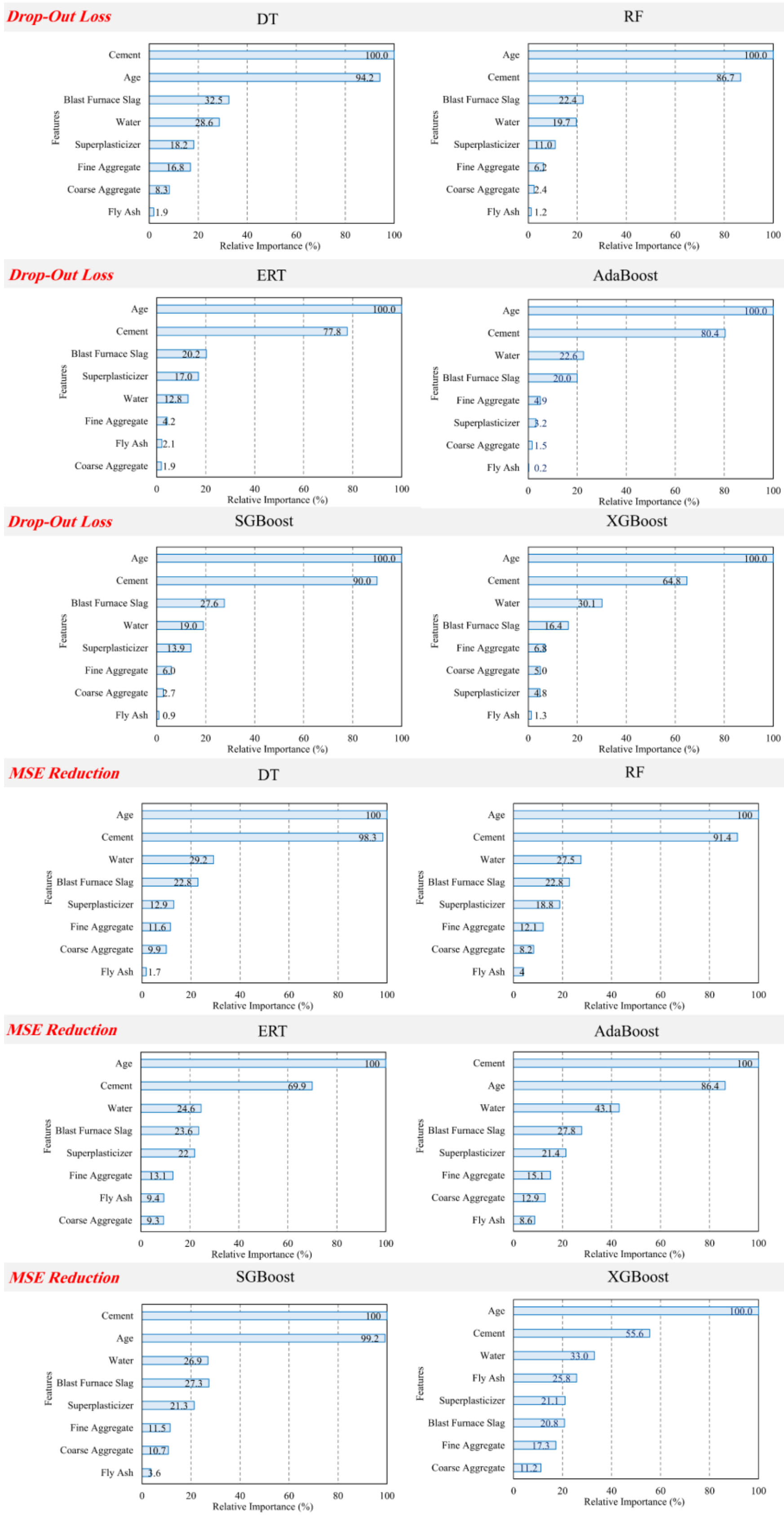

Fig. 7. Feature importance scoring for input parameters of the machine learning models.

On the other hand, it can be seen that using feature importance to conduct a parametric assessment of the concrete constituent materials is somehow misleading. It can be understood by comparing the knowledge available in the literature to the results presented in Table 3. For example, in most machine learning models, the fly ash ranked as the least essential variable, while blast furnace slag was ranked as the third or fourth in some cases. However, it was presented in many studies that the fly ash causes a similar or higher impact than the blast furnace slag [59,60].

Table 2
Comparison of drop-out loss and MSE reduction approaches to feature importance analysis.

Drop-Out Loss						
	DT	RF	RFT	Adaboost	SGBoost	XGBoost
R^2	0.77	0.86	0.89	0.86	0.88	0.93
Age	94.15	100.00	100.00	100.00	100.00	100.00
Blast Furnace Slag	32.50	22.37	20.23	20.03	27.65	16.37
Cement	100.00	86.68	77.77	80.38	89.98	64.77
Coarse Aggregate	8.26	2.43	1.91	1.55	2.73	4.95
Fine Aggregate	16.79	6.24	4.22	4.92	6.04	6.83
Fly Ash	1.87	1.18	2.13	0.20	0.91	1.30
Superplasticizer	18.23	11.02	17.02	3.19	13.94	4.80
Water	28.58	19.66	12.81	22.61	19.00	30.14
MSE Reduction						
	DT	RF	RFT	Adaboost	SGBoost	XGBoost
R^2	0.77	0.86	0.89	0.86	0.88	0.93
Age	100.00	100.00	100.00	86.40	99.20	100.00
Blast Furnace Slag	22.80	22.80	23.60	27.80	27.30	20.79
Cement	98.30	91.40	69.90	100.00	100.00	55.59
Coarse Aggregate	9.90	8.20	9.30	12.90	10.70	11.21
Fine Aggregate	11.60	12.10	13.10	15.10	11.50	17.29
Fly Ash	1.70	4.00	9.40	8.60	3.60	25.77
Superplasticizer	12.90	18.80	22.00	21.40	21.30	21.09
Water	29.20	27.50	24.60	43.10	26.90	33.00

Low Impact	***High Impact***

Table 3
Ranks of the input features using machine learning techniques.

Drop-Out Loss						
Rank	DT	RF	ERT	AdaBoost	SGBoost	XGBoost
1	Cement	Age	Age	Age	Age	Age
2	Age	Cement	Cement	Cement	Cement	Cement
3	Blast Furnace Slag	Blast Furnace Slag	Blast Furnace Slag	Water	Blast Furnace Slag	Water
4	Water	Water	Superplasticizer	Blast Furnace Slag	Water	Blast Furnace Slag
5	Superplasticizer	Superplasticizer	Water	Fine Aggregate	Superplasticizer	Fine Aggregate
6	Fine Aggregate	Fine Aggregate	Fine Aggregate	Superplasticizer	Fine Aggregate	Coarse Aggregate
7	Coarse Aggregate	Coarse Aggregate	Fly Ash	Coarse Aggregate	Coarse Aggregate	Superplasticizer
8	Fly Ash	Fly Ash	Coarse Aggregate	Fly Ash	Fly Ash	Fly Ash
MSE Reduction						
Rank	DT	RF	ERT	AdaBoost	SGBoost	XGBoost
1	Age	Age	Age	Cement	Cement	Age
2	Cement	Cement	Cement	Age	Age	Cement
3	Water	Water	Water	Water	Water	Water
4	Blast Furnace Slag	Blast Furnace Slag	Blast Furnace Slag	Blast Furnace Slag	Blast Furnace Slag	Fly Ash
5	Superplasticizer	Superplasticizer	Superplasticizer	Superplasticizer	Superplasticizer	Superplasticizer
6	Fine Aggregate	Fine Aggregate	Fine Aggregate	Fine Aggregate	Fine Aggregate	Blast Furnace Slag
7	Coarse Aggregate	Coarse Aggregate	Fly Ash	Coarse Aggregate	Coarse Aggregate	Fine Aggregate
8	Fly Ash	Fly Ash	Coarse Aggregate	Fly Ash	Fly Ash	Coarse Aggregate

3.3. Partial dependence analysis

Indeed, the partial dependency plots represent the marginal influence a certain input parameter has on a machine learning model's outcome and illustrate whether the interaction between the target and a feature is linear, monotonic, or more complicated. Moreover, it can answer questions on how the model's estimation results vary for a feature so that the impact of the predictor is clearly stated. The results of this analysis are shown in Figure 8 for the DT and RF models, Figure 9 for the ERT and AdaBoost ones, and Figure 10 for the SGBoost and XGBoost ones. It can be seen that the trends in all machine learning techniques are primarily similar in terms of positive and negative impacts. Generally, the cement, blast furnace slag, and superplasticizers positively impacted the concrete's compressive strength in all the investigated models with an average peak partial dependence of 30%, 10%, and 5%, respectively. Additionally, an exception to these observations is obtained from the XGBoost model for the case superplasticizer, which positively impacted specific content and then started reducing considerably. It simulates the practical scenario presented in the literature in which a high dosage of chemical admixtures harms the strength of concrete [61]. In contrast, the water content had a negative influence on all models.

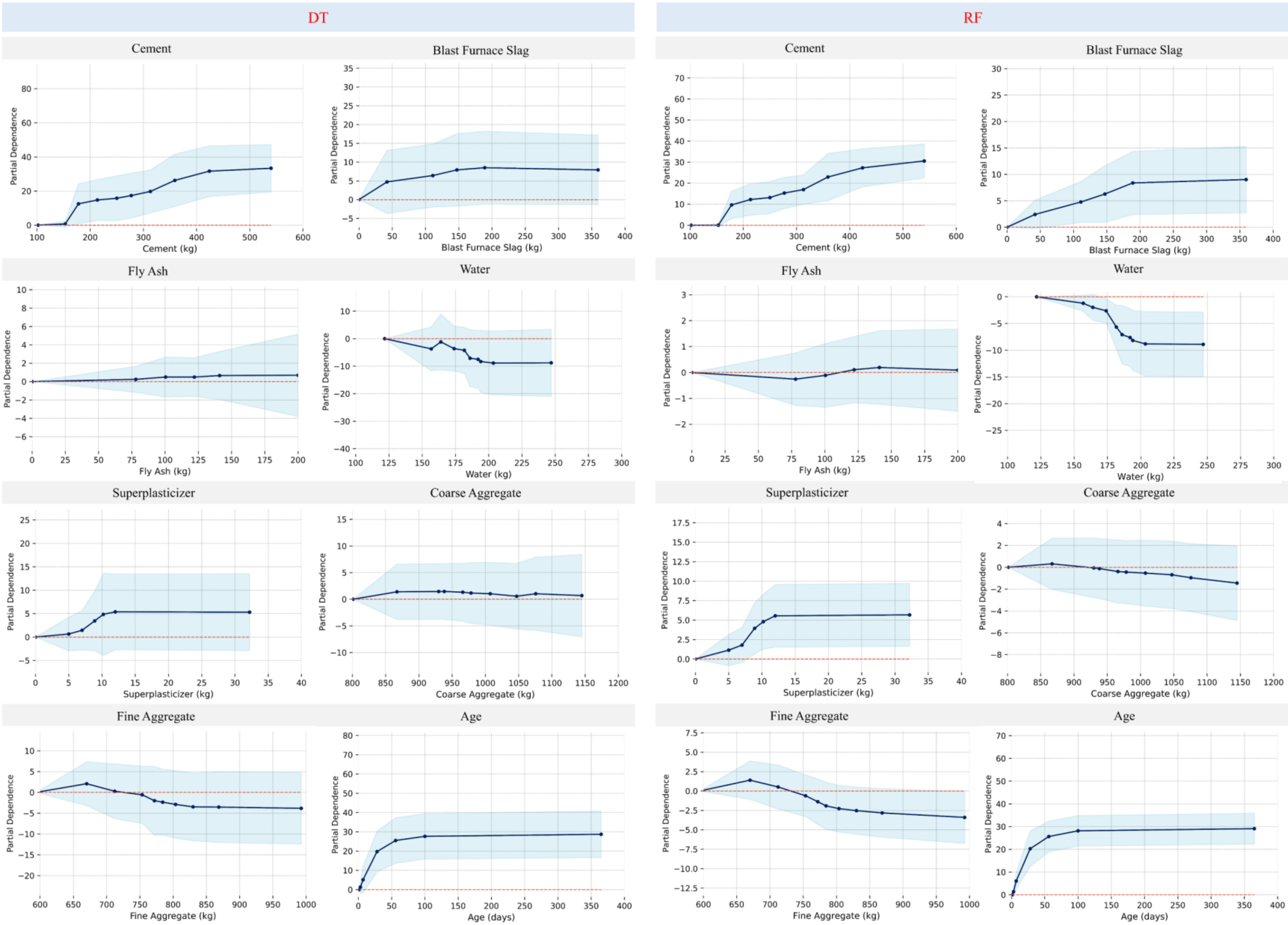

Fig. 8. Partial dependence curves for the DT and RF models.

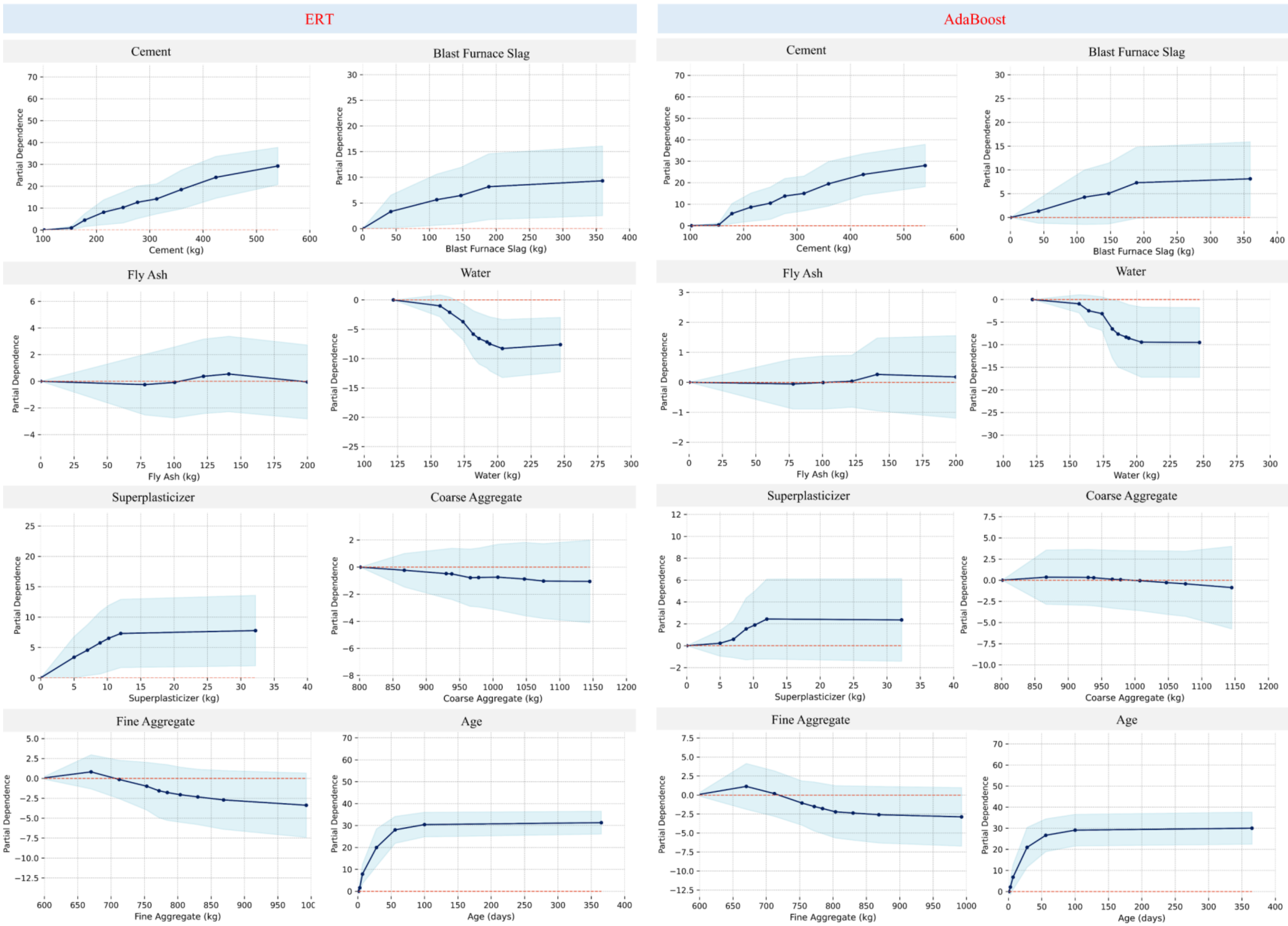

Fig. 9. Partial dependence curves for the ERT and AdaBoost models.

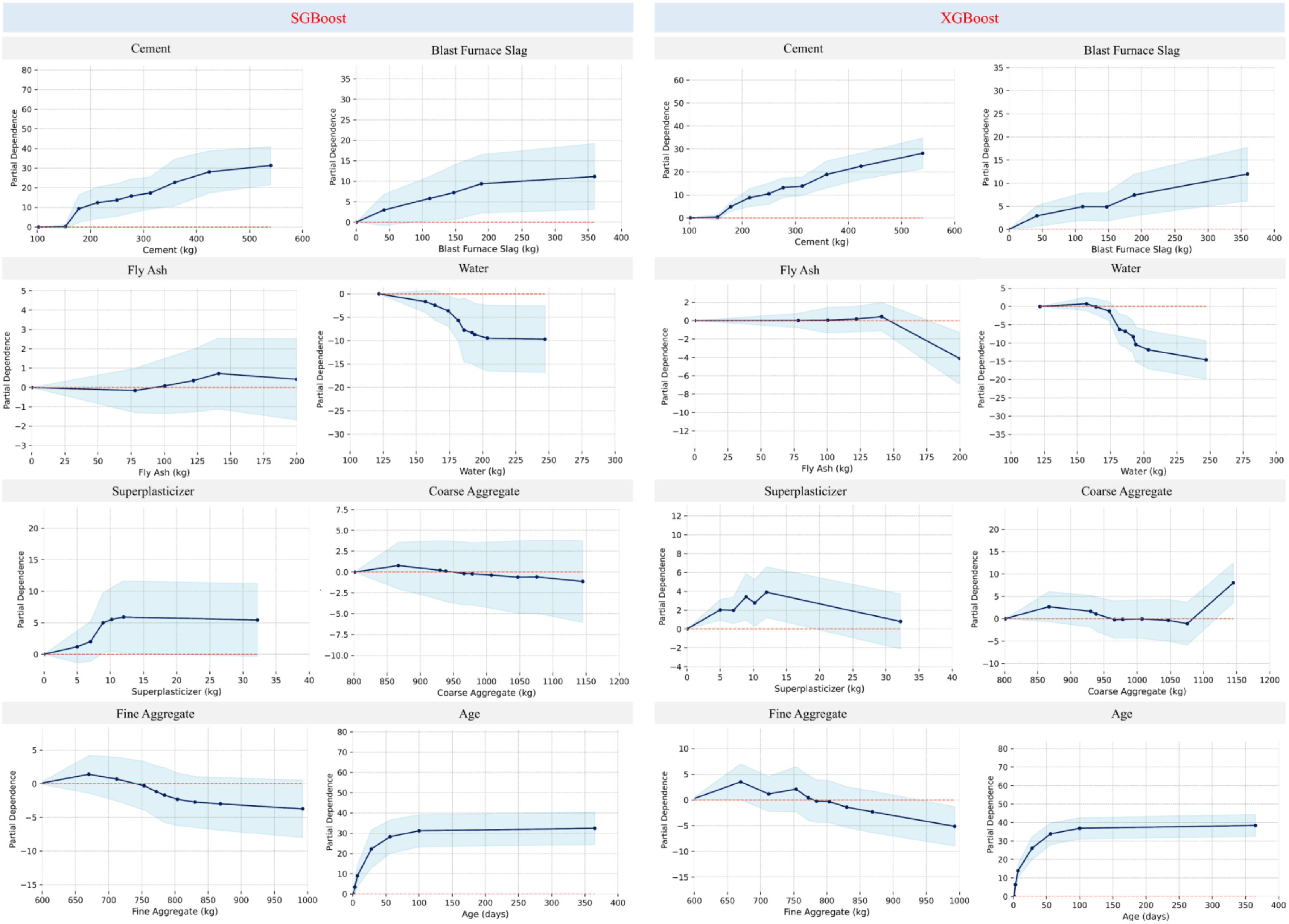

Fig. 10. Partial dependence curves for the SGBoost and XGBoost models.

Additionally, the influences of the contents of fine and coarse aggregates and fly ash fluctuated in most models. However, research in the literature concluded that the high content of fly ash harms the strength of the concrete. This point was only obtained in the case of the XGBoost model, while other approaches showed almost no impact at will.

4. Conclusion

In conclusion, this study has focused on evaluating the application of machine learning techniques for conducting a parametric assessment to evaluate the influence of concrete mixture's consistent materials on its compressive strength.

- Within the research context, six different machine learning models were developed and assessed their performance. After that, two different methods were used in conducting a parametric assessment in the artificial intelligence environment. The first one was the feature importance which tries to rank the inputs of the estimation model for their contribution to the model's accuracy. In contrast, the second one illustrates the effect of variation in each input feature on the model's outcomes.
- In general, the study results have shown that using the feature importance to understand which parameter affects the strength of concrete is somehow misleading in the drop-out loss, and MSE reduction approaches since their outcomes did not match the expectations based on the knowledge available in the literature.
- Nevertheless, the partial dependence plots provide a clear idea about the influence of the value of each feature on the prediction outcomes, especially in the case of the XGBoost, in which the results were the closest to the conclusions of previous research.
- Accordingly, this study recommends utilizing partial dependence analysis with the XGBoost model to evaluate the constituent materials' influence on the compressive strength of concrete.

Data availability

The raw data that were used to support the findings of the study can be obtained from the following link: "*https://archive.ics.uci.edu/ml/datasets/concrete+compressive+strength*"

References

[1] Aïtcin P-C. Cements of yesterday and today. Cem Concr Res 2000;30:1349–59. https://doi.org/10.1016/S0008-8846(00)00365-3.

[2] Habib A, Houri A AL, Habib M, Elzokra A, Yildirim U. Structural Performance and Finite Element Modeling of Roller Compacted Concrete Dams: A Review. Lat Am J Solids Struct 2021;18. https://doi.org/10.1590/1679-78256467.

[3] Shakhmenko G, Birsh J. Concrete mix design and optimization. Proc. 2nd Int. Symp. Civ. Eng., 1998, p. 1–8.

[4] Ni H-G, Wang J-Z. Prediction of compressive strength of concrete by neural networks. Cem Concr Res 2000;30:1245–50. https://doi.org/10.1016/S0008-8846(00)00345-8.

[5] Chopra P, Sharma RK, Kumar M, Chopra T. Comparison of Machine Learning Techniques for the Prediction of Compressive Strength of Concrete. Adv Civ Eng 2018;2018:1–9. https://doi.org/10.1155/2018/5481705.

[6] Young BA, Hall A, Pilon L, Gupta P, Sant G. Can the compressive strength of concrete be estimated from knowledge of the mixture proportions?: New insights from statistical analysis and machine learning methods. Cem Concr Res 2019;115:379–88. https://doi.org/10.1016/j.cemconres.2018.09.006.

[7] Feng D-C, Liu Z-T, Wang X-D, Chen Y, Chang J-Q, Wei D-F, et al. Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach. Constr Build Mater 2020;230:117000. https://doi.org/10.1016/j.conbuildmat.2019.117000.

[8] Kaloop MR, Kumar D, Samui P, Hu JW, Kim D. Compressive strength prediction of high-performance concrete using gradient tree boosting machine. Constr Build Mater 2020;264:120198. https://doi.org/10.1016/j.conbuildmat.2020.120198.

[9] Nguyen-Sy T, Wakim J, To Q-D, Vu M-N, Nguyen T-D, Nguyen T-T. Predicting the compressive strength of concrete from its compositions and age using the extreme gradient boosting method. Constr Build Mater 2020;260:119757. https://doi.org/10.1016/j.conbuildmat.2020.119757.

[10] Xu J, Zhou L, He G, Ji X, Dai Y, Dang Y. Comprehensive Machine Learning-Based Model for Predicting Compressive Strength of Ready-Mix Concrete. Materials (Basel) 2021;14:1068. https://doi.org/10.3390/ma14051068.

[11] Topçu İB, Sarıdemir M. Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic. Comput Mater Sci 2008;41:305–11. https://doi.org/10.1016/j.commatsci.2007.04.009.

[12] Chopra P, Sharma RK, Kumar M. Prediction of Compressive Strength of Concrete Using Artificial Neural Network and Genetic Programming. Adv Mater Sci Eng 2016;2016:1–10. https://doi.org/10.1155/2016/7648467.

[13] Lee S-C. Prediction of concrete strength using artificial neural networks. Eng Struct 2003;25:849–57. https://doi.org/10.1016/S0141-0296(03)00004-X.

[14] Habib A, Yıldırım U. Prediction of the dynamic properties in rubberized concrete. Comput Concr 2021;27:185–97.

[15] Barkhordari MS, Armaghani DJ, Fakharian P. Ensemble machine learning models for prediction of flyrock due to quarry blasting. Int J Environ Sci Technol 2022. https://doi.org/10.1007/s13762-022-04096-w.

[16] Han Q, Gui C, Xu J, Lacidogna G. A generalized method to predict the compressive strength of high-performance concrete by improved random forest algorithm. Constr Build Mater 2019;226:734–42. https://doi.org/10.1016/j.conbuildmat.2019.07.315.

[17] Guryanov A. Histogram-Based Algorithm for Building Gradient Boosting Ensembles of Piecewise Linear Decision Trees, 2019, p. 39–50. https://doi.org/10.1007/978-3-030-37334-4_4.

[18] Farooq F, Nasir Amin M, Khan K, Rehan Sadiq M, Faisal Javed MF, Aslam F, et al. A Comparative Study of Random Forest and Genetic Engineering Programming for the Prediction of Compressive Strength of High Strength Concrete (HSC). Appl Sci 2020;10:7330. https://doi.org/10.3390/app10207330.

[19] Jha AK, Adhikari S, Thapa S, Kumar A, Kumar A, Mishra S. Evaluation of Factors Affecting Compressive Strength of Concrete using Machine Learning. 2020 Adv. Comput. Commun. Technol. High Perform. Appl., IEEE; 2020, p. 70–4. https://doi.org/10.1109/ACCTHPA49271.2020.9213199.

[20] Anysz H, Brzozowski Ł, Kretowicz W, Narloch P. Feature Importance of Stabilised Rammed Earth Components Affecting the Compressive Strength Calculated with Explainable Artificial Intelligence Tools. Materials (Basel) 2020;13:2317. https://doi.org/10.3390/ma13102317.

[21] Ly H-B, Nguyen T-A, Thi Mai H-V, Tran VQ. Development of deep neural network model to predict the compressive strength of rubber concrete. Constr Build Mater 2021;301:124081. https://doi.org/10.1016/j.conbuildmat.2021.124081.

[22] Su M, Zhong Q, Peng H. Regularized multivariate polynomial regression analysis of the compressive strength of slag-metakaolin geopolymer pastes based on experimental data. Constr Build Mater 2021;303:124529. https://doi.org/10.1016/j.conbuildmat.2021.124529.

[23] Dao D Van, Ly H-B, Vu H-LT, Le T-T, Pham BT. Investigation and Optimization of the C-ANN Structure in Predicting the Compressive Strength of Foamed Concrete. Materials (Basel) 2020;13:1072. https://doi.org/10.3390/ma13051072.

[24] Mane KM, Kulkarni DK, Prakash KB. Prediction of flexural strength of concrete produced by using pozzolanic materials and partly replacing NFA by MS. J Soft Comput Civ Eng 2019;3:65–75.

[25] Nayak SC, Nayak SK, Panda SK. Assessing compressive strength of concrete with extreme learning machine. J Soft Comput Civ Eng 2021;5:68–85.

[26] Pandey S, Kumar V, Kumar P. Application and Analysis of Machine Learning Algorithms for Design of Concrete Mix with Plasticizer and without Plasticizer. J Soft Comput Civ Eng 2021;5:19–37.

[27] Naderpour H, Sharei M, Fakharian P, Heravi MA. Shear Strength Prediction of Reinforced Concrete Shear Wall Using ANN, GMDH-NN and GEP. J Soft Comput Civ Eng 2022;6:66–87.

[28] Amini A, Abdollahi A, Hariri-Ardebili MA, Lall U. Copula-based reliability and sensitivity analysis of aging dams: Adaptive Kriging and polynomial chaos Kriging methods. Appl Soft Comput 2021;109:107524. https://doi.org/10.1016/j.asoc.2021.107524.

[29] Rajan K. Materials informatics. Mater Today 2005;8:38–45. https://doi.org/10.1016/S1369-7021(05)71123-8.

[30] Himanen L, Geurts A, Foster AS, Rinke P. Data-Driven Materials Science: Status, Challenges, and Perspectives. Adv Sci 2019;6:1900808. https://doi.org/10.1002/advs.201900808.

[31] Marsland S. Machine learning: an algorithmic perspective. Chapman and Hall/CRC; 2011.

[32] Kotsiantis SB, Zaharakis I, Pintelas P. Supervised machine learning: A review of classification techniques. Emerg Artif Intell Appl Comput Eng 2007;160:3–24.

[33] Mahesh B. Machine learning algorithms-a review. Int J Sci Res (IJSR)[Internet] 2020;9:381–6.

[34] Sharma D, Kumar N. A review on machine learning algorithms, tasks and applications. Int J Adv Res Comput Eng Technol 2017;6:1323–2278.

[35] Myles AJ, Feudale RN, Liu Y, Woody NA, Brown SD. An introduction to decision tree modeling. J Chemom 2004;18:275–85. https://doi.org/10.1002/cem.873.

[36] Quinlan JR. Decision trees and decision-making. IEEE Trans Syst Man Cybern 1990;20:339–46. https://doi.org/10.1109/21.52545.

[37] Chen Y-L, Hung LT-H. Using decision trees to summarize associative classification rules. Expert Syst Appl 2009;36:2338–51. https://doi.org/10.1016/j.eswa.2007.12.031.

[38] Loh W. Classification and regression trees. Wiley Interdiscip Rev Data Min Knowl Discov 2011;1:14–23.

[39] Charbuty B, Abdulazeez A. Classification Based on Decision Tree Algorithm for Machine Learning. J Appl Sci Technol Trends 2021;2:20–8. https://doi.org/10.38094/jastt20165.

[40] Janikow CZ. Fuzzy decision trees: issues and methods. IEEE Trans Syst Man Cybern Part B 1998;28:1–14. https://doi.org/10.1109/3477.658573.

[41] Breiman L. Random forests. Mach Learn 2001;45:5–32.

[42] Micheletti N, Foresti L, Robert S, Leuenberger M, Pedrazzini A, Jaboyedoff M, et al. Machine Learning Feature Selection Methods for Landslide Susceptibility Mapping. Math Geosci 2014;46:33–57. https://doi.org/10.1007/s11004-013-9511-0.

[43] Williams G. Data mining with Rattle and R: The art of excavating data for knowledge discovery. Springer Science & Business Media; 2011.

[44] Rodriguez-Galiano V, Sanchez-Castillo M, Chica-Olmo M, Chica-Rivas M. Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. Ore Geol Rev 2015;71:804–18. https://doi.org/10.1016/j.oregeorev.2015.01.001.

[45] Zhang J, Ma G, Huang Y, Sun J, Aslani F, Nener B. Modelling uniaxial compressive strength of lightweight self-compacting concrete using random forest regression. Constr Build Mater 2019;210:713–9. https://doi.org/10.1016/j.conbuildmat.2019.03.189.

[46] Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Mach Learn 2006;63:3–42. https://doi.org/10.1007/s10994-006-6226-1.

[47] Freund Y, Schapire RE. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. J Comput Syst Sci 1997;55:119–39. https://doi.org/10.1006/jcss.1997.1504.

[48] Drucker H. Improving regressors using boosting techniques. ICML, vol. 97, Citeseer; 1997, p. 107–15.

[49] Chen T, Guestrin C. Xgboost: A scalable tree boosting system. Proc. 22nd acm sigkdd Int. Conf. Knowl. Discov. data Min., 2016, p. 785–94.

[50] Feng D-C, Wang W-J, Mangalathu S, Hu G, Wu T. Implementing ensemble learning methods to predict the shear strength of RC deep beams with/without web reinforcements. Eng Struct 2021;235:111979. https://doi.org/10.1016/j.engstruct.2021.111979.

[51] Fisher A, Rudin C, Dominici F. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. J Mach Learn Res 2019;20:1–81.

[52] Archer KJ, Kimes R V. Empirical characterization of random forest variable importance measures. Comput Stat Data Anal 2008;52:2249–60. https://doi.org/10.1016/j.csda.2007.08.015.

[53] Strobl C, Boulesteix A-L, Zeileis A, Hothorn T. Bias in random forest variable importance measures: Illustrations, sources and a solution. BMC Bioinformatics 2007;8:25. https://doi.org/10.1186/1471-2105-8-25.

[54] Biecek P. DALEX: explainers for complex predictive models in R. J Mach Learn Res 2018;19:3245–9.

[55] Yeh I-C. Modeling of strength of high-performance concrete using artificial neural networks. Cem Concr Res 1998;28:1797–808. https://doi.org/10.1016/S0008-8846(98)00165-3.

[56] Yeh I-C. Analysis of Strength of Concrete Using Design of Experiments and Neural Networks. J Mater Civ Eng 2006;18:597–604. https://doi.org/10.1061/(ASCE)0899-1561(2006)18:4(597).

[57] Asteris PG, Skentou AD, Bardhan A, Samui P, Pilakoutas K. Predicting concrete compressive strength using hybrid ensembling of surrogate machine learning models. Cem Concr Res 2021;145:106449. https://doi.org/10.1016/j.cemconres.2021.106449.

[58] Ke X, Duan Y. A Bayesian machine learning approach for inverse prediction of high-performance concrete ingredients with targeted performance. Constr Build Mater 2021;270:121424. https://doi.org/10.1016/j.conbuildmat.2020.121424.

[59] Ramezanianpour AA, Malhotra VM. Effect of curing on the compressive strength, resistance to chloride-ion penetration and porosity of concretes incorporating slag, fly ash or silica fume. Cem Concr Compos 1995;17:125–33. https://doi.org/10.1016/0958-9465(95)00005-W.

[60] Sengul O, Tasdemir MA. Compressive Strength and Rapid Chloride Permeability of Concretes with Ground Fly Ash and Slag. J Mater Civ Eng 2009;21:494–501. https://doi.org/10.1061/(ASCE)0899-1561(2009)21:9(494).

[61] Alsadey S. Influence of superplasticizer on strength of concrete. Int J Res Eng Technol 2012;1:164–6.