Civil Engineering Infrastructures Journal 2024, 57(2): 247-265
DOI: 10.22059/ceij.2023.354679.1910

RESEARCH PAPER

Machine Learning-Based Estimation of Concrete Compressive Strength: A Multi-Model and Multi-Dataset Study

Hoang, N.D.[1*] and Tran, D.V.[1]

[1] Ph.D. Instructor, Faculty of Civil Engineering, Duy Tan University, Da Nang, Vietnam.

Received: 31 Jan. 2023; Revised: 24 May 2023; Accepted: 12 Jun. 2023

ABSTRACT: Concrete is a commonly used construction material due to its favourable engineering properties, such as high compressive strength, good durability, and resistance to corrosion. Accurate predictions of the compressive strength of this material significantly reduce the time and effort required by laboratory tests. The current paper aims to compare the performance of prominent machine learning-based approaches used for predicting the compressive strength of concrete. In addition, 11 historical datasets, collected from the literature, are used. The diversity of the input features, the data dimensionality, and the number of instances can be helpful to evaluate the generalization capability of the employed machine learning models. Repetitive data sampling processes, consisting of 20 independent runs, are carried out to obtain the machine learning models' performances. Through experiments, it can be shown that the gradient boosting machines attain the best performance. Notably, the extreme gradient boosting machine has achieved the best outcome in five historical datasets.

Keywords: Concrete Compressive Strength, Gradient-Boosting Machine, Machine Learning, Regression Models, Comparative Study.

1. Introduction

Concrete has been commonly used in construction due to it is favorable engineering properties such as high compressive strength, good durability, resistance to corrosion, etc. (Chung et al., 2021). Basically, a concrete mix consists of four primary constituents: fine aggregate, coarse aggregate, cement, and water. These constituents can be easily accessed in the local market. These advantages allow concrete to be widely used in various forms of construction projects around the globe.

Moreover, in recent years, many studies have found that supplementary materials like Fly Ash (FA) (Gomaa et al., 2021), Ground Granulated Blast-furnace Slag (GGBS) (Kandiri et al., 2020), silica fume (Kang et al., 2021), and many other industrial/agricultural waste or by-products can be blended into concrete to meliorate its mechanical properties. Han et al. (2020), pointed out that the inclusion of those supplementary materials into concrete offers significant environmental benefits and also enhances the longevity and resiliency of concrete structures. Among various concrete properties, the Compressive Strength (CS) is apparently the most critical since this index directly governs the structural safety and must be

* Corresponding author E-mail: hoangnhatduc@duytan.edu.vn

specified to determine the performance of concrete structures throughout their lifecycles (Zhao et al., 2022).

When designing concrete mixes, one significant challenge is to select appropriate materials to achieve a targeted compressive strength. Therefore, it is of immense advantage to possess reliable predictive models that can yield accurate estimation of the CS based on the amount or proportion of the concrete components. These models can help to come up with meaningful predictions that can help to reduce the time and cost required for making and testing samples. Historical data plays a crucial role in constructing robust prediction models.

Recent studies with extensive data collection and model performance comparison have demonstrated the advantages of advanced Machine Learning (ML) models over conventional statistical regression analysis-based models (Ben Chaabene et al., 2020). Accordingly, various ML-based models have been proposed in the literature. Artificial Neural Network (ANN) and Genetic Programming (GP) was used in Chopra et al. (2016) to predict the CS at 28, 56, and 91 days. One advantage of the GP is its capability of constructing predictive formulas used for the CS prediction. However, using the collected dataset, the authors found that ANN is preferable to the GP with respect to predictive accuracy.

ANN was also utilized by Hocine (2018) to estimate the CS of limestone filler concrete and High-Performance Concrete (HPC), respectively. Although ANN-based models are capable nonlinear regressors, their performance substantially depends on the training algorithms. Current training methods of ANNs rely on stochastic gradient descent-based algorithms that are susceptible to being trapped in sub-optimal solutions. Gene expression programming was used by Shahmansouri et al. (2020) to estimate ground granulated blast-furnace slag blended concrete. Using a dataset consisting of 351 specimens, the authors successfully constructed predictive formulas with high degrees of data fitting. Zhang and Aslani (2021) proposed a data-driven approach based on a back-propagation neural network incorporating ultrasonic pulse velocity for estimating the CS of lightweight aggregate concretes.

Nguyen et al. (2021) developed predictive models based on Support Vector Machine regressor (SVM), ANN, Gradient Boosting Machine (GBM), and Extreme Gradient Boosting (XGBoost) for estimating the CS of HPC. The authors found that GBR and XGBoost perform better than SVM and ANN. Nevertheless, this paper did not explore the capability of XGBoost in predicting the CS of other widely-used concretes (e.g., self-compacting concrete and class F fly ash-blended concrete). An intelligent approach that hybridizes a genetic algorithm and a backpropagation neural network was proposed by Zhang et al. (2021) for predicting the CS of rubberized concrete. GBM was also used by Rathakrishnan et al. (2022) to model the CS of concrete mixes blended with ground granulated blast-furnace slag.

Ensemble learning models based on adaptive boosting machine, GBM, XGBoost, and random forest were proposed by Li and Song (2022). The mixtures included admixtures such as fly ash and silica fume. The authors observed good performance of GBM that achieved a coefficient of determination (R^2) up to 0.96. Naser et al. (2022) applied Multivariate Adaptive Regression Splines (MARS) for estimating the CS of green concrete; MARS obtained the most desired performance (with $R^2 = 0.89$) which is better than that of SVM and random forest.

Hoang (2022) reported superior performances of neural computing models and XGBoost over other data-driven approaches for predicting the CS of self-consolidating concrete; however, the predictive capabilities of MARS and piecewise linear regression models were not investigated. In general, recent reviews and comparative works (Ben Chaabene et al.,

2020; Khambra and Shukla, 2021; Mirrashid and Naderpour, 2020) pointed out an increasing trend of using advanced data-driven tools in estimating this crucial mechanical property of concrete. However, the inclusion of various mineral additions, supplementary materials, and admixtures increases the complexity of the concrete. Thus, it is beneficial for the research community and practitioners to obtain information regarding the predictive capability of prominent ML models in estimating the CS of samples stored in various historical databases.

The current paper aims to compare the capabilities of prominent ML models, including XGBoost, GBM, SVM, MARS, GP, ANN and Sequential Piecewise Linear Regression (SPLR). The selections of the first six models are based on reviewing recent works on ML-based CS prediction (Naser et al., 2022; Nguyen et al., 2021; Tanyildizi and Çevik, 2010; Ullah et al., 2022; Zhang and Aslani, 2021). In addition, the SPLR model has been shown to be a capable nonlinear regressor (Hoang, 2019); however, its performance in modeling the CS has not yet been investigated.

Furthermore, 11 historical datasets, gathered from previous experimental works, are employed to train and test the ML models. Repeated data sampling processes, consisting of 20 runs, are performed to obtain statistical criteria that express the performance of the models. The current study aims to report the prediction results of the employed ML models in estimating the CS of concretes in multiple datasets.

The outcomes of this paper may serve as initial guidance for researchers in selecting appropriate ML models for the task of interest. Since data samples are crucial for constructing reliable ML models, the scope of the paper is limited to the datasets that are openly accessed via data repositories or reported in reliable sources such as academic journal articles. Accordingly, the current work contributes to the body of knowledge in the following aspects:

i) This study investigates the performances of a wide range of ML models, including the powerful methods of gradient boosting machines, for predicting the CS of concretes.

ii) Although XGBoost has shown outstanding performances in modeling the mechanical properties of HPC, its capability in estimating the CS of other concretes (e.g., self-compacting concrete, class F fly ash-blended concrete, rubberized concrete) has not been fully explored.

iii) Datasets representing diverse types of concrete are gathered from previous works to construct and test the ML approaches.

iv) Through experiments, it can be shown that the gradient boosting machines cannot attain the best performance in all datasets. Nevertheless, SVM, MARS, and GP may outperform the gradient-boosting-based models in predicting the CS of certain types of concrete.

2. The Employed Machine Learning Models

2.1. Extreme Gradient Boosting Machine (XGBoost)

The XGBoost (Chen and Guestrin, 2016) is an improved version of the standard the gradient boosting algorithm. This method is essentially an ensemble of boosted regression trees. the model training phase of the ML method is fast since it can be executed in parallel (Zhang et al., 2019).

Let $D = \{(x_i, y_i)\}$ be a collected dataset including n samples and d predictor variables. The XGBoost employs Z additive functions for estimating the target variable of the CS as follows:

$$\hat{y} = \phi(x_i) = \sum_{z=0}^{Z-1} f_z(x_i) \quad (1)$$

where $f_z \in F$: is the space of classification trees and $\hat{y}$: is the estimated CS value.

The objective function used in the model training phase is given by Eq. (2).

$$L^t = \sum_{i=0}^{n-1} l(y_i, \hat{y}_i^{t-1} + f_t(x_i)) + \Omega(f_t) \quad (2)$$

where l: is a loss function which calculates the difference between the predicted (y_i) and the actual variable ($\hat{y}_i$) of the concrete CS at an iteration t. $\Omega(f)$: denotes a function that regularizes the model complexity. The regularization function $\Omega(f)$ is stated as follows:

$$\Omega(\mathrm{f}) = \gamma T + \frac{1}{2}\lambda \sum_{j=0}^{T-1} w_j^2 \quad (3)$$

where γ: is the minimum reduction coefficient, λ: denotes the regularization parameter, T: represents the number of leaves in a classification tree and w: is the weights associated with the leaves.

2.2. Gradient Boosting Machine (GBM)

The GBM iteratively combines a set of weak learners (e.g. trees) to attain a robust learner with enhanced fitting accuracy. This ML method can be viewed as a numerical optimization approach that establishes an additive model that minimizes a loss function (Friedman, 2001). For regression problems, the commonly used loss function is the mean squared error.

Hence, the GBM iteratively adds a new regression tree iteratively that helps reduce the used loss function. Via the process of fitting decision trees to the residuals, the overall ensemble model is enhanced in the regions where it did not well fit the data. The GBM operates by fitting a decision tree f^k at k^{th} iteration using the residual of the previous iteration r_{k-1}.

Accordingly, the updated model $f(x)$ is computed as follows:

$$f(x) = f(x) + \alpha f^k(x) \quad (4)$$

subsequently, the residual r_k is updated as follows:

$$r_k = r_{k-1} - \alpha f^k(x) \quad (5)$$

2.3. Support Vector Machine (SVM)

The SVM (Drucker et al., 1996) utilizes a margin of tolerance (ε) for fitting a nonlinear function that describes the mapping relationship between the CS and the concrete mix's constituents. This ML model minimizes the training error and concurrently searches for a hyper-plane that has a maximal margin. Additionally, the kernel function is used to cope with nonlinearity.

In detail, the kernel function has the role of mapping the data from its original input space to a high-dimensional space. In such high-dimensional space, a linear regression model can be established.

The training phase of a SVM model constructs a linear model $f(x)$ that minimizes the structural risk in the feature space:

$$f(x) = w^T \varphi(x) + b \quad (6)$$

where $\phi(x)$: denotes to a nonlinear mapping from the input space to the high-dimensional feature space; w and b: are the model parameters which are used to specify a SVM model. To compute them, the following constrained optimization problem needs to be solved:

$$\text{Min. } \frac{1}{2}\| w \|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*)$$

subjected to:

$$\begin{cases} y_i - (\langle w, \varphi(x_i)\rangle + b) \le \varepsilon + \xi_i \\ (\langle w, \varphi(x_i)\rangle + b) - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \quad (7)$$

where C: denotes the complexity coefficient; ξ_i and ξ_i^*: are the slack variables (Drucker et al., 1996); $i = 1, 2, \ldots, N$ and N is the number of data samples.

2.4. Multivariate Adaptive Regression Splines (MARS)

MARS (Friedman, 1991) constructs a nonlinear mapping relationship by dividing the high-dimensional learning space into sub-ranges of prediction variables. In

addition, this ML model employs piecewise linear functions and an adaptive training approach for model construction. A MARS model can be understood as a set of simple basis functions which describe the relationship between predictor variables and the CS.

A basis function can be expressed as follows:

$$b_m(x) = max(0, C - x) \quad \text{or} \quad b_m(x) = max(0, x - C) \tag{8}$$

where b_m: is a basis function; x: denotes an input variable; C: is a threshold parameter used to divide the original range of x into sub-ranges.

Using the concept of the basis function, the general model can be expressed as follows:

$$f(x) = \alpha_0 + \sum_{m=1}^{M} \alpha_m b_m(x) \tag{9}$$

where $\alpha_0, \alpha_1, ..., \alpha_M$: denote weighting coefficients of the model, $f(x)$: yields the output of the CS and M: represents the number of weighting coefficients.

2.5. Genetic Programming (GP)

The GP (Koza, 1994) is a ML approach inspired by real-world biological systems. This ML method is capable of generating mathematical equations (also called programs) to describe the behaviors of nonlinear systems. Thus, this method can be used to construct predictive equations used for estimating the CS without any assumptions about the prior form of the mapping relationships. GP is able to evolve both the model structure and its parameters according to the collected dataset.

The basic operations of a GP model are described as follows (Koza, 1994; Searson 2015): 1) The generation of a random population of programs; 2) The evaluation of programs with a specified fitness function; 3) The generation of new programs based on the processes of reproduction, mutation, and crossover; 4) The process of self-tuning and comparison of fitness; and 5) the selection of best program through evolutionary competition.

2.6. Artificial Neural Network (ANN)

The ANN is essentially an interconnected network of individual neurons (Haykin, 2008). This ML method is capable of simulating the information processing and knowledge generalization in the human brain. Each neuron uses a nonlinear activation function to process its input signal. To construct an ANN-based CS prediction model, a historical dataset is first collected. Subsequently, the back-propagation framework (Rumelhart et al., 1986) coupled with an optimizer is employed to fit the model's parameters including the weight matrix of the hidden layer (W_1), the weight matrix of the output layer (W_2), the bias vector of the hidden layer (b_1), and the bias vector of the output layer (b_2). The ANN model used for estimating the CS can be stated as follows:

$$f(x) = b_2 + W_2 \times \sigma(b_1 + W_1 \times x) \tag{10}$$

where x: is the matrix of input variables; σ: denotes the activation function.

In the case of nonlinear function approximation, the Mean Square Error (MSE) loss function is often employed. Additionally, the sigmoid activation function can be used. The adaptive moment estimation (Adam) (Kingma and Ba, 2015) is the state-of-the-art optimizer employed for training the ANN model.

The Adam is an effective first-order gradient-based optimization of stochastic objective functions. This algorithm harnesses the information obtained from the average of the second moments of the gradients to enhance the performance of the optimization process.

A Piecewise Linear Regression Model (PLRM) is a data modeling method that uses Individual linear models to fit a subset of the training data. The transition location

between separated domains of input features is often called a breakpoint or a knot (Breiman, 1993).

The appropriate value of a knot is estimated from the training dataset. SPLR, described by Hoang (2019), employs a sequential algorithm to compute the knots of a PLRM. The model training phase of the SPLR relies on a set of hinge hydrophobic characteristics functions (Breiman, 1993).

This function basically separates the training data into separate domains in which individual linear models can be used to fit the dataset locally. A SPLR model with one predictor variable X and one break point or knot b is given by:

$$\begin{aligned} Y = {} & \beta_0 + \beta_{11}\max(0, \text{sign}(X - b)) + \\ & \beta_{12}\max(0, \text{sign}(b - X)) + \\ & \beta_{21}\max(0, X - b) + \\ & \beta_{22}\max(0, b - X) \end{aligned} \tag{11}$$

where β_0, β_{11} and β_{12}: denote the bias parameters, β_{21} and β_{22}: represent the slope parameters of the two linear models separated by a knot. A general SPLR model used for estimating the CS values is expressed as follows:

$$Y = \sum_{d=1}^{D}\sum_{v=1}^{V_d} LF_{d,v}(X_d) \tag{12}$$

where d: is the index of predictor variables (e.g. the components of a concrete mix), D: is number of predictor variables, v: denotes the index of the hinge function of the d^{th} predictor variable, V_d: represents the number of hinge functions of the d^{th} explanatory variable.

3. The Collected Datasets

To assess the capability of the employed ML models, this study has selected 11 historical datasets compiled by the previous works. In these datasets, the number of features ranges from 4 to 10. The number of data samples is from 70 to 1030. The selected datasets include normal concrete (Al-Jamimi et al., 2022), high-strength concrete (Al-Shamiri et al., 2019), self-compacting concrete (Kovacevic et al., 2022), lightweight concrete (Tanyildizi and Çevik 2010; Ullah et al., 2022), and high-performance concrete (Videla and Gaedicke 2004; Yeh 1998). In addition, concrete with the alternative binder of GGBS (Shahmansouri et al., 2020) and the alternative aggregate of rubber (Gesoglu et al., 2009) are also considered. The diversity of the features and the number of data instances can be helpful to reveal the overall predictive capability of the ML approaches. The compiled datasets are summarized in Table 1 that provides information regarding the number of features, number of samples, descriptions, and the sources of the data.

Table 1. The employed datasets

Dataset	Number of input features	Number of samples	Description	Reference
1	7	108	Plain and blended cement concretes	Al-Jamimi et al. (2022)
2	5	324	High-strength concrete	Al-Shamiri et al. (2019)
3	8	70	Rubberized concretes	Gesoglu et al. (2009)
4	7	262	Self-Compacting Concrete with class F fly ash	Kovačević et al. (2022)
5	8	144	Concrete containing fly ash and silica fume	Pala et al. (2007)
6	5	117	Concrete containing GGBS	Shahmansouri et al. (2020)
7	6	96	Lightweight concrete containing silica fume	Tanyildizi and Cevik (2010)
8	4	191	Lightweight foamed concrete	Ullah et al. (2022)
9	10	195	Portland blast-furnace slag cement high-performance concrete	Videla and Gaedicke (2004)
10	8	1030	High performance concrete	Yeh (1998)
11	10	323	Concrete with manufactured sand	Zhao et al. (2017)

Table 2 provides an overview of the CS is influencing factors used in each dataset. Furthermore, the frequency of the predictor variables is demonstrated in Figure 1. It is noted that to standardize the range of the variables, this study relied on the Z-score normalization method. The Z-score normalization equation is given by:

$$X_Z = \frac{X_O - \mu_X}{\sigma_X} \tag{13}$$

where *d*: is the index of predictor variables (e.g. the components of a concrete mix), *D*: is number of predictor variables, *v*: denotes the index of the hinge function of the d^{th} predictor variable, V_d: represents the number of hinge functions of the d^{th} explanatory variable.

Table 2. The employed datasets

Input variables	Note	Dataset										
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Water content	X_1	x	x	x	x	x	o	o	o	x	x	x
Cement content	X_2	x	x	x	x	x	o	x	x	x	x	o
Water to cement ratio	X_3	o	o	o	o	o	o	x	x	o	o	x
Water to binder ratio	X_4	o	o	o	o	o	o	o	o	o	o	x
Silica fume content	X_5	x	o	x	o	x	x	x	o	x	o	o
Fly ash content	X_6	x	o	o	x	x	o	o	o	o	x	o
Coarse aggregate content	X_7	x	x	x	x	x	o	o	o	x	x	o
Fine aggregate content	X_8	x	x	x	x	x	o	o	x	x	x	o
Superplasticizer content	X_9	o	x	x	x	o	o	x	o	x	x	o
Crump rubber content	X_{10}	o	o	x	o	o	o	o	o	o	o	o
Tire chips content	X_{11}	o	o	x	o	o	o	o	o	o	o	o
High-rate water reducing agent content	X_{12}	o	o	o	o	x	o	o	o	x	o	o
NAOH concentration	X_{13}	o	o	o	o	o	x	o	o	o	o	o
Natural zeolite content	X_{14}	o	o	o	o	o	x	o	o	o	o	o
Ground granulated blast-furnace slag content	X_{15}	o	o	o	o	o	x	o	o	o	o	o
Temperature	X_{16}	o	o	o	o	o	o	x	o	o	o	o
Pumice aggregate	X_{17}	o	o	o	o	o	o	x	o	o	o	o
Foam	X_{18}	o	o	o	o	o	o	o	x	o	o	o
Entrapped air content	X_{19}	o	o	o	o	o	o	o	o	x	o	o
Blast furnace slag	X_{20}	o	o	o	o	o	o	o	o	o	x	o
Compressive strength of cement	X_{21}	o	o	o	o	o	o	o	o	o	o	x
Tensile strength of cement	X_{22}	o	o	o	o	o	o	o	o	o	o	x
D_{max} of crushed stone	X_{23}	o	o	o	o	o	o	o	o	o	o	x
Stone powder content in sand	X_{24}	o	o	o	o	o	o	o	o	o	o	x
Fineness modulus of sand	X_{25}	o	o	o	o	o	o	o	o	o	o	x
Sand ratio	X_{26}	o	o	o	o	o	o	o	o	o	o	x
Slump	X_{27}	o	o	o	o	o	o	o	o	x	o	o
Concrete age	X_{28}	x	o	o	x	x	x	o	o	x	x	x

4. Experimental Results and Discussion

The performance of the ML models with respect to the datasets of concrete strength samples is reported in this section of the article. For each dataset, 90% of the samples are used for training the prediction models; 10% of the dataset is used for testing the models predictive capability. To evaluate the ML models, the Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE) and coefficient of determination (R^2) are computed. The equations used to calculate those indices are presented in the following manner:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - t_i)^2} \tag{14}$$

$$MAPE = \frac{100}{N} \times \sum_{i=1}^{N}\frac{|y_i - t_i|}{y_i} \tag{15}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(t_i - y_i)^2}{\sum_{i=1}^{N}(t_i - \bar{t})^2} \tag{16}$$

where t_i and y_i: are the experimental and estimated CS of the i^{th} sample, respectively, N: denotes the number of samples, $\bar{t}$: is the mean of the actual CS. the RMSE measures the deviations between the experimental and estimated CS values.

It is actually the square root of the second sample moment of the deviations between estimated and actual values. This index aims to aggregate the magnitudes of the residuals in predictions for various data points into a single measurement, indicating the prediction error of a CS prediction model. The RSME is always non-negative and a RMSE of 0 implies a perfect fit to the collected data. Generally, the lower the RMSE is, the better the ML model is.

However, since the RMSE is scale-dependent, it is only valid to compare models fitting one dataset. The MAPE expresses the relative error of the model prediction. Similar to the RMSE, a small value of the MAPE indicates a good ML model. In addition, the R^2 represents the proportion of the variation in the CS of concrete that can be captured by the ML models (Mendenhall and Sincich, 2011).

A R^2=1 demonstrates a perfect regression model. Generally, the higher the R^2 is, the better the ML model is. In this study, the XGBoost model is constructed with the built-in functions provided in (XGBoost, 2021). The GBM, SVM, and ANN models are built with the Scikit-Learn library (Pedregosa et al., 2011). The MARS and GP are developed using the MATLAB toolboxes provided by Jekabsons (2016) and Searson (2015), respectively. The SPLR model is constructed in MATLAB by the author. It is noted that the five-fold cross validation processes (Wong and Yeh, 2020) were employed to set the free parameters of the ML models. The performances of the employed ML models in each dataset are presented in Tables 3 and 4. In Table 3, the model accuracy is presented in terms of the average RMSE obtained from the testing phase.

As can be observed from the experimental results the XGBoost model has achieved the best performances in 5 out of 11 datasets. The GBM model is the second best model with 5 times being the 1st rank.

The SVM model has been ranked as the best model twice. Meanwhile, each of the MARS and GP models attains the best outcome in one dataset. The model ranking is further demonstrated by Figure 2. Table 3 reports the average computation time of each model with respect to different datasets. It can be seen that the XGBoost's training phases are fast, with the average training time ranging from 0.03 to 0.06 s.

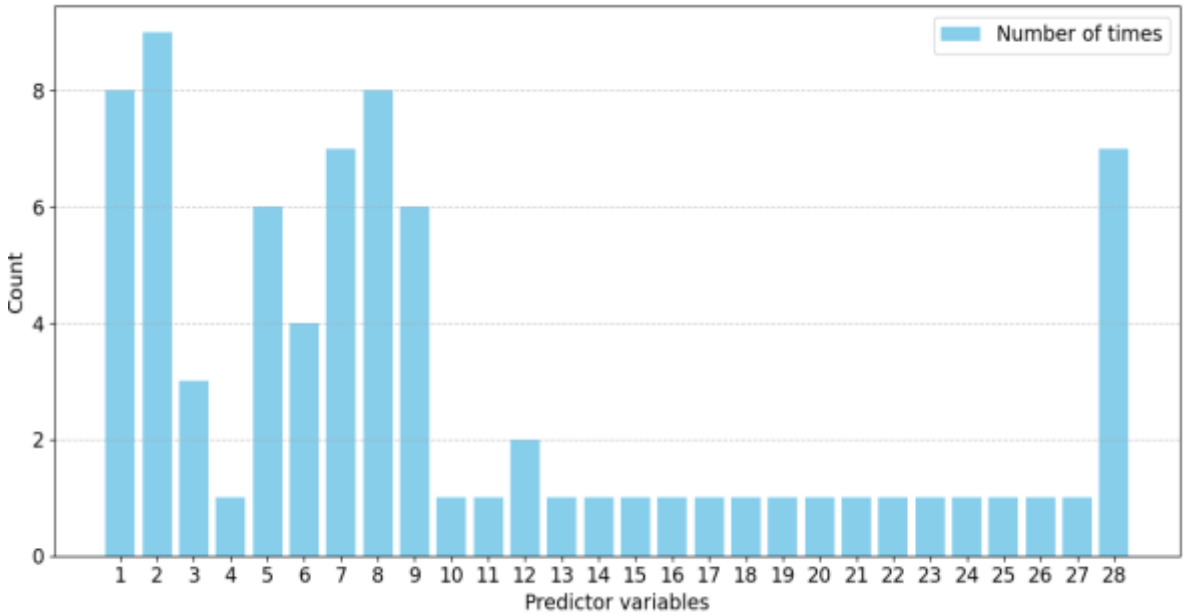


Fig. 1. The frequency of the predictor variables

Table 3. Prediction performance of the models in terms of RMSE

Data	Models						
	XGBoost	GBM	SVM	MARS	GP	ANN	SPLR
1	1.57	1.78	1.33	1.58	1.24	1.97	2.45
2	0.31	0.47	0.92	0.93	1.55	1.62	2.02
3	2.61	2.34	1.2	1.36	1.92	1.96	2.27
4	5.74	5.21	7.11	7.52	8.09	8.15	6.35
5	2.78	2.91	6.02	3.07	3.9	6.01	5.6
6	2.85	2.72	2.24	3.08	3.09	3.67	3.45
7	2.33	2.24	2.43	2.15	2.24	2.85	2.5
8	3.54	3.14	4.51	5.28	5.77	4.52	5.04
9	3.6	3.6	12.02	4.12	4.27	6.72	4.23
10	4.14	4.19	5.26	6.21	6.48	5.93	6.33
11	2.58	2.63	4.74	2.99	5.44	5.47	4.96

Table 4. Computational time (s)

Data	Concrete strength prediction models						
	XGBoost	GBM	SVM	MARS	GP	ANN	SPLR
1	0.03	0.03	0.01	0.12	96.38	0.04	0.05
2	0.03	0.08	0.14	11.33	222.92	0.08	0.05
3	0.03	0.04	0.01	0.10	1.70	0.05	0.06
4	0.04	0.13	0.04	21.15	227.53	0.14	0.70
5	0.03	0.04	0.01	3.80	101.20	0.04	0.03
6	0.03	0.02	0.02	1.50	103.18	0.21	0.06
7	0.03	0.01	0.01	0.06	101.44	0.07	0.08
8	0.04	0.02	0.01	0.13	103.64	0.11	0.03
9	0.03	0.03	0.01	0.66	100.40	0.15	0.75
10	0.06	0.20	0.15	7.30	968.45	0.40	0.29
11	0.03	0.10	0.02	10.87	19.00	0.17	0.14

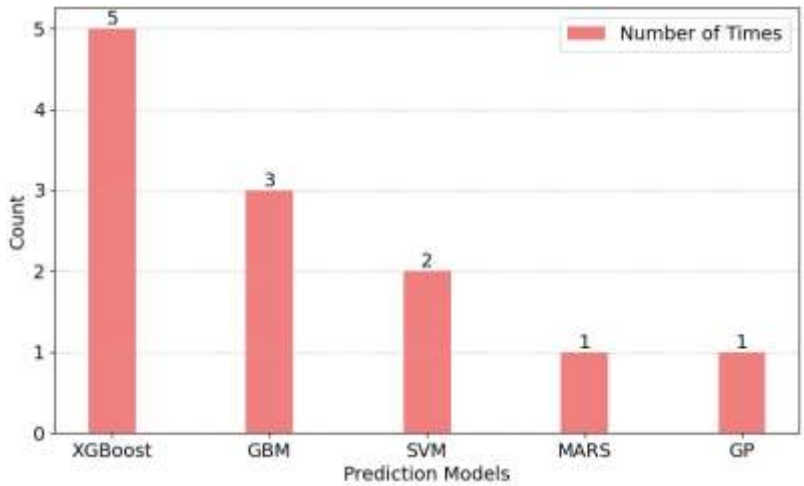


Fig. 2. The number of times that the model achieves the 1st rank

On the contrary, the GP requires much longer computational cost for model training; its training time can go up to 968 s in the Dataset 10. It is understandable because the training phase of the XGBoost model can be carried out in parallel. Meanwhile, the evolutionary operations performed by the GP's populations require much higher computational cost to accomplish. In addition, the detailed performance of the best model as sociated with each dataset is presented in Tables 5 and 6. As can be seen from the experimental results, the ML models are able to fit the datasets to a high degree. These results clearly demonstrate the capability of the ML models in CS prediction of various types of concrete. In general, the MAPE of the CS estimations can be as low as 2.48% in the case of the XGBoost used for

predicting the high-strength concrete samples provided in the Dataset 2 (Al-Shamiri et al., 2019). The R^2 values in all datasets are higher than 0.90 which indicates a sufficient degree of variance explanation. Additionally, in 8 out of 11 datasets, the R^2 is greater than or equal to 0.95. The SVM model used for predicting the CS of rubberized concrete achieves the R^2 of roughly 1 which indicates a nearly perfect fit. Herein, the red straight line represents a perfect fit. The scatter plots providing the overview of the data fitting results are presented in Figure 3.

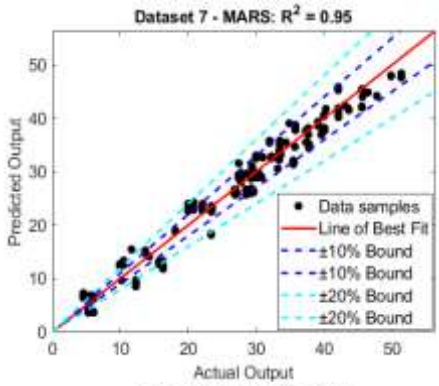
The nearer the data points (denoted as black circles) to the red line, the better they are fitted by the ML models. The lines of ±10% and ± 20% bounds are also added to inspect the magnitude of the models' residuals. Most of the prediction errors lie within the ±20% bound. The Dataset 4 (self-compacting concrete blended with class F fly ash), the Dataset 8 (lightweight foamed concrete), and the Dataset 10 (high-performance concrete) have high proportions of data beyond the ±20% bound. One possible reason for this phenomenon is that the complexity of the mapping functions between the CS and its influencing factors hidden in those data is high.

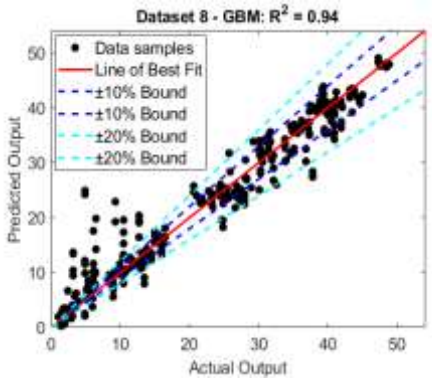
Table 5. Performances of the best models (from Dataset 1 to Dataset 6)

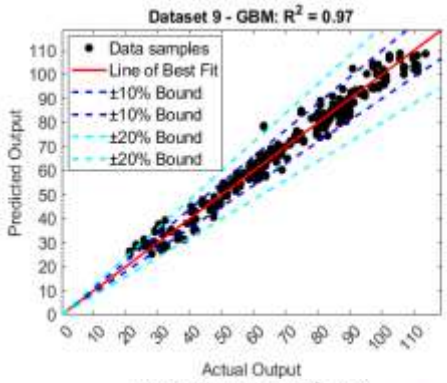
Dataset	The best model	Performance indices	Mean	Std.
1	GP	RSME	1.24	0.36
		MAPE (%)	3.25	1.02
		R^2	0.97	0.03
2	XGBoost	RSME	1.55	0.16
		MAPE (%)	2.48	0.35
		R^2	0.97	0.01
3	SVM	RSME	1.20	0.34
		MAPE (%)	2.93	1.47
		R^2	1.00	0.00
4	GBM	RSME	5.21	0.88
		MAPE (%)	12.34	2.45
		R^2	0.91	0.03
5	XGBoost	RSME	2.78	0.62
		MAPE (%)	5.13	1.33
		R^2	0.98	0.01
6	SVM	RSME	2.24	0.67
		MAPE (%)	2.72	0.78
		R^2	0.93	0.05

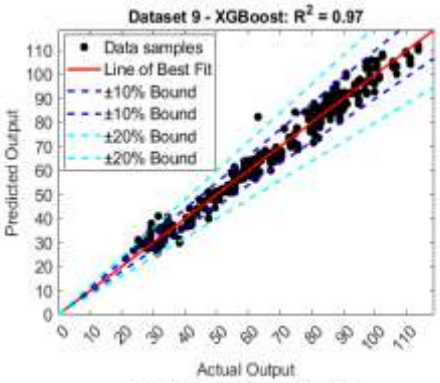
Table 6. Performances of the best models (from Dataset 7 to Dataset 11)

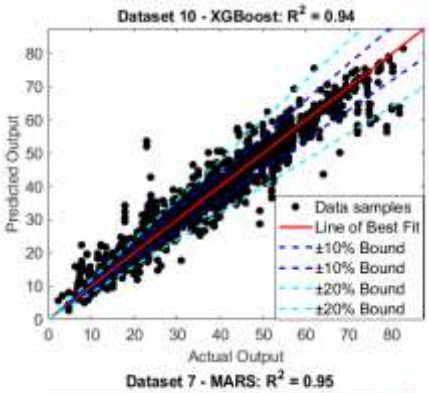
Dataset	The best model	Performance indices	Mean	Std.
7	MARS	RSME	2.15	0.39
		MAPE (%)	8.28	2.85
		R^2	0.95	0.03
8	GBM	RSME	3.14	0.95
		MAPE (%)	19.20	10.30
		R^2	0.94	0.05
9	XGBoost	RSME	3.60	0.63
		MAPE (%)	4.90	1.03
		R^2	0.97	0.01
9	GBM	RSME	3.60	0.69
		MAPE (%)	4.87	1.12
		R^2	0.97	0.01
10	XGBoost	RSME	4.14	0.44
		MAPE (%)	9.97	1.14
		R^2	0.94	0.02
11	XGBoost	RSME	2.58	0.85
		MAPE (%)	5.10	5.00
		R^2	0.97	0.02

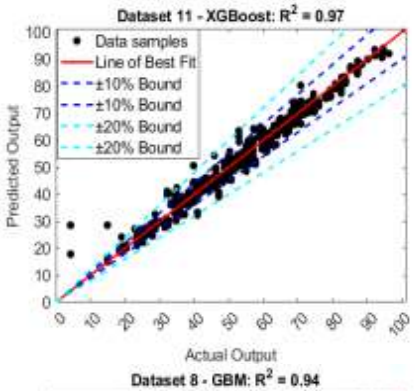

Dataset 7 - MARS: R^2 = 0.95
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound

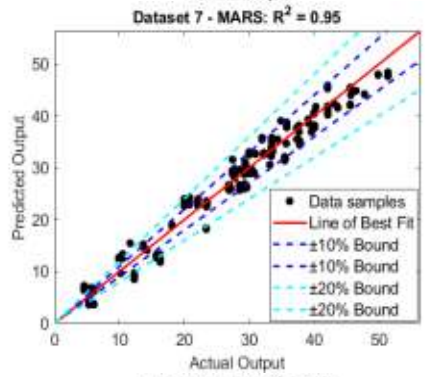

Dataset 8 - GBM: R^2 = 0.94
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound

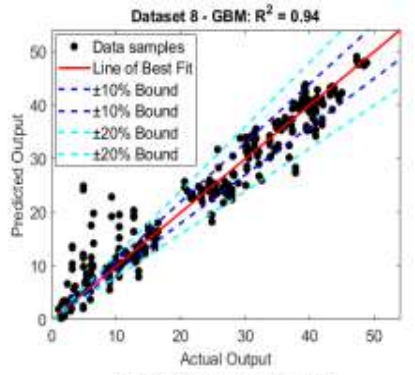

Dataset 9 - GBM: R^2 = 0.97
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound

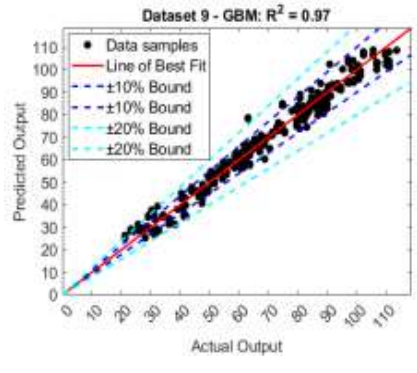

Dataset 9 - XGBoost: R^2 = 0.97
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound

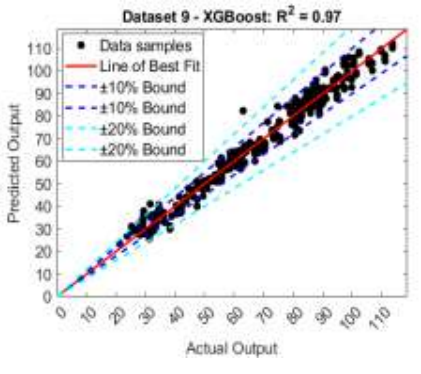

Dataset 10 - XGBoost: R^2 = 0.94
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound


Dataset 11 - XGBoost: R^2 = 0.97
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound


Dataset 7 - MARS: R^2 = 0.95
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound


Dataset 8 - GBM: R^2 = 0.94
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound


Dataset 9 - GBM: R^2 = 0.97
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound


Dataset 9 - XGBoost: R^2 = 0.97
Predicted Output
Actual Output
Data samples
Line of Best Fit
±10% Bound
±10% Bound
±20% Bound
±20% Bound

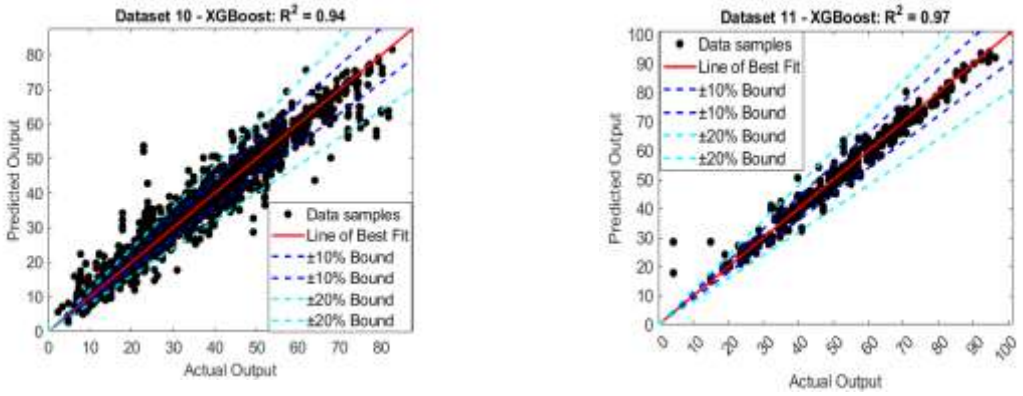


Fig. 3. Line of best fit plots

Table 7. Summary of the models' rank

Models	Dataset										
	1	2	3	4	5	6	7	8	9	10	11
XGBoost	3	1	7	2	1	3	4	2	1	1	1
GBM	5	2	6	1	2	2	2	1	1	2	2
SVM	2	3	1	4	7	1	5	3	7	3	4
MARS	4	4	2	5	3	4	1	6	3	5	3
GP	1	5	3	6	4	5	3	7	5	7	6
ANN	6	6	4	7	6	7	7	4	6	4	7
SPLR	7	7	5	3	5	6	6	5	4	6	5

Notably, the number of influencing factors used by the Dataset 8 is 4 which is quiet limited. It is possible that the CS values of the lightweight foamed concrete samples are affected by other explanatory factors that are not yet covered by the current work. Datasets 1, 2, 3, 5, 6, and 9 have the major proportion of the samples lying within the ±10% bound. This fact indicates a strong correlation between the estimated and the observed CS values.

Table 7 and Figure 4 summarize the ranks of the ML models with respect to different datasets. Apparently, the XGBoost is the best model with five 1st ranks, two 2nd ranks and two 3rd ranks. The GBM model is the second best one with three 1st ranks and six 2nd ranks. The SVM model succeeds the GBM model with two 1st ranks, one 2nd ranks, and three 3rd ranks. Each of the GP and MARS models achieves one 1st rank. However, MARS achieves the 2nd rank in one dataset. In addition, the highest rank of SPLR is the 3rd; the rank of the ANN model never goes higher than the 4th.

Thus, the outcomes of this study are in line with the previous works of Nguyen et al. (2021) and Kang et al. (2021) which points out the advantage of the XGBoost and GBM models. However, the SVM, MARS and GP models can also be the models of choice in the tasks of predicting the CS of the rubberized concrete, concrete containing GGBS and lightweight concrete containing silica fume. More details regarding the performances of the ML models are reported in Appendix 1 (boxplots of the model performance) and Appendix 2 (detailed model ranking).

5. Concluding Remarks

CS is considered the most important mechanical property of concrete. This index serves as a crucial indicator of the concrete quality. This study carried out a large-scale comparative study which investigates the performance of the prominent ML models used in estimating the CS of 11 historical datasets. The number of explanatory variables in these datasets ranged from 4 to 10. The number of samples ranged from 70 to 1030.

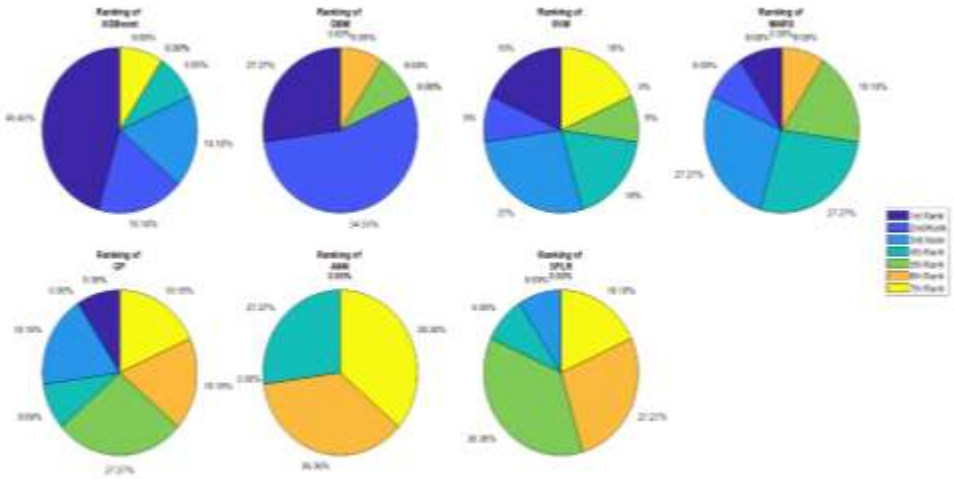


Fig. 4. Percentages of the model rankings

XGBoost, GBM, SVM, MARS, GP, ANN and SPLR were employed and their performances were benchmarked with the indices of RMSE, MAPE and R^2. Repetitive data sampling processes, consisting of 20 independent runs, were used for reliably assessing the model predictive capability.

Experimental results pointed out that the XGBoost model was achieved the most desired outcomes with 5 times of 1st rank. Its performance was followed by the GBM, SVM, MARS and GP. The highest rank that the SPRL achieves was 3rd and its performance was slightly better than that of ANN.

In general, XGBoost and GBM are the models of choice when dealing with the task of CS estimation. However, SVM, MARS, and GP should also be attempted for estimating the CS of the rubberized concrete, concrete containing GGBS, and lightweight concrete containing silica fume.

The prediction of the CS of diverse concrete types is a highly active research topic. Therefore, there are various datasets of concrete strength that are documented in the literature. In addition, researchers continuously compile, analyze, and report the testing records of the CS of concrete.

Hence, due to the limited time frame of the current study, the selected datasets in the current work cannot be comprehensive and cover all relevant datasets. In addition to the selected ML models, many other advanced methods (e.g., sophisticated ensembles of decision trees, light gradient boosting machines, neural networks trained by novel metaheuristic algorithms, etc.) also have potential for the task of interest.

Accordingly, the current work can be extended in multiple ways such as:

i) The investigation of other advanced ML methods such as deep learning regression (Zeng et al., 2022), hybrid ensemble learning (Cao et al., 2022), metaheuristic-trained ANN (Zhang et al., 2021) and ensemble deep neural networks (Barkhordari and Massone, 2022),

ii) The collection of more experimental datasets used for model validation.

iii) The applications of advanced feature selection for enhancing the model performance.

6. Supplementary Material

The datasets used to support the findings of this study are deposited in GitHub repository as follows: https://github.com/nhddtuedu/cs ml.

7. References

Al-Jamimi, H.A., Al-Kutti, W.A., Alwahaishi, S. and Alotaibi, K.S. (2022). "Prediction of compressive strength in plain and blended cement concretes using a hybrid artificial intelligence model", *Case Studies in Construction Materials*, 17, e01238, https://doi.org/10.1016/j.cscm.2022.e01238.

Al-Shamiri, A.K., Kim, J.H., Yuan, T.F. and Yoon,

Y.S. (2019). "Modeling the compressive strength of high-strength concrete: An extreme learning approach", *Construction and Building Materials*, 208, 204-219, https://doi.org/10.1016/j.conbuildmat.2019.02.165.

Barkhordari, M.S. and Massone, L.M. (2022). "Failure mode detection of reinforced concrete shear walls using ensemble deep Neural Networks, *International Journal of Concrete Structures and Materials"*, 16, 33, https://doi:10.1186/s40069-022-00522-y.

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

Bishop, CM. (2011). *Pattern recognition and machine learning (Information science and statistics)*, Springer, ISBN-10: 0387310738.

Breiman, L. (1993). "Hinging hyperplanes for regression, classification and function approximation", *IEEE Transactions on Information Theory*, 39, 999-1013, https://doi:10.110 9/18.256506.

Cao, M.T., Nguyen, N.M. and Wang, W.C. (2022). "Using an evolutionary heterogeneous ensemble of artificial neural network and multivariate adaptive regression splines to predict bearing capacity in axial piles, *Engineering Structures*, 268, 114769, https://doi.org/10.1016/j.engstruct.2022.114769.

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*, San Francisco, California, USA, 785-794, https://doi.org/10.1145/2939672.29385.

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, 7648467, https://doi.org/10.1155/2016/7648467.

Chung, K.L., Wang, L., Ghannam, M., Guan, M. and Luo, J. (2021). "Prediction of concrete compressive strength based on early-age effective conductivity measurement", *Journal of Building Engineering*, 35, 101998, https://doi.org/10 .1 016/j.jobe.2020.101998.

Drucker, H., Burges, CJC., Kaufman, L., Smola, A. and Vapnik, V. (1996). "Support vector regression machines", *Proceedings of the 9th International Conference on Neural Information Processing Systems*, Denver, Colorado.

Friedman, JH. (1991). "Multivariate adaptive regression splines", *The Annals of Statistics*, 19(1), 1-67, https://doi.org/10.1214/aos/1176347963.

Friedman, JH. (2001). "Greedy function approximation: A Gradient Boosting Machine", *The Annals of Statistics*, 29(5), 1189-1232.

Friedman, J.H. (2002). "Stochastic gradient boosting", *Computational Statistics and Data Analysis*, 38(4), 367-378, https://doi.org/10.1016/S0167-9473(01)00065-2.

Gesoglu, M., Güneyisi, E., Özturan, T. and Özbay, E. (2009). "Modeling the mechanical properties of rubberized concretes by neural network and genetic programming", *Materials and Structures*, 43, 31, https://doi:10.1617/s11527-009-9468-0.

Gomaa, E., Han, T., ElGawady, M., Huang, J. and Kumar, A. (2021). "Machine learning to predict properties of fresh and hardened alkali-activated concrete", *Cement and Concrete Composites*, 115, 103863, https://doi.org/10.1016/j.cemconcomp.2020.103863.

Han, T., Siddique, A., Khayat, K., Huang, J. and Kumar, A. (2020). "An ensemble machine learning approach for prediction and optimization of modulus of elasticity of recycled aggregate concrete", *Construction and Building Materials*, 244, 118271, https://doi.org/10.1016/j.conb uild mat.2020.118271.

Haykin, SO. (2008). *Neural Networks and Learning Machines*, Pearson, London, United Kingdom.

Hoang, N.D. (2019). "Estimating punching shear capacity of steel fibre reinforced concrete slabs using sequential piecewise multiple linear regression and Artificial Neural Network", *Measurement*, 137, 58-70, https://doi.org/10.1016/j.measurement.2019.01.035.

Hoang, N.D. (2022). "Machine learning-based estimation of the compressive strength of self-compacting concrete", *A Multi-Dataset Study Mathematics*, 10, 3771.

Hocine, A. (2018). "Compressive strength prediction of limestone filler concrete using artificial neural networks", *Advances in Computational Design*, 3(3), 289-302, https://doi.org/10.12989/acd.2018.3.3.289.

Jekabsons, G. (2016). ARESLab: *Adaptive regression splines toolbox for Matlab/Octave*, Technical Report, Riga Technical University, available at http://wwwcsrtulv/jekabsons/.

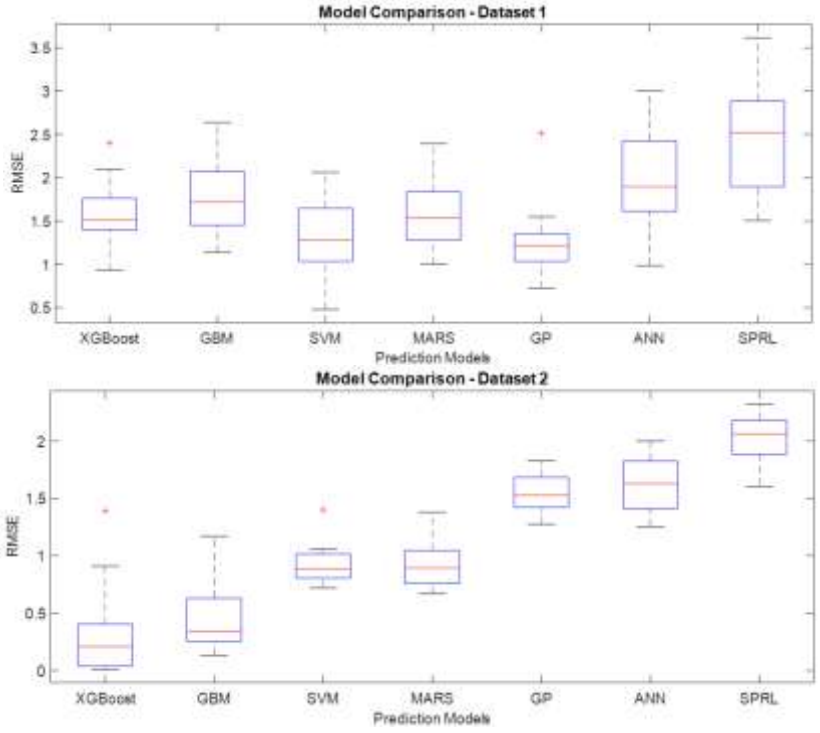
Kandiri, A., Mohammadi Golafshani, E. and Behnood, A. (2020). "Estimation of the compressive strength of concretes containing ground granulated blast furnace slag using hybridized multi-objective ANN and salp swarm algorithm", *Construction and Building Materials*, 248, 118676, https://doi.org/10.1016/j.conbuildmat.2020.118

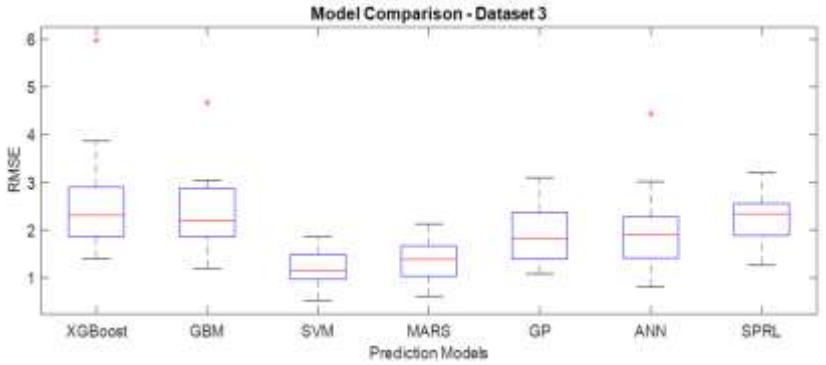
676.
Kang, M.C., Yoo, D.Y., Gupta, R. (2021). "Machine learning-based prediction for compressive and flexural strengths of steel fiber-reinforced concrete", *Construction and Building Materials*, 266, 121117. https://doi.org/10.1016/j.conbuildmat.2020.121117.
Khambra, G. and Shukla, P. (2021). "Novel machine learning applications on fly ash based concrete", *An Overview Materials Today: Proceedings*, https://doi.org/10.1016/j.matpr.2021.07.262.
Kingma, D.P. and Ba, J. (2015). "Adam: *A method for stochastic optimization*, arXiv:14126980 [csLG], https://doi.org/10.48550/arXiv.1412.6980.
Kovacevic, M., Lozancic, S., Nyarko, E.K. and Hadzima-Nyarko, M. (2022). "Application of artificial intelligence methods for predicting the compressive strength of self-compacting concrete with class f, Fly Ash", *Materials*, 15, 4191.
Koza, J.R. (1994). "Genetic programming as a means for programming computers by natural selection", *Statistics and Computing*, 4, 87-112, https://doi:10.1007/BF00175355.
Li, Q.F. and Song, Z.M. (2022). "High-performance concrete strength prediction based on ensemble learning", *Construction and Building Materials*, 324, 126694, https://doi.org/10.1016/j.conbuildmat.2022.126694.
Mendenhall, W. and Sincich, T.T. (2011). *A second course in statistics: Regression analysis*, 7th Edition, Pearson, ISSN 978-0321691699.
Mirrashid, M., Naderpour, H. (2020). "Recent trends in prediction of concrete elements behavior using soft computing (2010-2020)", *Archives of Computational Methods in Engineering*, *https*://doi:10.1007/s11831-020-09500-7.
Naser, A.H., Badr, A.H., Henedy, S.N., Ostrowski, K.A. and Imran, H. (2022). "Application of Multivariate Adaptive Regression Splines (MARS) approach in prediction of compressive strength of eco-friendly concrete", *Case Studies in Construction Materials*, 17, e01262, https://doi.org /10.101 6/j.cscm.2022.e01262.
Nguyen, H., Vu, T., Vo, T.P. and Thai, H.T. (2021). "Efficient machine learning models for prediction of concrete strengths", *Construction and Building Materials*, 266, 120950, https://doi.org/10.1016/j.conbuildmat.2020.120950.
Pala, M., Ozbay, E., Oztas, A. and Yuce, MI. (2007). Appraisal of long-term effects of fly ash and silica fume on compressive strength of concrete by neural networks", *Construction and Building Materials*, 21, 384-394, https://doi.org/10.1016/j.conbuildmat.2005.08.009.
Pedregosa, F. et al. (2011). "Scikit-learn: Machine learning in Python", *Journal of Machine Learning Research*, 12, 2825-2830.
Rathakrishnan, V.Bt., Beddu, S., Ahmed, A.N. (2022). "Predicting compressive strength of high-performance concrete with high volume ground granulated blast-furnace slag replacement using boosting machine learning algorithms", *Scientific Reports*, 12, 9539, https://doi:10.1038/s41598-02212890-2.
Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986). "Learning representations by back-propagating errors", *Nature*, 323, 533-536, https://doi:10.1038/323533a0.
Ryan, S.E. and Porth, L.S. (2007). "A tutorial on the piecewise regression approach applied to bedload transport data", General Technical Report RMRS-GTR-189, Fort Collins, CO: US Department of Agriculture, Forest Service, Rocky Mountain Research Station, 41p, https://doi.org/10.2737/RMRS-GTR-189.
Searson, D.P. (2015), GPTIPS 2: An open-source software platform for symbolic data mining", In: Gandomi, A.H., Alavi, A.H., Ryan, C. (Eds.), *Handbook of Genetic Programming Applications*, Springer International Publishing, Cham, 22 p., 551-573, https://doi:10.1007/978-3-319-20883-1.
Shahmansouri, A.A., Akbarzadeh Bengar, H. and Ghanbari, S. (2020), "Compressive strength prediction of eco-efficient GGBS-based geopolymer concrete using GEP method", *Journal of Building Engineering*, 31, 101326, https://doi.org/10.1016/j.jobe.2020.101326.
Tanyildizi, H. and Çevik, A. (2010). "Modeling mechanical performance of lightweight concrete containing silica fume exposed to high temperature using genetic programming", *Construction and Building Materials*, 24, 2612-2618, https://doi.org/10.1016/j.conbuildmat.201005.001.
Touzani, S., Granderson, J. and Fernandes, S. (2018). "Gradient boosting machine for modeling the energy consumption of commercial buildings", *Energy and Buildings*, 158, 1533-1543, https://doi.org/10.1016/j.enbuild.2017.11.039.
Ullah, HS., Khushnood, R.A., Ahmad, J. and Farooq, F. (2022). "Predictive modelling of sustainable lightweight foamed concrete using machine learning novel approach", *Journal of Building Engineering*, 56, 104746. https://doi.org/1 0.1016/j.jobe.2022.104746.
Videla, C. and Gaedicke, C. (2004). "Modeling portland blast-furnace slag cement high-performance concrete", ACI, *Materials Journal*, 101, 365-375.
Wong, T. and Yeh, P. (2020). "Reliable accuracy estimates from k-fold cross validation", *IEEE Transactions on Knowledge and Data Engineering*, 32, 1586-1594,

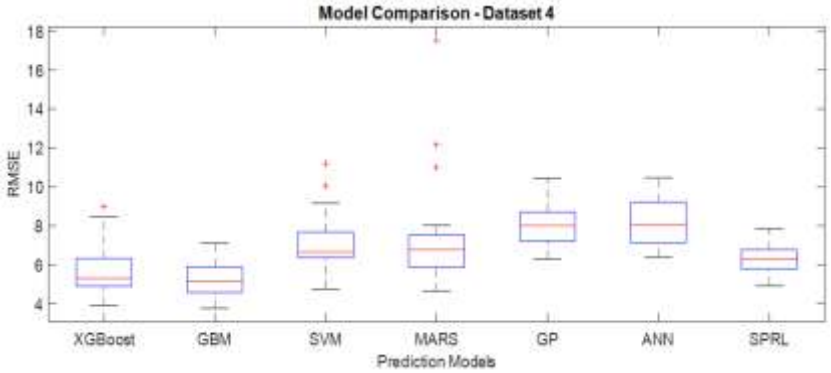
https://doi:10.1109/TKDE.2019.2912815.
XGBoost, (2021). "XGBoost documentation", https://xgboostreadthedocsio/en/stable/indexhtml" (Last access: 12/30/2021).
Yeh, I.C. (1998). Modeling concrete strength with Augment-Neuron Networks", *Journal of Materials in Civil Engineering*, 10, 263-268, https://doi:10.1061/(ASCE)08991561(1998)10:4(263).
Zeng, Z., Zhu, Z., Yao, W., Wang, Z., Wang, C., Wei, Y., Wei, Z. and Guan, X. (2022). "Accurate prediction of concrete compressive strength based on explainable features using deep learning", *Construction and Building Materials*, 329, 127082, https://doi.org/10.1016/j.conbuildmat.2022127082.
Zhang, X., Nguyen, H., Bui, X.N., Tran, Q.H., Nguyen, D.A., Bui, D.T. and Moayedi, H. (2019). "Novel soft computing model for predicting blast-induced ground vibration in open-pit mines based on Particle Swarm Optimization and XGBoost", *Natural Resources Research*, https;//doi:10.1007/s11053-019-9492-7.
Zhang, Y. and Aslani, F. (2021). "Compressive strength prediction models of lightweight aggregate concretes using ultrasonic pulse velocity", *Construction and Building Materials*, 292, 123419, https://doi.org/10.1016/j.conbuildmat.2021.123419.
Zhang, Y., Aslani, F. and Lehane, B. (2021). "Compressive strength of rubberized concrete: Regression and GA-BPNN approaches using ultrasonic pulse velocity", *Construction and Building Materials*, 307, 124951, https://doi.org/10.1016/j.conbuildmat.2021.124951.
Zhao, S., Hu, F., Ding, X., Zhao, M., Li, C. and Pei, S. (2017). "Dataset of tensile strength development of concrete with manufactured sand" Data in Brief, 11, 469-472, https://doi.org/10.1016/j.dib.2 017.02.043.
Zhao, Y., Hu, H., Song, C. and Wang, Z. (2022). "Predicting compressive strength of manufactured-sand concrete using conventional and metaheuristic-tuned Artificial Neural Network", *Measurement*, 194, 110993, https://doi.org/10.1016/j.measurement.2022.110993.

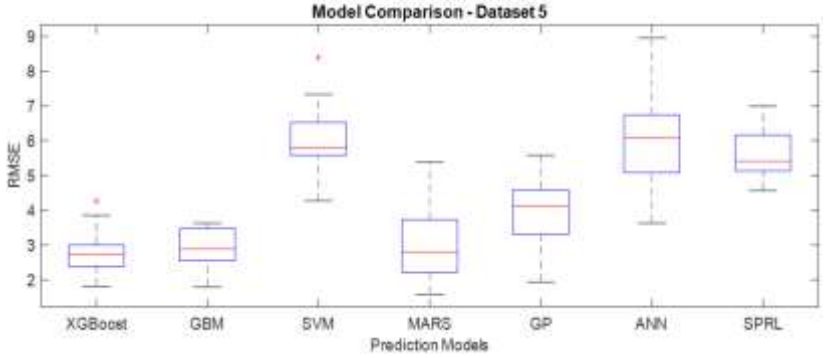
Appendix 1

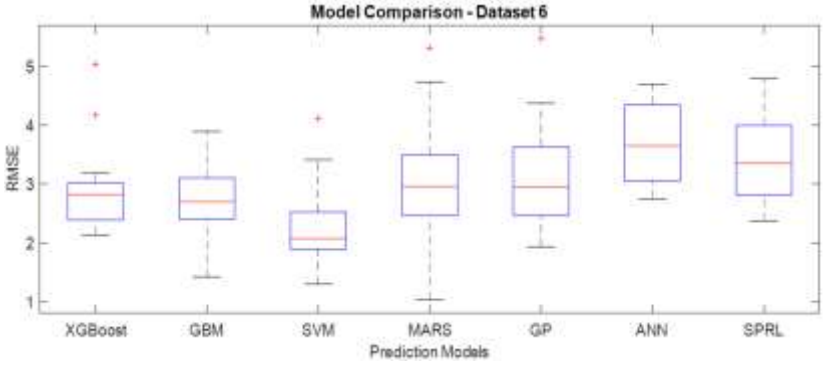
Boxplots of the Model Performance

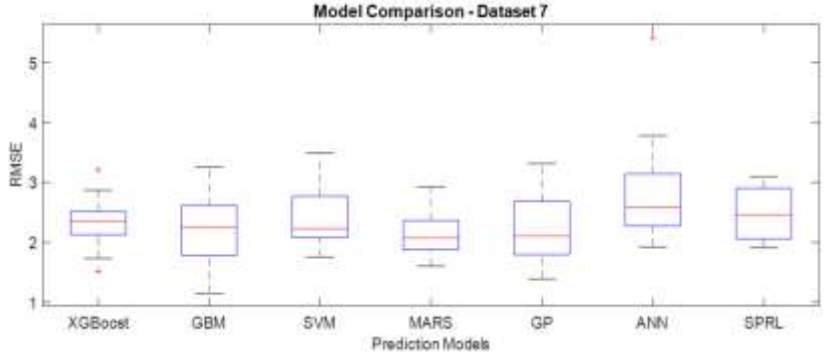



Model Comparison - Dataset 3
RMSE
XGBoost
GBM
SVM
MARS
GP
ANN
SPRL
Prediction Models


Model Comparison - Dataset 4
RMSE
XGBoost
GBM
SVM
MARS
GP
ANN
SPRL
Prediction Models


Model Comparison - Dataset 5
RMSE
XGBoost
GBM
SVM
MARS
GP
ANN
SPRL
Prediction Models


Model Comparison - Dataset 6
RMSE
XGBoost
GBM
SVM
MARS
GP
ANN
SPRL
Prediction Models


Model Comparison - Dataset 7
RMSE
XGBoost
GBM
SVM
MARS
GP
ANN
SPRL
Prediction Models

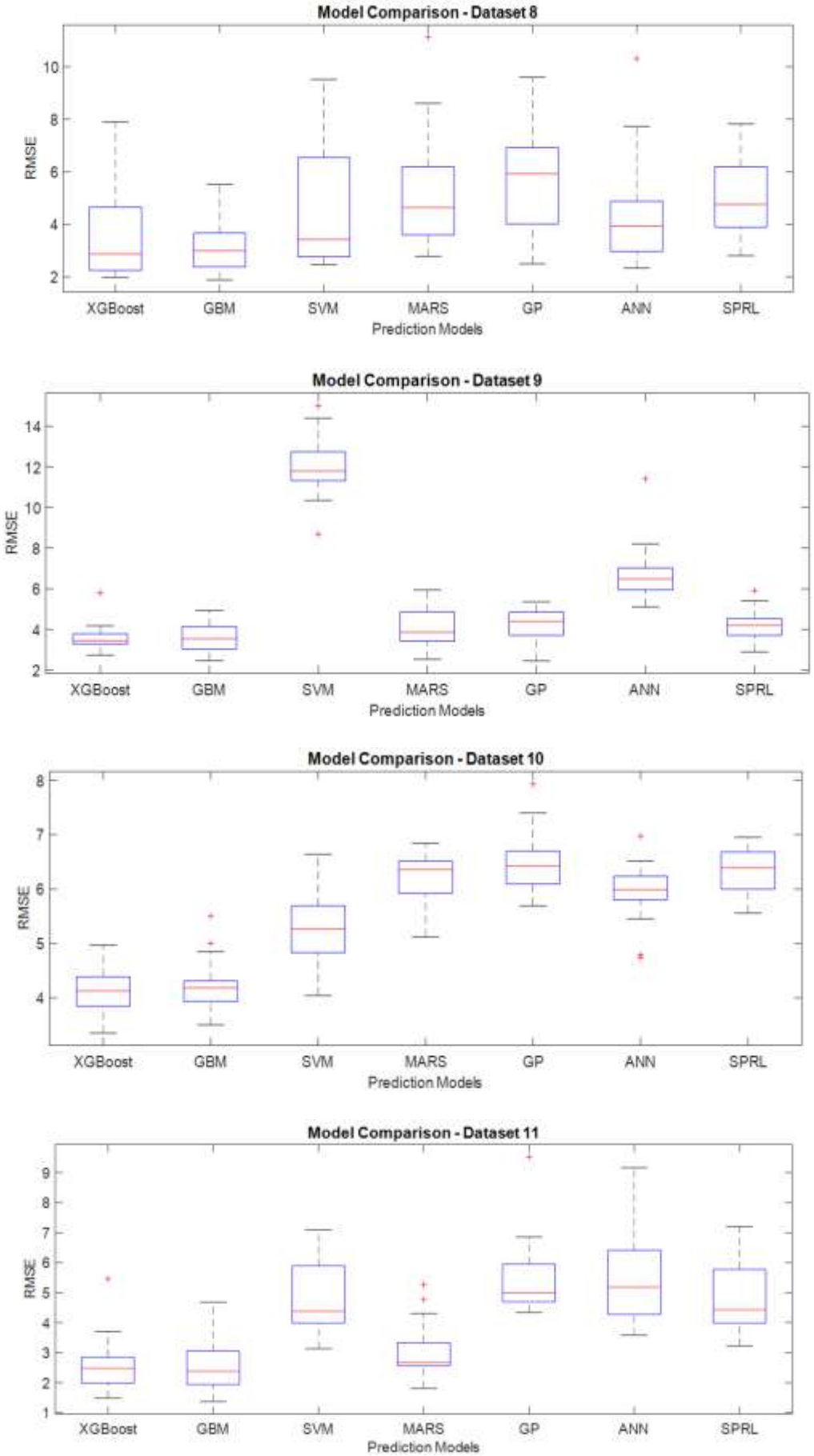


Fig. A1-1. Boxplots of the model performance

Appendix 2

Models' Ranking

Table A2-1. Ranking of the models

Dataset	Models	RMSE	Ranking
1	GP	1.24	1
	SVM	1.33	2
	XGBoost	1.57	3
	MARS	1.58	4
	GBM	1.78	5
	ANN	1.97	6
	SPRL	2.45	7
2	XGBoost	0.31	1
	GBM	0.47	2
	SVM	0.92	3
	MARS	0.93	4
	GP	1.55	5
	ANN	1.62	6
	SPRL	2.02	7
3	SVM	1.20	1
	MARS	1.36	2
	GP	1.92	3
	ANN	1.96	4
	SPRL	2.27	5
	GBM	2.34	6
	XGBoost	2.61	7
4	GBM	5.21	1
	XGBoost	5.74	2
	SPRL	6.35	3
	SVM	7.11	4
	MARS	7.52	5
	GP	8.09	6
	ANN	8.15	7
5	XGBoost	2.78	1
	GBM	2.91	2
	MARS	3.07	3
	GP	3.9	4
	SPRL	5.6	5
	ANN	6.01	6
	SVM	6.02	7