



OPEN

Soft computing models for prediction of bentonite plastic concrete strength

Waleed Bin Inqiad¹, Muhammad Faisal Javed^{2,3}✉, Kennedy Onyelowe^{4,5}✉, Muhammad Shahid Siddique¹, Usama Asif⁶, Loai Alkhatabi⁷ & Fahid Aslam⁸

Bentonite plastic concrete (BPC) is extensively used in the construction of water-tight structures like cut-off walls in dams, etc., because it offers high plasticity, improved workability, and homogeneity. Also, bentonite is added to concrete mixes for the adsorption of toxic metals. The modified design of BPC, as compared to normal concrete, requires a reliable tool to predict its strength. Thus, this study presents a novel attempt at the application of two innovative evolutionary techniques known as multi-expression programming (MEP) and gene expression programming (GEP) and a boosting-based algorithm known as AdaBoost to predict the 28-day compressive strength (f'_c) of BPC based on its mixture composition. The MEP and GEP algorithms expressed their outputs in the form of an empirical equation, while AdaBoost failed to do so. The algorithms were trained using a dataset of 246 points gathered from published literature having six important input factors for predicting. The developed models were subject to error evaluation, and the results revealed that all algorithms satisfied the suggested criteria and had a correlation coefficient (R) greater than 0.9 for both the training and testing phases. However, AdaBoost surpassed both MEP and GEP in terms of accuracy and demonstrated a lower testing RMSE of 1.66 compared to 2.02 for MEP and 2.38 for GEP. Similarly, the objective function value for AdaBoost was 0.10 compared to 0.176 for GEP and 0.16 for MEP, which indicated the overall good performance of AdaBoost compared to the two evolutionary techniques. Also, Shapley additive analysis was done on the AdaBoost model to gain further insights into the prediction process, which revealed that cement, coarse aggregate, and fine aggregate are the most important factors in predicting the strength of BPC. Moreover, an interactive graphical user interface (GUI) has been developed to be practically utilized in the civil engineering industry for prediction of BPC strength.

Keywords Plastic concrete, Genetic programming, Bentonite, Compressive strength, AdaBoost, Shapley additive explanation

Abbreviations

BPC	Bentonite plastic concrete
f'_c	28-Day compressive strength
MEP	Multi expression programming
GEP	Gene expression programming
R	Coefficient of correlation
RMSE	Root mean squared error
OF	Objective function
ML	Machine learning

¹Military College of Engineering (MCE), National University of Science and Technology (NUST), Islamabad 44000, Pakistan. ²Department of Civil Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan. ³Western Caspian University, Baku, Azerbaijan. ⁴Department of Civil Engineering, Michael Okpara University of Agriculture, Umudike 440109, Nigeria. ⁵Department of Civil Engineering, Kampala International University, Kampala, Uganda. ⁶Department of Civil and Environmental Engineering, School of Engineering and Digital Sciences, Nazarbayev University, Nur-Sultan 010000, Kazakhstan. ⁷Department of Civil and Environmental Engineering, College of Engineering, University of Jeddah, Jeddah 23890, Saudi Arabia. ⁸Department of Civil Engineering, College of Engineering in Al-Kharj, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia. ✉email: arbabfaisal@giki.edu.pk; kennedychibuzor@kiu.ac.ug

ANN	Artificial neural network
AI	Artificial intelligence
XGB	Extreme gradient boosting
GP	Genetic programming
ET	Expression tree
DT	Decision tree
R ²	Coefficient of determination
PI	Performance index
MAE	Mean absolute error
RRMSE	Relative root mean squared error
SHAP	Shapely additive explanatory analysis
RF	Random forest
SVM	Support vector machine
DL	Deep learning
RHA	Rice husk ash
GUI	Graphical user interface

Metal plating and mineral extraction industries cause water pollution by releasing huge quantities of water contaminated with toxic metals such as Pb (II), Fe (III) and Cr (III). These metals do not disintegrate easily and are accumulated by the organisms drinking contaminated water or products of the contaminated water like crops etc¹. The most widely used method of mitigating water pollution is adsorption². In recent years, the adsorption of hazardous water by using clays has been increased because of the advantages of clays as compared to other adsorbing materials such as high adsorption, good accessibility, lesser cost, higher surface area, and higher capacity for ionic exchange³. The adsorption of heavy metals using bentonite has also been investigated by many researchers^{4–6}. Bentonite is a naturally occurring clay mineral which has found numerous applications in concrete industry due to its special properties. The incorporation of bentonite in concrete results in enhancement of several key properties of concrete and elimination of toxic metals⁷. The mixture of conventional concrete and bentonite is called bentonite plastic concrete (BPC), and it is widely used to construct cut-off walls for preventing seepage under dams^{8–10}. The swift prediction of compressive strength of BPC is important to reduce the time and cost required by concrete testing procedures in constructing dams with BPC due to the modified mix design of BPC as compared to normal concrete. Thus, having accurate value of 28-day compressive strength (f_c') of BPC is important for making key decisions in construction of large dam projects^{11–14}.

During the construction of dams and other water-tight structures, two important factors to consider include water tightness and seepage control. The construction of cut-off walls is one of the most commonly used methods to prevent seepage from the dams. However, due to the rigid diaphragm of cut-off walls, a slight deformation in earth embankment can lead to the failure of cut-off wall. Also, cracks can develop in cut-off walls due to the deformation of embankment caused by seismic activity or fluctuation in impounded reservoir levels^{15–17}. This issue can be resolved by using plastic concrete whose deformation characteristics are similar to as of embankment soils^{18–20}. However, this plastic concrete must be water-tight to allow the soil and wall to deform without separation. This plastic concrete has higher formability and lower strength in contrast to normal concrete due to use of clay in concrete²¹. The mixture composition of BPC generally includes aggregate (coarse and fine), cement, water, bentonite, and clay formed into a mixture of higher water-to-cement ratio than conventional concrete to result into a ductile material²¹. The researchers have also suggested using a mixture of bentonite and kaolinite for constructing impermeable clay coatings at places where bentonite isn't readily available. The study conducted by Karunaratne et al.²² reveals that the consolidating and hydraulic conductivity properties of concrete made with equal amounts of bentonite and kaolinite are almost the same as of concrete made with 100% bentonite. Thus, it shows that bentonite is a versatile material that aids in development of plastic concrete to be used in water-tight structures such as dams and also for adsorbing contaminated water. It is worth mentioning that 28-day compressive strength is an important parameter used for quality control of concrete. In case of BPC, there are various factors which can affect f_c' like chemical properties of concrete constituents, water-to-cement ratio, and curing time etc. During the use of BPC, the samples taken from various mixtures must be rigorously tested for f_c' and other important parameters. However, it is not possible to cast, store, cure, and test a large number of concrete samples of BPC at construction site due to resource, time, and effort limitations^{23–25}. Therefore, a tool for accurate and fast prediction of BPC strength is needed to overcome the above-mentioned limitations and foster the widespread use of BPC for water-tight structures and also for adsorption of toxic metal-contaminated water. Thus, this study aims to develop prediction models that makes use of concrete mixture compositions to estimate f_c' of BPC accurately.

In recent years, machine learning (ML) techniques have captured the attention of researchers in different fields to predict different properties^{26–30}. Initially, the use of artificial neural networks (ANNs) was preferred due to their impressive accuracy and because they eliminated the need for a pre-determined mathematical model^{31–33}. However, with the passage of time, other ML models such as random forest (RF), support vector machine (SVM), and different boosting algorithms were developed to further increase the accuracy of ML models^{34–36}. However, the evolutionary algorithm techniques such as GEP and MEP are advantageous in the sense that they represent their output in the form of an empirical equation thus giving more transparency and insight into the prediction process³⁷. The other ML algorithms don't have this ability and thus are called black-box models³⁸. Also, GEP and MEP does not need optimization of model architecture like ANN etc., thus reducing the time and computing power required to make predictions³⁹. Table 1 summarizes the main advantages of MEP and GEP algorithms.

Gene expression programming (GEP)	Multi expression programming (MEP)
Uses linear chromosomes for encoding expression trees	Uses linear chromosomes with multiple expressions
High flexibility in evolving complex structures	High flexibility due to multiple expressions
Translates chromosomes into expression trees for evaluation	Evaluates multiple expressions from a single chromosome
Efficient in discovering optimal or near-optimal solutions	Efficient in discovering optimal or near-optimal solutions
Can generate diverse solutions through genetic diversity	Naturally promotes solution diversity with multiple expressions
Good convergence properties due to the efficient search space	Good convergence by evaluating multiple solutions simultaneously
Capable of handling complex symbolic regression problems	Handles complex problems with multiple potential solutions
Scales well with problem complexity	Scales well due to the compact representation
Supports parallel evaluation of individuals	Inherently supports parallel evaluation of multiple expressions
Allows for problem-specific tuning of genetic operators	Provides more options for problem-specific tuning due to multiple expressions

Table 1. Advantages of MEP and GEP algorithms.

Previous studies regarding concrete strength prediction using ML techniques

It has been observed that during past few years, the subject of predicting strength and other properties of different types of concrete has been popular among researchers. To discuss a few studies, Amin et al.⁴⁰ used different bagging algorithms such as AdaBoost and boosting algorithms like RF to predict strength of concrete containing rice husk ash (RHA) in place of cement. The authors utilized a dataset for RHA-modified concrete from literature and used it to predict strength of concrete. The developed models were checked for their accuracy by using various error metrics such as average error, correlation coefficient and root mean squared error. The study concluded that RF is the most accurate algorithm which predicted strength values having more than 92% accuracy. In the same way, Li et al.⁴¹ predicted splitting tensile strength of concrete containing metakaolin using ANN, SVR, RF and gradient boosting decision tree (GBDT). The authors compared the efficiency of algorithms and revealed that GBDT proved to be the most accurate out of all the algorithms having a mere average error of 0.14. Similarly, Nazar et al.⁴² employed ANN and GEP for the purpose of predicting strength and slump of geopolymer concrete made with fly ash. The authors also conducted sensitivity analysis on the GEP model since it proved to be more accurate than ANN based on the evaluation using different error metrics. Also, Mahmood et al.⁴³ utilized a laboratory dataset of 133 points for Self-compacting concrete made by replacing cement with marble powder and RHA. The authors utilized variety of algorithms including SVM, RF, decision tree, gradient boosting etc. along with different deep learning techniques to predict compressive strength. The authors measured the relative accuracy of algorithms by means of mean absolute percentage error and average error and concluded that deep learning techniques predicted strength with greater accuracy as compared to other ML algorithms with higher accuracy and lesser errors. Iqbal et al.⁴⁴ applied MEP to predict split tensile strength and modulus of elasticity of concrete which contained waste foundry sand in addition to natural sand. The authors highlight an important advantage of using MEP that it presented its output in the form of empirical equations which can be effectively utilized by professionals and researchers for calculating modulus of elasticity and split tensile strength of sustainable concrete containing waste foundry sand as a replacement of natural sand. The authors also conducted a parametric analysis to determine the relationship between the input variables and the two output variables which provided further insights into the MEP model and enhanced its effectiveness. Furthermore, Shahab et al.⁴⁵ predicted electrical resistivity and compressive strength of concrete modified with graphene nanoplatelets. The authors used GEP, SVM and RF for this purpose and also developed a graphical user interface that can be used to predict strength of graphene nanoplatelets-based concrete efficiently. As evident from above discussion, significant work has been done on the subject of predicting strength properties of different types of concrete using various ML techniques but there is still a lack of work focusing on strength prediction of concrete containing clay and bentonite. Thus, this research aims to fill the gap in the literature by employing two evolutionary techniques i.e., MEP and GEP and AdaBoost to predict f_c' of BPC.

Overview of machine learning in civil engineering

Artificial intelligence (AI) in realm of civil engineering chiefly refers to the use of advanced algorithms and ML techniques for promoting efficient infrastructure design, optimizing construction processes, and increasing overall efficiency and safety of construction projects^{46–50}. ML is a subset of AI by virtue of which machines can learn from the data without explicit human intervention. ML has different types including supervised, unsupervised and reinforcement learning. ANNs are commonly used tool in deep learning (DL), which in turn is a subset of ML as shown in (Fig. 1). ML and DL differ from each other in a way that DL automatically extract features and does classification while in case of ML, the user must identify features on which the algorithm performs classification and calculates predictions. Machine learning essentially refers to the kind of AI in which computers can recognize patterns in the data and use the information from data to make accurate predictions without requiring extensive programming. It has been extensively utilized in civil engineering for structural analysis, materials optimization, construction management and maintenance etc. In particular, different evolutionary techniques, random forests, decision trees, neural networks etc. have been used to predict different properties of concrete and cement composites^{51–55}, soil liquefaction and compaction parameters^{56–59}, and slope stability^{60–63}.

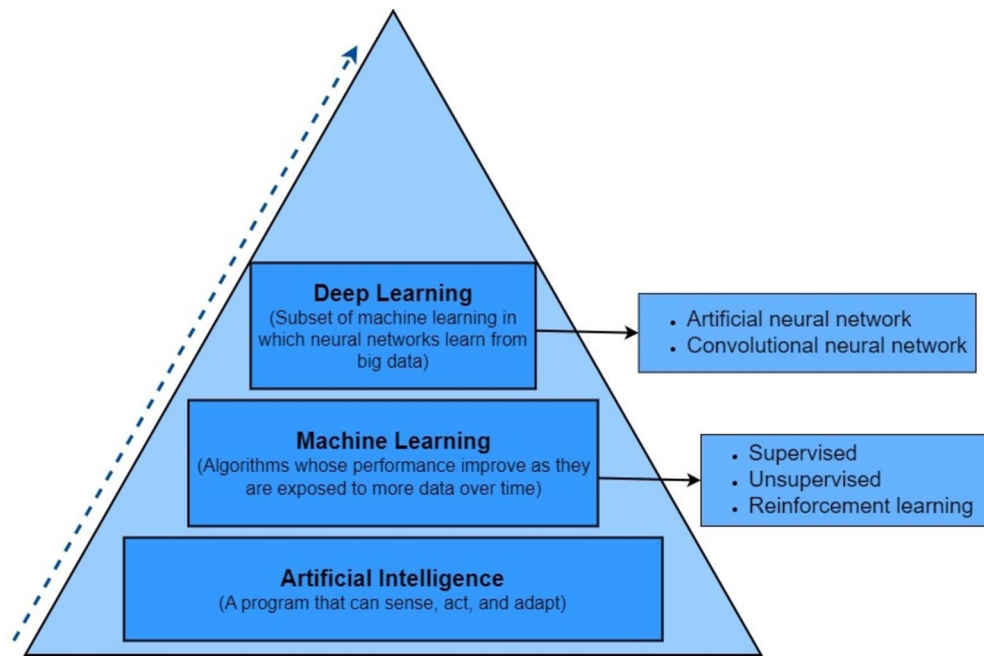


Figure 1. Artificial intelligence (AI) and its subtypes.

Research significance

It is evident from overview of relevant literature that there have been some studies regarding prediction of concrete containing various admixtures like RHA, metakaolin, marble powder, and fly ash etc. However, the subject of prediction of mechanical properties particularly compressive strength is relatively unexplored. Thus, this study is conducted as an attempt to fill this gap in the literature by utilizing evolutionary programming techniques like GEP and MEP and a boosting-based technique known as AdaBoost. The motivation behind using MEP and GEP along with AdaBoost for predicting f_c' of BPC stems from the fact that they are widely regarded as grey-box models in contrast to black-box nature of ANN and AdaBoost etc. The MEP and GEP algorithms have the ability to depict the output as an empirical expression. This expression can be seamlessly used by researchers across the globe to calculate the desired value. It also provides transparency to the prediction process and fosters the widespread use of developed models³⁸. Also, the selection of AdaBoost in place of other techniques like SVM and ANNs is due to its increased accuracy, simple algorithm architecture, and efficient implementation⁶⁴. Moreover, this study is novel in the sense that it compares two evolutionary techniques MEP and GEP with one of the most famous boosting techniques i.e., AdaBoost. Furthermore, to get increased insights into the importance of different variables in predicting f_c' of BPC, this study utilizes shapley explanatory analyses on the model of highest accuracy.

Methodology

This study is attributed to predicting f_c' of BPC using three soft computing models including GEP, AdaBoost and MEP. The overall methodology followed in the study is given in (Fig. 2). After the identification of the problem, the next step was to collect data from the literature to be used for building the models. A comprehensive literature search was performed, and 246 data points were collected to be used for training the algorithms. Different descriptive statistical analysis techniques were also performed on the data to get useful information about the database on which the models will be constructed. Also, according to the suggestions of previous studies, different statistical evaluation metrics were used to check the performance of developed models. The actual model development process involved finding the appropriate set of hyperparameters which gave maximum accuracy. Furthermore, a comparison was drawn between the developed models and the model proving to be most accurate was selected for further explanatory analysis to gain insight into the model prediction process. This whole process is explained step by step in (Fig. 2).

Prediction models

Out of the three ML models employed in current research, two algorithms (MEP and GEP) are subtypes of genetic programming (GP) which is based on the use of evolutionary algorithms. The idea of genetic programming GP was given by John Koza for the first time⁶⁵ by integrating the concept of genetics with natural selection⁶⁶. GP is a useful instrument for mathematical modelling due to its parse-trees which are non-linear, rather than binary strings of a fixed length. GP utilizes Darwin's theory of evolution to solve problems using processes such as mutation, crossover, and reproduction in the same way these processes function inside the genes of human beings⁶⁵. These processes favour the formation of individuals with good results and thus helps in convergence on the algorithm⁶⁷. Both MEP and GEP are subtypes of the same parent technique i.e., GP. It uses evolutionary

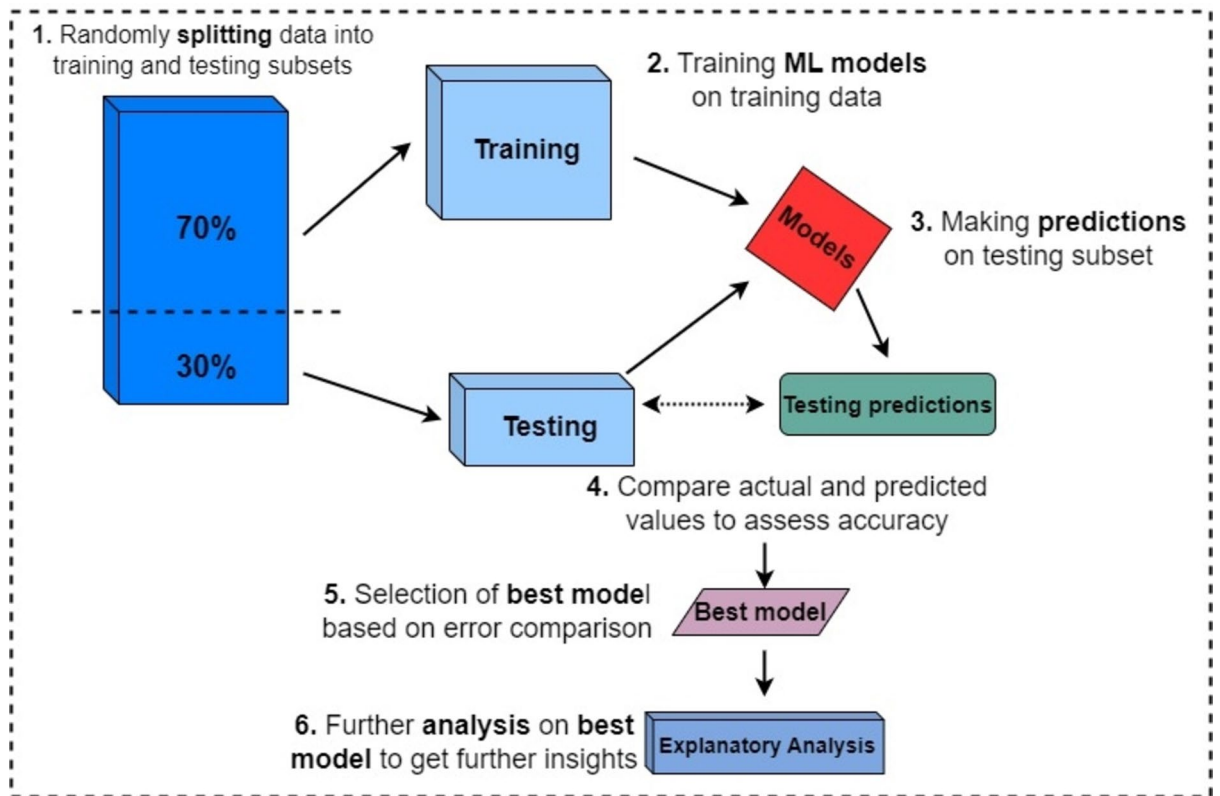


Figure 2. Overall methodology followed in the study.

algorithms to create and modify computer programs to solve a problem. The basis of GP lies in Darwin's theory of natural selection and uses an iterative process to generate increasingly complex programs to improve the accuracy^{68–70}. The basic process of GP involves creating random programs according to pre-defined rules and parameters. These programs are evaluated using fitness function and the accurate programs are used to create more programs. This process repeats several times before reaching at the most accurate and efficient program as the solution to the problem⁷¹. GP has the advantage that it can discover solutions to the problems which are too difficult to solve for human brain. This ability of GP makes it a useful tool for solving problems that require large computing power. GP has many variants having different ways to represent the individuals in the population, but the basic working principle is same for all variants. This study utilizes the two most common and most accurate variants of GP named as MEP and GEP for modelling strength of bentonite plastic concrete.

Gene expression programming (GEP)

GEP was proposed as an alternative algorithm to GP by Ferreira⁶⁶ which is based on the evolutionary population theorem. It also utilizes parse trees and fixed-length chromosomes. However, GEP considers fixed length parse trees in contrast to GP which considers parse trees of variable length. It is the same as separating genotype from phenotype⁶⁶. GEP eliminates the possibility of modifying the whole structure since all mutations are taking place inside a single linear framework. It is because GEP only allows to pass the genotype to the next generations. Also, in the populations made by the GEP algorithm, only one chromosome is responsible for creating populations with many genes.

The overall methodology followed by the GEP algorithm is illustrated in (Fig. 4). The basic methodology of GEP revolves around using genes which represent a smaller segment of the code. These genes are built using mathematical functions known as primitive functions. These primitive functions can range from simple arithmetic functions like addition and subtraction etc. to complex functions like trigonometric and logarithmic functions. Various evolutionary processes such as mutation and recombination⁷² are used to combine these genes as illustrated in (Fig. 3). The algorithm starts by generating a population of chromosomes and calculates the fitness of these chromosomes on the basis of a pre-determined fitness function. The chromosomes which exhibit good accuracy based on the fitness function are selected to be included in the next generation while the worst performing ones are discarded from the population^{73–75}. This whole process of generating chromosomes, evaluating their fitness, and using them to create next generation is repeated for many times which enables the algorithm to converge towards the actual solution⁷⁶.

Multi expression programming (MEP)

MEP is a subtype of GP that uses linear chromosomes to make predictions. The basic working principle of MEP is similar to GEP. However, it can encode multiple results in a single chromosome which makes it more accurate

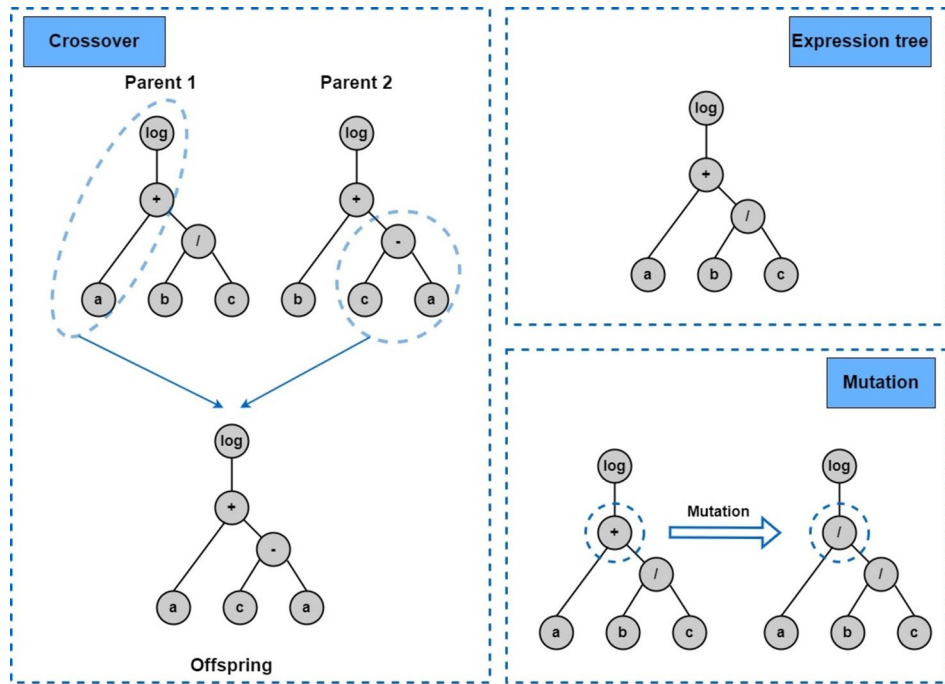


Figure 3. Visual representation of expression tree and genetic processes.

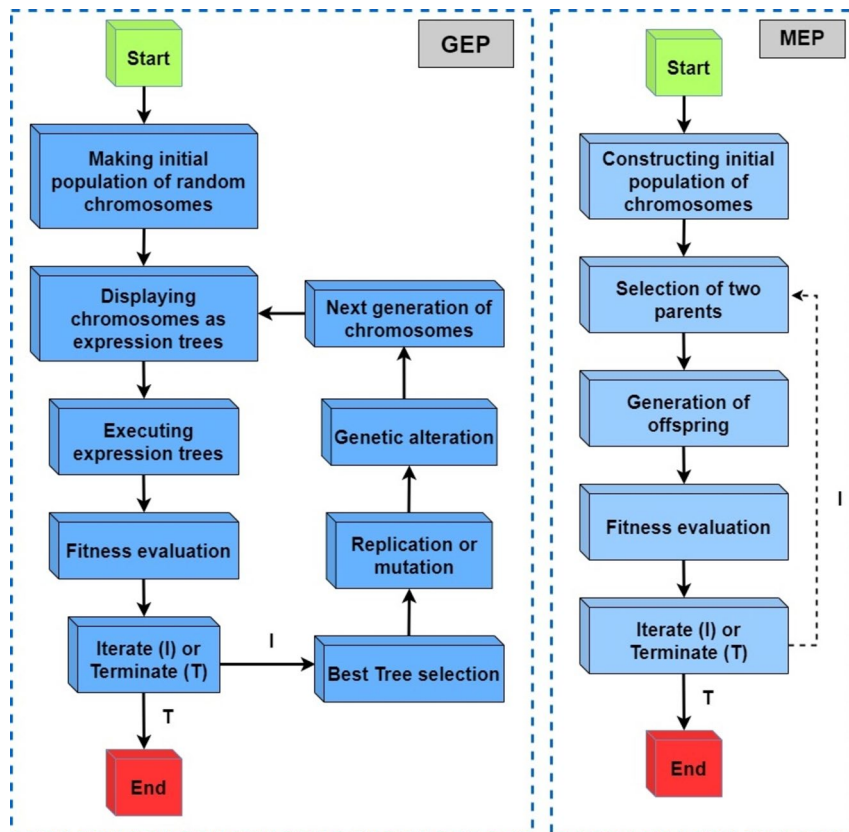


Figure 4. Methodology followed by the genetic programming techniques.

than other variants of GP⁷⁷. The final answer of MEP is derived from the chromosome which performs best according to the fitness function. A binary tournament procedure is used to select two parents and these parents are used to generate an offspring. The offspring is altered using genetic processes and this process repeats over several generations until the most accurate representation of the given problem is discovered as illustrated by (Fig. 4). The prediction process by MEP starts by creating a population of expressions. These expressions are displayed as expressions trees (as illustrated in Fig. 3). The generated expressions are classified according to their ability to solve the problem accurately by means of a fitness function. The expressions having good accuracy are used to generate another population of expressions while others are discarded from the population. The genetic processes of mutation and crossover are used on existing expressions to create new expressions. The visual representation of both these processes is given in (Fig. 3). The process of mutation involves altering the structure of an existing expression while crossover means generating a whole new expression from two parent expressions⁷⁸.

AdaBoost regression

In 1990, Schapire argued that a single DT has limited capabilities and thus is a weak learner. He combined several weak learners in series and made a strong learner which laid the groundwork for development of boosting algorithm⁷⁹. In AdaBoost algorithm, every time the algorithm adds a new tree, the general tree is eliminated and only the strongest tree is added. The repetition of this process over many iterations led to the improvement of algorithmic accuracy⁸⁰. This algorithm involves a simple weak regression algorithm improvement process, which improves its accuracy by virtue of continuous training. The first weak learner is obtained by learning from the training data⁸¹. However, after the creation of first basic tree, some data samples are correctly predicted while some are not. So, the wrong training samples are combined with the untrained data to make a new training sample and the second tree is obtained by training on this newly constructed sample. The wrong samples now are combined again with the untrained data to form yet another training sample, and it is trained to obtain the third weak learner tree. This process is repeated many times until an accurate learner is obtained. The algorithm also gives different weight to the samples to increase the accuracy of learners⁸². The accurate learners are given less weights while less accurate learners are given more weight so that the algorithm can focus more on the less accurate learners to make them accurate⁸³. The overall process of AdaBoost algorithm is given in (Fig. 5). It is important to mention that while the training process of basic tree model is carried out, the weight distribution of each sample present in the dataset must be adjusted. The training data changes, and hence the training results will also vary, which are summed to get the final result.

Data collection and analysis

Collection of a reliable database to be used for training the algorithms is one of the most important steps in building data-driven models⁸⁴. Thus, an extensive dataset consisting of 246 instances for BPC strength has been collected from internationally published literature⁸⁵ given in appendix. The collected dataset has six input parameters that have been identified crucial for predicting f_c of bentonite plastic concrete. These input parameters include cement and water content, quantities of coarse and fine aggregates, bentonite content, and clay content. These six input parameters will be utilized towards the prediction of single output i.e., compressive strength. The input parameters used in this study are measured in kilograms while the output parameter is measured in megapascals (MPa).

Data description and splitting

It has been rigorously suggested that the dataset to be used for model development should be divided into two or three sets and the algorithms should be trained on one set while the other dataset should be used for checking

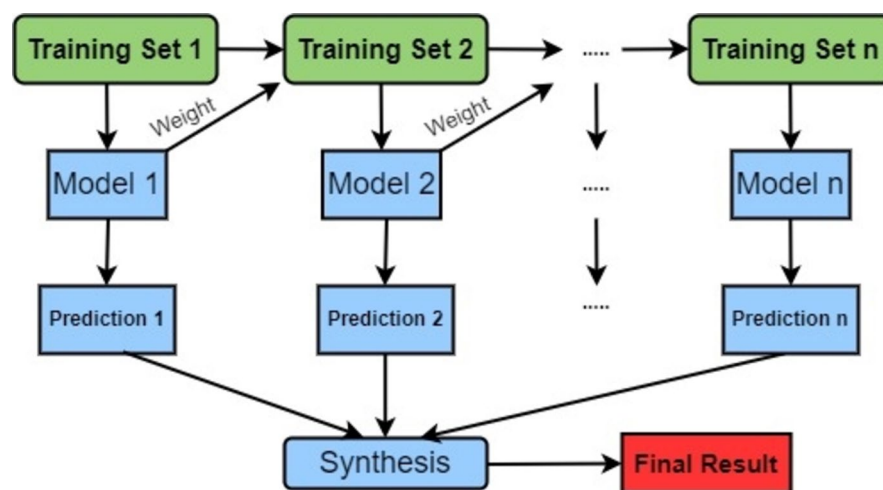


Figure 5. Methodology followed by AdaBoost regression.

accuracy of the developed algorithms. Thus, the collected data to be used in this study has been divided into training set containing 172 points (70%) and testing set having rest of the 74 points (30%). The ML models were trained using training data then they will be used to make predictions on the unseen test data to make sure that they can perform well when tested against unseen data and are not overfitted to the training data. Moreover, the summary of statistical analysis of the collected data is given in (Table 2). It can be seen from Table 2 that the input and output variables are spread across a wide range and this type of data distribution is imperative for developing widely applicable models.

Data correlation

It is useful to check the interdependence of selected variables before starting the model development process. This is due to the fact that if the input variables are highly correlated with each other, it might cause a problem during algorithm development known as multi-collinearity⁸⁶. The interdependence between variables employed in the study can be checked by using a statistical analysis tool known as correlation matrix. It quantifies the regression between different variables in the form of coefficient of correlation (R) and thus can be used to check the effect of explanatory variables on each other. The extent of dependence of one variable on another can be investigated by looking at the value of R between them. This R value can be positive or negative indicating positive and negative correlation between two variables respectively. Generally, R value greater than 0.8 between two variables signifies the presence of a good correlation between them⁸⁷. Notice from the correlation matrix developed for the data used in current study given in (Fig. 6) that the correlation values between different variables is less than 0.8 for most of the cases. It is an indication that the risk of multi-collinearity will not arise during model development process.

	Cement	Coarse aggregate	Fine aggregate	Water	Bentonite	Clay	Compressive strength
Units	kg	kg	kg	kg	kg	kg	MPa
Symbols	x_0	x_1	x_2	x_3	x_4	x_5	f_c
Maximum	400	1179.1	1305	500	320	380	29.4
Minimum	50	295	524	118.1	5	0	0.80
Mean	148.77	625.23	895.31	326.2	87.63	143.43	7.03
Standard deviation	63.43	203.87	247.54	89.86	55.63	95.87	6.403
25%	108.1	452.08	695.40	295.7	49.80	45.03	2.485
50%	134	665.55	838.41	341.2	79.26	182.62	4.33
75%	170.03	775	1116.29	390.9	112	220.47	9.81

Table 2. Descriptive analysis of the dataset.

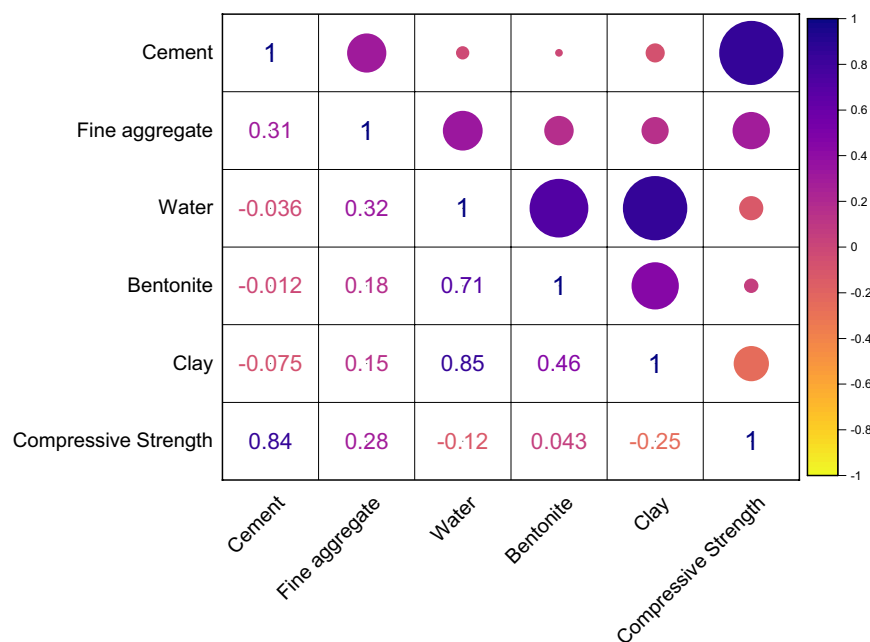


Figure 6. Correlation matrix of variables.

Data distribution

Another important factor to consider while dealing with data-driven models is the distribution of the data on which the models will be based. The performance of a ML model depends on distribution of the data on which it has been trained⁸⁸. Thus, the distribution of the data used in current study is shown by contour plots given in (Fig. 7). The frequency distribution density of variables is also shown by density plots shown above and on the right side of the contour plots. The thorough distribution of data is evident from the contour and distribution density plots. These plots provide useful information about the range of each variable involved in the ML model and this type of wide distribution is important for development of robust and accurate models⁸⁹.

Performance assessment

The performance of ML models should be checked to make sure that the developed models can accurately solve the problem⁹⁰. Thus, several error metrics suggested by researchers were used for performance assessment of models developed in the current study. These metrics include MAE, RMSE and objective function etc. The metrics MAE and RMSE are the most widely used and are crucial for checking the performance of any ML model⁹¹. These two metrics should be kept as close to zero as possible because MAE gives the absolute average difference between experimental and ML predicted values while RMSE is used as an indicator of larger errors. This is because residuals are squared before taking average in RMSE thus giving more weight to larger errors. Also, coefficient of correlation (R) is a commonly used metric for evaluating general accuracy of models. However, it is suggested to never use R as a sole indicator of model's effectiveness because it is insensitive towards the division or multiplication of output with a constant⁹². Generally, the correlation value greater than 0.8 is considered satisfactory for a model with good performance⁹⁰. Similarly, performance index (PI) and objective function (OF) are commonly used metrics for evaluating model's overall performance. Performance index makes use of relative root mean square error (RRMSE) and correlation coefficient while objective function integrates the relative number of data points in addition to RRMSE and R to predict model's accuracy. The value of PI and OF being less than 0.2 indicates that the model has superior overall performance⁹³. Moreover, a20-index is another recently developed metric which gives the proportion of predictions deviating more than $\pm 20\%$ from the actual values⁹⁴. The summary of the error evaluation metrics used in current study along with the suggested criteria is given in (Table 3).

Results and discussion

Model development

MEP model development

The MEP algorithm was employed with the help of a software developed specifically for this purpose known as MEPX 2021.05.18.0. Before starting the training of the algorithm, there are various parameters that need to

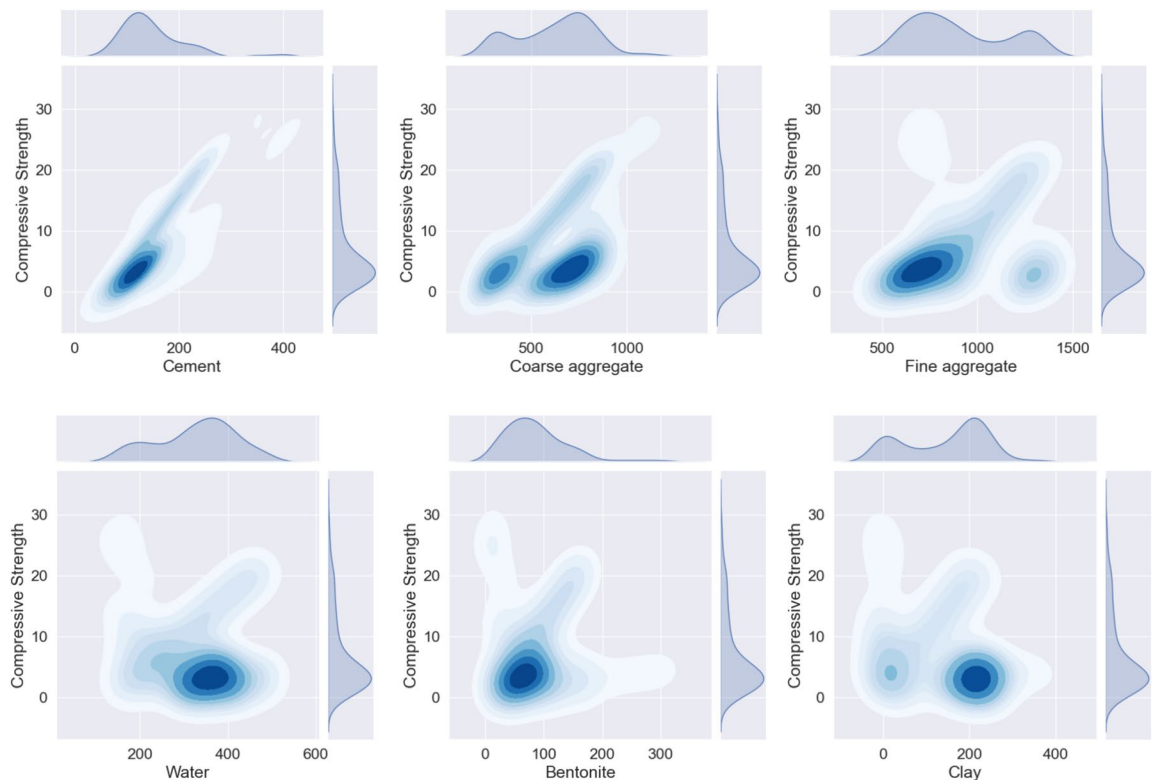


Figure 7. Contour plots of input variables.

No	Metric	Abbreviation	Formula	Range	Suggested value
1	Mean absolute error	MAE	$\frac{\sum x - y }{n}$	0 to $+\infty$	Close to zero
2	Root mean square error	RMSE	$\sqrt{\frac{\sum (x - y)^2}{n}}$	0 to $+\infty$	Close to zero
3	Coefficient of determination	R	$\frac{(n \sum y - (\sum x)(\sum y))}{\sqrt{(n \sum x^2 - (\sum x)^2)(n \sum y^2 - (\sum y)^2)}}$	0 to 1	R > 0.8
4	Performance index	PI	$\frac{RRMSE}{1+R}$	0 to $+\infty$	PI < 0.2
5	Objective Function	OF	$\left(\frac{n_{\text{Training}} - n_{\text{Testing}}}{n}\right) PI_{\text{Training}} + 2 \left(\frac{n_{\text{Testing}}}{n}\right) PI_{\text{Testing}}$	0 to $+\infty$	OF < 0.2
6	a20-index	a20	$\frac{n20}{n}$	0 to 1	Close to one

Table 3. Summary of error evaluation metrics. x = experimental value, y = model predicted value, n = total number of data samples, n20 = data samples having x/y ratio between 0.80 and 1.20.

be tuned. These parameters should be chosen carefully because they affect the convergence of the algorithm towards the result. These parameters were chosen using a trial-and-error approach and also with the help of recommendations from previous studies^{95–97}. The important MEP fitting parameters include subpopulation size and number of subpopulations. The algorithm was started using 10 as subpopulation size and it was varied until the model with highest accuracy was reached at a value of 100. Similar was the case with number of subpopulations. Increase in these two parameters generally increases the accuracy of the model however it also makes the resulting equation complex and increases the computing power of the model. Also, an abnormally high value of these parameters can also lead to overfitting of the model⁹⁸. Thus, these parameters should be carefully chosen. Code length and crossover probability are another two important parameters. Code length is directly related to the length and complexity of the resulting MEP equation while crossover probability indicates the probability of crossover of two parents to produce an offspring⁹⁹. In the current study, crossover probability is chosen as 0.9 indicating that 90% of the parents will go under crossover to produce offspring. The combination of MEP parameters used in this study is given in (Table 4).

GEP model development

Similar to the MEP algorithm, GEP was also employed using a special software called GeneXpro Tools¹⁰⁰. This software allowed to vary different parameters and monitor the change in algorithm's accuracy effectively. The GEP fitting parameters were also chosen using a trial-and-error approach. The parameters head size and number of chromosomes control the model's accuracy and convergence towards the answer. These parameters were varied across a range of possible values until highest model accuracy was reached using number of chromosomes as 30 and number of genes as 10. The variation in these parameters was terminated when there was no considerable

Parameters	Settings
GEP parameters	
Constants per gene	5
No. of genes	4
Linking function	Addition
Head size	10
No. of chromosomes	30
Functions	+, -, ×, ÷, sqrt
MEP parameters	
Number of generations	500
Subpopulation size	100
Runs	10
Crossover probability	0.9
No. of subpopulations	200
Functions	+, -, ×, ÷, sqrt
Code length	40
AdaBoost parameters	
Max. depth	8
n_estimators	100
Learning rate	0.01

Table 4. Algorithm fitting parameters used in the study.

change in accuracy with varying these parameters. The combination of parameters that gave the most accurate GEP model presented in this study is given in (Table 4).

AdaBoost model development

The AdaBoost model was implemented with the help of python programming language in the Anaconda software. The optimal combination of AdaBoost hyperparameters used in current study is shown in (Table 4). These parameters were chosen using grid search approach. In this approach, a parameter is selected and varied across a range of possible values while keeping all other parameters constant. In this way, the optimal value of every parameter was found, and the set of values given in (Table 4) resulted in a model with the lowest RMSE and highest R value. The n-estimators and max. depth in AdaBoost algorithm signify with the number of trees constructed and depth of each tree respectively. Learning rate is a factor that is applied to each new prediction of the tree and its value ranges from 0.01 to 0.1¹⁰¹. A smaller learning rate ensures that corrections are made gradually, thus resulting in a larger number of trees. For this reason, the value of learning rate is set to 0.01.

Algorithm results

MEP result

The MEP algorithm provided its output in the form of a C++ code which was decoded to get the equation to predict f_c' given by Eq. (1). Several trials were performed using the MEP software to develop an equation which is simple and accurate. The resultant MEP equation relates f_c' of bentonite plastic concrete with six most influential parameters using simple arithmetic functions. Only simple mathematical functions were chosen to be part of the equation in order to keep the resulting equation simple and not computationally challenging. The predictive capabilities of MEP equation are displayed in the form of a curve fitting plot in (Fig. 8). Notice that the MEP equation predicted strength with great accuracy for both training and testing sets except at a few points. Thus, it is evident from Fig. 8 that the developed MEP equation can be used to accurately forecast f_c' of BPC.

$$f_c' = \left(\frac{\frac{x_5}{x_3} - \frac{(x_4)^2}{(x_3)^3}}{\frac{x_1 - x_4}{(x_3)^2} - x_2} / \frac{x_3}{x_1 - x_4} \right) + \frac{x_0}{\sqrt{x_3}} - \sqrt{\frac{x_0}{x_3}} - \frac{x_3}{x_1 - x_4} \quad (1)$$

GEP result

The GEP model was constructed using addition as a linking function. Thus, the subexpression from every gene was added to get the final GEP equation given by Eq. (2). Notice that the GEP equation is also constructed using simple functions for simplification purposes. It is noteworthy that the GEP algorithm expressed the result in the form of four subexpression trees as shown in (Fig. 9). These trees were decoded, and the resulting expressions were added to get the final result. Also, the values of constants used in the expression trees are given in (Table 5).

$$f_c' = \left[x_4 - \left\{ \frac{x_4}{\frac{x_2 - (x_1 - x_3)}{\sqrt{x_3/1.512}}} \right\} \right] + \left[\left(\frac{\sqrt{2.581x_1}}{x_3} \right) \left(\frac{x_5 + (x_1/x_4)}{x_3} \right) \right] + \left[x_0 \left\{ \frac{(-38.93x_5/x_2)}{4.44 - \sqrt{x_3}} \right\} \right] + \left[\left\{ \left(\frac{\sqrt{x_5 + x_0}}{(6.04/x_0)x_3} \right) \sqrt{x_4} \right\} - x_4 \right] \quad (2)$$

Moreover, the prediction capacity of the GEP equation is visualized with the help of a curve fitting plot represented in (Fig. 10). It can be seen that GEP also predicted strength with great accuracy in both testing and

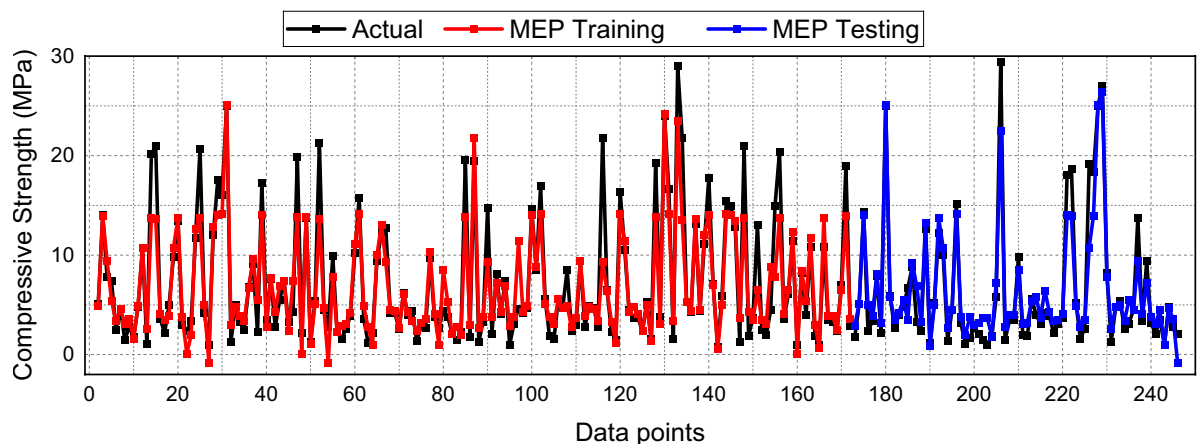
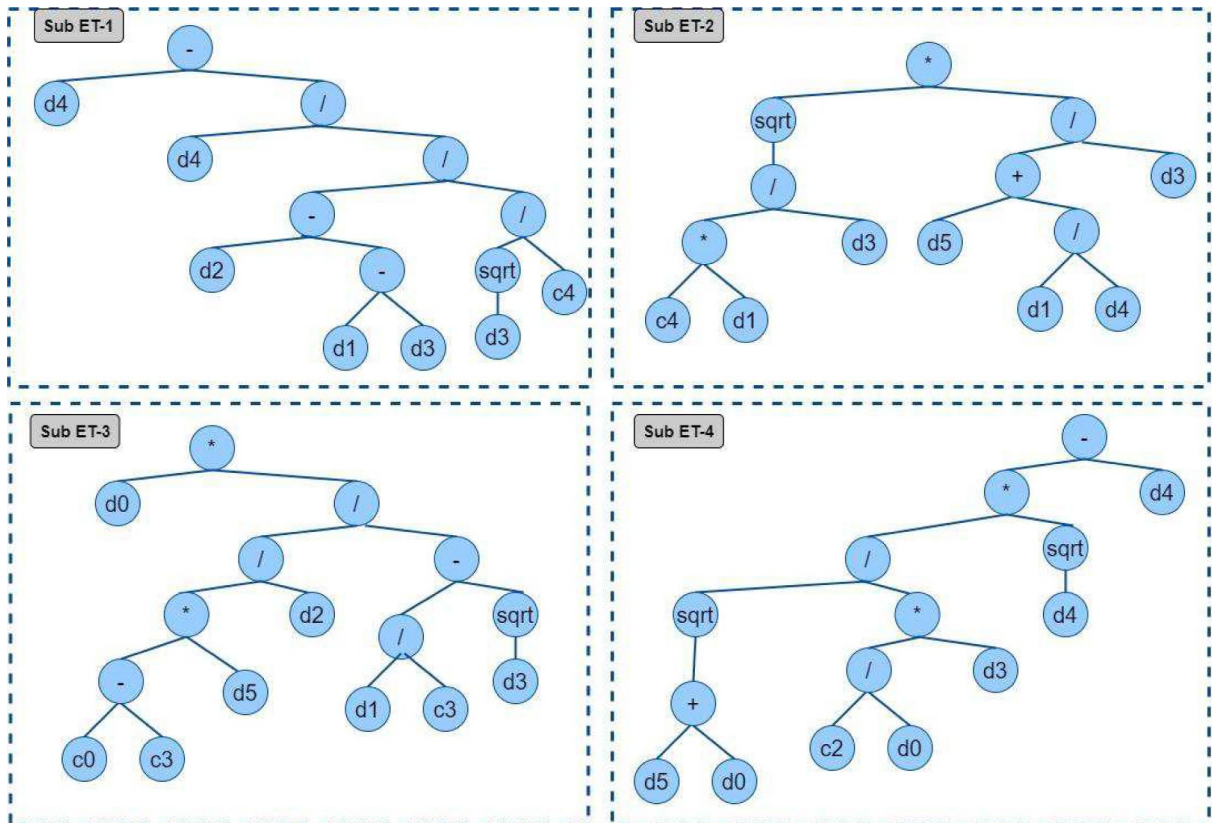


Figure 8. Curve fitting plot of MEP model.



Note: The variables d_0, d_1, d_2, \dots etc. in the expression trees correspond to x_0, x_1, x_2, \dots in the equation.

Figure 9. Expression tree representation of GEP equation. The variables d_0, d_1, d_2, \dots etc. in the expression trees correspond to x_0, x_1, x_2, \dots in the equation.

Name	Value
G1C4	1.512
G2C4	2.581
G3C3	4.441
G3C0	-8.776
G4C2	6.047

Table 5. Values of constants used in GEP expression trees.

training phases. There is some difference between actual and predicted values at some points but the overall performance of GEP looks satisfactory from the curve fitting plot.

AdaBoost result

The results of AdaBoost algorithm to predict BPC strength are shown in (Fig. 11). It can be well understood from curve fitting plot in Fig. 11 that AdaBoost predicted strength values with superior accuracy. The curve of predicted values practically lies above the actual value curve at most of the points. The difference between actual and predicted values at other points is also not as large as in GEP and MEP models. Moreover, from the error evaluation summary given in (Table 5), it can be seen that AdaBoost has the highest correlation with actual values in both training and testing phases and subsequently the lowest values for errors like RMSE and average error. Moreover, the objective function and performance index values are lowest for the AdaBoost model. However, it must be noted that while AdaBoost is more accurate than other two techniques, it failed to express its output in the form of an empirical relation like MEP and GEP algorithms.

Error assessment of models

It is necessary to evaluate the performance of ML models with the help of error evaluation metrics to validate their performance. Thus, the developed models were tested for their accuracy using previously described error

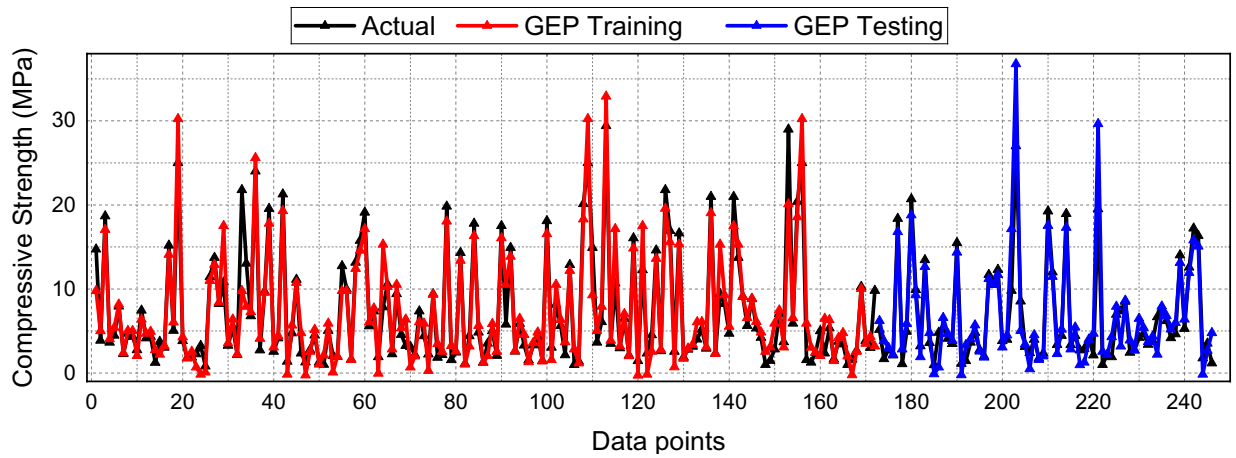


Figure 10. Curve fitting plot of GEP model.

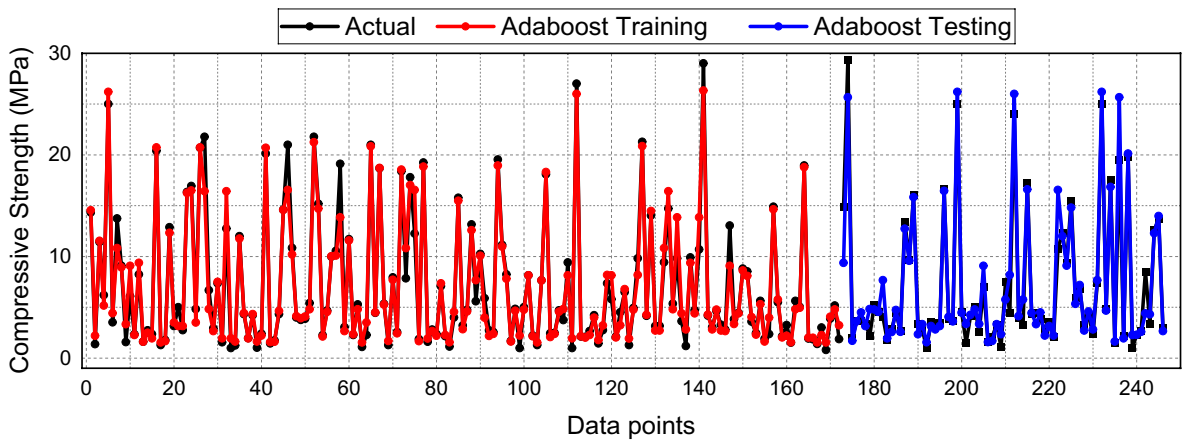


Figure 11. Curve fitting plot of AdaBoost model.

metrics in Sect. 2.3. The summary of this error evaluation is given in Table 6 for both training and testing phases of data-driven models. Notice that the correlation between experimental and predicted values by MEP, GEP, and AdaBoost in both stages is greater than the recommended value of 0.8 and even surpasses 0.9. It indicates the good predictive abilities of all models. Similarly, RMSE and mean absolute error of the models is also very close to zero. Moreover, the values of performance index and objective function are also less than the upper threshold of 0.2, which indicates that the models exhibit overall good performance.

However, it is necessary to evaluate each model’s strengths and weaknesses. To start with, the MEP model exhibited largest absolute error of 1.56 in training phase between actual and predicted values compared to 1.51 for GEP and 0.70 for AdaBoost. Also, the root mean squared error values are highest for MEP training followed by GEP training and lastly AdaBoost. Similarly, the correlation between actual and predicted values is highest for AdaBoost model. Moreover, the value of objective function which indicates the overall performance of a model is lowest for AdaBoost model, then MEP and lastly GEP. Thus, it can be inferred that AdaBoost gave predictions

	GEP	MEP	AdaBoost			
	Training	Testing	Training	Testing	Training	Testing
MAE	1.51	1.49	1.56	1.31	0.70	0.95
RMSE	2.36	2.38	2.57	2.02	1.22	1.66
R	0.936	0.932	0.923	0.959	0.962	0.936
p	0.16	0.187	0.18	0.16	0.088	0.12
a20-index	0.639	0.625	0.540	0.513	0.761	0.702
OF	0.176		0.16		0.107	

Table 6. Error metrics of developed models.

closest to the real values, followed by GEP and MEP gave the least accurate predictions. The two evolutionary techniques expressed their output in the form of empirical equations given by Eqs. (1) and (2). Both of the equations are simple and easy to solve since they involve only basic arithmetic functions.

Although the error evaluation in Table give the values of individual error metrics and reveals that all models are good for prediction purposes according to the evaluation criteria suggested by the literature. However, it is beneficial to test a model by simultaneously considering all of the error metrics employed in the study. Thus, the ability of developed algorithms to predict $f_{c'}$ of BPC is tested using radar plots and given in (Fig. 12). The radar plot for training phase of algorithms is given in Fig. 12a while Fig. 12b represents radar plot of algorithms for testing dataset. Notice from Fig. 12a that AdaBoost exhibits the least values of almost all error metrics. The values of error metrics are same for GEP and MEP except for RMSE. GEP exhibits a lower RMSE in training phase than MEP. But the overall performance of both models is almost same in training phase. However, for the testing phase in (Fig. 12b), the MEP algorithm exhibits lower average error and RMSE than GEP while AdaBoost continues to have the least values of MAE and RMSE than both of the evolutionary techniques. It means that AdaBoost has the lowest values of error metrics in both phases. However, MEP and GEP exhibits same performance in training phase, but MEP performs better than GEP when tested on unseen data.

Comparison of models

The radar plots given in Fig. 12 indicate that AdaBoost algorithm has lesser error values than the other two models in both training and testing phase. However, an interesting way to compare the relative performance of models to predict the given output is by means of a Taylor diagram¹⁰². A Taylor diagram is a useful technique to compare different ML models on the basis of standard deviation and coefficient of correlation (R). The accuracy of an algorithm can be check by checking its distance from the reference point. An accurate algorithm will lie close to the point of actual data and vice versa. A Taylor plot to compare the algorithms deployed in current research is given in (Fig. 13) and it can be seen from (Fig. 13) that the AdaBoost point lies closest to reference correlation, indicating the superior accuracy of AdaBoost. The MEP and GEP points lie on the same level in regard to correlation with actual values, but since GEP point lies closer to the standard deviation line of actual data than MEP, it can be concluded that GEP values deviate lesser from actual values than MEP. It can also be verified from

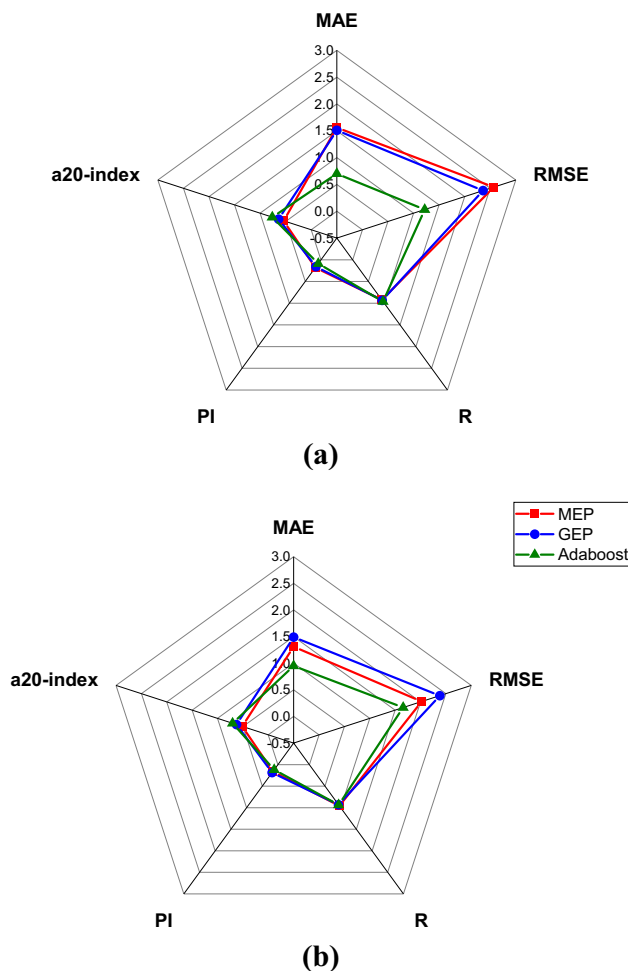


Figure 12. Radar plot-based comparison of algorithms; (a) training phase; (b) testing phase.

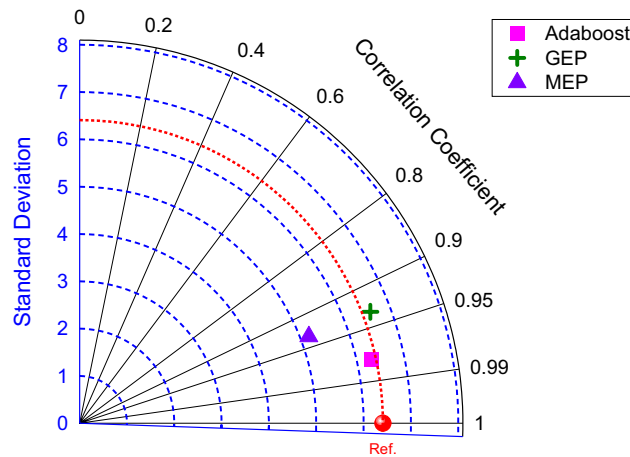


Figure 13. Taylor plot-based comparison of developed models.

the a20-index values of algorithms in (Table 6). GEP has higher values of a20-index indicating lesser deviation from actual values than MEP. However, the a20-index values are highest for the AdaBoost model surpassing both its competitors. Thus, from the Taylor plot, the overall order of accuracy is AdaBoost > GEP > MEP.

Explanatory analysis

After the development of models and their successful validation using the error metrics, the next step is to carry explanatory analysis to get more information about the model prediction process and relative importance of different input variables. Carrying out this type of analysis on developed models is important for making sure that the model is generalized and can perform well on various data configurations⁸⁹. Thus, shapley analysis is carried out on AdaBoost model since in Sect. 3.4, we established that it is more robust than other two models for prediction of $f_{c,t}$ of bentonite plastic concrete. SHAP is a method that explains how predictions are made by a model and is increasingly catching attention of researchers to enhance the interpretability of soft computing models. It is a reliable method to provide simple and interpretable explanation of ML models to people who are not experts in the field and to different stakeholders. The explanation provided by SHAP analysis can then be used to make informed decisions and further improve the ML models by varying the set of input variables etc. It divides a prediction into sum of contributions offered by each of the input variables. The SHAP analysis results in a dataset having same dimensions as the original training data and it represents each prediction made by the ML model as a sum of these SHAP values and a base value¹⁰³. With the recent advancements in the field of machine learning to predict various material properties, the need for reliable methods for enhancing interpretability of ML models has also been increased and SHAP successfully caters this need of researchers by providing a reliable method for both overall interpretation of the data-driven model and also the individual explanations for each prediction made by the model¹⁰⁴.

SHAP global interpretation

The results of shapley analysis of AdaBoost model to predict BPC strength are given in (Fig. 14). The variables are ranked according to their mean absolute SHAP values for the entire dataset in the bar plot given in (Fig. 14a). Notice from (Fig. 14a) that cement is the most contributing factor to predict strength having absolute shap value of 12.82 followed by coarse aggregate (6.05), fine aggregate (0.43) and the least contributing factor is bentonite (0.12). Although Fig. 14a describes the mean absolute shap values of variables, it does not give an idea about which variables are positively or negatively correlated with the output. To overcome this limitation, the SHAP summary plot is given in (Fig. 14b) in which the SHAP values for input variables considered to develop AdaBoost model for predicting $f_{c,t}$ of BPC are given. The values given in SHAP summary plot signify the contribution of each variable in strength prediction and are arranged in the order of decreasing importance from top to bottom. In shapley summary plot, the features having the broadest range of SHAP values are the important ones and vice versa. Each dot on the plot shows contribution of a particular feature with blue showing lower values and red representing higher values¹⁰⁵. It can be well visualized from Fig. 14b that cement has the broadest range of red dots, thus it is the most important feature to predict strength followed by coarse aggregate, clay, and fine aggregate while bentonite has the least red dots over a wide range which means it is the least contributing factor. It is depicted by the summary plot that cement, coarse aggregate, and fine aggregate contribute towards an increase in the strength values while clay, water, and bentonite cause a decrease in output values.

SHAP local interpretation

After the identifying most important features to predict $f_{c,t}$ of BPC by means of global interpretability plots, the next step is to check the combined effect of input variables through partial dependence plots as shown in (Fig. 15). Notice that as coarse aggregate content increases, the cement content also increases along with the increase in corresponding SHAP values of cement. It is because the incorporation of greater quantities of coarse aggregate call for higher cement content so that the cement paste can cover all the aggregate particles and result

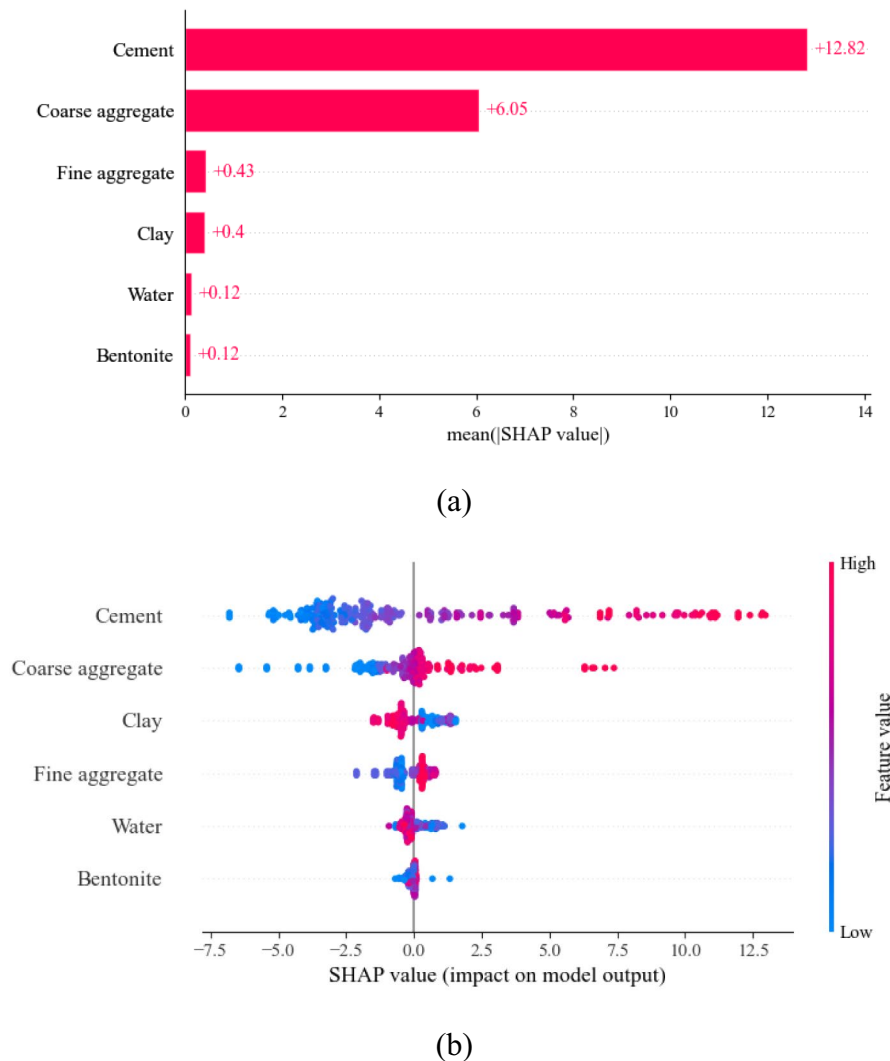


Figure 14. Shapley analysis of AdaBoost algorithm; (a) mean absolute SHAP values; (b) SHAP summary plot.

in a homogeneous concrete matrix. Similarly, the increase in cement content from 80 to 240 kg increases SHAP values of coarse aggregate from -6 to 8. However, the change in SHAP values of fine aggregate with cement content is much more scattered. Same trend is exhibited by water and coarse aggregate. Moreover, the SHAP values of bentonite shows an initial increase with increase in cement content but becomes almost constant when cement content reaches 120 kg. Furthermore, the relation between fine aggregate and SHAP values of clay is much more randomized and does not exhibit a specific trend.

Figure 16 depicts three SHAP force plots to show how model generates individual predictions. The force plot tool of shapley additive explanatory analysis is widely used to explain the phenomenon of generating predictions by ML model. The importance of each variable in force plot is proportional to its size in the force plot and the intersection of blue and red arrows show the predicted value by the ML model. The centre point of the plot is base value, which is the average prediction made by the model and each arrow in the force plot represents an input variable which acts as predictor for final output. Each arrowhead represents the direction and magnitude of the effect of a particular input variable on the output. The force plots for predicted strength values of 3.86 and 8.22 MPa are shown in (Fig. 16a,b) respectively. Notice from these figures that all variables offer some contribution in the prediction, however cement, coarse aggregate, and fine aggregate are the most contributing factors to BPC strength. The contribution of other variables is relatively small.

While the force plots given in Fig. 16 provide a condensed interpretation of individual predictions, a more detailed explanation can be offered by means of SHAP waterfall plots as shown in (Fig. 17) which depict how different features have significant effect on the AdaBoost predictions. These plots represent a simple yet useful way of explaining local interpretability of ML model plots by providing the contribution of each input variable in yielding output. The predicted values by the model i.e., $f(x)$ are shown at the top of the plot and the base value, which is the average predicted value of the model and same for all predictions is shown at bottom of the plot. The parameters in waterfall plots are arranged according to their impact on the output from top to bottom. The numbers written in grey alongside the names of input variables denote the variable values for the particular instance

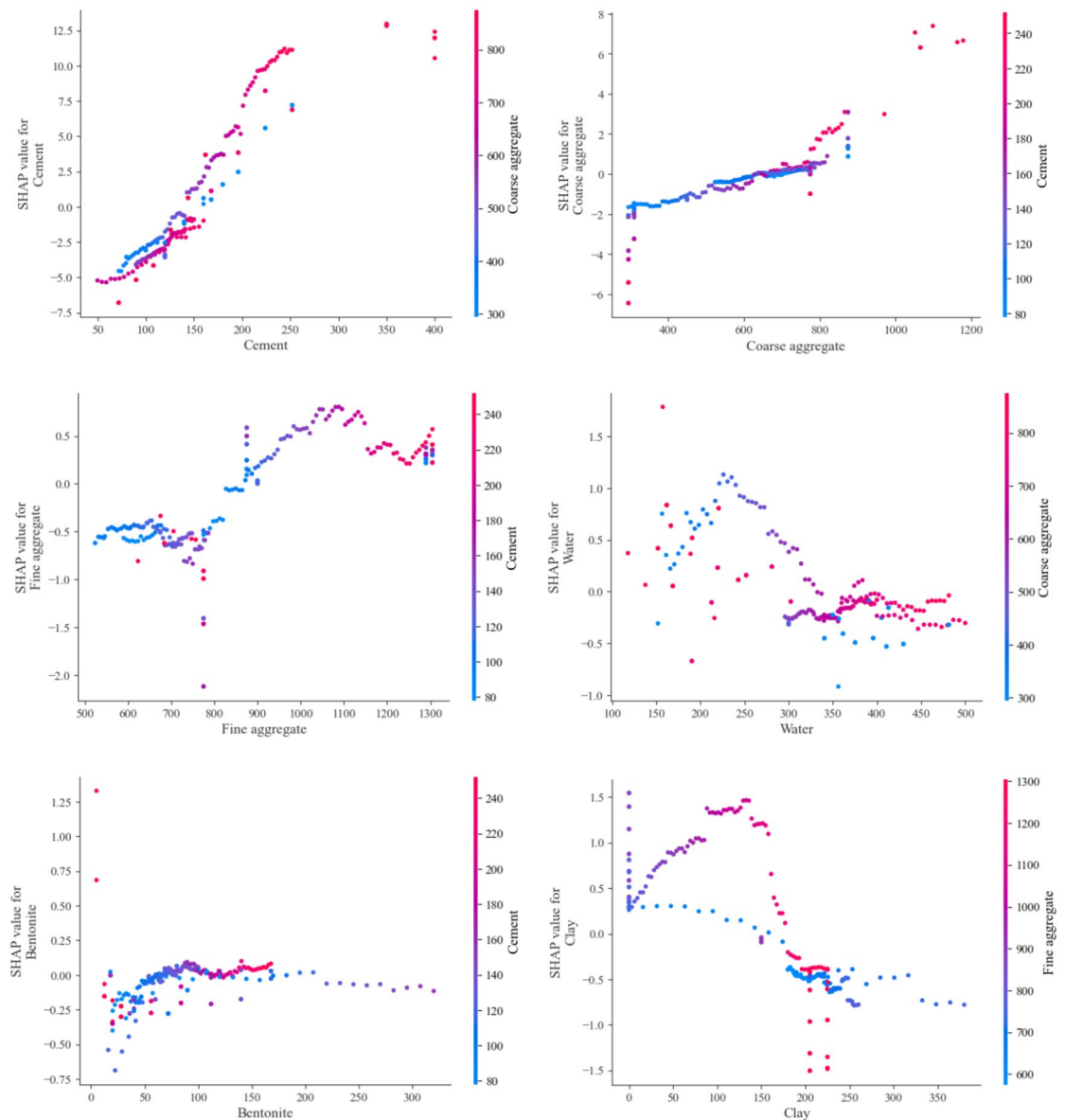


Figure 15. SHAP local interpretability; partial dependence plots of variables.

under consideration. The direction of arrow indicates the direction in which each input affects the prediction. The arrows are colour coded in such a way that red colour corresponds to a push towards higher strength value while blue colour represents a push towards lower strength values. The final prediction of the model is a sum of base value and all shap values from input variables. The waterfall plots for predicted strength values of 8.315 and 12.19 MPa are shown in (Fig. 17a,b) respectively. Notice from Fig. 17a that only water affects strength negatively for this instance while all other inputs positively influence the output. However, the effect of cement, clay, and aggregate is more pronounced than other input variables. Similarly, in Fig. 17b only clay negatively affects the strength. In this way, shapley additive explanatory analysis can be used to check how the algorithm generated individual predictions for more transparency in the model building process.

Notice from the global and local SHAP interpretability that cement, clay, and aggregate content are the most important variables for predicting BPC strength. These findings are also in line with the findings of previous studies. The importance of cement content in BPC has been investigated by Abbaslou et al.¹⁰⁶ in which the authors identified that cement is one of the main factors in determining strength of BPC. Similarly, the importance of aggregate content in the BPC has been highlighted in various studies^{107–109}. Moreover, the reduction in strength due to incorporation of bentonite and clay minerals in concrete was studied by Guan et al.¹¹⁰ in which the authors highlighted that addition of bentonite and clay minerals results in loss of compressive strength of concrete. This is because the bentonite and clay present in BPC absorbs water due to their excellent absorptive properties which results in less water being available for hydration of cement paste and loss in concrete strength. This is also validation by the results of SHAP analysis where clay and bentonite have the smallest contribution towards concrete strength. Thus, it can be concluded that the AdaBoost is the feasible model for predicting strength of

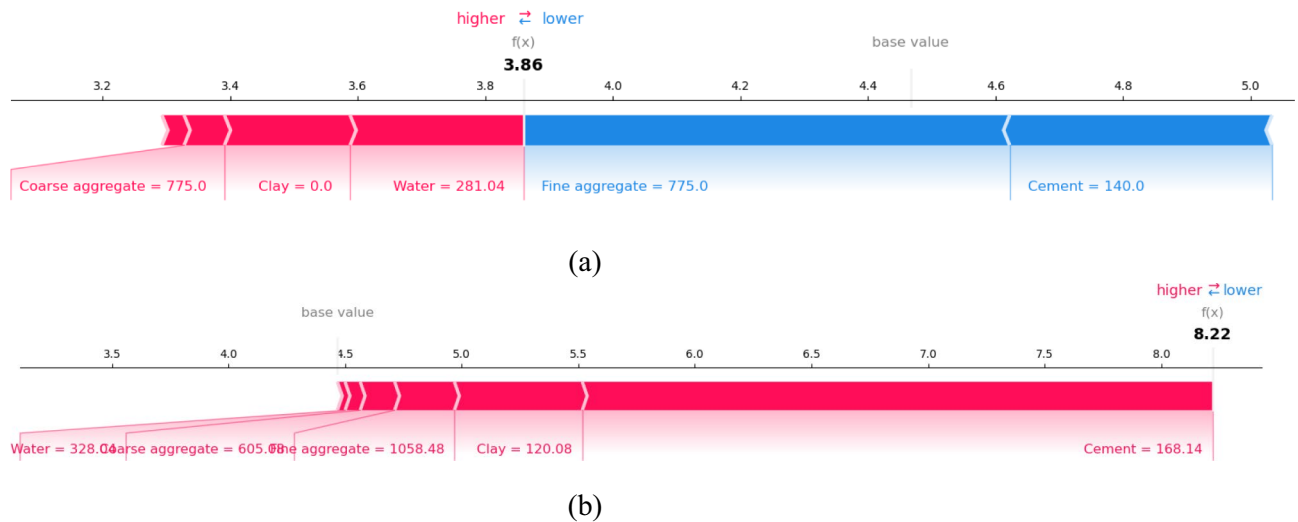


Figure 16. SHAP force plots for different strength values.

BPC which is robust and accurate and the insights into the prediction process by virtue of shapley analysis reveals that it aligns perfectly with the findings of previous studies.

Graphical user interface (GUI)

As discussed earlier in introduction section, the accurate determination BPC strength is crucial for fostering its widespread adoption in the civil engineering industry. Thus, this study was done to develop predictive models for BPC strength using MEP, GEP, and AdaBoost algorithms. The AdaBoost algorithm predicted the desired outcome with the highest accuracy. However, being a black-box model, its widespread utilization for professionals in the industry is quite challenging. Similarly, the GEP and MEP algorithms yielded equations to predict BPC strength, but both of the equations are computationally challenging. Therefore, a GUI has been developed using python programming language to help compute the BPC strength automatically based on the set of input variables considered in this as shown in Fig. 18. The user can enter desired input values and upon pressing the predict button, result will be obtained instantly in the output box specified in the GUI. This GUI can serve as an invaluable tool for professionals in the civil engineering industry and different stakeholders to effectively predict the BPC strength by entering the mixture composition without requiring extensive technical expertise. Also, with the increasing shift of civil engineering industry towards informed decision making, this GUI is an important tool which reflects the practical application of ML utilization in civil engineering industry.

Conclusions

This study outlines the successful application of ML techniques to foster the use of BPC in construction industry by providing empirical models to predict its 28-day compressive strength by using MEP, AdaBoost and GEP. The dataset of 246 points gathered from published literature was used for this purpose having six important input factors. The gathered dataset was divided into training and testing sets to be used for model training and testing respectively. The main conclusions of this study are:

- The two evolutionary techniques employed in this study i.e., MEP and GEP offered the advantage of expressing their output in the form of an empirical equation that can be used to effectively calculate f_c' of BPC while AdaBoost has no such advantage.
- The error evaluation revealed that all algorithms satisfied the criteria suggested in literature for good models by exhibiting a correlation coefficient greater than 0.9 between real and predicted values for both training and testing phases.
- The AdaBoost algorithm proved to be more accurate than GEP and MEP by demonstrating a higher correlation of 0.962 compared to 0.936 of GEP and 0.926 of MEP. Similarly, the value of objective function was 0.10 for AdaBoost, 0.16 for MEP and 0.176 for GEP which indicated that AdaBoost is the most accurate algorithm. Moreover, the performance index values were also lesser for AdaBoost model. However, it doesn't yield empirical equations like MEP and GEP.
- Shapley additive explanatory analysis was carried out on AdaBoost model to gain insights into the prediction process and the results indicated that cement, coarse aggregate, and fine aggregate are the most important factors to predict BPC strength.

Recommendations

Although the empirical prediction models developed in current study showcased good performance and can be effectively used for predicting f_c' of BPC, but it is important to highlight the limitations of current study and give suggestions for future research endeavours.

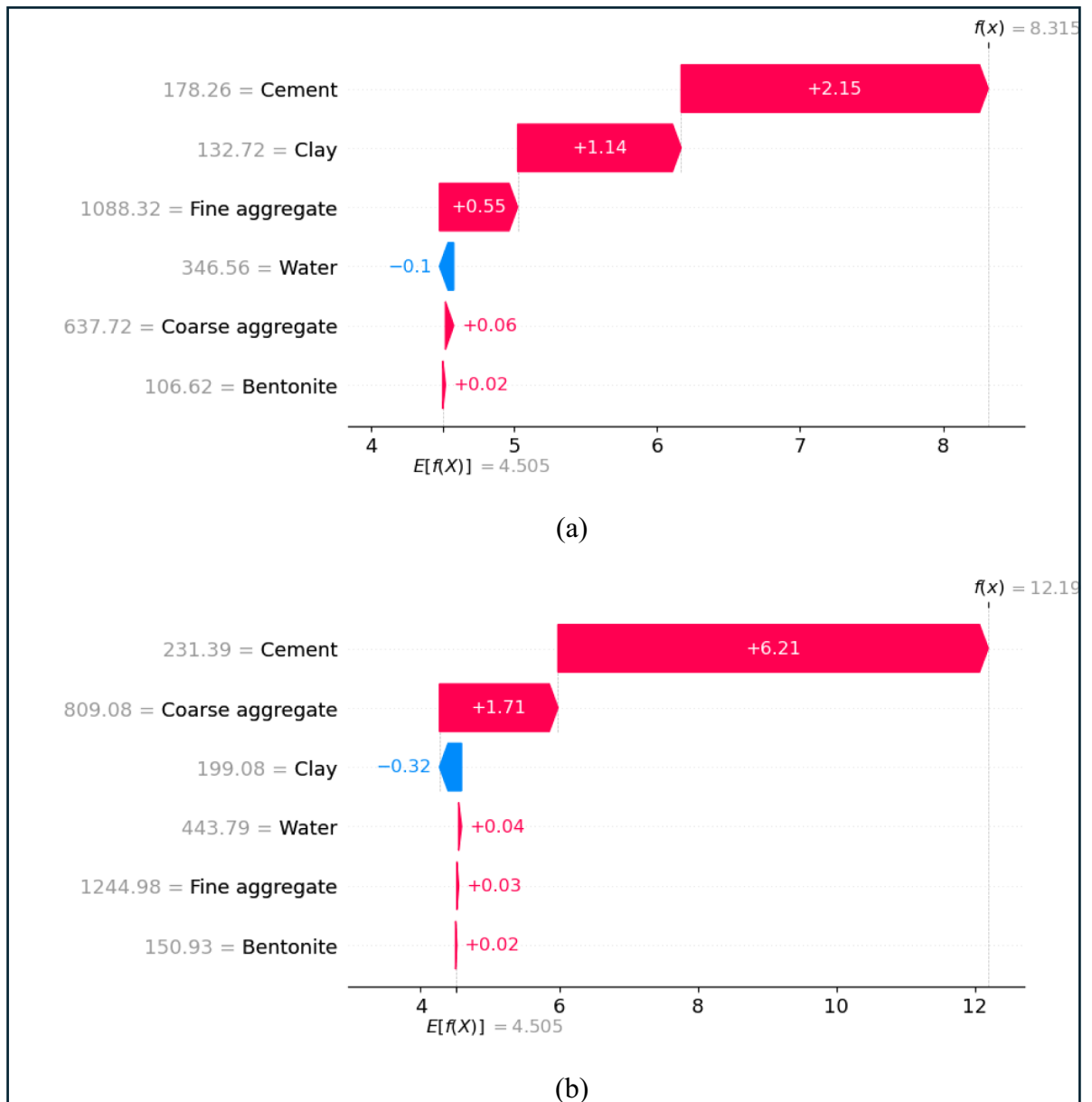


Figure 17. SHAP waterfall plots for different strength values.

- It should be noted that the models in current study were developed using a dataset of 246 variables as given in Table A. However, it is recommended to consider even larger datasets from variable sources for future studies to develop more generalized models.
- The input and output variables used in current study had a limited range as given in (Table 1). It is advised to use input and output variables with much wider range in future studies for development of widely applicable and more robust models.
- Different material properties like nominal diameter of aggregates, type of cement, different properties of bentonite and clay etc. can affect compressive strength of bentonite plastic concrete. This study doesn't explicitly incorporate the effect of these variables but recommends considering them in future research.
- Prediction models for other mechanical properties of bentonite plastic concrete like flexural strength, split tensile strength etc. should be developed to foster its widespread use.

Soft Computing Models for Prediction of Bentonite Plastic Concrete Strength

Input parameters

Cement	Kg/m ³	<input style="width: 95%; height: 20px;" type="text"/>
Water	Kg/m ³	<input style="width: 95%; height: 20px;" type="text"/>
Coarse Aggregate	Kg/m ³	<input style="width: 95%; height: 20px;" type="text"/>
Fine Aggregate	Kg/m ³	<input style="width: 95%; height: 20px;" type="text"/>
Clay	Kg/m ³	<input style="width: 95%; height: 20px;" type="text"/>
Bentonite	Kg/m ³	<input style="width: 95%; height: 20px;" type="text"/>

Prediction

Output parameters

Compressive strength	MPa	<input style="width: 95%; height: 20px;" type="text"/>
-----------------------------	-----	--

Clear

Figure 18. Graphical user interface to predict BPC strength.

Data availability

The dataset used in this study is available with the authors and available upon reasonable request.

Received: 21 February 2024; Accepted: 2 August 2024

Published online: 05 August 2024

References

- Inglezakis, V. J., Stylianou, M. A., Gkantzou, D. & Loizidou, M. D. Removal of Pb (II) from aqueous solutions by using clinoptilolite and bentonite as adsorbents. *Desalination* **210**, 248–256 (2007).
- Chaari, I., Fakhfakh, E., Chakroun, S. & Bouzid, J. Lead removal from aqueous solutions by a Tunisian smectitic clay. *J. Hazard. Mater.* **156**, 545–551 (2008).
- Wang, K. *et al.* Experimental study of mechanical properties of hot dry granite under thermal-mechanical couplings. *Geothermics* **119**, (2024).
- Li, Z. *et al.* Ternary cementless composite based on red mud, ultra-fine fly ash, and GGBS: Synergistic utilization and geopolymerization mechanism. *Case Stud. Constr. Mater.* **19**, e02410 (2023).
- Cai, Y. *et al.* A review of monitoring, calculation, and simulation methods for ground subsidence induced by coal mining. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00595-4> (2023).
- Chen, C., Han, D. & Chang, C. C. MPCCT: Multimodal vision-language learning paradigm with context-based compact transformer. *Pattern Recognit.* **147**, 110084 (2024).
- Amlashi, A. T., Abdollahi, S. M., Goodarzi, S. & Ghanizadeh, A. R. Soft computing based formulations for slump, compressive strength, and elastic modulus of bentonite plastic concrete. *J. Clean. Prod.* **230**, 1197–1216 (2019).
- He, H., Qiao, H., Sun, T., Yang, H. & He, C. Research progress in mechanisms, influence factors and improvement routes of chloride binding for cement composites. *J. Build. Eng.* <https://doi.org/10.1016/j.job.2024.108978> (2024).
- Chen, C., Han, D. & Shen, X. CLVIN: Complete language-vision interaction network for visual question answering. *Knowl. Based Syst.* **275**, 110706 (2023).
- Shi, S., Han, D. & Cui, M. A multimodal hybrid parallel network intrusion detection model. *Conn. Sci.* <https://doi.org/10.1080/09540091.2023.2227780> (2023).
- Sun, L. *et al.* Experimental investigation on the bond performance of sea sand coral concrete with FRP bar reinforcement for marine environments. *Adv. Str. Eng.* <https://doi.org/10.1177/13694332221131153> (2022).
- Thiruchittampalam, S., Singh, S. K., Banerjee, B. P., Glenn, N. F. & Raval, S. Spoil characterisation using UAV-based optical remote sensing in coal mine dumps. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00622-4> (2023).
- Liu, Y. *et al.* Time-shift effect of spontaneous combustion characteristics and microstructure difference of dry-soaked coal. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00616-2> (2023).
- Wang, H., Han, D., Cui, M. & Chen, C. NAS-YOLOX: A SAR ship detection using neural architecture search and multi-scale attention. *Conn. Sci.* **35**, 1–32 (2023).
- Zheng, H., Jiang, B., Wang, H. & Zheng, Y. Experimental and numerical simulation study on forced ventilation and dust removal of coal mine heading surface. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-024-00667-z> (2024).
- Asif, U. *et al.* Predicting the mechanical properties of plastic concrete: An optimization method by using genetic programming and ensemble learners. *Case Stud. Construction Mater.* **20**, e03135 (2024).
- Dou, J. *et al.* Surface activity, wetting, and aggregation of a perfluoropolyether quaternary ammonium salt surfactant with a hydroxyethyl group. *Molecules* **28**, 7151 (2023).
- Hu, L., Gao, D. & Li, Y. Analysis of the influence of long curing age on the compressive strength of plastic concrete. *Adv. Mater. Res.* **382**, 200–203 (2012).
- Asif, U., Javed, M. F., Alyami, M. & Hammad, A. W. Performance evaluation of concrete made with plastic waste using multi-expression programming. *Mater. Today Commun.* **39**, 108789 (2024).
- Ifukhar, B. *et al.* Experimental study on the eco-friendly plastic-sand paver blocks by utilising plastic waste and basalt fibers. *Heliyon* **9**, 6 (2023).
- Callari, C. *Embankment Dams with Bituminous Concrete Upstream Facing: Review and Recommendations, International Commission on Large Dams (ICOLD) Bulletin n. 114.* <https://www.researchgate.net/publication/347228515>.
- Karunaratne, G. & Chew, S. Bentonite: Kaolinite clay liner. *Geosynth. Int.* **8**, 113–133 (2001).
- Farooq, F., Ahmed, W., Akbar, A., Aslam, F. & Alyousef, R. Predictive modeling for sustainable high-performance concrete from industrial wastes: A comparison and optimization of models using ensemble learners. *J. Clean. Prod.* **292**, 126032 (2021).
- Jiao, H. *et al.* A novel approach in forecasting compressive strength of concrete with carbon nanotubes as nanomaterials. *Mater. Today Commun.* **35**, 106335 (2023).
- Zheng, W. *et al.* Sustainable predictive model of concrete utilizing waste ingredient: Individual algorithms with optimized ensemble approaches. *Mater. Today Commun.* **35**, 105901 (2023).
- Khawaja, L. *et al.* Indirect estimation of resilient modulus (Mr) of subgrade soil: Gene expression programming vs multi expression programming. *Structures* **66**, 106837 (2024).
- Asif, U., Memon, S. A., Javed, M. F. & Kim, J. Predictive modeling and experimental validation for assessing the mechanical properties of cementitious composites made with silica fume and ground granulated blast furnace slag. *Buildings* **14**(4), 1091 (2024).
- Fei, R., Guo, Y., Li, J., Hu, B. & Yang, L. An Improved BPNN method based on probability density for indoor location. *IEICE Trans. Inf. Syst.* **E106.D**, 773–785 (2023).
- Zhao, Y. *et al.* Intelligent control of multilegged robot smooth motion: A review. *IEEE Access* **11**, 86645–86685. <https://doi.org/10.1109/ACCESS.2023.3304992> (2023).
- Chen, D. L., Zhao, J. W. & Qin, S. R. SVM strategy and analysis of a three-phase quasi-Z-source inverter with high voltage transmission ratio. *Sci. Ch. Technol. Sci.* **66**, 2996–3010 (2023).
- Eldin, N. N. & Senouci, A. B. Measurement and prediction of the strength of rubberized concrete. *Cem. Concr. Compos.* **16**, 287–298 (1994).
- Zhu, C., Li, X., Wang, C., Zhang, B. & Li, B. Deep learning-based coseismic deformation estimation from InSAR interferograms. *IEEE Trans. Geosci. Remote Sens.* **62**, 1–10 (2024).
- Ahmad, A. *et al.* Compressive strength prediction via gene expression programming (Gep) and artificial neural network (ann) for concrete containing rca. *Buildings* **11**, 324 (2021).
- Jalal, F. E. *et al.* ANN-based swarm intelligence for predicting expansive soil swell pressure and compression strength. *Sci. Rep.* <https://doi.org/10.1038/s41598-024-65547-7> (2024).
- Alyousef, R. *et al.* Forecasting the strength characteristics of concrete incorporating waste foundry sand using advance machine algorithms including deep learning. *Case Stud. Constr. Mater.* **19**, e02459 (2023).
- Zhang, G. *et al.* Electric-field-driven printed 3D highly ordered microstructure with cell feature size promotes the maturation of engineered cardiac tissues. *Adv. Sci.* <https://doi.org/10.1002/advs.202206264> (2023).

37. Zare Naghadehi, M., Samaei, M., Ranjbaria, M. & Nourani, V. State-of-the-art predictive modeling of TBM performance in changing geological conditions through gene expression programming. *Meas. (Lond.)* **126**, 46–57 (2018).
38. Shahin, M., Jaksa, M. B. & Maier, H. R. *Physical Modeling of Rolling Dynamic Compaction View Project Artificial Neural Networks-Pile Capacity Prediction View Project. Artificial Neural Networks in Geotechnical Engineering Article in Electronic Journal of Geotechnical Engineering* <https://www.researchgate.net/publication/228364758> (2008).
39. Wang, G. *et al.* Research and practice of intelligent coal mine technology systems in China. *Int J Coal Sci Technol* **9**, 1–17 (2022).
40. Amin, M. N. *et al.* Prediction of sustainable concrete utilizing rice husk ash (RHA) as supplementary cementitious material (SCM): Optimization and hyper-tuning. *J. Mater. Res. Technol.* **25**, 1495–1536 (2023).
41. Li, Q. *et al.* Splitting tensile strength prediction of Metakaolin concrete using machine learning techniques. *Sci. Rep.* <https://doi.org/10.1038/s41598-023-47196-4> (2023).
42. Nazar, S. *et al.* Machine learning interpretable-prediction models to evaluate the slump and strength of fly ash-based geopolymer. *J. Mater. Res. Technol.* **24**, 100–124 (2023).
43. Mahmood, M. S. *et al.* Enhancing compressive strength prediction in self-compacting concrete using machine learning and deep learning techniques with incorporation of rice husk ash and marble powder. *Case Stud. Constr. Mater.* **19**, e02557 (2023).
44. Iqbal, M. F. *et al.* Sustainable utilization of foundry waste: Forecasting mechanical properties of foundry sand based concrete using multi-expression programming. *Sci. Total Environ.* **780**, 146524 (2021).
45. Shahab, Z. *et al.* Experimental investigation and predictive modeling of compressive strength and electrical resistivity of graphene nanoplatelets modified concrete. *Mater. Today Commun.* **38**, 107639 (2024).
46. Ahmad, A. *et al.* Comparative study of supervised machine learning algorithms for predicting the compressive strength of concrete at high temperature. *Materials* **14**, 4222 (2021).
47. Ahmad, A., Farooq, F., Ostrowski, K. A., Śliwa-Wieczorek, K. & Czarnecki, S. Application of novel machine learning techniques for predicting the surface chloride concentration in concrete containing waste material. *Materials* **14**, 2297 (2021).
48. Hu, Y. *et al.* Strength evaluation sustainable concrete with waste ingredients at elevated temperature by employing interpretable algorithms: Optimization and hyper tuning. *Mater. Today Commun.* **36**, 106467 (2023).
49. Ahmad, A. *et al.* Prediction of compressive strength of fly ash based concrete using individual and ensemble algorithm. *Materials* **14**, 1–21 (2021).
50. Song, H. *et al.* Predicting the compressive strength of concrete with fly ash admixture using machine learning algorithms. *Constr. Build. Mater.* **308**, 125021 (2021).
51. Wu, H. *et al.* Stability analysis of rib pillars in highwall mining under dynamic and static loads in open-pit coal mine. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-022-00504-1> (2022).
52. Bin Inqiad, W., Ali Raza, M. & Asim, M. Predicting 28-day compressive strength of self-compacting concrete (SCC) using gene expression programming (GEP). *Arch. Adv. Eng. Sci.* <https://doi.org/10.47852/bonviewaaes32021606> (2023).
53. Yao, X. *et al.* AI-based performance prediction for 3D-printed concrete considering anisotropy and steam curing condition. *Constr. Build. Mater.* **375**, 130898 (2023).
54. Lu, D. *et al.* A dynamic elastoplastic model of concrete based on a modeling method with environmental factors as constitutive variables. *J. Eng. Mech.* **149**, 04023102 (2023).
55. Meng, S., Meng, F., Chi, H., Chen, H. & Pang, A. A robust observer based on the nonlinear descriptor systems application to estimate the state of charge of lithium-ion batteries. *J. Frankl. Inst.* **360**, 11397–11413 (2023).
56. Lu, D., Zhou, X., Du, X. & Wang, G. 3D dynamic elastoplastic constitutive model of concrete within the framework of rate-dependent consistency condition. *J. Eng. Mech.* [https://doi.org/10.1061/\(ASCE\)EM.1943-7889.0001854](https://doi.org/10.1061/(ASCE)EM.1943-7889.0001854) (2020).
57. Shi, M. L., Lv, L. & Xu, L. A multi-fidelity surrogate model based on extreme support vector regression: Fusing different fidelity data for engineering design. *Eng. Comput.* **40**, 473–493 (2023).
58. Wang, M. *et al.* Sulfate diffusion in coal pillar: Experimental data and prediction model. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00575-8> (2023).
59. Xie, X. *et al.* Fluid inverse volumetric modeling and applications from surface motion. *IEEE Trans. Vis. Comput. Graph.* <https://doi.org/10.1109/TVCG.2024.3370551> (2024).
60. Qi, Q., Yue, X., Duo, X., Xu, Z. & Li, Z. Spatial prediction of soil organic carbon in coal mining subsidence areas based on RBF neural network. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00588-3> (2023).
61. Wu, Y. *et al.* A study on the ultimate span of a concrete-filled steel tube arch bridge. *Buildings* **14**(4), 896 (2024).
62. Soleimani, F., Si, G., Roshan, H. & Zhang, J. Numerical modelling of gas outburst from coal: a review from control parameters to the initiation process. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00657-7> (2023).
63. Huang, F. *et al.* Slope stability prediction based on a long short-term memory neural network: comparisons with convolutional neural networks, support vector machines and random forest models. *Int. J. Coal Sci. Technol.* **10**(1), 18 (2023).
64. Wang, C., Xu, S. & Yang, J. Adaboost algorithm in artificial intelligence for optimizing the IRI prediction accuracy of asphalt concrete pavement. *Sensors* **21**, 5682 (2021).
65. Koza, J. R. Genetic programming as a means for programming computers by natural selection. *Stat. Comput.* **4**, 87–112 (1994).
66. Ferreira, C. *Gene Expression Programming: Mathematical Modeling by an Artificial Intelligence*. **21**, (Springer-Verlag Berlin Heidelberg, Springer, 2006).
67. Lu, S., Zhao, J., Song, J., Chang, J. & Shu, C. M. Apparent activation energy of mineral in open pit mine based upon the evolution of active functional groups. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00650-0> (2023).
68. Crina, M. O. & Gros, an, G. A Comparison of Several Linear GP Techniques A Comparison of Several Linear Genetic Programming Techniques. www.mep.cs.ubbcluj.ro. (2003).
69. Li, J., Tang, H., Li, X., Dou, H. & Li, R. LEF-YOLO: A lightweight method for intelligent detection of four extreme wildfires based on the YOLO framework. *Int. J. Wildland Fire* <https://doi.org/10.1071/WF23044> (2023).
70. Chang, X. *et al.* Single-objective and multi-objective flood interval forecasting considering interval fitting coefficients. *Water Resour. Manag.* <https://doi.org/10.1007/s11269-024-03848-2> (2024).
71. Sonebi, M. & Cevik, A. Genetic programming based formulation for fresh and hardened properties of self-compacting concrete containing pulverised fuel ash. *Constr. Build. Mater.* **23**, 2614–2622 (2009).
72. Koza, J. R. & Jacks Hall, M. *Survey of genetic algorithms and genetic programming*. <http://www-cs-faculty.stanford.edu/~koza/>.
73. Gholampour, A., Gandomi, A. H. & Ozbakkaloglu, T. New formulations for mechanical properties of recycled aggregate concrete using gene expression programming. *Constr. Build. Mater.* **130**, 122–145 (2017).
74. Xin, J., Xu, W., Cao, B., Wang, T. & Zhang, S. A deep-learning-based MAC for integrating channel access, rate adaptation and channel switch. <https://doi.org/10.48550/arXiv.2406.02291> (2024).
75. Liao, L. *et al.* Color image recovery using generalized matrix completion over higher-order finite dimensional algebra. *Axioms* **12**, 954 (2023).
76. Bin Inqiad, W. *et al.* Comparison of boosting and genetic programming techniques for prediction of tensile strain capacity of engineered cementitious composites (ECC). *Mater. Today Commun.* **39**, 109222 (2024).
77. Wang, H. L. & Yin, Z. Y. High performance prediction of soil compaction parameters using multi expression programming. *Eng. Geol.* **276**, 105758 (2020).
78. Mousavi, S. M., Alavi, A. H., Gandomi, A. H., Esmaeili, M. A. & Gandomi, M. A Data mining approach to compressive strength of CFRP-confined concrete cylinders. *Str. Eng. Mech.* **36**(6), 759–783 (2010).

79. Schapire, R. E. The strength of weak learnability. *Mach. Learn.* **5**, 197–227 (1990).
80. He, X. *et al.* Excellent microwave absorption performance of LaFeO₃/Fe₃O₄/C perovskite composites with optimized structure and impedance matching. *Carbon* **213**, 118200 (2023).
81. Guo, J. *et al.* Study on optimization and combination strategy of multiple daily runoff prediction models coupled with physical mechanism and LSTM. *J. Hydrol. (Amst.)* **624**, 129969 (2023).
82. Freund, Y. & Schapire, R. E. A short introduction to boosting. *J. Jpn. Soc. Artif. Intell.* **14**, 771–780 (1999).
83. Ying, C., Qi-Guang, M., Jia-Chen, L. & Lin, G. Advance and prospects of AdaBoost algorithm. *Acta Automat. Sin.* **39**, 745–758 (2013).
84. Inqiad, W. B. *et al.* Comparative analysis of various machine learning algorithms to predict 28-day compressive strength of Self-compacting concrete. *Heliyon* **9**, 22036 (2023).
85. Thapa, I., Kumar, N., Ghani, S., Kumar, S. & Gupta, M. Applications of bentonite in plastic concrete: A comprehensive study on enhancing workability and predicting compressive strength using hybridized AI models. *Asian J. Civil Eng.* <https://doi.org/10.1007/s42107-023-00966-x> (2024).
86. Iftikhar, B. *et al.* Predictive modeling of compressive strength of sustainable rice husk ash concrete: Ensemble learner optimization and comparison. *J. Clean. Prod.* **348**, 131285 (2022).
87. Sarveghadi, M., Gandomi, A. H., Bolandi, H. & Alavi, A. H. Development of prediction models for shear strength of SFRCB using a machine learning approach. *Neural Comput. Appl.* **31**, 2085–2094. <https://doi.org/10.1007/s00521-015-1997-6> (2019).
88. Ilyas, I. *et al.* Advanced machine learning modeling approach for prediction of compressive strength of FRP confined concrete using multiphysics genetic expression programming. *Polymers (Basel)* **14**, 1789 (2022).
89. Li, P. *et al.* Sustainable use of chemically modified tyre rubber in concrete: Machine learning based novel predictive model. *Chem. Phys. Lett.* **793**, 139478 (2022).
90. Rostami, A. *et al.* Rigorous framework determining residual gas saturations during spontaneous and forced imbibition using gene expression programming. *J. Nat. Gas Sci. Eng.* **84**, 103644 (2020).
91. Despotovic, M., Nedic, V., Despotovic, D. & Cvetanovic, S. Evaluation of empirical models for predicting monthly mean horizontal diffuse solar radiation. *Renew. Sustain. Energy Rev.* **56**, 246–260 (2016).
92. Jalal, F. E., Xu, Y., Iqbal, M., Jamhiri, B. & Javed, M. F. Predicting the compaction characteristics of expansive soils using two genetic programming-based algorithms. *Transp. Geotech.* **30**, 100608 (2021).
93. Gandomi, A. H. & Roke, D. A. Assessment of artificial neural network and genetic programming as predictive tools. *Adv. Eng. Softw.* **88**, 63–72 (2015).
94. Asteris, P. G., Roussis, P. C. & Douvika, M. G. Feed-forward neural network prediction of the mechanical properties of sandcrete materials. *Sensors* **17**, 1344 (2017).
95. Ma, D. *et al.* Water–rock two-phase flow model for water inrush and instability of fault rocks during mine tunnelling. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00612-6> (2023).
96. Saberi, F. & Hosseini-Barzi, M. Effect of thermal maturation and organic matter content on oil shale fracturing. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-024-00666-0> (2024).
97. Inqiad, W. Estimation of 28-day compressive strength of self-compacting concrete using multi expression programming (MEP): An artificial intelligence approach †. *Eng. Proc.* **56**, 212 (2023).
98. Chu, H. H. *et al.* Sustainable use of fly-ash: Use of gene-expression programming (GEP) and multi-expression programming (MEP) for forecasting the compressive strength geopolymer concrete. *Ain Shams Eng. J.* **12**, 3603–3617 (2021).
99. Deng, W., He, P. & Huang, Z. Multi-expression based gene expression programming. In *Lecture Notes in Electrical Engineering* (eds Sun, Z. & Deng, Z.) (Springer, 2013).
100. Khan, M. *et al.* Forecasting the strength of graphene nanoparticles-reinforced cementitious composites using ensemble learning algorithms. *Results Eng.* **21**, 101837 (2024).
101. Luo, T. *et al.* Quantitative characterization of the brittleness of deep shales by integrating mineral content, elastic parameters, in situ stress conditions and logging analysis. *Int. J. Coal Sci. Technol.* **11**, 1–13 (2024).
102. Wang, L. *et al.* Effect of long reaction distance on gas composition from organic-rich shale pyrolysis under high-temperature steam environment. *Int. J. Coal Sci. Technol.* **11**, 1–18 (2024).
103. Nohara, Y., Matsumoto, K., Soejima, H. & Nakashima, N. Explanation of machine learning models using shapley additive explanation and application for real data in hospital. *Comput. Methods Prog. Biomed.* **214**, 106584 (2022).
104. Wang, S., Guo, J., Yu, Y., Shi, P. & Zhang, H. Quality evaluation of land reclamation in mining area based on remote sensing. *Int. J. Coal Sci. Technol.* **10**, 1–10 (2023).
105. Ekanayake, I. U., Meddage, D. P. P. & Rathnayake, U. A novel approach to explain the black-box nature of machine learning in compressive strength predictions of concrete using Shapley additive explanations (SHAP). *Case Stud. Constr. Materials* **16**, 01059 (2022).
106. Abbaslou, H., Ghanizadeh, A. R. & Amlashi, A. T. The compatibility of bentonite/sepiolite plastic concrete cut-off wall material. *Constr. Build. Mater.* **124**, 1165–1173 (2016).
107. Jiang, Y. *et al.* Mechanical properties and acoustic emission characteristics of soft rock with different water contents under dynamic disturbance. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-024-00682-0> (2024).
108. Ghanizadeh, A. R., Abbaslou, H., Amlashi, A. T. & Alidoust, P. Modeling of bentonite/sepiolite plastic concrete compressive strength using artificial neural network and support vector machine. *Front. Struct. Civil Eng.* **13**, 215–239 (2019).
109. Zhang, C., Wang, P., Wang, E., Chen, D. & Li, C. Characteristics of coal resources in China and statistical analysis and preventive measures for coal mine accidents. *Int. J. Coal Sci. Technol.* <https://doi.org/10.1007/s40789-023-00582-9> (2023).
110. Guan, Q. Effect of clay dosage on mechanical properties of plastic concrete. *Adv. Mater. Res.* <https://doi.org/10.4028/www.scientific.net/AMR.250-253.664> (2011).

Author contributions

Waleed Bin Inqiad: Conceptualization, Methodology, Software, Machine learning, Data Curation, Investigation, Validation, Writing—Original Draft, Writing—Review and Editing, Visualization Muhammad Faisal Javed: Conceptualization, Methodology, Validation, Investigation, Writing—Review and Editing, Supervision, Project administration, Funding acquisition, Resources. Muhammad Shahid Siddique: Methodology, Project administration, Funding acquisition, Resources. Kennedy Onyelowe: Project administration, Funding acquisition, Resources. Usama Asif: Conceptualization, Methodology, Software, Machine learning, Investigation, Validation, Writing—Original Draft, Writing—Review and Editing, Visualization. Fahid Aslam: Writing—Review and Editing, Supervision Loai Alkhattabi: Conceptualization, Funding acquisition, Resources.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-024-69271-0>.

Correspondence and requests for materials should be addressed to M.F.J. or K.O.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024