Civil Engineering Infrastructures Journal 2023, 56(1): 159-172
DOI: 10.22059/CEIJ.2022.337777.1811

RESEARCH PAPER

Predicting Compressive Strength of Concrete Using Histogram-Based Gradient Boosting Approach for Rapid Design of Mixtures

Al Adwan, J.[1], Alzubi, Y.[1*], Alkhdour, A.[1] and Alqawasmeh, H.[1]

[1] Assistant Professor, Civil Engineering Department, Faculty of Engineering Technology, Al-Balqa Applied University, Amman, Jordan.

Received: 21 Jan. 2022; Revised: 14 Mar. 2022; Accepted: 04 Apr. 2022

ABSTRACT: Applications of machine learning techniques in concrete properties' prediction have great interest to many researchers worldwide. Indeed, some of the most common machine learning methods are those based on adopting boosting algorithms. A new approach, histogram-based gradient boosting, was recently introduced to the literature. It is a technique that buckets continuous feature values into discrete bins to speed up the computations and reduce memory usage. Previous studies have discussed its efficiency in various scientific disciplines to save computational time and memory. However, the algorithm's accuracy is still unclear, and its application in concrete properties estimation has not yet been considered. This paper is devoted to evaluating the capability of histogram-based gradient boosting in predicting concrete's compressive strength and comparing its accuracy to other boosting methods. Generally, the results of the study have shown that the histogram-based gradient boosting approach is capable of achieving reliable prediction of concrete compressive strength. Additionally, it showed the effects of each model's parameters on the accuracy of the estimation.

Keywords: Compressive Strength, Concrete, Histogram-Based Gradient Boosting, Machine Learning.

1. Introduction

There is no doubt that concrete is one of the most widely used materials all over the world (Elzokra et al., 2020; Habib et al., 2021). Generally, it has high compressive strength and stiffness, making it suitable for various construction works (Malkawi et al., 2020). Concrete mixture design as a vital and recondite problem is the method of identifying the kind and amount of individual components to produce concrete that fits desired properties, such as workability and strength, for a particular purpose and is also economically acceptable (Yeh, 2007; Wardeh et al., 2015). Optimal design of concrete composition proportions with the lowest cost and needed performance is considered a challenge of designers and decision-makers (DeRousseau et al., 2018). In reality, the strength of this material is one of its significant parameters for mixture design and optimization, yet obtaining it at a mature age requires a relatively long experimental process (Ni and Wang, 2000; Al Houri et al., 2020).

Mathematical modeling of concrete

* Corresponding author E-mail: yazan.alzubi@yahoo.com

characteristics has become a fundamental issue in recent years, owing to the rapid developments of novel concrete types stimulated by the ever-challenging requirements of the construction industry (Chaabene et al., 2020). The used techniques to design concrete characteristics are separated into two categories: conventional and artificial intelligence-augmented methods (Ziolkowski and Niedostatkiewicz, 2019; Gholamzadeh Chitgar and Berenjian, 2021). In general, various data-driven mathematical methods have been applied to estimate the properties of concrete rapidly, including multiple linear regression (Khademi et al., 2017), fuzzy logic (Topcu and Sarıdemir, 2008), genetic programming (Chopra et al., 2016), artificial neural networks (Lee, 2003; Habib and Yildirim, 2021), bagging machine learning techniques (Han et al., 2019; Farooq et al., 2020), boosting machine learning methods (Nguyen-Sy et al., 2020; Salimbahrami and Shakeri, 2021).

Due to its superior accuracy, the gradient-boosting decision tree is a commonly used machine learning algorithm in many disciplines (Guryanov, 2019). For instance, Feng et al. (2020) utilized an adaptive boosting approach to estimate concrete's compressive. Kaloop et al. (2020) used the gradient tree boosting machine to predict the compressive strength of high-performance concrete. Moreover, Nguyen-Sy et al. (2020) developed a machine learning model using extreme gradient boosting to design concrete mixtures using constituent materials and age at testing. Even though the gradient boosting algorithm performs very well, massive ensembles will likely be slow in training and inference (Lu et al., 2020). This difficulty in developing gradient-boosting models for large datasets, tens of thousands of observations or more, emerges from the time consumed to find the best split points (Shepovalov and Akella, 2020).

In contrast, a histogram-based technique can decrease computation time and memory usage by converting continuous data into discrete bins to build attribute histograms instead of finding the split points (Cai et al., 2021). Recently, three major Python coding libraries that wrap up many modern efficiency approaches for training gradient-boosting algorithms have permitted the development of histogram-based models, including scikit-learn (Pedregosa et al., 2011), XGBoost (Chen and Guestrin, 2016), and LightGBM (Ke et al., 2017). The importance of histogram-based gradient boosting (HGBoost) comes from its ability to considerably reduce the computational efforts and memory usage required to train machine learning models for large datasets. Previous studies have focused on the technique's performance regarding the time and memory needed to develop the estimation models. Besides, the capability of HGBoost in predicting concrete characteristics is still unclear, and the studies investigating its accuracy for adaptation in the civil engineering discipline are scarce.

Therefore, this article evaluates the accuracy and reliability of the histogram-based gradient boosting technique in predicting the compressive strength of concrete mixtures. As a part of the study, a parametric assessment will be conducted to highlight the influence of each of the method's parameters on the model's accuracy. In addition, the results of the HGBoost will be compared against that of various boosting approaches to benchmark and understand the algorithm's performance. This information is missing from the literature and is helpful for many engineers and researchers working in the field.

2. Materials and Methods

Researchers have invented artificial intelligence-based solutions for many real-time scenarios in recent years, thanks to advances in the application of machine learning algorithms across a wide range of fields (Amidi et al., 2021; Ahmad et al., 2022; Kim et al., 2022). The roots of gradient boosting may be traced back to the discovery by statistician Leo Breiman that it can help diminish bias, resulting in increased performance (Mease and Wyner, 2008). Adaptive Boosting (AdaBoost) was the first convenient embodiment of the boosting concept. It was developed by Yoav Freund and Robert Schapire in 1995, and it has been shown to help enhance the behavior of various learning techniques (Freund and Schapire, 1997). When it comes to training machine learning algorithms for massive datasets, HGBoost is very valuable because of its ability to minimize the amount of computing effort and memory needed significantly (Guryanov, 2019). The general methodology followed in this research for quantifying the accuracy of HGBoost is illustrated in Figure 1.

The fundamental purpose of concrete mixture design is to determine the appropriate quantity of constituents in the mix. Therefore, the mixture components must be chosen precisely to obtain the highest possible concrete performance while keeping costs down. This behavior is manifested by several features in which compressive strength is the most important. In this research, machine learning techniques will be applied to design a concrete mixture using the HGBoost model and an enormous dataset for estimating the strength of concrete. The histogram-based gradient boosting method predicts the compressive strength of a concrete mix relied on the quantity of the seven primary ingredients and age at testing of concrete, more accurately cement, fine and coarse aggregate, superplasticizer, fly ash, silica fume, and water, as shown in Figure 2.

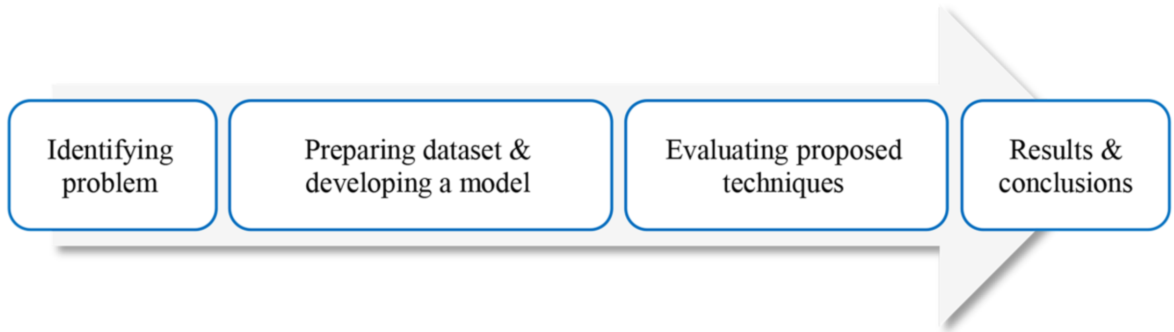

Fig. 1. Graphical description of the research methodology

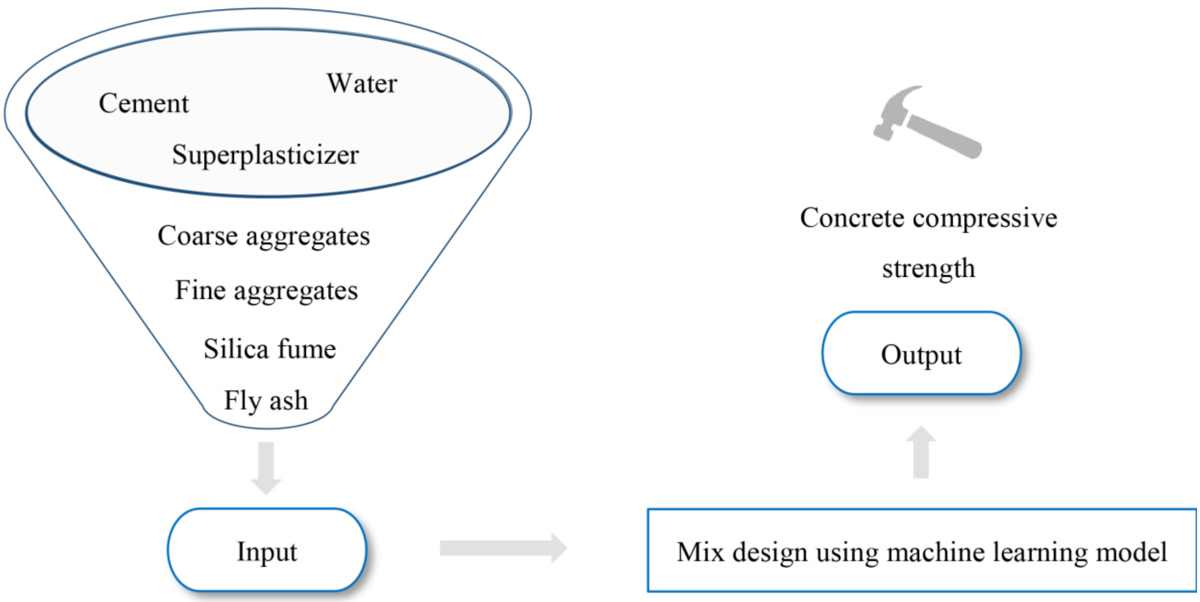

Fig. 2. Stages of a process for concrete mixture design using a machine learning

2.1. Dataset Acquisition

As stated before, this research focuses solely on the accuracy of HGBoost rather than its time and memory consumption. Accordingly, developing a significantly large dataset with tens of thousands of observations is not required. Thus, a relatively large set of experimental results for various concrete mixtures was utilized to investigate the HGBoost model's capability against other machine learning techniques. This dataset was first developed by Yeh (1998) from many resources and then used in various researches for developing numerical models (Yeh, 2006; Asteris et al., 2021; Ke and Duan, 2021). Generally, the database comprises 1005 observations (Table 1 and Figure 3). The compressive strength of the mixtures is accounted for 15 cm cylinder specimens. Yeh (1998) stated that the dataset often contains some unexpected inaccuracies regarding the class of fly ash or that of the cement, the gradation of the aggregates, and the type of superplasticizer. While this issue might cause difficulty to the machine learning model due to the fluctuation of the materials' sources, it will simulate the cause of constructing a generalized machine learning model for designing concrete mixtures, which is one of the major potentials of the HGBoost model. Indeed, the concrete's constituent materials and age were used as the input parameters, while its compressive strength was the output of the estimation models. Additionally, it can be noticed from Table 1 that the fly ash and age have higher standard deviations than the average values, which is attributed to having high vibration between the values and abnormal distribution of the data.

2.2. Boosting Machine Learnings Techniques

A decision tree is a learning technique that is widely used in data mining. It is applied for classification and regression issues. According to this process, the estimation model is often created by recursively splitting the dataset, fitting a basic predicting algorithm in each of these divisions, and ultimately depicting each model as a decision tree (Loh, 2011). At a given node m, the data is represented by Q_m with N_m samples. This data is partitioned into two subsets $Q_m^{left}(\theta)$, Eq. (1), and $Q_m^{right}(\theta)$, Eq. (2), where for each candidate split $\theta = (j, t_m)$ compose of a j feature and t_m threshold.

$$
\begin{aligned}
Q_m^{left}(\theta) &= \{(x, y) | x_i \leq t_m\} \\
Q_m^{right}(\theta) &= Q_m / Q_m^{left}(\theta)
\end{aligned}
\tag{1}
$$

The quality of the candidate split at the given node m is calculated through the loss function $H()$.

$$
\begin{aligned}
&G(Q_m, \theta) \\
&= \frac{N_m^{left}}{N_m} H\left(Q_m^{left}(\theta)\right) \\
&+ \frac{N_m^{right}}{N_m} H\left(Q_m^{right}(\theta)\right)
\end{aligned}
\tag{2}
$$

Thereafter, those parameters that minimize the loss are selected using Eq. (3), and the process is repeated for $Q_m^{left}(\theta)$ and $Q_m^{right}(\theta)$ until reaching the maximum allowable depth $N_m < min_{samples}$ or $N_m = 1$.

Table 1. Descriptive statistics of the utilized database

	Variable	Average	Standard deviation	Min.	First quintile	Median	Third quintile	Max.
Input	Cement (kg/m^3)	278.63	104.34	102	190.7	265	349	540
	Blast furnace slag (kg/m^3)	72.04	86.17	0	0	20	142.5	359.4
	Fly ash (kg/m^3)	55.54	64.21	0	0	0	118.3	200.1
	Water (kg/m^3)	182.08	21.34	121.8	166.6	185.7	192.9	247
	Superplasticizer (kg/m^3)	6.033	5.92	0	0	6.1	10	32.2
	Coarse aggregate (kg/m^3)	974.38	77.58	801	932	968	1032	1145
	Fine aggregate (kg/m^3)	772.69	80.34	594	724.3	780	823.1	992.6
	Age (days)	45.86	63.73	1	7	28	56	365
Output	Compressive strength (MPa)	35.25	16.285	2.33	23.52	33.8	44.975	82.6

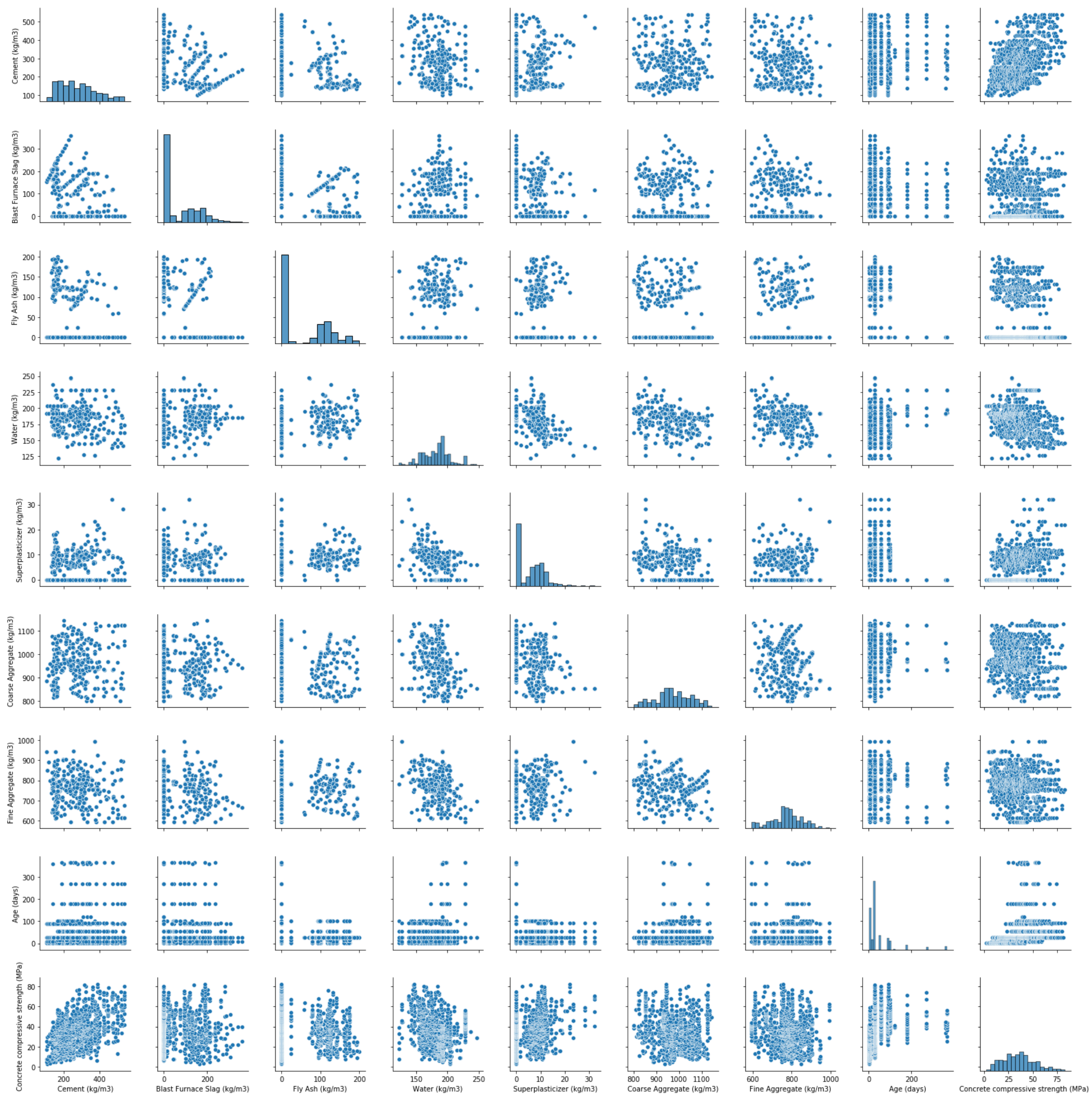

Fig. 3. Visualization of the utilized database

$$\theta^* = argmin_\theta G(Q_m, \theta) \tag{3}$$

On the other hand, Adaptive boosting, as described by Freund and Schapire (1997), is a method that works by fitting new copies of the regression model using the same training dataset through adjusting its weights from the error of the prior trial. The implantation of this algorithm is performed according to Drucker (1997). In the beginning, the AdaBoost is trained on a base estimator (weak learner) $f(x)$, and the error e_i is obtained for the entire set. Thereafter, a series of weak learners $f_k(x), k = 1,2, \dots, N$ is produced and combined to develop a strong model $H(x)$ through the strategy in Eq. (4).

$$H(x) = v \sum_{k=1}^{N} \left(ln \frac{1}{\alpha_k} \right) g(x) \tag{4}$$

where v: is the learning rate, α_k: is the weight of the base estimators calculated from Eq. (5), and $g(x)$: is median of all $\alpha_k f_k(x)$.

$$\alpha_k = \frac{e_i}{1 - e_i} \tag{5}$$

Stochastic gradient boosting (SGBoost) is similar to the AdaBoost approach that works by adding a new model ensemble but with a significant difference based on minimizing the learner's loss function. In addition, the weak estimator in the gradient boosting is a larger decision tree with multiple levels compared to the regressor in the AdaBoost model. The training set $\{(x_1, y_1), \dots, (x_n, y_n)\} \subset \chi \times R$ will be determined with a sample size n and space of input variables χ. Specify $\hat{y}_i$ as the prediction of a gradient boosting machine method with x_i input variable as indicated in Eq. (6).

$$\hat{y}_i = F_M(x_i) = \sum_{m=1}^{M} h_m(x_i) \tag{6}$$

in which M: is the total number of predictors provided in the algorithm and h_m: is a weak learner.

The gradient boosting is developed greedily, as stated in Eq. (7).

$$F_m(x) = F_{m-1}(x) + \underset{h \in H}{argmin} \sum_{i=1}^{n} L[y_i, F_{m-1}(x_i) + h(x_i)] \tag{7}$$

where $h(x)$: is the base estimator, $L(\cdot)$: is the loss function with the negative gradient identified in Eq. (8).

$$g_m = -\frac{\partial L[y, F_{m-1}(x)]}{\partial F_{m-1}(x)} \tag{8}$$

The SGBoost is formulated by modifying the gradient boosting model to fit the base learner randomly on a subsample with fraction $f < 1$.

Histogram-based gradient boosting (HGBoost) is a recently introduced machine learning approach (Chen and Guestrin, 2016). Unlike other techniques, it buckets continuous feature values into discrete bins and uses these bins to construct feature histograms during training. This method speeds up the training stage and reduces the memory consumption of the model. The algorithm developed herein is based on the one available in the scikit-learn library (Cai et al., 2021).

eXtreme gradient boosting (XGBoost) is an efficient and scalable machine learning algorithm applied for tree boosting (Pedregosa et al., 2011). In general, both gradients boosting and XGBoost follow the principle of gradient boosting, but XGBoost uses a more regularized model to control the over-fitting issues to achieve better results. Additionally, the XGBoost model uses the exact greedy tree method compared to the optimized approximate one adopted in the HGBoost model. The objective function of this algorithm is given in Eq. (9).

$$Obj = \sum_{i=1}^{n} L[\hat{y}_i, y_i)] + \sum_{i=1}^{n} \omega(f_t) \tag{9}$$

where $L(\cdot)$: is the loss function for the model's bias, and ω: is a regular term used for suppressing the complexity of the model.

2.6. Model Development and Hyperparameters Tunning

The machine learning algorithm's performance is greatly affected by the chosen hyperparameter values that must be tuned. This article used the grid search method with k-fold cross-validation in the training stage to optimize the approaches' hyperparameters. Hence, the proposed process for developing the machine learning techniques, Figure 4, has started by first dividing the dataset into 70% training and 30% testing ones. The proper parameter selection of each approach was performed utilizing a 10-folds cross-validation procedure. Once the hyperparameters of each technique are defined, the performance of the final tuned algorithm is validated on the test dataset by comparing different scoring parameters.

2.7. Quality Assessment

Statistical measures and visual representations are adopted for analyzing the HGBoost performance, as revealed in Figure 5. The goodness-of-fit is checked using the coefficient of determination, Eq. (10), and A20-index, Eq. (11). The Root Mean Square Error (RMSE), Eq. (12), and Mean Absolute Error (MAE), Eq. (13), were used for the error analysis.

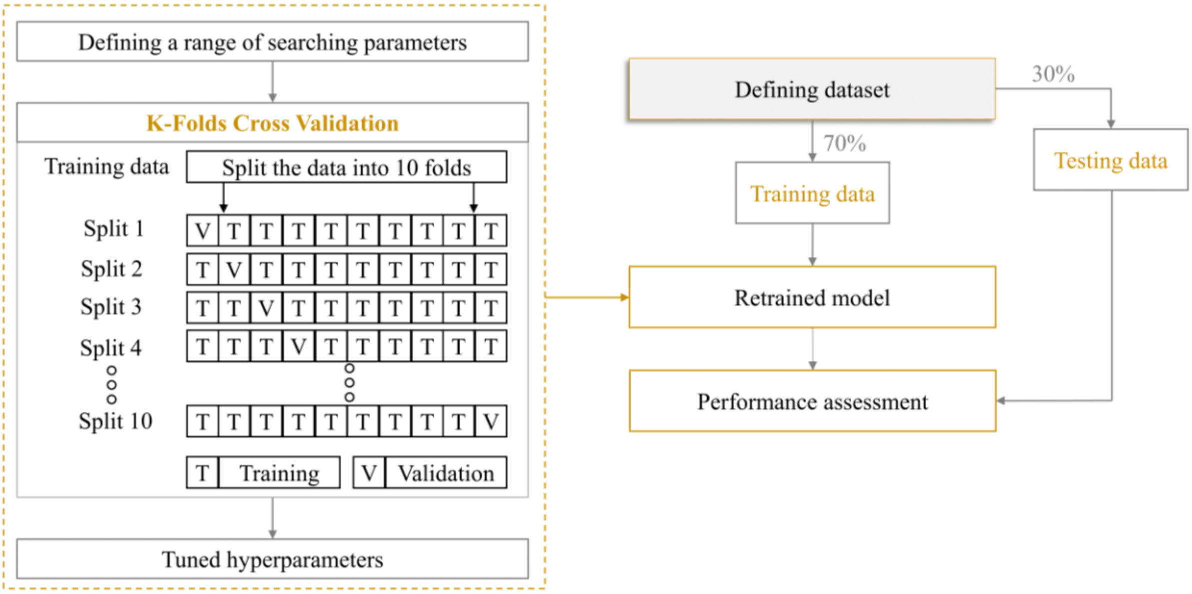

Fig. 4. Schematic diagram for developing machine learning algorithms

$$R^2 = 1 - \frac{\sum(x_i - y_i)^2}{\sum(x_i - \bar{x}_i)^2} \tag{10}$$

$$A20 - Index = \frac{m_{20}}{n} \tag{11}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(x_i - y_i)^2}{n}} \tag{12}$$

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

where x_i: is the measured value, $\bar{x}_i$: is the mean of the measured values, y_i: is the predicted value $\bar{y}_i$: 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

Indeed, hyperparameter tuning of machine learning models is critical for achieving superior estimation capabilities. In this section, a parametric study was conducted to evaluate the impacts of each of the HGBoost parameters on the model performance. The effect of the tree depth is clarified in Figure 6. Generally, it can be seen that shallow trees result in reduced performance of the HGBoost model. Whereas, deep ones, especially over ten edges from the root to the deepest leaf, provide superior testing results in which the A10-index, R^2, RMSE, and MAE 0.85, 0.92, 22, and 3.2, respectively. Similar outcomes were reported for the unconstrained trees. Accordingly, the unconstrained tree is adopted for this study. Also, in Figure 7, increasing the number of bins beyond ten affects the estimation results slightly and yields high accuracy with an average R^2 of 0.92. Thus, any value after 10 is suitable for the HGBoost model, yet in this study, the one with the least error value was used. On the other hand, the least essential variable while generating the model was the L2 regularization parameter, Figure 8, since it showed a fluctuation at low values and slightly impacted both the fitting rates and errors of the algorithm and later ones. Furthermore, for the investigated dataset, the learning rate with a value of 0.3 yields the best performance of the HGBoost method with an A10-index of about 1, as indicated in Figure 9. Additionally, the least-squares loss function improved the model's accuracy compared to the slightest absolute deviation, as shown in Figure 10. Therefore, the least-squares loss function was utilized for developing the HGBoost model.

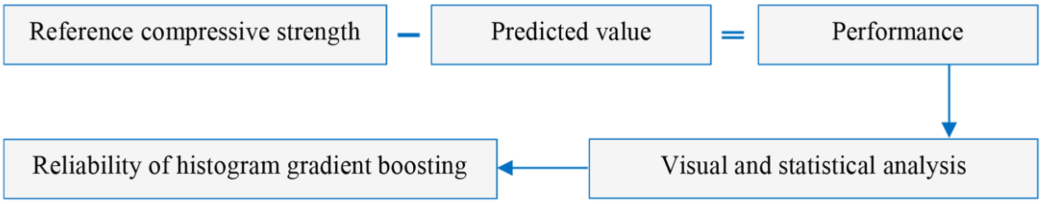

Fig. 5. Benchmarking proposed model

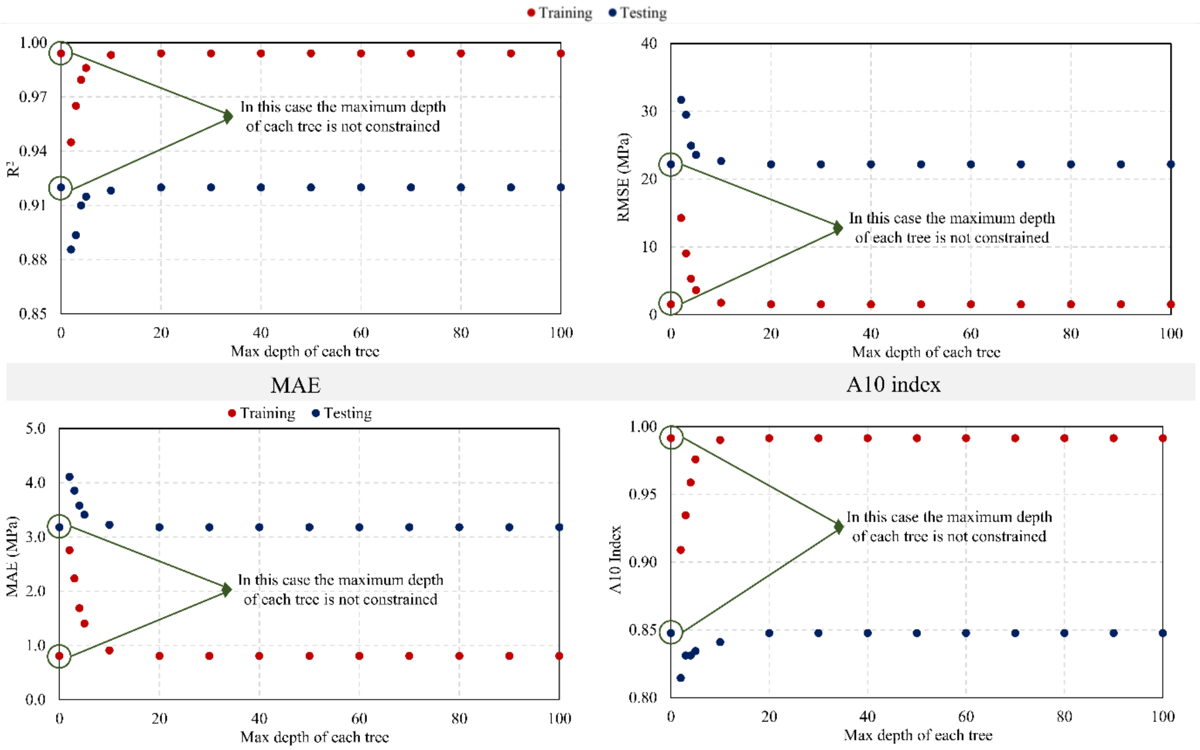

Fig. 6. Influence of the depth of each tree on the HGBoost performance

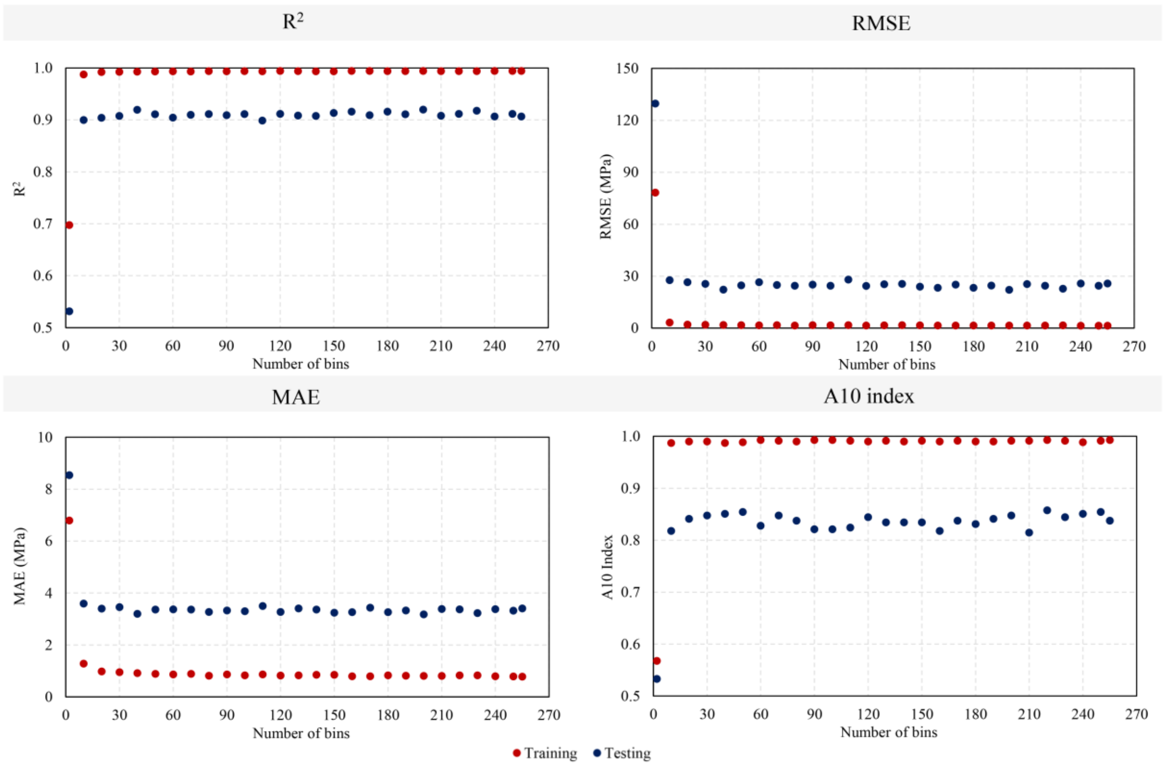

Fig. 7. Role of number of bins in HGBoost on the model performance

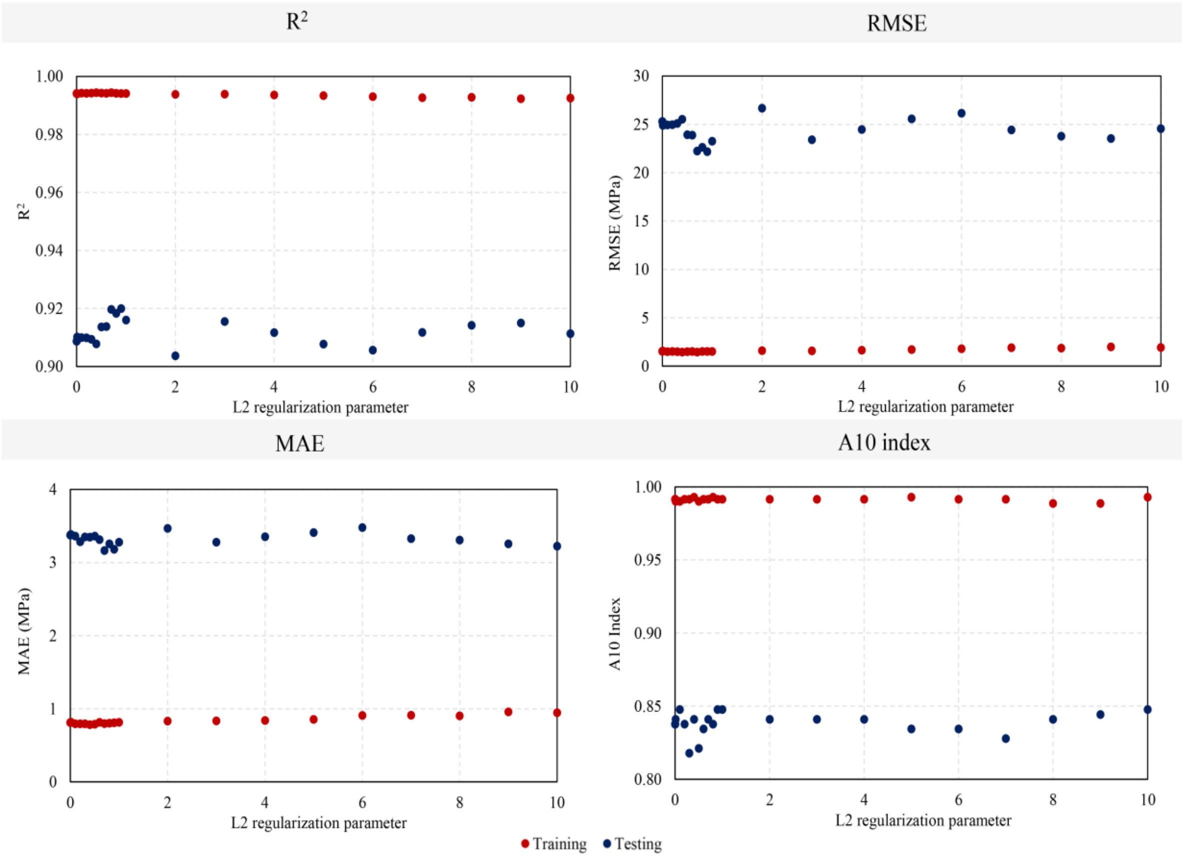

Fig. 8. Impact of the L2 regularization parameter on the HGBoost quality

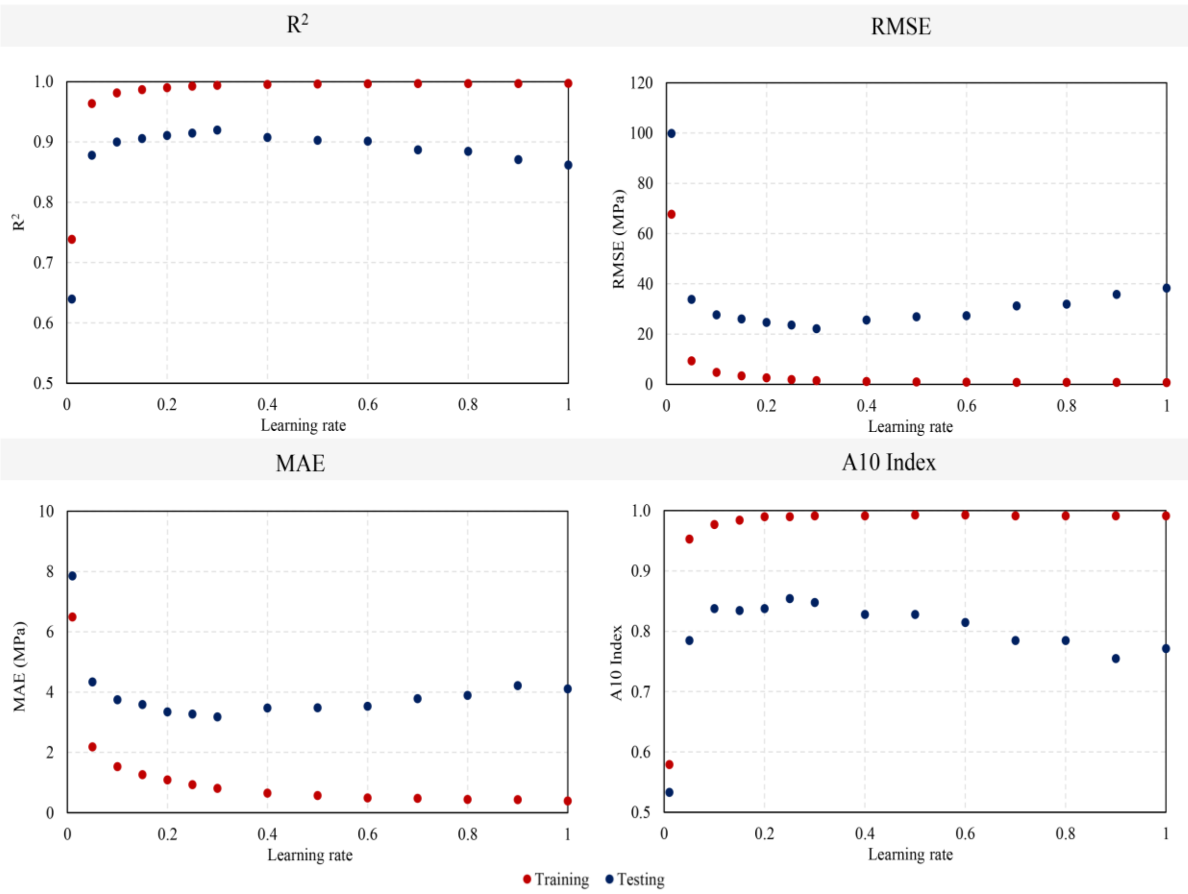

Fig. 9. Influence of learning rate in HGBoost on the performance of the model

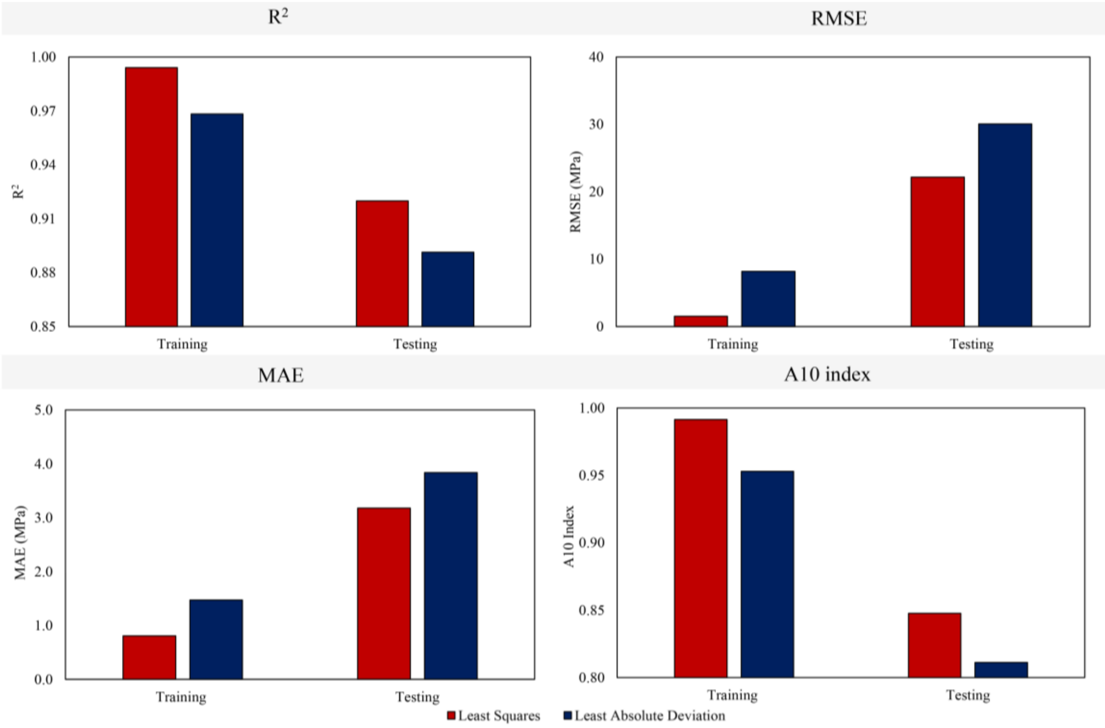

Fig. 10. Influence of the type of loss function on the performance of the HGBoost model

As mentioned previously, several machine learning and statistical models were proposed for predicting the compressive strength of concrete mixtures. However, the literature still lacks a study that evaluates the accuracy of HGBoost models. Therefore, a comparative assessment of boosting algorithms in machine learning will be presented. The estimation results, Figure 11, show that the AdaBoost model gives a minimal performance. At the same time, SGBoost, HGBoost, and XGBoost approaches yielded similar outcomes. This observation is further illustrated in the residual plot as shown in Figure 12a. As a matter of fact, the box plots in Figure 12b highlights that for the case of the training dataset, the average, median, first quartile, and third quartile were all predicted accurately in all models. On the other hand, these values were best estimated in the testing dataset by the HGBoost model followed by the SGBoost and XGBoost. Indeed, the capabilities of the methods are not affected by the value of the compressive strength in which the models revealed good efficiency for both low and high strength mixtures.

Table 2 represents the fitting rates and the errors values of the addressed algorithm. In general, the coefficient of determination for the training dataset is higher than that of the testing data. The performance of the AdaBoost technique in the testing case has dropped significantly compared to the training set directed towards an overfitting issue. Also, the A20 index of HGBoost has reduced by 4.9% and 6% against SGBoost and XGBoost models, respectively, while the AdaBoost generates the lowest A20 index and highest testing errors. On the other hand, the HGBoost achieves higher testing errors than the SGBoost and XGBoost because the latter techniques use the exact greedy tree method while the HGBoost adopts an optimized approximate one.

Table 2. Performance assessment of the models

Model	**Training**				**Testing**			
	R^2	**RMSE**	**MAE**	**A20 Index**	**R^2**	**RMSE**	**MAE**	**A20 Index**
AdaBoost	1.00	1.41	0.54	0.97	0.87	35.54	4.17	0.78
SGBoost	1.00	0.54	0.22	0.99	0.92	23.01	2.96	0.88
HGBoost	0.99	1.51	0.79	0.99	0.91	24.97	3.28	0.84
XGBoost	0.99	1.60	0.84	1.00	0.93	20.61	2.94	0.89

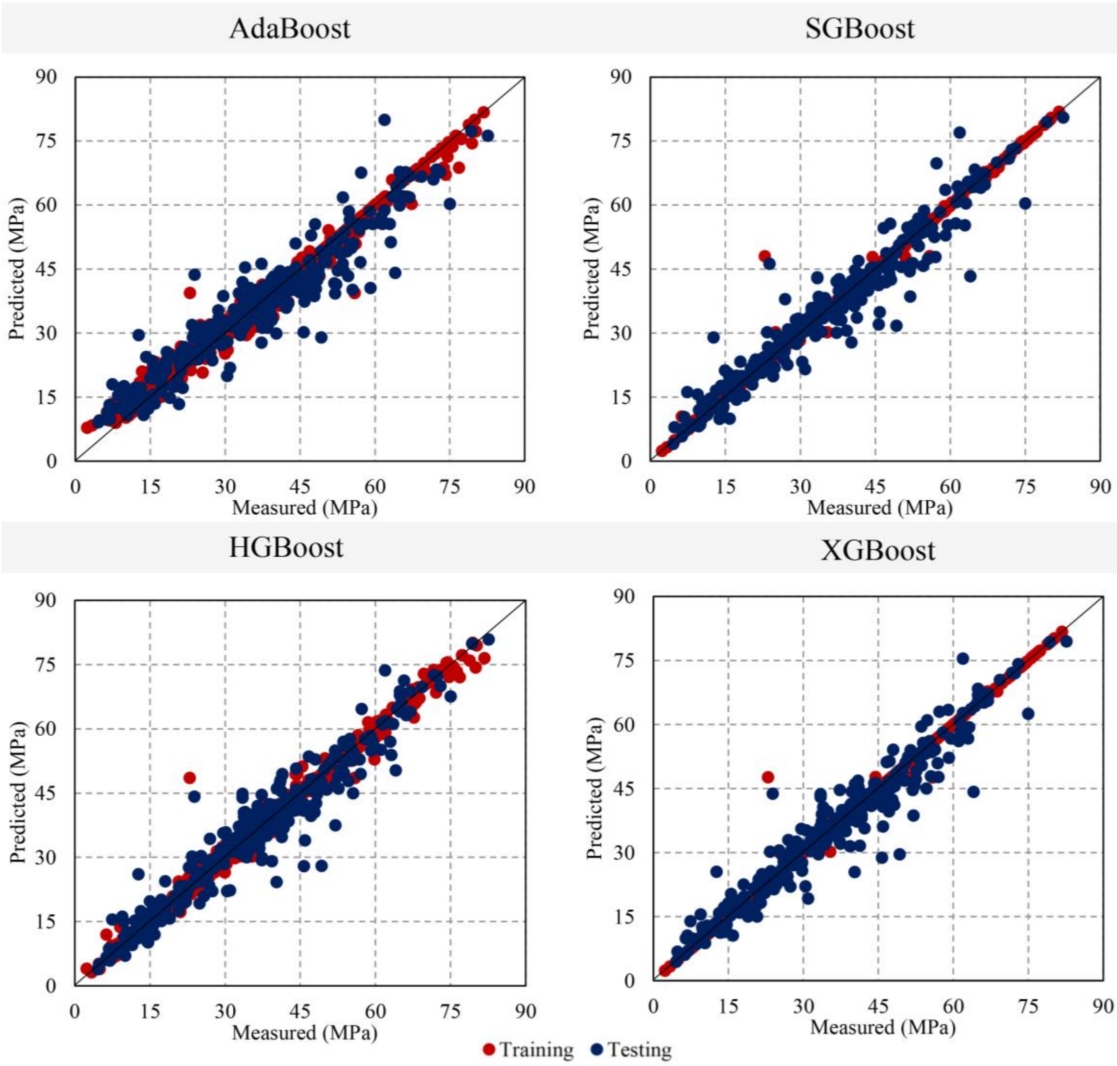

Fig. 11. Compressive strength estimation using boosting technique

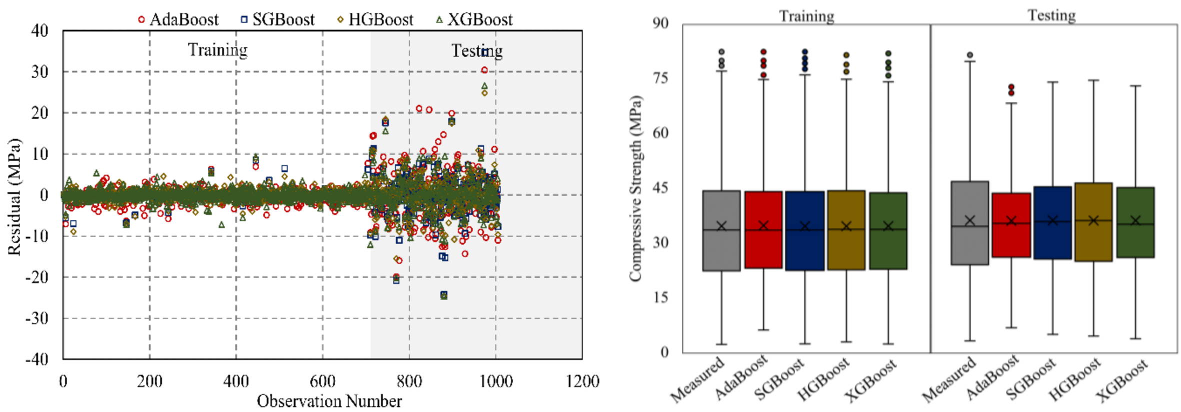

Fig. 12. a) Residual plot; and b) Box plots of the machine learning techniques

4. Conclusions

Traditionally, decision-makers have been mainly concerned with meeting needed objectives for specific concrete characteristics, such as obtaining a defined early-age compressive strength besides keeping a workable concrete when developing concrete mixtures. At present, this situation has changed after emerging modern technologies that offer advanced methods to quantify and fine-tune the properties of concrete. This investigation is primarily focused on using machine learning in concrete mixture design and developing a helpful tool to be applied in engineering practices. It evaluated the quality of histogram-based gradient boosting algorithms in estimating the compressive strength of the concrete mix. The performance analysis of the HGBoost model indicated that it returns high accuracy, but a slight reduction was produced compared to SGBoost and XGBoost in the testing case, as previously stated. Further works are still needed in this field to propose more generalized models capable of estimating other properties of concrete, such as its durability and dynamic behavior. In addition, it is crucial to report the performance of other newly developed models in the rapidly growing field of machine learning, including those approaches that can be built to serve specified tasks.

5. References

Ahmad, A., Ahmad, W., Aslam, F. and Joyklad, P. (2022). "Compressive strength prediction of fly ash-based geopolymer concrete via advanced machine learning techniques", *Case Studies in Construction Materials*, 16(1 June), e00840, https://doi.org/10.1016/j.cscm.2021.e00840.

Al Houri, A., Habib, A., Elzokra, A. and Habib, M. (2020). "Tensile testing of soils: History, equipment and methodologies", *Civil Engineering Journal*, 6(3), 591-601, https://doi.org/10.28991/cej-2020-03091494.

Amidi, Y., Nazari, B., Sadri, S. and Yousefi, A. (2021). "Parameter estimation in multiple dynamic synaptic coupling model using Bayesian Point Process State-Space Modeling framework", *Neural Computation*, 33(5), 1269-1299, https://doi.org/10.1162/neco_a_01375.

Asteris, P.G., Skentou, A.D., Bardhan, A., Samui, P. and Pilakoutas, K. (2021). "Predicting concrete compressive strength using hybrid ensembling of surrogate machine learning models", *Cement and Concrete Research*, 145(1 July), 106449, https://doi.org/10.1016/j.cemconres.2021.106449.

Cai, J., Li, X., Tan, Z. and Peng, S. (2021). "An assembly-level neutronic calculation method based on LightGBM algorithm", *Annals of Nuclear Energy*, 150(1 January), 107871, https://doi.org/10.1016/j.anucene.2020.107871.

Chaabene, W.B., Flah, M. and Nehdi, M.L. (2020). "Machine learning prediction of mechanical properties of concrete: Critical review", *Construction and Building Materials*, 260(10 November), 119889, https://doi.org/10.1016/j.conbuildmat.2020.119889.

Chen, T., and Guestrin, C. (2016). "Xgboost: A scalable tree boosting system", *Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining* (pp. 785-794), San Francisco, California, USA.

Chopra, P., Sharma, R.K. and Kumar, M. (2016). "Prediction of compressive strength of concrete using artificial neural network and genetic programming", *Advances in Materials Science and Engineering*, 2016(10 January), Article ID 7648467, https://doi.org/10.1155/2016/7648467.

DeRousseau, M.A., Kasprzyk, J.R. and Srubar III, W.V. (2018). "Computational design optimization of concrete mixtures: A review", *Cement and Concrete Research*, 109(1 July), 42-53, https://doi.org/10.1016/j.cemconres.2018.04.007.

Drucker, H. (1997). Improving regressors using boosting techniques", 97th International Conference on Machine Learning (ICML), Vol. 97, pp. 107-115, https://doi.org/10.1155/2016/7648467.

Elzokra, A., Al Houri, A., Habib, A., Habib, M. and Malkawi, A.B. (2020). "Shrinkage behavior of conventional and nonconventional concrete: A review", *Civil Engineering Journal*, 6(9), 1839-1851, https://doi.org/10.28991/cej-2020-03091586.

Farooq, F., Nasir Amin, M., Khan, K., Rehan Sadiq, M., Faisal Javed, M., Aslam, F. and Alyousef, R. (2020). "A comparative study of random forest and genetic engineering programming for the prediction of compressive strength of high strength concrete (HSC)", *Applied Sciences*, 10(20), 7330, https://doi.org/10.3390/app10207330.

Feng, D.C., Liu, Z.T., Wang, X.D., Chen, Y., Chang, J.Q., Wei, D.F. and Jiang, Z.M. (2020). "Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach", *Construction and Building Materials*, 230(10 January), 117000, https://doi.org/10.1016/j.conbuildmat.2019.117000.

Freund, Y., and Schapire, R.E. (1997). "A decision-theoretic generalization of on-line learning and an application to boosting", *Journal of Computer and System Sciences*, 55(1), 119-139, https://doi.org/10.1006/jcss.1997.1504.

Gholamzadeh Chitgar, A. and Berenjian, J. (2021). "Performance evaluation of RBF networks with various variables to forecast the properties of SCCs", *Civil Engineering Infrastructures Journal*, 54(1), 59-73.

Guryanov, A. (2019). "Histogram-based algorithm for building gradient boosting ensembles of piecewise linear decision trees", *International Conference on Analysis of Images, Social Networks and Texts* (pp. 39-50), Springer, Cham, https://doi.org/10.1007/978-3-030-37334-4_4.

Habib, A., Houri, A.A., Habib, M., Elzokra, A. and Yildirim, U. (2021). "Structural performance and Finite Element modeling of roller compacted concrete dams: A review", *Latin American Journal of Solids and Structures*, 18(13 May), https://doi.org/10.1590/1679-78256467.

Habib, A. and Yildirim, U. (2021). "Prediction of the dynamic properties in rubberized concrete", *Computers and Concrete*, 27(3), 185-197, https://doi.org/10.12989/cac.2021.27.3.185.

Han, Q., Gui, C., Xu, J. and Lacidogna, G. (2019). "A generalized method to predict the compressive strength of high-performance concrete by improved random forest algorithm", *Construction and Building Materials*, 226(30 November), 734-742, https://doi.org/10.1016/j.conbuildmat.2019.07.315.

Kaloop, M.R., Kumar, D., Samui, P., Hu, J.W. and Kim, D. (2020). "Compressive strength prediction of high-performance concrete using gradient tree boosting machine", *Construction and Building Materials*, 264(20 December), 120198, https://doi.org/10.1016/j.conbuildmat.2020.120198.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y. (2017). "Lightgbm: A highly efficient gradient boosting decision tree", *NeurIPS Proceedings*, *Advances in Neural Information Processing Systems (NIPS 2017)*, 30, 3146-3154.

Ke, X., and Duan, Y. (2021). "A Bayesian machine learning approach for inverse prediction of high-performance concrete ingredients with targeted performance", *Construction and Building Materials*, 270(8 February), 121424, https://doi.org/10.1016/j.conbuildmat.2020.121424.

Khademi, F., Akbari, M., Jamal, S.M. and Nikoo, M. (2017). "Multiple linear regression, artificial neural network, and fuzzy logic prediction of 28 days compressive strength of concrete", *Frontiers of Structural and Civil Engineering*, 11(1), 90-99, https://doi.org/10.1007/s11709-016-0363-9.

Kim, B., Lee, D. E., Hu, G., Natarajan, Y., Preethaa, S. and Rathinakumar, A.P. (2022). "Ensemble machine learning-based approach for predicting of FRP-concrete interfacial bonding", *Mathematics*, 10(2), 231, https://doi.org/10.3390/math10020231.

Lee, S.C. (2003). "Prediction of concrete strength using artificial neural networks", *Engineering Structures*, 25(7), 849-857, https://doi.org/10.1016/S0141-0296(03)00004-X.

Loh, W.Y. (2011). "Classification and regression trees", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1), 14-23.

Lu, H., Karimireddy, S.P., Ponomareva, N. and Mirrokni, V. (2020). "Accelerating gradient boosting machines", *International Conference on Artificial Intelligence and Statistics,* (pp. 516-526), PMLR.

Malkawi, A.B., Habib, M., Aladwan, J. and Alzubi, Y. (2020). "Engineering properties of fibre reinforced lightweight geopolymer concrete using palm oil biowastes", *Australian Journal of Civil Engineering*, 18(1), 82-92, https://doi.org/10.1080/14488353.2020.172195.

Mease, D. and Wyner, A.J. (2008). "Evidence contrary to the statistical view of boosting", *Journal of Machine Learning Research*, 9, 131- 156.

Nguyen-Sy, T., Wakim, J., To, Q.D., Vu, M.N., Nguyen, T.D. and Nguyen, T.T. (2020). "Predicting the compressive strength of concrete from its compositions and age using the extreme gradient boosting method", *Construction and Building Materials*, 260(10 November), 119757, https://doi.org/10.1016/j.conbuildmat.

Ni, H.G. and Wang, J.Z. (2000). "Prediction of compressive strength of concrete by neural networks", *Cement and Concrete Research*, 30(8), 1245-1250, https://doi.org/10.1016/S0008-8846(00)00345-8.

Salimbahrami, S.R. and Shakeri, R. (2021). "Experimental investigation and comparative machine-learning prediction of compressive strength of recycled aggregate concrete", *Soft Computing*, 25(2), 919-932, https://doi.org/10.1007/s00500-021-05571-1.

Shepovalov, M. and Akella, V. (2020). "FPGA and GPU-based acceleration of ML workloads on Amazon cloud-A case study using gradient boosted decision tree library", *Integration*, 70(1 January), 1-9, https://doi.org/10.1016/j.vlsi.2019.09.007.

Topcu, I.B. and Sarıdemir, M. (2008). "Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic", *Computational Materials Science*, 41(3), 305-311, https://doi.org/10.1016/j.commatsci.2007.04.009.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. and

Vanderplas, J. (2011). "Scikit-learn: Machine learning in Python", *The Journal of Machine Learning Research*, 12(1 November), 2825-2830.

Wardeh, G., Ghorbel, E. and Gomart, H. (2015). "Mix design and properties of recycled aggregate concretes: applicability of Eurocode 2", *International Journal of Concrete Structures and Materials*, 9(1), 1-20, https://doi.org/10.1007/s40069-014-0087-y.

Yeh, I.C. (1998). "Modeling of strength of high-performance concrete using artificial neural networks", *Cement and Concrete Research*, 28(12), 1797-1808, https://doi.org/10.1016/j.commatsci.2007.04.009.

Yeh, I.C. (2006). "Analysis of strength of concrete using design of experiments and neural networks", *Journal of Materials in Civil Engineering*, 18(4), 597-604, https://doi.org/10.1061/(ASCE)0899-1561(2006)18:4(597).

Yeh, I.C. (2007). "Computer-aided design for optimum concrete mixtures", *Cement and Concrete Composites*, 29(3), 193-202, https://doi.org/10.1016/j.cemconcomp.2006.11.001.

Ziolkowski, P. and Niedostatkiewicz, M. (2019). "Machine learning techniques in concrete mix design", *Materials*, 12(8), 1256, https://doi.org/10.3390/ma12081256.