Machine learning regression and classification algorithms utilised for strength prediction of OPC/by-product materials improved soils.

Eyo E. U.[a]*, Abbey S. J.[a]

(*Corresponding author Email: eyo.eyo@uwe.ac.uk)

[a]*Faculty of Environment and Technology, Department of Geography and Environmental Management, Civil Engineering Cluster, University of the West of England, Bristol, United Kingdom.*

Abstract

In this study, stand-alone machine (ML) models (Bayesian regressor (BLR), least square linear regressor (REG), artificial neural networks (ANN), and logistic regression (LR)), tree-ensemble ML models (boosted decision tree (BDT), random decision forest (RDF) decision jungle (DJ)) and meta-ensemble ML models (voting (VE) and stacking (SE)) are applied to predict the strength of different soils improved by part-substitution of OPC with PFA and GGBS in various combinations and proportions. Multiclass elements of these proposed ML models are also deployed to provide analysis across multiple cross-validation methods. Results of regression analysis indicated higher statistical variance of OPC-substituted predictor variables compared to soils improved by OPC alone when using both stand-alone and tree-based algorithms. On average, the REG model produced strength predictions with higher accuracy (RMSE of 0.39 and R^2 of 0.86) compared to ANN (RMSE of 0.44 and R^2 of 0.82), but with comparatively lower accuracy compared to tree-based models (average RMSE of 0.33 and R^2 of 0.90) and meta-ensemble models (average RMSE of 0.06 and R^2 of 0.91). For ML classification, multiclass neural network algorithm (*m*ANN) produced higher accuracy (0.78), precision (0.67) and rate of recall (0.67) compared to tree-based models but fell short to meta-ensemble models (average accuracy of 0.80, precision of 0.70 and recall of 0.71). Diagnostic tests across different validation methods indicated better performance of the VE model compared to its SE ML counterpart when adopting the train-validation split technique. Overall, the ensemble methods were more versatile on regression and multiclass classification problems because they aggregated multiple learners to provide robust predictions.

Keywords: Machine learning; cement; PFA; GGBS; OPC; Bayesian regressor; linear regression; artificial neural networks; logistic regression; ensembles

Nomenclature

List of abbreviations

ANN	Artificial neural network
ASTM	American system of testing for materials
AUC	Area under curve
BDT	Boosted decision tree.
BLR	Bayesian linear regression
BP	Back propagation
CEM I	Cement
CRISP-DM	CRoss industry standard process data mining
CV	Cross validation
DAG	Directed acyclic graph.
DJ	Decision jungle
EML	Extreme machine learning
FN	Functional networks
FPR	False positive rate
GA	Genetic algorithm
GGBS	Ground granulated blast furnace slag.
K-FCV	k-fold cross validation
KNN	k-Nearest neighbours
LR	Logistic regression
MAE	Mean absolute error.
MARS	Multivariate adaptive regression splines
MCCV	Monte Carlo cross validation
MGP	Multi-genetic programming
ML	Machine learning
OPC	Ordinary Portland Cement
PFA	Pulverised fuel ash
PI	Plasticity index
RDF	Random decision forest
REG	Linear regression
RMSE	Root mean square error.
ROC	Receiver operating characteristic
SE	Stacking ensemble
SVM	Support vector machine
TPR	True positive rate
TVS	Train-validation split.
UCS	Unconfined compressive strength
VE	Voting ensemble

List of symbols

α_m	neural network activities
Y	predictor variable
Y_{n}	normalised UCS influential factor
Y_0	raw UCS influential factor
m_{y}	mean of distribution
σ_{y}	standard deviation
N	dataset points
n_v	size of validation dataset
n_t	size of training dataset
X_n	independent variable
β_0	regression constant
t_m	sample target value
ε_m	additive noise
w	weight vector
β	Bayesian precision parameter
α	hyper-parameter controlling distribution
$\mathbf{\Sigma}$	posterior variance
$\boldsymbol{\mu}$	mean of weights
w_{ij}	weight between two neurons
y_n	neural network output signal
x	activation of *nth* neuron
σ	neural network activation function
t	hypothesis test value
SE	standard error
X_m	mean of actual observations.
SS_{xx}	explained variation.

1. Introduction

The method of soil stabilisation that involves treatment with pozzolanic binders, continues to remain one of the most effective and economical means of ground improvement. The temptation to utilise Ordinary Portland Cement (CEM-I or OPC) as a traditional binder to stabilise weak soils seems nearly unavoidable in certain projects given their good hydraulic and binding qualities. However, the negative effects on the environment due to the continuous production and usage of OPC cannot be over-emphasized [1]. Hence, a complete replacement or part-substitution of OPC with relatively low-carbon secondary alternatives or by-products such as ground granulated blast furnace slag (GGBS) and pulverised fuel ash (PFA) has become inevitable in soil stabilisation [2–10].

Determination of the mechanical properties of stabilised soils based on some composite binder mixture parameters is undoubtedly an important first step towards the development of design mix guidelines for subsequent field application [11–13]. For a stabilised soil with multiple binder combinations, the challenges of establishing a property such as compressive strength may involve some time-consuming and laborious laboratory trial batching (soil-binder type quantities and optimum combinations), choice of curing duration, selection and testing of other related properties which can have the potential of affecting the target variable. Meanwhile, on the basis of theory, conventional models of forecasting the compressive strength of stabilised soils consist essentially of relationships that are developed empirically from statistical methods whereby, linear, and sometimes nonlinear regression techniques are applied [14,15]. The analytical equations generated through these models tend to determine unknown coefficients that affect the relationship of other variables and the compressive strength. These models, though effective in some cases, are riddled with shortcomings such as those associated with the complexities of the stabilised soil mentioned above.

In recent times, machine learning (ML) techniques have been introduced to compensate for the limitations of traditional methods of compressive strength prediction of soils [16,17]. However, the adoption of ML models for performance evaluation of improved ground properties has been very slow and only reported in few studies as follows: strength, dry density, moisture content additive content, resilient modulus modelling and prediction using artificial neural networks (ANN), support vector machines and regression (SVM & R), meta-ensembles (voting, stacking, tiering & bagging), functional networks (FN), multivariate adaptive regression splines (MARS), Logistic regression (LR), k-nearest neighbours (KNN), Genetic algorithm (GA), multi-genetic programming (MGP) [16–30]. These authors have used materials such as cement, lime, fly ash, fibres and geopolymers to strengthen the weak soils.

It is obvious that the application of multiple ML algorithms and a critical analysis that compares the relative performances of each one is not plentiful in literature. Moreover, an application of ML models to predict the unconfined compressive strength (UCS) of soils stabilised by partial substitution of OPC with cementitious by-products (PFA and GGBS) has not been done.

In this study, stand-alone ML models (Bayesian regression, linear regression, artificial neural networks, and logistic regression), tree-ensemble ML models (boosted decision tree, random decision forest and decision jungle) and meta-ensemble ML models (voting and stacking) are applied to investigate and predict the strength properties of five different soils stabilised by partial replacement of OPC with PFA and GGBS in various combinations and proportions.

Until recently, most ML predictions reported in literature have relied on conventional statistical metrics such as coefficient of determination and other standard error analyses for performance assessment [31–39]. However, this study aims to extend the scope of investigation, evaluation and prediction to include other diagnostic tests to support, confirm

and validate the commonly used statistical measures. This ensures that adequate sensitivity analysis of performance is accounted for while also emphasising the effect of weight variables or features in the prediction process.
Also, the complex combination of binders used in stabilisation of the soils considered in this study can easily be regarded as a ML classification problem. Hence, the multiclass elements of the proposed ML models are further deployed to provide analysis by considering multiple cross-validation methods, an aspect which has not been considered in most previous studies.
The structure and framework of this study are as follows: a statement of the method involving database construction, development and the experimental procedures adopted for soil stabilisation is presented section in 2. This shall consist of a series of steps and processes required for the preparation of collated datasets. Subsequent development and implementation of the proposed models are given in section 3. Detailed analyses and discussions of the performance of ML models are carried out in section 4. This section includes an evaluation of ML regression, classification, and sensitivity analysis of the prediction problems. In section 5, the significance of this study and recommendations for ML model deployment are laid out. Following this is the concluding section where the main points and highlights of the study are given.

2. Research Method

2.1. Database development and stabilisation procedure

High quality and original experimental data of unconfined compressive strength (UCS) tests on soils stabilised using OPC and a combination of cementitious by-product materials were compiled from literature and used to train ML models [14]. Soil-binder-water reactions under a prescribed or natural curing environment can play a very significant role in the hardening rate of the stabilised mixed. In this regard, 5 different soil types of varying initial properties were improved by OPC, a blend of OPC and PFA and OPC-PFA-GGBS. OPC contents of 5%, 10%, 15% and 20% (by weight of dry soil) were applied to stabilise the weak soils. The OPC-PFA-stabilised soils were first composed of a 50% reduction in the OPC content used. The OPC was then further reduced to 33.33% and substituted by equal amounts of PFA and GGBS to produce a stabilised soil of OPC-PFA-GGBS mixes. In all the soil mixtures, the total proportion of the binder in the stabilised soil remained constant at 5%, 10%, 15% and 20%. The ratio of water-OPC in the stabilised soil mixture was unity. Three series of UCS tests (ASTM D2166) were carried out following curing at 7, 14, 28 and 56 days to assess strength developments of the soil with varying initial moisture and plasticity properties. The first series was performed to study the influence of OPC addition alone on UCS while the second and third series of tests were carried out to investigate the effect of OPC-PFA and OPC-PFA-GGBS addition respectively. Compared to the natural soils, there was an improvement in the UCS of the soils when stabilised by the binders details of which are given in Abbey et al. [14]. However, in keeping with the main goal of this research, the datasets will be used to train ML models for UCS prediction.

2.2.Featurization and hyper-parameter optimisation

Stabilised soil datasets used in the supervised training of ML models in this study did not contain any missing feature and attributes. Nevertheless, there was need for other forms of feature engineering to be carried out to enable a reduction in unnecessary redundancy and improvement of the integrity of data used for training of the ML models.

2.2.1. Data normalization

Without any distortion of the differences in the range of values of the UCS dataset, these values were transformed into a common scale by using the Z-score standardisation method to ensure outliers were avoided. The Z-score transformation is described mathematically as:

$$Y_n = \frac{Y_0 - m_y}{\sigma_y} \quad (1)$$

Where Y_n and Y_0 represent the normalised and raw UCS influential factors respectively, while m_y and σ_y denote corresponding values of the mean and standard deviation, respectively.

2.2.2. Cross-validation (CV)

For average sized datasets such as those used in this study, it was necessary to apply cross-validation techniques to improve the reliability of the training sets and reduce the chances of certain coincidental features receiving more importance [40]. Moreover, an overfitted model is highly undesirable since it lessens the predictive performance on some ‘‘unseen” tested data [41]. The following cross-validation techniques were used to optimise hyper-parameters: *k*-fold cross-validation (*k*-FCV), Monte Carlo cross-validation (MCCV) and train-validation split (T-VS) method.

The *k*-fold cross-validation (*k*-FCV) technique tends to divide the dataset (N) points into some *k*- subsets of equal sizes. The process then treats one of the *k*-subsets as a training subset and the remaining as validation subset. This process then repeats *k* number of times by excluding one of the *k*- subsets in each cycle. In this study, 10-fold CV was adopted.

Monte Carlo cross-validation (MCCV) method splits the dataset (N) points into two subsets by sampling and without replacement of one of the data points. The training is then performed on the subset that was not replaced and validation on the replaced subset. Even though there exist a rather unique training set, MCCV tends to avoid the need to run any form of iterations unlike the *k*-FCV.

In summary, if we consider both MCCV and *k*-FCV and then assume N to represent the size of the dataset, *k* denoting the number of *k*-fold subsets, n_v the size of validation set, and n_t the size of training set then:

$$k\text{-FCV, N} = k \text{ x } n_v \quad (2)$$

$$\text{MCCV, N} = n_t + n_v. \quad (3)$$

Train-validation split (T-VS) is a simple method of randomised dataset splitting whereby each of the subsets are used for training and testing purposes, respectively. In this research, 80% of the parent dataset were used to meticulously train the ML models (i.e., selection and optimisation of hyper-parameters and functions) while the remaining 20% of the dataset were used to test and assess the prediction performance of the ML models. Previous studies have recommended that the testing data subset may not be less than 10% nor more than 30% of the entire data records [32,40].

3. Machine learning models.

3.1. Multiple linear regression (REG)

This is one of the most common mathematical methods employed for supervised ML. The least square function of REG technique establishes several correlations between one or more independent or explanatory variables and predictor or dependent variables. Changes in the predictor variable say, *Y* are often triggered by the nature of the independent variable *X* as the following general equation shows:

$$Y_n = \beta_0 + \beta_1 X_{1n} + \beta_2 X_{2n} + \cdots + \beta_m X_{mn} + \epsilon_n \quad (4)$$

Where Y_n = predictor variable; $X_{1n}, X_{2n,\ldots}, X_{mn}$ = independent variables; β_0 = constant; and $\beta_1, \beta_{2,\ldots}, \beta_m$ = coefficients of regression; and ε = error term.

3.2. Logistic regressor (LR)

This is a nonlinear model where the deviation or variance of the predictor variable is a function of its mean. In other words, unlike REG, the value of the predictor variable depends on the probability that it belongs to a certain class. LR tends to add an exponential function on top of REG in order to restrain the predictor of response $Y_n \in [0,1]$, instead of $Y_n \in \Re$ as in REG [16]. The LR model for say 'y' distinct predictors could be represented as:

$$y(X) = P_r(Y = 1|X) = \frac{e^{\beta_0+\beta_1 X_{1n}+\beta_2 X_{2n}+\cdots+\beta_m X_{mn}+\epsilon_n}}{1+ e^{\beta_0+\beta_1 X_{1n}+\beta_2 X_{2n}+\cdots+\beta_m X_{mn}+\epsilon_n}} \quad (5)$$

Where Pr (Y=1|X) = probability that the response Y = 1 or 0 given $X_{1n}, X_{2n,\ldots}, X_{mn}$ = independent variables, $X_{1n}, X_{2n,\ldots}, X_{mn}$ = independent variables; β_0 = constant; and $\beta_1, \beta_{2,\ldots}, \beta_m$ = coefficients of regression; and ε = error term. Because of its extreme validity in classification as well as regression problems, LR unlike its REG counterpart is mostly preferred as the default first option [40].

3.3. Bayesian linear Regressor (BLR)

This is a special case of linear regression whereby the model analysis is undertaken within the context of a statistical inference of the "Bayes" theorem. This is then used to update the probability of a given hypothesis as more information or evidence become available. Bayes theorem describes the probability of an event taking place as a result of having prior knowledge of certain conditions that might be related to such event. If it is considered that a target value say, t_m, is sampled from an experiment then the relationship between this value and the predictor variable $y(x_m;w)$ can be given as [42]:

$$t_m = y(\boldsymbol{x}_m; \boldsymbol{w}) + \varepsilon_m \quad (6)$$

Where ε_m = an additive noise (modelled as Gaussian distribution $\varepsilon \sim \mathcal{N}(0,\beta^{-1})$ with a random zero-mean variable); w = weight vector; β = precision parameter

Hence, following Gaussian distribution: we have the target value t_m as:

$$p(t_m|\boldsymbol{w},\beta) = \mathcal{N}(t_m|y(\boldsymbol{x}_m;\boldsymbol{w}),\beta^{-1}) \quad (7)$$

With the input parameter given as $\boldsymbol{x}$, the likelihood or probability for the target vector $\mathbf{t}$ then:

$$p(t_m|\boldsymbol{w},\beta) = \prod_{m-1}^{K}\mathcal{N}(t_m|y(\boldsymbol{x}_m;\boldsymbol{w}),\beta^{-1}) = \left(\frac{\beta}{2\pi}\right)^{N/2} exp\left\{-\frac{\beta}{2}||\boldsymbol{t}-\Phi\boldsymbol{w}||^2\right\} \quad (8)$$

Where $\boldsymbol{x} = (x_1, x_2, x_3, \ldots, x_N)^T$; $\boldsymbol{t} = (t_1, t_2, t_3, \ldots t_N)^T$.

To prevent any model complexity and over-fitting of the maximum likelihood directed against $\boldsymbol{w}$, a prior distribution is then defined as follows:

$$p(\boldsymbol{w}|\alpha) = \prod_{m-1}^{L}\mathcal{N}(w_m|0,\alpha^{-1}) \quad (9)$$

Where α = hyper-parameter controlling the distribution of w_m, $p(w_m|\alpha) = \mathcal{N}(w_m|0,\alpha^{-1})$.
the posterior distribution over $\boldsymbol{w}$ can also be obtained as follows:

$$p(\boldsymbol{w}|\boldsymbol{t},\alpha,\beta) = \frac{p(\boldsymbol{t}|\boldsymbol{w},\beta)p(\boldsymbol{w}|\alpha)}{p(\boldsymbol{t}|\alpha,\beta)} = \mathcal{N}(\boldsymbol{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}) \quad (10)$$

Where $p(t|\alpha, \beta)$ = normalising factor; $\boldsymbol{\Sigma}$ = posterior variance; $\boldsymbol{\mu}$ = mean of weights

3.4. Tree-ensembles

Tree-based models or ensemble of decision trees (Fig. 1) are a ML paradigm whereby formal rules are obtained from detected patterns in the datasets hence, the tree-based models must be trained in a rigorous manner on the data in order to be able to predict the properties presented by a query [43]. Depending on the application, differences exist of how the tree-based ML models are built and for the purposes of his study the random decision forest (RDF), boosted decision trees (BDT) and decision jungles (DJ) shall be considered.

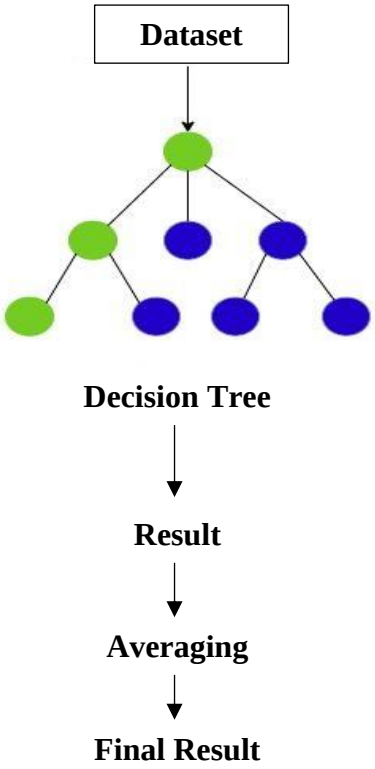


Fig. 1. Structure of a single regression (decision) tree.

3.4.1. Random decision forest (RDF)

This is a decision ensemble that is created to reduce the instability or fluctuations of single regression trees. It utilises the "bootstrap aggregation" (bagging) concept to generate various "bootstrap aggregation" (bagging) concept to generate various similar data records that are hitherto sampled from the same parent source. Bagging is simply a technique of aggregating a multiple tree models in a 'bag' for data training [37]. One of the disadvantages RDF is that it tends to fall victim of overfitting due to its small biases and wide variance.

3.4.2. Boosted decision trees (BDT)

Like RDF, these are an ensemble means of solving the problems of instability and poor performance of a single regression tree. In general, the idea of "boosting" is a strategy that is used to improve the performance of weaker learning regression tree algorithms. The step of boosted model building is often repeated through a set of iterations. Unlike RDF where all the trees are of equal importance, the BDT are rather hierarchical, and each tree layer is created recursively [41]. BDT tend to possess high performance ability especially on nonlinear datasets. However, one of the disadvantages of BDT is that its interpretability capacity is low and that makes it difficult to gain sufficient intuition of the patterns learned by the model.

3.4.3. *Decision jungles (DJ)*

These are a somewhat recent extension to RDF. A DJ is composed of an ensemble of rooted decision directed acyclic graphs (DAGs) as a method to obtain compact and accurate ML classifiers [44]. By permitting the merging of trees, a decision DAG traditionally has low memory footprint and are great at generalisation performance. DJs have the advantage that they are non-parametric ML models that can represent nonlinear decision boundaries. They are capable of selecting integrated features and performing classifications while also being very resilient to noisy features.

3.5. *Artificial Neural Network (ANN)*

Just like the tree-ensemble ML models, ANN will have to be trained on a dataset to be able to predict the properties of a presented query. ANN are a family of data-processing ML models that are basically inspired by the human brain or neural networks whereby the neurons are interconnected through synapses (Fig. 2a) [45]. This network of neurons receive inputs, processes them and then make decisions or predictions. The neuron which is the processing element has the ability of filtering functions to make sure that inputted data to a specific node does not affect the network. The neuron also has an adaptive learning capability to adjust the weights that are connected between the nodes. ANN has a basic input layer, a hidden layer, and an output layer (Fig. 2b). When a processing neuron or element provides an input to another unit, the output is received as an input by the successive processing unit. This interconnectedness can be mathematically expressed as:

$$\alpha_m = \sigma\left(\sum_j \omega_{ij} y_j\right), \quad \sigma(x) = \frac{1}{1-e^{-x}} \tag{11}$$

Where α_m = ANN activities; w_{ij} = weight between two neurons; y_n = output signal; x = activation of *nth* neuron; $\sigma(x)$ = activation function facilitating input transformation to output by multiplication of the inputs from the processing neuron by corresponding weights.

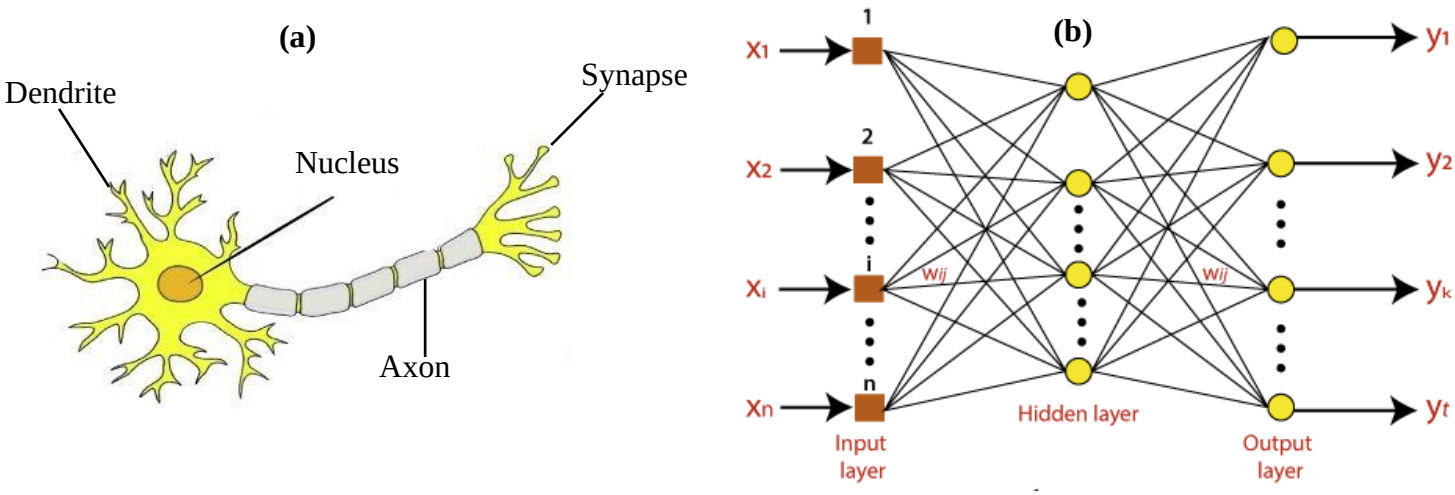


Fig. 2. Neural network architecture (a) human neuron (b) artificial neural network (ANN).

3.6. Meta-ensembles

Compared to stand-alone and tree-based models stated in the forgoing, meta-ensembles (or model of models) are used to further improve the accuracy of prediction by combining some of the above-mentioned models. Both voting and stacking meta-ensembles are proposed for this study.

3.6.1. Voting (VE)

When considering a regression problem, VE calculates the average predictions from the combined models. Meanwhile, for classification problems, predictions for each class label are added and the label that has a majority vote is predicted [24]. Fig. 3a depicts the VE ML model whereby the mean inputs are obtained from a combination of several models as the value of prediction.

3.6.2. Stacking (SE)

This is an extension of the VE whereby the ML model learns how much and when to rely on each model to make generalised multistage predictions. The result of predictions obtained from say previous combined models ($X_{m=1\sim j}$) serves as input Y of the next stage as further predictions are being made (X_{pred}) as shown in Fig. 3b [24,46].

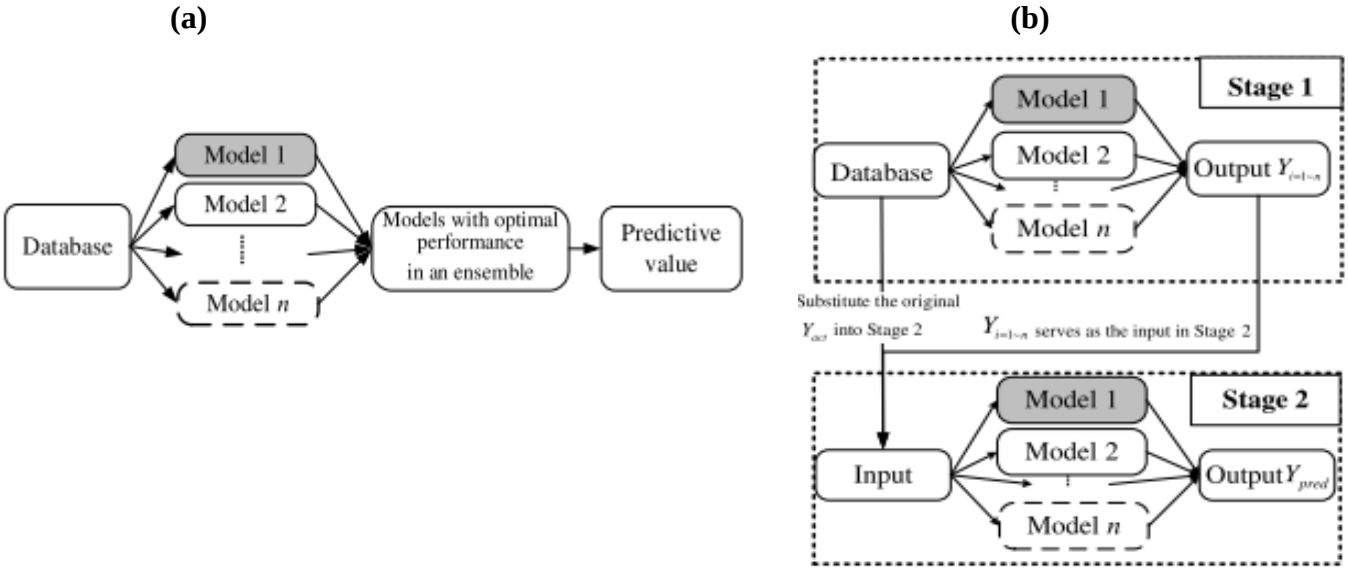
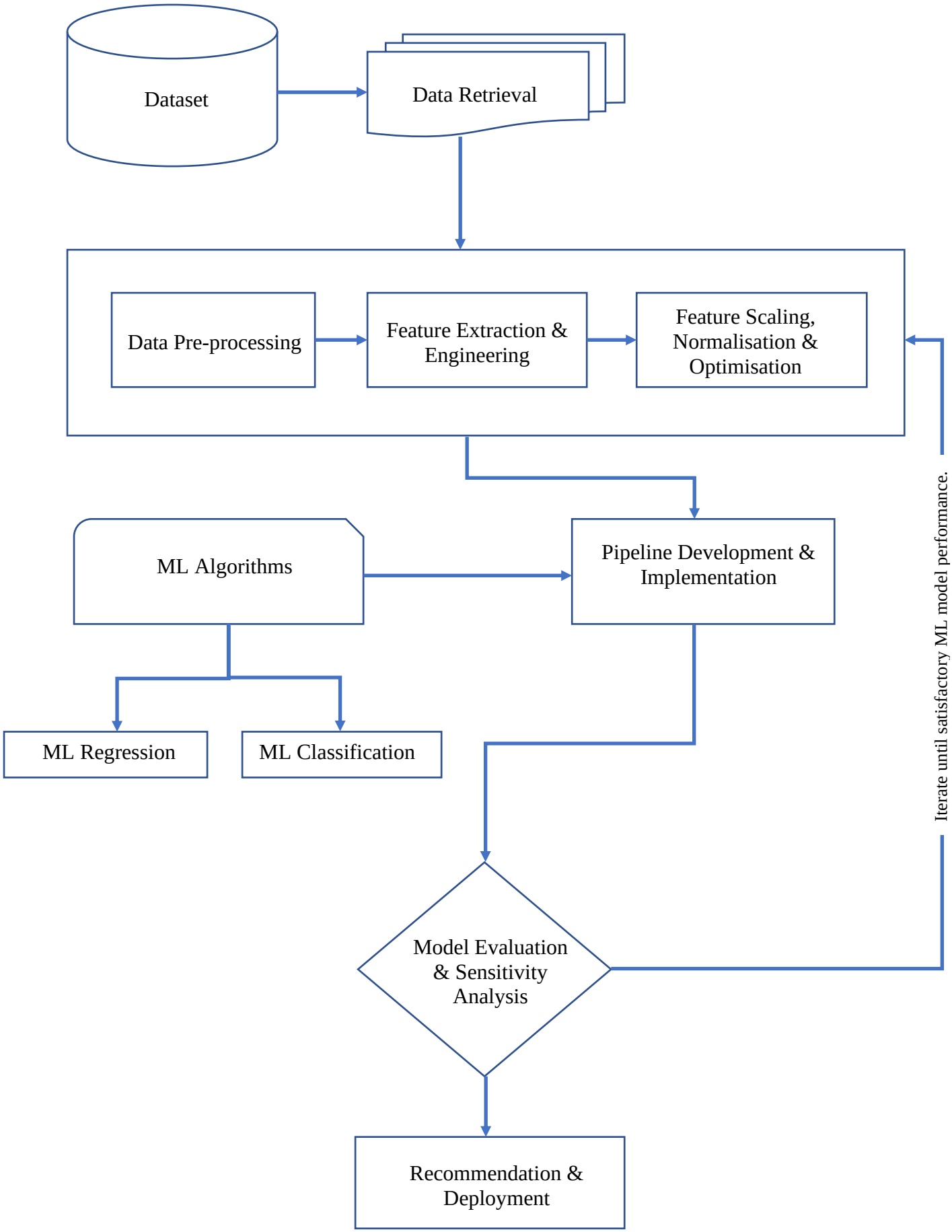


Fig. 3. Structure of meta-ensemble models (a) voting ensemble (b) stacking ensemble.

3.7. ML model pipeline development and implementation

Implementation of the algorithms was carried out on a designer platform which supports Python programming language (that includes the numpy, scipy and scikit-learn libraries) and ML pipeline developments. Considering the nature of the dataset used in this research, both

ML regression and classification were conducted to investigate and evaluate the performance of the algorithms. Important features and parameters used in the ML models and datasets respectively are given in Table 1 and Table 2. Fig. 4 depicts the flowchart of the methodology followed in developing the desired ML pipeline and subsequent evaluation of the models in this study.



338

339 **Fig. 4.** Machine learning methodology flowchart.

Table 1

Parameter settings of ML models

Model	Parameter	Option/value
REG	Method of solution	Ordinary least squares
	L2 regularization weight	0.001
LR	Trainer mode	Single parameter
	L1 regularization weight	1
	L2 regularization weight	1
BLR	Regularization weight	1
RDF	Resampling method	Bagging
	Trainer mode	Single parameter
	Number of decision trees	8
	Max. decision tree depth	32
	Number of random splits per node	128
	Min. sample no. per leaf node	1
BDT	Trainer mode	Single parameter
	Max. no of leaves per tree	20
	Training instances to form a tree	10
	Rate of learning	0.2
	Number of trees constructed	100
DJ	Resampling method	Bagging
	Trainer mode	Single parameter
	No. of DAGs	8
	DAGs max. depth	32
	DAGs max. width	128
	No. of optimization steps per DAG layer	2048
ANN	Hidden layer spec.	Fully connected
	Number of hidden nodes	100
	Rate of learning	0.005
	Number of learning iterations	100
	Initial learning weight diameter	0.1
	Normaliser	min-max

Table 2
Data features and attributes

Feature	Attributes	Data type
Binder combinations	C C-PFA C-GGBS-PFA	String
Duration	7 14 28 56	Integer
Soil type	Soil 1 Soil 2 Soil 3 Soil 4 Soil 5	String
Binder quantity	5 10 15 20	Integer
Plasticity Index	PI	Integer

3.8. Performance evaluation of ML models

For an examination and evaluation of the precision of prediction and subsequent performance of the ML models studied in this research, three indicators are considered namely, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and coefficient of determination (R^2). Detailed mathematical formulae of these performance criteria are given in literature [31,47]. Further assessment of the robustness and integrity of the regression models are considered by considering their prediction intervals. Prediction interval (**Eq. 12**) (unlike confidence interval) defines a range that may have a likelihood of containing the value of a dependent variable for a single future observation given some specific values of the independent variables.

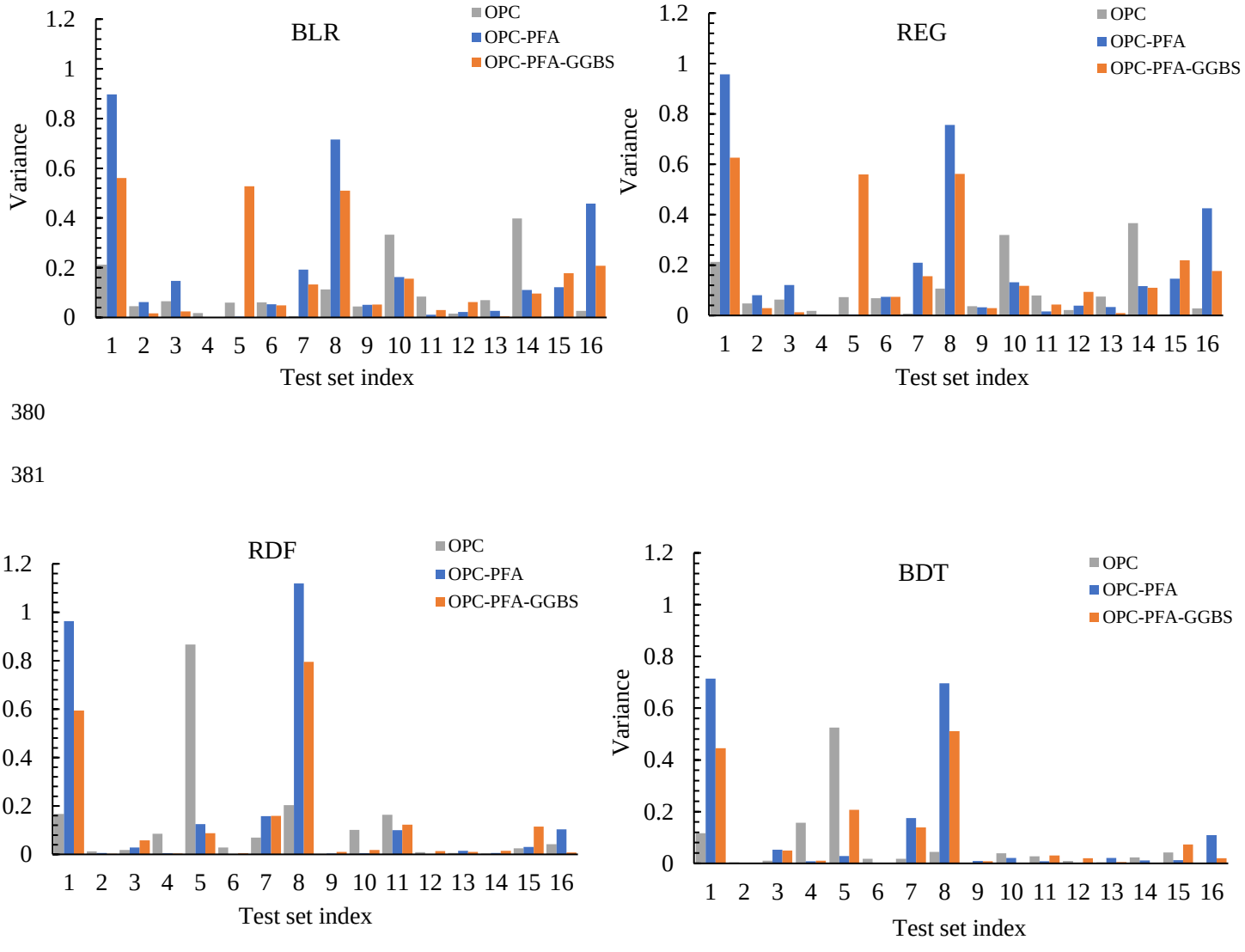
$$X = Y \pm t_{\alpha} SE \sqrt{1 + \frac{1}{n} + \frac{(X - X_m)^2}{SS_{xx}}} \quad (12)$$

Where Y = predicted values; t = hypothesis test value based on the percentage confidence; SE = standard error; n = size of dataset or number of observations; X_m = mean of actual observations ; SS_{xx} = explained variation.

4. Results and discussion

The ML models (stand-alone algorithms – REG, BLR, ANN; tree-based algorithms – BDT and RDF and the meta-ensemble models, VE and SE) proposed for regression analysis will be considered first in this section. Subsequently, analyses of classification problems using the multiclass ML models (LR, ANN, RDF, DJ and the meta-ensembles) will be given.

Fig. 5 compares the statistical variance (from the mean of the target variable) of each predictor components (OPC, OPC-PFA and OPC-PFA-GGBS) for ML test data sets. Generally, across the algorithms tested on, higher deviations are experienced by both the OPC-PFA and OPC-PFA-GGBS predictor variables compared to the soil stabilised by OPC alone. However, compared to the other models, RDF seems to register the highest possible variance (about 0.87) followed by BDT (about 0.53) for the soil stabilised using only OPC. Further examination of the performance of the ML models and an analysis of the independent variables are given in the following sections.



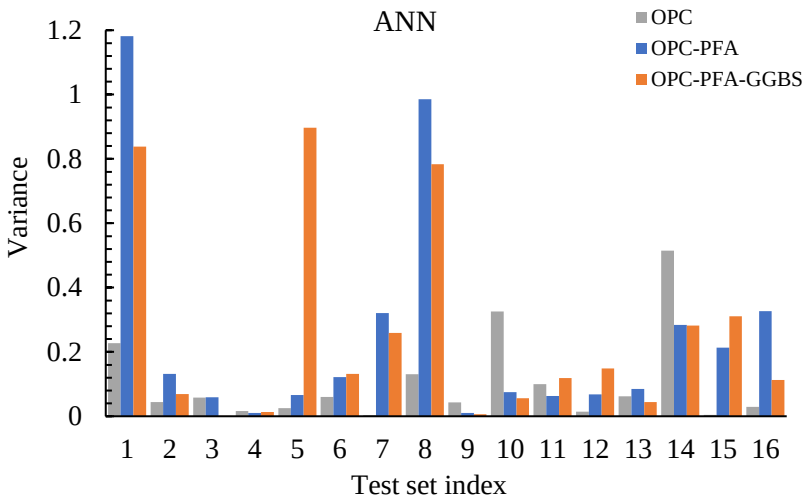


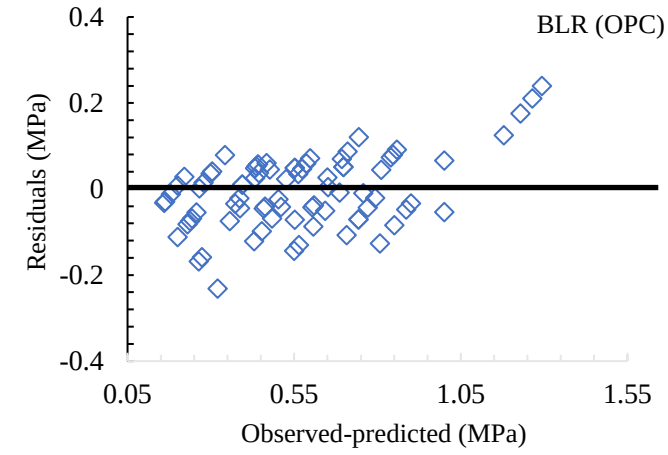
Fig. 5. Statistical variance of predictor components of improved soils (a) BLR (b) REG (c) RDF (d) BDT (e) ANN

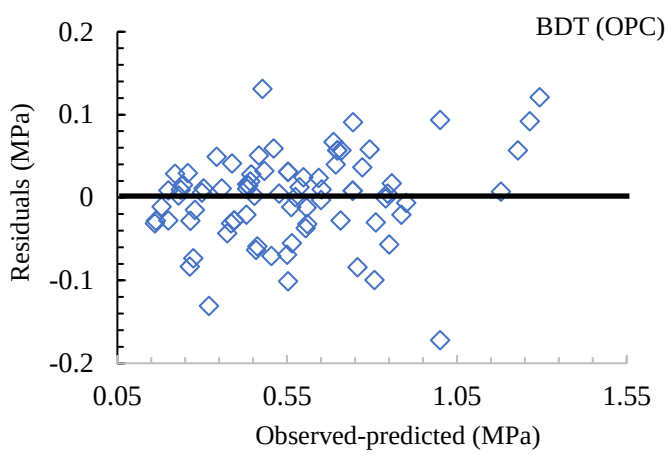
4.1. ML Regression

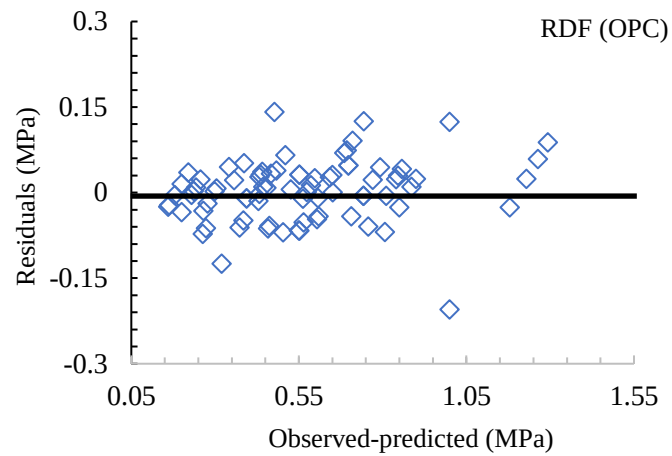
4.1.1. Quality Assessment of ML regression models

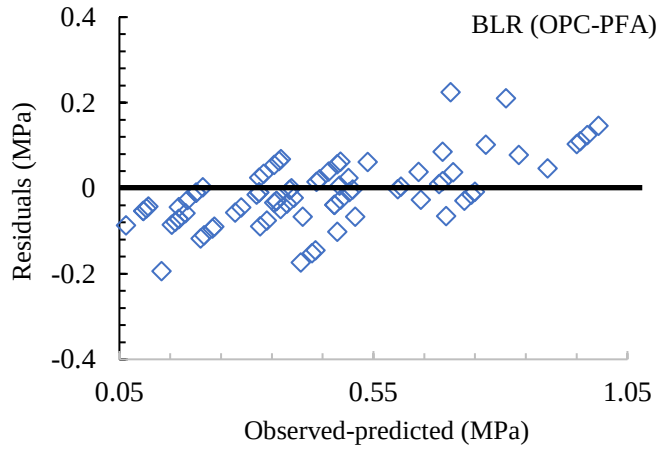
Residual lag plot provides the basis for an evaluation of the quality of the algorithms used to perform the regression analysis in this research. Furthermore, it does allow for an examination of any underlying statistical assumptions especially when considering the independence of features or variables and normality of distribution [48]. In order for any assumptions to hold true for a given regression model, the residuals will have to be distributed randomly around zero [49]. For a good model, the residual's scatter plot will show a disorderly pattern of the data hence, without indicating any form of trends. In other words, if there are any forms of trends observed in the data, this will indicate that the residuals are not entirely independent.

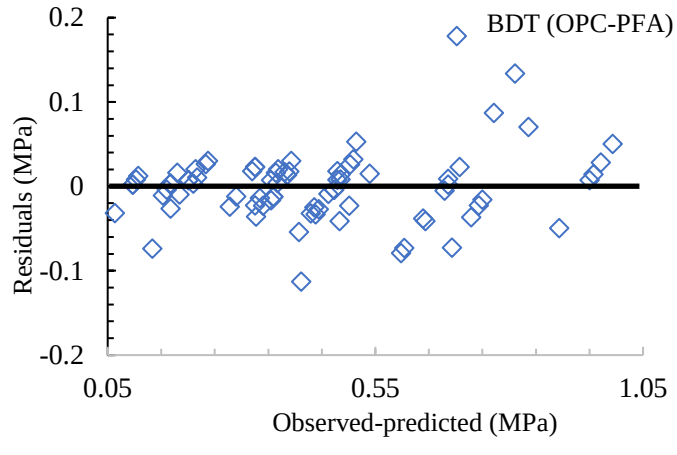
While performing feature engineering as discussed previously, it is important to note that the data used in the regression modelling were first normalised before being validated (train-validation split method) in order to prevent overfitting and imbalance. From Fig. 6, it is observed that the models used to perform the regression analysis all seem to indicate some measure of distribution about zero. However, a closer examination shows that the tree-ensemble and meta-ensembles all appear to exhibit more scatter and rather better randomness in the positioning of the data compared to the stand-alone models. Also, much better independence of error terms is exhibited by the tree-based and meta-ensembles. However, in terms of the features or the dependent variables used for the improvement of the soils, it is observed across the models that there is little or no difference in the degree of randomness of the 3 different combinations of the binders used (Fig. 6).

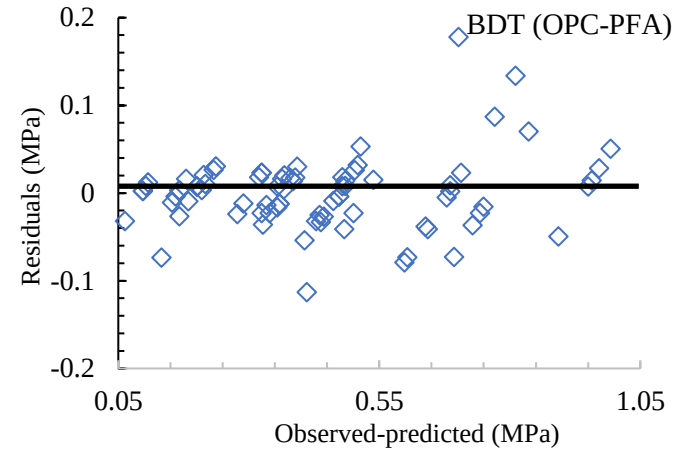

BLR (OPC)
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05
1.55

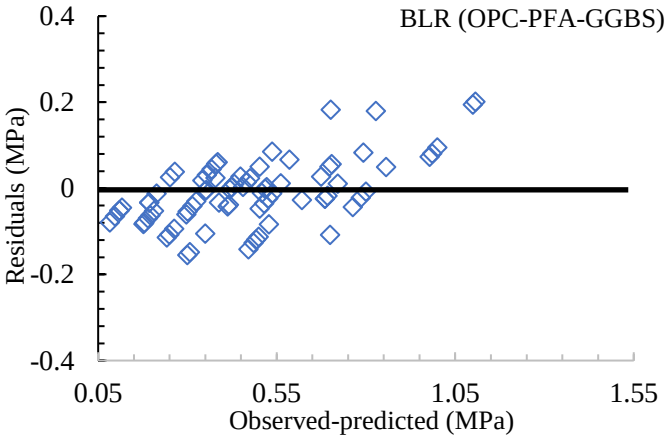

BDT (OPC)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.05
0.55
1.05
1.55

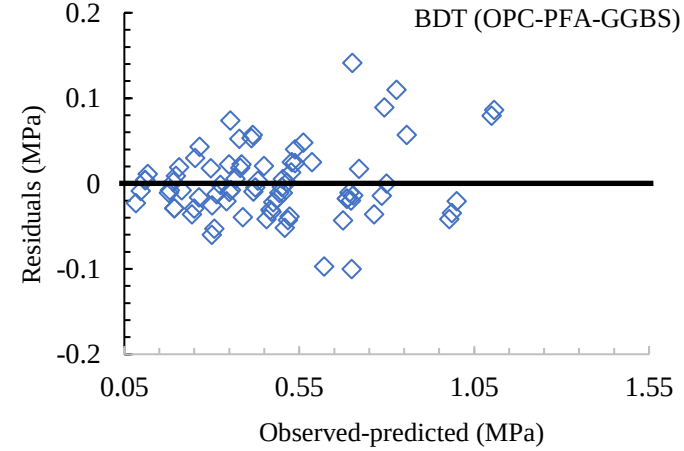

RDF (OPC)
Residuals (MPa)
Observed-predicted (MPa)
0.3
0.15
0
-0.15
-0.3
0.05
0.55
1.05
1.55

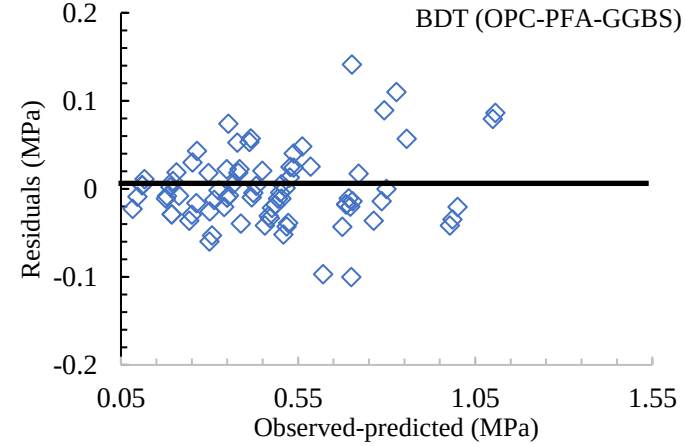

BLR (OPC-PFA)
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05

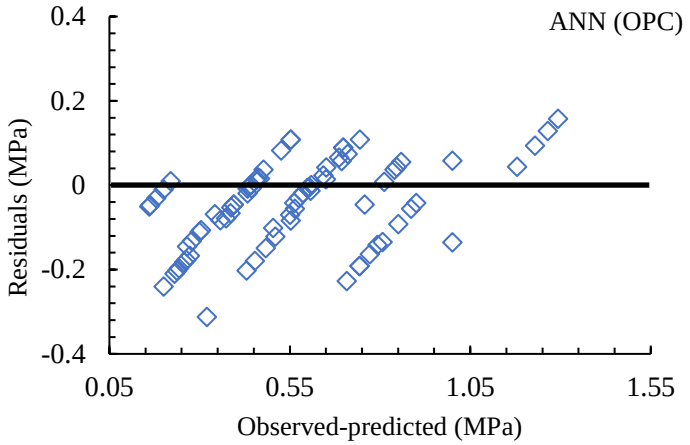

BDT (OPC-PFA)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.05
0.55
1.05

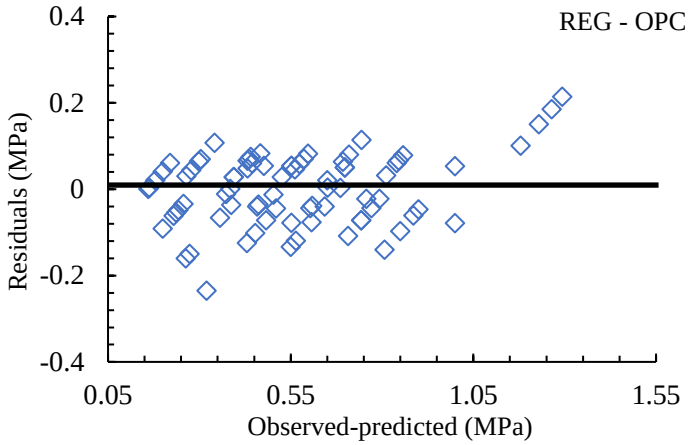

BDT (OPC-PFA)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.05
0.55
1.05

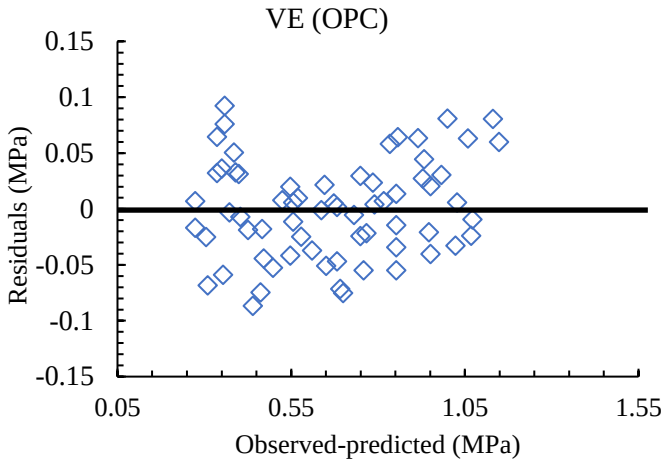

BLR (OPC-PFA-GGBS)
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05
1.55

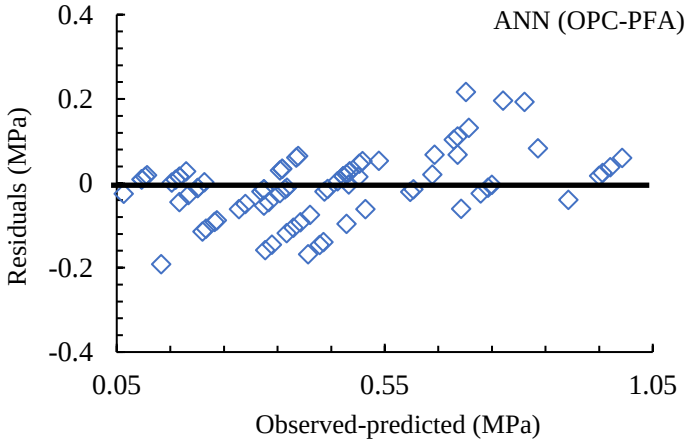

BDT (OPC-PFA-GGBS)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.05
0.55
1.05
1.55

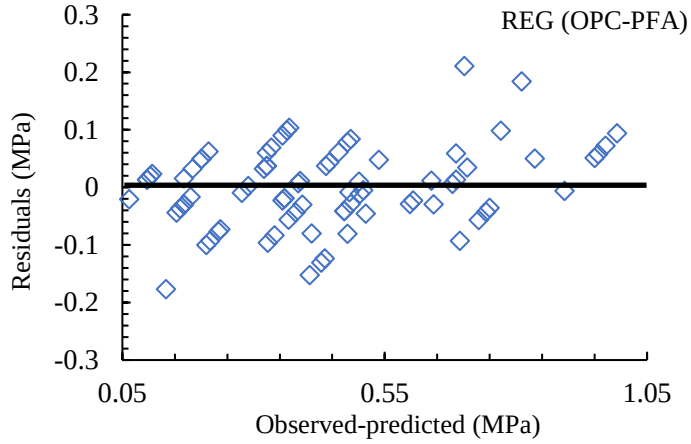

BDT (OPC-PFA-GGBS)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.05
0.55
1.05
1.55

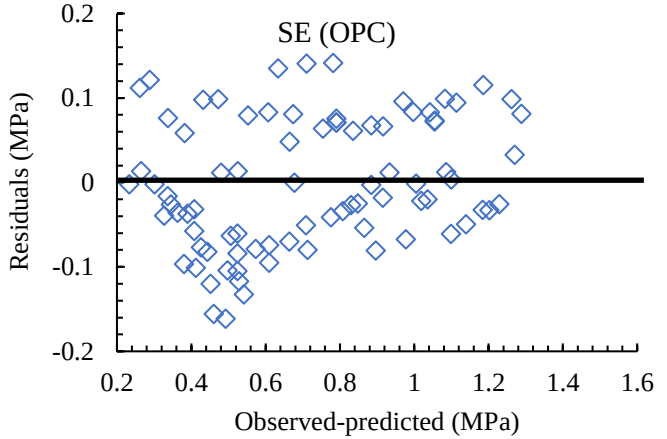

ANN (OPC)
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05
1.55

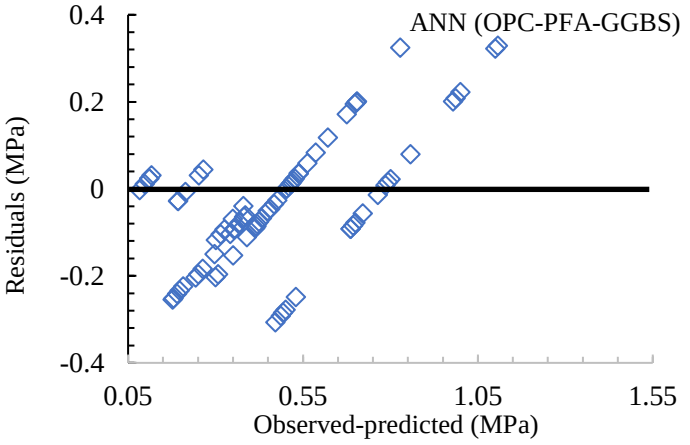

REG - OPC
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05
1.55


VE (OPC)
Residuals (MPa)
Observed-predicted (MPa)
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0.05
0.55
1.05
1.55


ANN (OPC-PFA)
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05


REG (OPC-PFA)
Residuals (MPa)
Observed-predicted (MPa)
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
0.05
0.55
1.05


SE (OPC)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6


ANN (OPC-PFA-GGBS)
Residuals (MPa)
Observed-predicted (MPa)
0.4
0.2
0
-0.2
-0.4
0.05
0.55
1.05
1.55

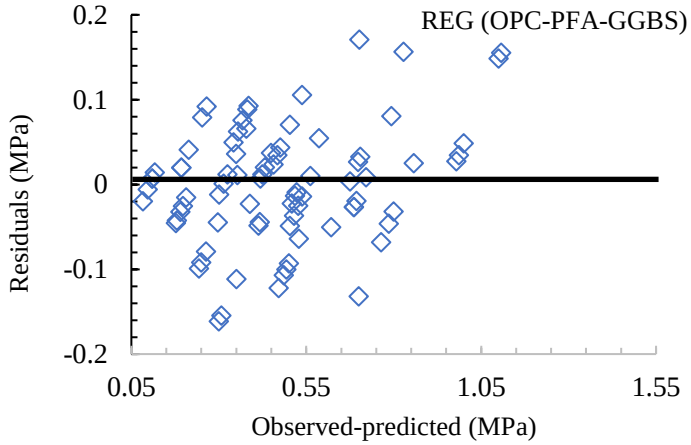

REG (OPC-PFA-GGBS)
Residuals (MPa)
Observed-predicted (MPa)
0.2
0.1
0
-0.1
-0.2
0.05
0.55
1.05
1.55

Fig. 6. Residual lag plots on trained dataset of ML models

4.1.2. Performance forecast

Indicators of the ML models' predictive performance are presented in Table 3 however, during the discussions herein, more attention will be given to the RMSE and R^2 metrics all of which are highlighted (bold-face fonts) in Table 3. It is observed that all ML algorithms provided predictions with some measured degrees of accuracy. The coefficient of determination ranges from approximately 0.78 to 0.96, the RMSE varies from as low as 0.02 kPa to a high of 0.50 kPa. Interestingly, it could be observed that the REG model gives predictions of the mixed soil's UCS with higher accuracy compared to the ANN, but with broadly lower accuracy compared to the BLR, RDF and BDT. The later outcome may not be entirely surprising with the REG model given that the data may not have fitted this algorithm's underlying assumptions as they would with the BLR and the tree-ensemble models (RDF and BDT). This behaviour was previously highlighted from the non-normality of the residual lag plots showing that this model may be incapable of approximating some unobserved phenomena of the mixed soil materials. It is pertinent to state here that the R^2 values obtained using the REG model are quite comparable to those of a previous study which relied on similar dataset for its prediction [14]. Albeit the R^2 values of Table 3 are only slightly lower because unlike the methodology of prediction adopted in the said previous study where all the datasets were used in the training of the model, 80% and 20% of the overall dataset were set aside and used in the training and testing of the REG model respectively in this research following the train-validation split method. Of particular interest, is the relatively worst prediction performance exhibited by the ANN model even though several previous studies have indicated that this algorithm can predict the strength of stabilised soils with reasonably high accuracy [24,27]. However, the ANN's inferior performance could be explained by also leveraging some of the theories advanced in prior studies [45,50], some of which are relevant to this research.

One major drawback of ANN is that the process of training is performed by relying on a search and optimisation algorithm (Levenberg-Marquardt or gradient descent) to constantly update its weights and biases over an error space that includes or that converges to local minima instead of a more global one, an approach regarded as backpropagation (BP) [32]. Low performances in strength prediction that is sometimes demonstrated by ANN could be mitigated by using Extreme ML (EML) algorithms whose training process also involves single hidden-layer feed-forward mechanism [50]. In other words, ELMs provide even more simplicity given that stopping criteria and learning rates (as given in Table 1 for ANN) may not have to be taken into consideration. But ELMs must also be used with caution because such models could require many more hidden neurons due to the random need to determine input weights and biases [51,52]. Moreover, excessive number of hidden layers of neurons (or black-boxes) could lead to overfitting meaning that the complex nature of a stabilised soil's mechanical property can be overestimated by feed-forward mechanisms like the ANN [53].

In order to rectify or overcome some of the deficiencies mentioned above, the tree-based ensemble ML algorithms could be used. As depicted in Table 3, these models seem to produce relatively much better predictions. The RDF has R^2 of 0.89 and RMSE values of 0.34 and 0.35 for soil stabilised by OPC alone and OPC substituted by PFA and GGBS respectively. Hence a slightly low prediction accuracy is given for the stabilised soil using a combination of OPC and PFA. A further explanation and appreciation of the difference in class predictions are given in subsequent sections in ML classification. On the other hand, BDT seems to produce the most superior prediction performance of both the stand-alone and

tree-ensemble models with the highest R^2 of 0.94 and a corresponding RMSE of 0.19 for the soil stabilised by using OPC alone. Unlike the RDF, the "boosting" strategy aided in an improved performance of weaker regression tree learner algorithms. Also, it may be suggested that the RDF performed slightly below the capacity of the BDT because for RDF, all trees are of equal importance, hence it has the potential of being subjected to overfitting given its minimum biases and wider variance.

It may not be completely unexpected that the tree-ensemble methods (BDT and RDF) have provided higher degrees of accuracy and performance compared to the stand-alone models in terms of the statistical metrics used in their assessment. The superior accuracy obtained from the tree-based models are mainly attributed to their structure and architecture. The tree-based ensemble methods are simply an aggregation or composition of single regression trees. This combination or boosting of trees are needed because low predictions and overfitting could be down to instability of a single regression tree when used alone [41]. Like the ANN models, the formal rules needed for training and subsequent predictions by tree-ensembles are learnt from patterns in the data. However, for the tree-ensembles, a series of tests or training are required to be performed on the data in order to logically partition them and by so doing, inconsistent variable features are learnt. To put it differently, the predictor variables are so repeatedly partitioned such that each successive final partition generates different sets of output value. Moreover, without having to smoothen or prune so-called "deep" trees, generalisation errors are thus also reduced, and overfitting mitigated [32].

It may be possible that lower performance is achieved in tree-ensemble models. In this case, the ensemble may have learnt from some interference due to the noise that ensues from the residuals rather than signals that emanate from within the data [41]. This phenomenon is attributed to a somewhat "greedy" construction process whereby at each step, an aggregation of single best performing variable and optimal point of split is selected which invariably means that there may also be a multi-step lookahead which takes into account variable combinations with even better results. Another drawback which Dreiseitl and Ohno-Machado [54] appear to agree with may have been the loss of information in the process resulting from continuous discretisation of variables by the splitting process,

As demonstrated in this research, such weakness could be slightly reduced by creating a further ensemble of models – meta-ensemble models (or model of models) through VE and SE techniques. This method was used to aggregate other models (stand-alone and tree-ensembles) with classifiers to enable an optimisation of the overall machine learning predictive performance. The statistical performance metrics from the meta-ensembles were obtained by calculating the weighted average of predictions from the combined models.

From Table 3, it is observed that the RMSE of the meta-ensemble models (VE and SE) fluctuates between approximately 0.04 and 0.09. Hence, compared to the stand-alone and tree-based models, RMSE values for the meta-ensemble models are about 4-5 folds lower. The meta-ensemble models (most especially the VE) also seem to have very high accuracy of prediction as observed from the R^2 values with the lowest being 0.80 and the highest, 0.96. Regarding the coefficient of determination, the meta-ensemble models (though slightly higher), seem slightly comparable with the algorithms derived from the tree-ensembles. Comparing both meta-ensemble models, Table 3 indicates that the VE has a slightly better performance than the SE given the later has higher RMSE and lower R^2 values in general. With better decrease in the component variance of the prediction errors, an aggregation of the ML algorithms is thus able to improve performance through the voting mechanism. Table 3 also reveals that predictions given for the stabilised soils with OPC substituted by equal amounts of PFA and GGBS combinations are the worst.

The improvement observed in the performance of the meta-ensemble ML models stems from these models' capacity to incorporate predictions from the stand-alone and tree-ensemble algorithms meta-heuristically so that the outcome is even more accurate.

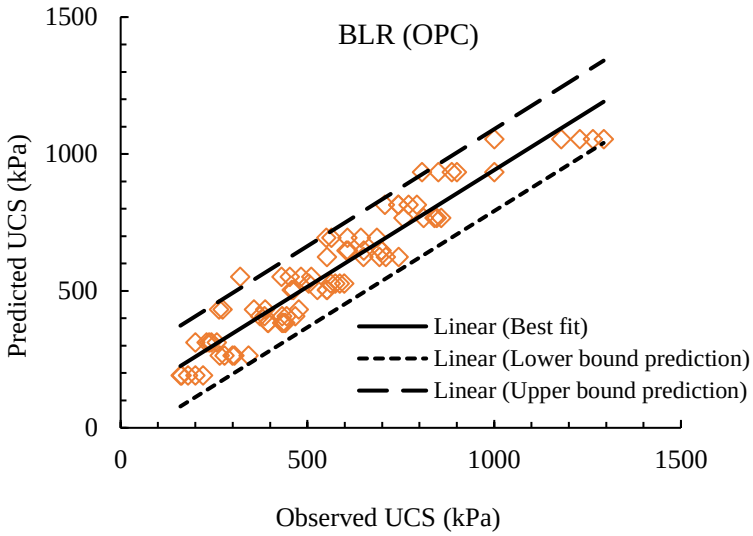
Table 3.
Statistical metrics indicating the performance of ML models utilised in strength prediction.

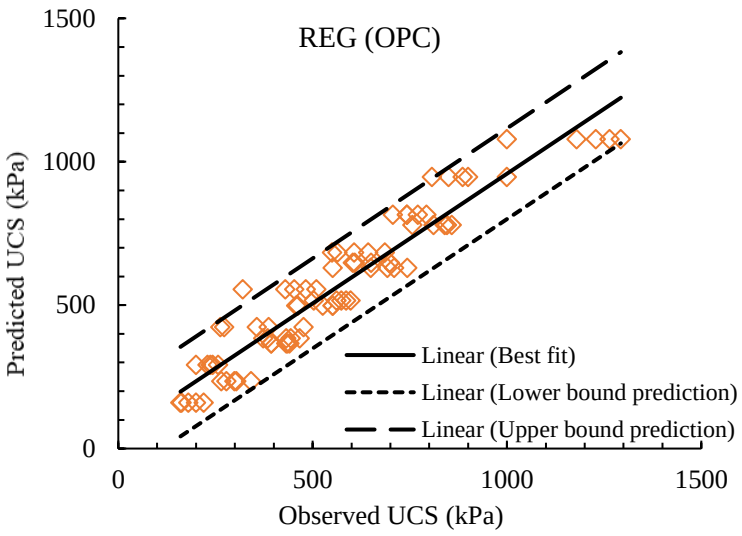
Model	Binder mix	Statistical metrics		
		R^2	RMSE	MAE
		-	kPa	kPa
BDT	OPC	**0.94**	**0.26**	**0.19**
	OPC-PFA	0.90	0.34	0.23
	OPC-PFA-GGBS	0.92	0.31	0.22
RDF	OPC	**0.89**	**0.34**	**0.26**
	OPC-PFA	0.85	0.41	0.27
	OPC-PFA-GGBS	**0.89**	**0.35**	**0.26**
BLR	OPC	**0.91**	**0.31**	**0.27**
	OPC-PFA	0.83	0.44	0.35
	OPC-PFA-GGBS	0.86	0.40	0.34
REG	OPC	**0.91**	**0.31**	**0.27**
	OPC-PFA	0.83	0.44	0.36
	OPC-PFA-GGBS	0.85	0.42	0.35
ANN	OPC	**0.90**	**0.32**	**0.27**
	OPC-PFA	0.78	0.50	0.42
	OPC-PFA-GGBS	0.78	0.50	0.42
SE	OPC	0.89	0.07	0.06
	OPC-PFA	**0.94**	**0.05**	**0.04**
	OPC-PFA-GGBS	0.79	0.09	0.08
VE	OPC	0.93	0.05	0.04
	OPC-PFA	0.93	0.05	0.04
	OPC-PFA-GGBS	**0.96**	**0.04**	**0.03**

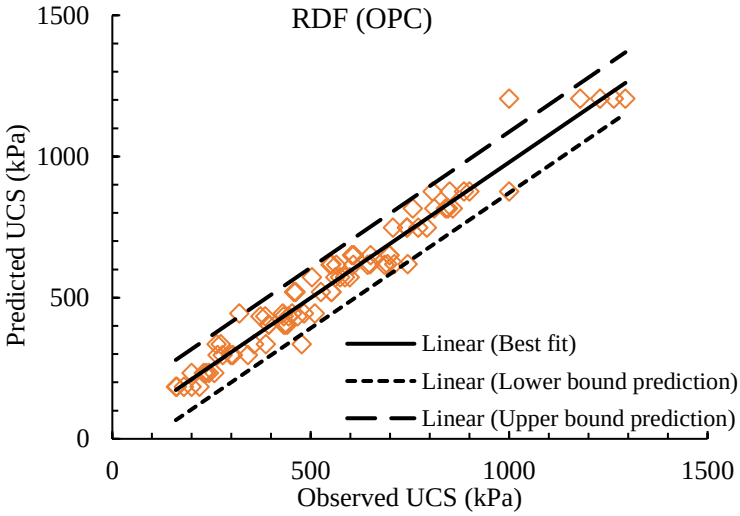
4.1.3. Uncertainty checks

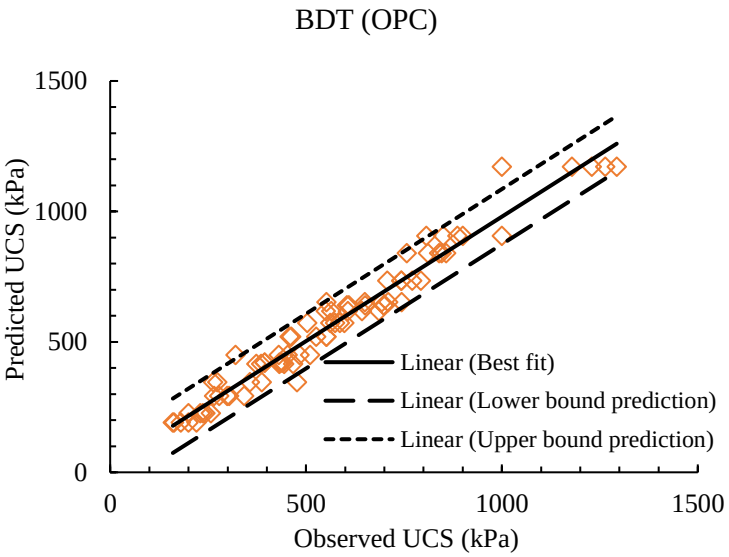
An assessment of each of the best performing regression ML models is established herein by considering their prediction intervals with 95% confidence. Following a transformation of the datasets using Eq. 1 to enable uniformity and linearity of prediction, Eq. 12 was further applied to derive both upper bound and lower bound interval of prediction based on the 2-

tailed t-test (at 2 degrees of freedom). Regardless of the differences in the range of data and their distribution around the true predictor (the trendline), as indicated in Fig. 7, virtually all the models (both stand-alone and ensembles) possess the interval that contains the dependent variable with a confidence level of 95%. This observation mirrors those of the coefficient of determination whereby the models exhibited relatively high values with the least being about 0.78. Moreover, the narrower ranges of prediction exhibited by the ensemble models (BDT, RDF, VE and SE) proves further, their superior prediction capabilities.









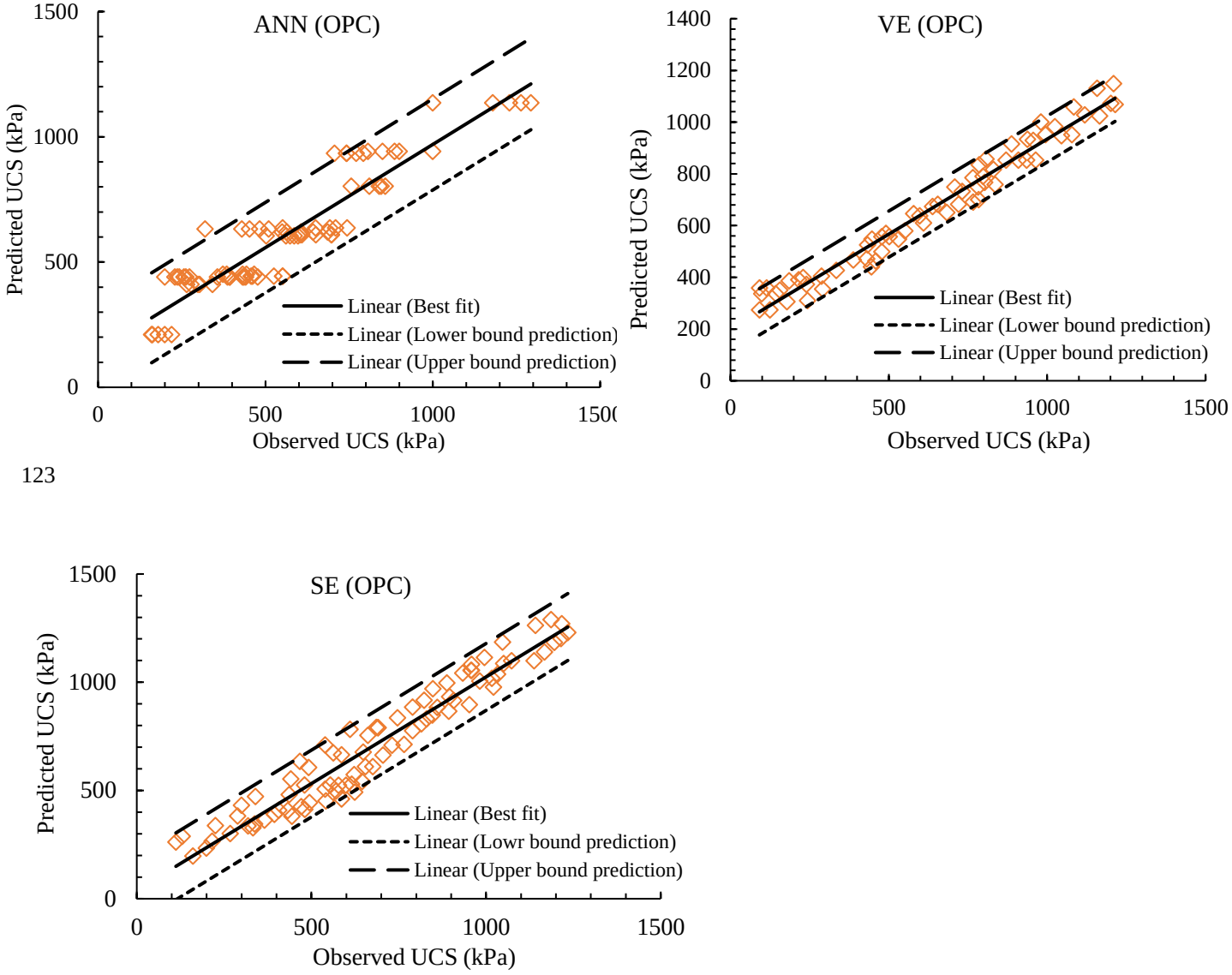
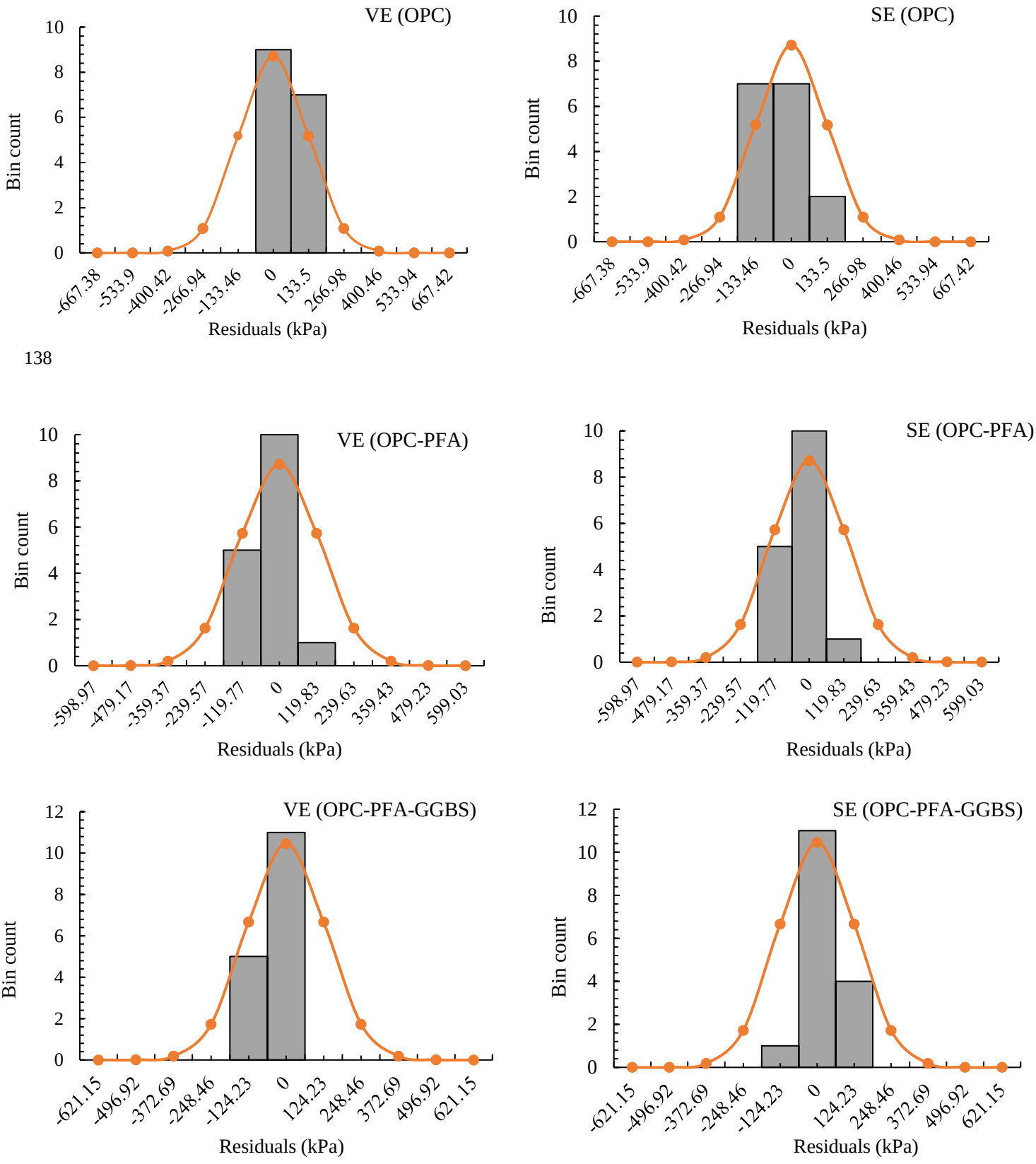


Fig. 7. Comparison of UCS strength predictions of the ML algorithms.

4.1.4. Normality of meta-ensembles

A diagnosis of normality of variance for the best performing meta-ensemble models were considered on the stabilised soils. The normality of assumptions for random error should hold true if on a histogram of the residuals, a symmetric bell-shaped curve or distribution is obtained [48]. It is observed from Fig. 8 that the residuals for both VE and SE prediction considering all the binder combinations appear to peak at zero and with less adjacent residuals. This phenomenon should indicate better performance; however, it is interesting to note that the distribution experienced from predictions on the soils stabilised by using only OPC seem slightly skewed and with lower bin counts compared to the soils stabilised with the OPC substituted by PFA and GGBS. Reasons for this behaviour cannot be advanced here except that the predictive features may not have been sufficient, thus making any further explanations rather inconclusive.



138

139

140 **Fig. 8**. Residuals and normal distribution plots of performance of meta-ensemble models

141

4.2. ML classification

The regression analyses presented in the foregoing were carried out on the data with three class features of binder combinations serving as dependent variables. Given the nature of these class features (each predicting the same output, the unconfined compressive strength), the result could be an arbitrary value. It is then very necessary that a classifier boundary between classes be determined by a threshold value.

First and foremost, looking at the classification metrics (Table 4) of the multiclass models employed for prediction of the unconfined compressive strength of the stabilised soils, it is observed that on average, the meta-ensemble models seem to perform better than some of the remaining 4 multiclass models. The multiclass logistics regression (*m*LR) model has the worst performance with an average accuracy of about 0.61, average precision of 0.42 and recall average of 0.42 probably due to an assumption of linearity even though there are instances of multi-collinearity between the dependent and predictor labels. Also, inaccurate predictions may have been increasing given the inability of *m*LR to sufficiently learn the categorical features. Interestingly, among the tree-based and stand-alone models, the multiclass *m*ANN seems to produce the highest accuracy (0.78), precision (0.67) and recall (0.67). It should be recalled that when used previously in the regression analysis, the neural network algorithm gave the least performance which in the case of classification; that is, within the context of this study, the previously stated setbacks may have been rectified thus making *m*ANN more suited to the complexities of non-linearity compared to *m*LR. The tree-ensembles (*m*RDF and *m*DJ) seem to have performed well on the multiclass problem due to their relatively high degree of accuracy in the prediction. Overall, the VE model appears to outperform all the other models given it possesses the highest classification metrics presented in Table 4. Further discussions of the meta-ensembles are provided below.

Table 4.
Classification metrics of multiclass ML models

Multiclass ML models	Average accuracy	Average precision	Average recall
*m*RDF	0.72	0.58	0.58
*m*DJ	0.67	0.50	0.5
*m*LR	0.61	0.42	0.42
*m*ANN	0.78	0.67	0.67
VE	0.88	0.79	0.83
SE	0.72	0.60	0.58

4.2.1. Sensitivity and multiclass prediction capability of the best algorithms

In order to assess the performance of the best meta-ensemble models, both the Receiver Operating Characteristic (ROC) and Lift curves were applied for some diagnostic tests of sensitivity and prediction capacity respectively. The best models are compared across some validation techniques applied to the normalised data before training by adopting 10-fold, Monte Carlo and Train-validation split cross-validation methods. This was done to provide an unbiased evaluation and estimate of the algorithm's' calibration and discrimination during the classification process.

4.2.2. Receiver operating characteristic (ROC)

Within the context of analysis in this section, sensitivity or recall may be regarded as a measure of how well the best prediction models can identify the true positives belonging to either of the three independent class variables. Receiver operating characteristics (ROC) curve is one of the most important probability evaluation metrics for assessing the best performing classification model by indicating the relationship between true positive rate (TPR) and false positive rate (FPR) during the course of any change in the decision threshold. Along with AUC (Area under the curve) which is a measure of separability, the ROC indicates how much an algorithm is able to distinguish between classes. The higher the AUC, the better the performance of a given model. For the ROC curve, an excellent or perfect classification is indicated by a point on the upper left corner with coordinates of 0 and 1 on the TPR vs FPR graph (Fig. 9). That is also to say that this represents 100% sensitivity or recall (no false negatives). A rather random act of guessing would produce a point along the blue diagonal line (i.e., line of no-discrimination) that runs from the origin (0,0) to the top right corners irrespective of the negative and positive base rates). This will indicate the worst possible situation. That will mean the AUC is approximately 0.5 and that the model possesses no discrimination capacity to distinguish between negative and positive classes.

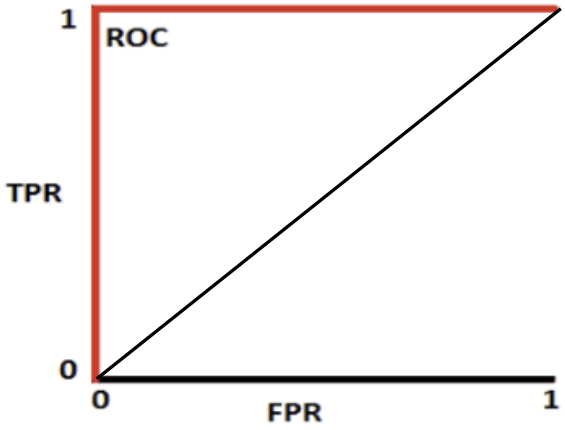


Fig. 9 Receiver operating curve (ROC)

Table 5 shows the AUC for the meta-ensemble models across three hyper-parameter tunning techniques. It is observed that the meta-ensemble models have an AUC value above 0.5. This means that the meta-ensemble models at the least, do have a discrimination ability to distinguish between negative and positive classes. The corresponding ROC curves (Fig. 10) confirms this characteristic even though not exactly perfect or an ideal situation. Stated in another way, it could be concluded that the nature of overlap has minimised any type 1 or type 2 error. Across the VE ML model, adopting the train-validation split technique appears to produce the best performance compared to SE. However, using the *k*-fold cross-validation technique seems to give the worst performance for the VE ML model. On average, it could be concluded from Table 5 that the Monte Carlo cross-validation method provides a middle ground for hyper-parameter tunning between the other two techniques.

Table 5

Comparison of Area under curve (AUC) values for different cross-validation methods

Meta-ensemble models	Cross-validation methods		
	k-fold	Monte Carlo	Train-validation split
VE	0.79	0.86	0.91
SE	0.81	0.83	0.67

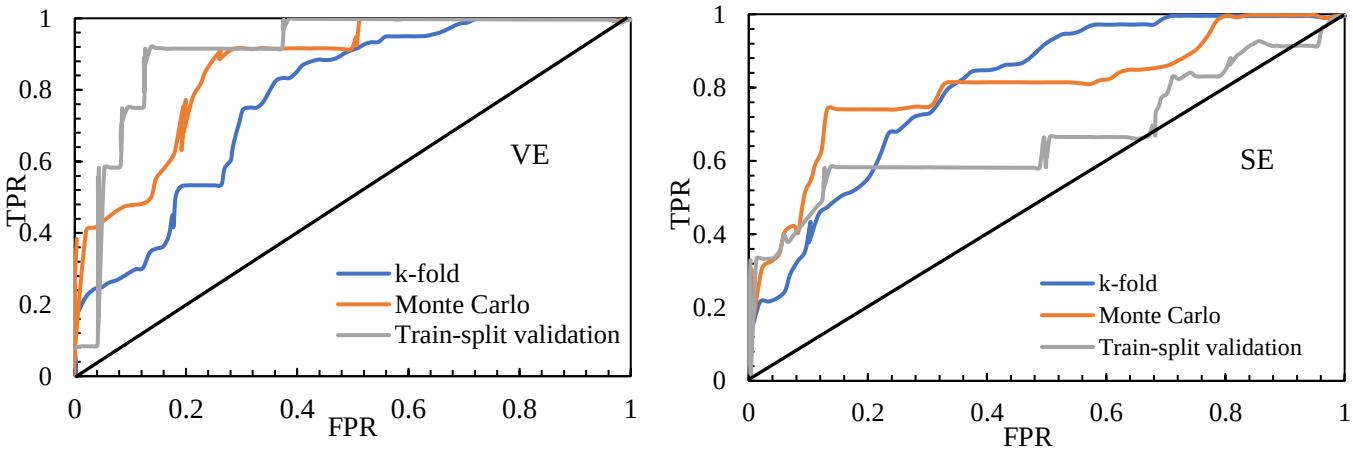


Fig. 10. Receiver operating curve of meta-ensemble models

4.2.3. Lift curve

Since this study deals with more than a binary (two-class) classification problem, further proves of how a better model could perform when compared to a random model is provided by the lift curve. This relative performance stems from the theory that a random model is likely to make an incorrect prediction of a multiclass classification problem compared to a better model with higher fractions of the sampled data. Hence, given a random model, the lift curve is a visual representation of the ratio of cumulative gains to the cumulative gains for that random model. The corresponding baseline lift curve is the horizontal or percentile axis. The greater the area between the baseline and the lift curve, the better the model. Fig. 11 represents the lift curve of the stabilised soils using the meta-ensemble models for predictive classifications following the application of *k*-fold, Monte Carlo and train-validation split cross-validations. Again, it is observed here that the overall area of the curves rising from the baseline indicates VE is the best performing model. Also, when comparing the performance of the hyper-parameter tunning techniques, it could be observed that train-validation split method provides the best validation. It is interesting to note how the *k*-fold method compares to its Monte Carlo counterpart given that both do possess almost the same trend and area under the lift curve rising from the percentile baseline.

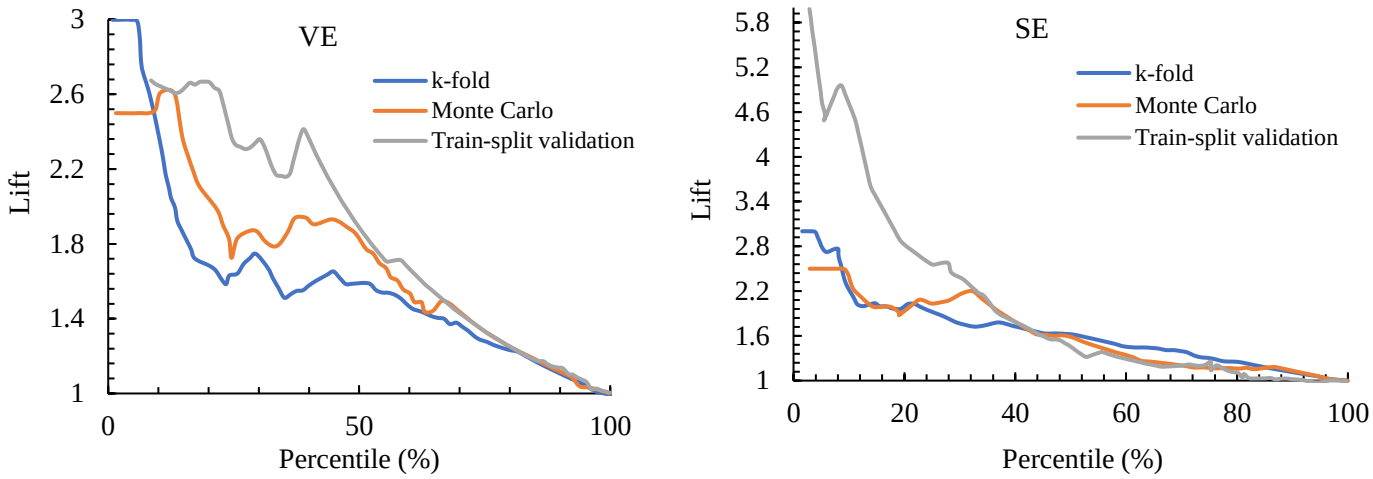


Fig. 11. A comparison of baseline lift performance of meta-ensemble models.

5. Significance of study, recommendations, and deployment of ML models

Computers with better processing speeds, higher computation power, and larger storage are some of the factors that characterise what is now termed the "age of information" or the "age of data". Accordingly, researchers, data scientists, developers, and engineers have been working assiduously to study and develop tools, algorithms, techniques, frameworks, and methodologies to build intelligent systems and models that can predict events, perform complex analyses, automate tasks, detect anomalies, ensure autonomous or self-healing failures, and even understand as well as respond to human inputs. Hence, data-driven decision making by leveraging machine learning paradigms is quite beneficial in modern times for the following reasons [55] :

- Insufficient human knowledge and expertise in a domain (e.g., simulating navigations in unknown or uncharted territories or even spatial planets).
- The rapid flux in system behaviour over time (e.g., availability of organisational infrastructure, network connectivity, etc).
- The inability for humans to formally explain or translate a well-known domain expertise into computational tasks (e.g., speech recognition, transformation, cognitive tasks, scene recognition, etc).
- Addressing domain specific challenges at scale with large volumes of data characterised by lots of complex conditions and constraints.

The present study has successfully built upon while also enhancing the evolving concepts and ideas of artificial intelligence and ML most especially those reported in recent research within the realms of soil stabilisation and geotechnical engineering in general. Only but a few of the properties or features known to influence the unconfined compressive strength of stabilised soils have been considered herein. Hence, it is recommended that for even better and effective data-driven decision, an evaluation of various other factors (e.g., compaction condition, polymers, wastes, mineralogy, soil-water chemistry, soil structure, fabric, etc) and environmental constraints (temperature, groundwater movement, drainage, and other climatic conditions) which could potentially affect soil strength should be taken into account in the future based on the techniques and framework already proposed in this research.

Furthermore, it is pertinent to state that the methods and techniques of evaluation adopted in this study represent a significant aspect of the end-to-end data mining lifecycle as suggested by a typical CRISP-DM model which is depicted in Fig. 12. The CRISP-DM model is an abbreviation for CRoss Industry Standard Process for Data Mining. CRISP-DM indicates the necessary processes, steps, and workflows for implementing any project right from formalising business requirements up to and including testing and deployment of a solution to transform data into valuable insights. This model does serve as a pointer to the tremendous amount of interest and investments in the Data Science discipline across industries, enterprises, companies, and domains. It also reinforces the earlier stated proposition that intelligent ML systems and data-driven organisations are becoming a reality with the advancements in tools and techniques only aiding in their further expansion. Hence, within the context of this study, it is suggested that for a practical application of the concepts developed, the 'Deployment' phase will ensure that the insights proposed are seamlessly transferred to production in a real-life setting. Accordingly, the models and their predictions as well as the background coding derived from this research can be deployed as saved files on an organisation's server, hardware or software resource and the proposed best meta-ensemble models reloaded while predictions are offered for new data samples on both the studied regression and classification problems. This can be applied either during preliminary stages of a geotechnical site investigation or the design and construction phases to predict and assess the strength performance of a stabilised soil.

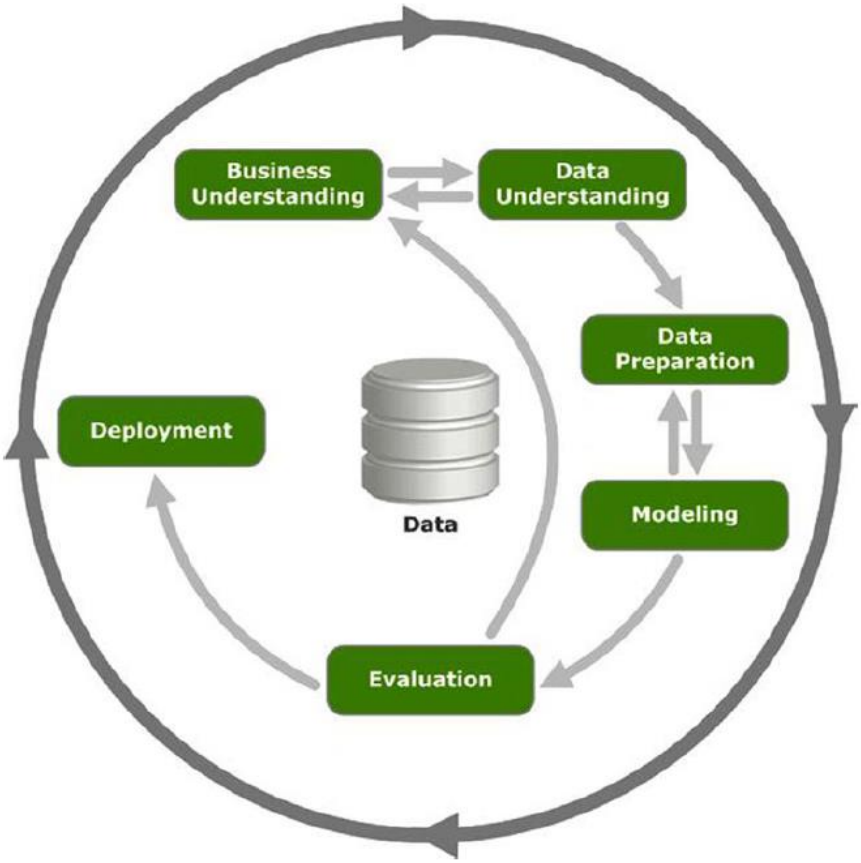


Fig. 12. Data mining lifecycle of the CRISP-DM model [55].

6. Conclusions

In this study, an analysis of ML algorithms applied to regression and multiclass classification problems of soil improvement was carried out. The summary of strength prediction of soil stabilised by OPC and part-substitution of OPC with equal amounts and combinations of PFA and GGBS using stand-alone, tree-based and meta-ensemble ML algorithms are as follows:

- Using the stand-alone (REG, BLR, ANN) and tree-ensemble models (RDF and BDT), higher statistical variance are experienced by both the OPC-PFA and OPC-PFA-GGBS predictor variables compared to the soil stabilised by OPC alone. However, RDF appears to register the highest possible variance (about 0.82) followed by BDT (about 0.53) for the soil improved using only OPC.
- Quality assessment of the ML algorithms indicated that the tree-based and meta-ensembles (VE and SE) produced much better independence of error terms. However, in terms of the features or the dependent variables used, it was observed across the models that there was relatively little difference in the degree of randomness about the zero axis of the residuals plot of 3 different combinations of the binders.
- With regards an analysis of regression, on average, REG model produced predictions of the mixed soil's UCS with higher accuracy (RMSE of 0.39 and R^2 of 0.86) compared to the ANN (RMSE of 0.44 and R^2 of 0.82), but with comparatively lower accuracy compared to the tree-based models (average RMSE of 0.33 and R^2 of 0.90) and meta-ensemble models (average RMSE of 0.06 and R^2 of 0.91).
- For ML multiclass classification, multiclass neural network algorithm (*m*ANN) gave the highest accuracy (0.78), precision (0.67) and recall (0.67) compared to tree-based and the remaining stand-alone models while only falling short to the meta-ensemble models (average accuracy of 0.80, precision of 0.70 and recall of 0.71).
- Sensitivity analysis from the receiver operating curve (ROC) and lift curves carried out across different validation techniques showed further prove of better performance of the meta-ensemble (VE) ML model compared to its SE ML counterpart when adopting the train-validation split technique as against the *k*-fold and Monte Carlo cross-validation methods.

References

[1] European Commission, Reference Document on Best Available Techniques in Cement , Lime and Magnesium Oxide Manufacturing Industries, (2010) 459.

[2] E.U. Eyo, S. Ng'ambi, S.J. Abbey, Incorporation of a nanotechnology-based additive in cementitious products for clay stabilisation, Journal of Rock Mechanics and Geotechnical Engineering. (2020). https://doi.org/10.1016/j.jrmge.2019.12.018.

[3] E.U. Eyo, S.J. Abbey, S. Ngambi, E. Ganjian, E. Coakley, Incorporation of a nanotechnology-based product in cementitious binders for sustainable mitigation of sulphate-induced heaving of stabilised soils, Engineering Science and Technology, an International Journal. (2020). https://doi.org/10.1016/j.jestch.2020.09.002.

[4] E.U. Eyo, S. Ng'ambi, S.J. Abbey, Performance of clay stabilized by cementitious materials and inclusion of zeolite/alkaline metals-based additive, Transportation Geotechnics. 23 (2020) 100330. https://doi.org/10.1016/j.trgeo.2020.100330.

[5] E.U. Eyo, S. Ngambi, S.J. Abbey, Investigative study of behaviour of treated expansive soil using empirical correlations, in: International Foundation Congress and Equipment Expo 5-10 March, Orlando, Florida, 2018: pp. 373–384.

[6] E.U. Eyo, S. Ngambi, S.J. Abbey, Investigative modelling of behaviour of expansive soils

improved using soil mixing technique., International Journal of Applied Engineering Research. 12 (2017) 3828–3836.
[7] S.J. Abbey, E.U. Eyo, J. Oti, S.Y. Amakye, S. Ngambi, Mechanical Properties and Microstructure of Fibre-Reinforced Clay Blended with By-Product Cementitious Materials, Geosciences:MPDI. 10 (2020).
[8] S.J. Abbey, A.O. Olubanwo, S. Ngambi, E.U. Eyo, B. Adeleke, Effect of Organic Matter on Swell and Undrained Shear Strength of Treated Soils, Journal of Civil, Construction and Environmental Engineering. 4 (2019) 48–58. https://doi.org/10.11648/j.jccee.20190402.12.
[9] S.J. Abbey, S. Ngambi, E. Coakley, Effect of cement and by-product material inclusion on plasticity of deep mixing improved soils, International Journal of Civil Engineering and Technology. 7 (2016) 265–274.
[10] S.J. Abbey, S. Ngambi, A.O. Olubanwo, Effect of overlap distance and chord angle on performance of overlapping soil-cement columns, International Journal of Civil Engineering and Technology. 8 (2017) 627–637.
[11] M. Kitazume, M. Terashi, The deep mixing method, 1st editio, CRC Press, 2013.
[12] EuroSoilStab, Design Guide Soft Soil Stabilisation, Ct97-0351. (1997) 95.
[13] A.E. Emeka, A.J. Chukwuemeka, M.B. Okwudili, Deformation behaviour of erodible soil stabilized with cement and quarry dust, Emerging Science Journal. 2 (2018) 383–387. https://doi.org/10.28991/esj-2018-01157.
[14] S.J. Abbey, S. Ng'ambi, E. Ganjian, Development of strength models for prediction of unconfined compressive strength of cement/by-product material improved soils, Geotechnical Testing Journal. 40 (2017) 928–935. https://doi.org/10.1520/GTJ20160138.
[15] S. Motamedi, K. Il Song, R. Hashim, Prediction of unconfined compressive strength of pulverized fuel ash–cement–sand mixture, Materials and Structures/Materiaux et Constructions. 48 (2015) 1061–1073. https://doi.org/10.1617/s11527-013-0215-1.
[16] A. Gajurel, P.S. Mukherjee, B. Chittoori, Estimating Optimal Additive Content for Soil Stabilization Using Machine Learning Methods, in: Geo-Congress 2019, 2019: pp. 662–672. https://doi.org/10.1061/9780784482124.067.
[17] S. Soleimani, S. Rajaei, P. Jiao, A. Sabz, S. Soheilinia, New prediction models for unconfined compressive strength of geopolymer stabilized soil using multi-gen genetic programming, Measurement: Journal of the International Measurement Confederation. 113 (2018) 99–107. https://doi.org/10.1016/j.measurement.2017.08.043.
[18] J. Tinoco, A. Gomes Correia, P. Cortez, Support vector machines applied to uniaxial compressive strength prediction of jet grouting columns, Computers and Geotechnics. 55 (2014) 132–140. https://doi.org/10.1016/j.compgeo.2013.08.010.
[19] K.W. Liao, J.C. Fan, C.L. Huang, An artificial neural network for groutability prediction of permeation grouting with microfine cement grouts, Computers and Geotechnics. 38 (2011) 978–986. https://doi.org/10.1016/j.compgeo.2011.07.008.
[20] E. Tekin, S.O. Akbas, Artificial neural networks approach for estimating the groutability of granular soils with cement-based grouts, Bulletin of Engineering Geology and the Environment. 70 (2011) 153–161. https://doi.org/10.1007/s10064-010-0295-x.
[21] E. Bachtiar, Mustaan, F. Jumawan, M. Artayani, Tahang, M.J. Rahman, A. Setiawan, M. Ihsan, Examining polyethylene terephthalate (Pet) as artificial coarse aggregates in concrete, Civil Engineering Journal (Iran). 6 (2020) 2416–2424. https://doi.org/10.28991/cej-2020-03091626.
[22] Z.N. Canbolat, G. Silahtaroğlu, Ö. Doğuç, N. Yılmaztürk, A machine learning approach to predict creatine kinase test results, Emerging Science Journal. 4 (2020) 283–296. https://doi.org/10.28991/esj-2020-01231.
[23] R.A. Mozumder, A.I. Laskar, M. Hussain, Empirical approach for strength prediction of geopolymer stabilized clayey soil using support vector machines, Construction and Building Materials. 132 (2017) 412–424. https://doi.org/10.1016/j.conbuildmat.2016.12.012.
[24] J.-S. Chou, K.-H. Yang, J.-Y. Lin, Peak Shear Strength of Discrete Fiber-Reinforced Soils Computed by Machine Learning and Metaensemble Methods, Journal of Computing in Civil Engineering. 30 (2016) 04016036. https://doi.org/10.1061/(asce)cp.1943-5487.0000595.
[25] S. Suman, M. Mahamaya, S.K. Das, Prediction of Maximum Dry Density and Unconfined

Compressive Strength of Cement Stabilised Soil Using Artificial Intelligence Techniques, International Journal of Geosynthetics and Ground Engineering. 2 (2016) 1–11. https://doi.org/10.1007/s40891-016-0051-9.
[26] S. Hanandeh, A. Ardah, M. Abu-Farsakh, Using artificial neural network and genetics algorithm to estimate the resilient modulus for stabilized subgrade and propose new empirical formula, Transportation Geotechnics. 24 (2020) 100358. https://doi.org/10.1016/j.trgeo.2020.100358.
[27] R.A. Mozumder, A.I. Laskar, Prediction of unconfined compressive strength of geopolymer stabilized clayey soil using Artificial Neural Network, Computers and Geotechnics. 69 (2015) 291–300. https://doi.org/10.1016/j.compgeo.2015.05.021.
[28] A.H. Alavi, A.H. Gandomi, A. Mollahassani, A.A. Heshmati, A. Rashed, Modeling of maximum dry density and optimum moisture content of stabilized soil using artificial neural networks, Journal of Plant Nutrition and Soil Science. 173 (2010) 368–379. https://doi.org/10.1002/jpln.200800233.
[29] S.K. Das, P. Samui, A.K. Sabat, Application of Artificial Intelligence to Maximum Dry Density and Unconfined Compressive Strength of Cement Stabilized Soil, Geotechnical and Geological Engineering. 29 (2011) 329–342. https://doi.org/10.1007/s10706-010-9379-4.
[30] J. Tinoco, C. António Alberto Santos, P. Da Venda, A.G. Correia, L. Lemos, A Data-driven Approach for qu Prediction of Laboratory Soil-cement Mixtures, Procedia Engineering. 143 (2016) 566–573. https://doi.org/10.1016/j.proeng.2016.06.073.
[31] U.K. Sevim, H.H. Bilgic, O.F. Cansiz, M. Ozturk, C.D. Atis, Compressive strength prediction models for cementitious composites with fly ash using machine learning techniques, Construction and Building Materials. (2020) 121584. https://doi.org/10.1016/j.conbuildmat.2020.121584.
[32] T. Han, A. Siddique, K. Khayat, J. Huang, A. Kumar, An ensemble machine learning approach for prediction and optimization of modulus of elasticity of recycled aggregate concrete, Construction and Building Materials. 244 (2020) 118271. https://doi.org/10.1016/j.conbuildmat.2020.118271.
[33] M. Jalal, Z. Grasley, C. Gurganus, J.W. Bullard, Experimental investigation and comparative machine-learning prediction of strength behavior of optimized recycled rubber concrete, Construction and Building Materials. 256 (2020) 119478. https://doi.org/10.1016/j.conbuildmat.2020.119478.
[34] H.L. Wang, Z.Y. Yin, P. Zhang, Y.F. Jin, Straightforward prediction for air-entry value of compacted soils using machine learning algorithms, Engineering Geology. 279 (2020) 105911. https://doi.org/10.1016/j.enggeo.2020.105911.
[35] H. Nguyen, T. Vu, T.P. Vo, H.T. Thai, Efficient machine learning models for prediction of concrete strengths, Construction and Building Materials. 266 (2021) 120950. https://doi.org/10.1016/j.conbuildmat.2020.120950.
[36] A. Marani, M.L. Nehdi, Machine learning prediction of compressive strength for phase change materials integrated cementitious composites, Construction and Building Materials. 265 (2020) 120286. https://doi.org/10.1016/j.conbuildmat.2020.120286.
[37] M.C. Kang, D.Y. Yoo, R. Gupta, Machine learning-based prediction for compressive and flexural strengths of steel fiber-reinforced concrete, Construction and Building Materials. 266 (2021) 121117. https://doi.org/10.1016/j.conbuildmat.2020.121117.
[38] A. Milad, S.A. Majeed, N.I.M. Yusoff, Comparative study of utilising neural network and response surface methodology for flexible pavement maintenance treatments, Civil Engineering Journal (Iran). 6 (2020) 1895–1905. https://doi.org/10.28991/cej-2020-03091590.
[39] A. Sujatha, L. Govindaraju, N. Shivakumar, V. Devaraj, Fuzzy Knowledge Based System for Suitability of Soils in Airfield Applications, Civil Engineering Journal. 7 (2021) 140–152.
[40] A. Joshi, Machine Learning and Artificial Intelligence, 1st ed., Springer International Publishing, 2020. https://doi.org/10.1007/978-3-030-26622-6.
[41] M.A. DeRousseau, E. Laftchiev, J.R. Kasprzyk, B. Rajagopalan, W. V. Srubar, A comparison of machine learning methods for predicting the compressive strength of field-placed concrete, Construction and Building Materials. 228 (2019) 116661. https://doi.org/10.1016/j.conbuildmat.2019.08.042.

[42] D. Kong, J. Zhu, C. Duan, L. Lu, D. Chen, Bayesian linear regression for surface roughness prediction, Mechanical Systems and Signal Processing. 142 (2020) 106770. https://doi.org/10.1016/j.ymssp.2020.106770.
[43] M.A. DeRousseau, J.R. Kasprzyk, W. V. Srubar, Computational design optimization of concrete mixtures: A review, Cement and Concrete Research. 109 (2018) 42–53. https://doi.org/10.1016/j.cemconres.2018.04.007.
[44] J. Shotton, S. Nowozin, T. Sharp, J. Winn, P. Kohli, A. Criminisi, Decision jungles: Compact and rich models for classification, Advances in Neural Information Processing Systems. (2013) 1–9.
[45] S.K. Das, Artificial Neural Networks in Geotechnical Engineering: Modeling and Application Issues, First Edit, Elsevier Inc., 2013. https://doi.org/10.1016/B978-0-12-398296-4.00010-6.
[46] D.H. Wolpert, Stacked generalization, Neural Networks. 5 (1992) 241–259. https://doi.org/https://doi.org/10.1016/S0893-6080(05)80023-1.
[47] K. Mamudur, M.R. Kattamuri, Application of Boosting-Based Ensemble Learning Method for the Prediction of Compression Index, Journal of The Institution of Engineers (India): Series A. 101 (2020) 409–419. https://doi.org/10.1007/s40030-020-00443-7.
[48] N. Galwey, Introduction to mixed modelling : beyond regression and analysis of variance, 2nd ed.., Chichester, [England] : Wiley, 2014.
[49] S. Chatterjee, A. Hadi, Regression analysis by example, 5th ed.., Hoboken, New Jersey : Wiley, 2012.
[50] W. Ben Chaabene, M. Flah, M.L. Nehdi, Machine learning prediction of mechanical properties of concrete: Critical review, Construction and Building Materials. 260 (2020) 119889. https://doi.org/10.1016/j.conbuildmat.2020.119889.
[51] Q.Y. Zhu, A.K. Qin, P.N. Suganthan, G. Bin Huang, Evolutionary extreme learning machine, Pattern Recognition. 38 (2005) 1759–1763. https://doi.org/10.1016/j.patcog.2005.03.028.
[52] L. Zhang, D. Zhang, Evolutionary Cost-Sensitive Extreme Learning Machine, IEEE Transactions on Neural Networks and Learning Systems. 28 (2017) 3045–3060.
[53] A. Behnood, E.M. Golafshani, Predicting the compressive strength of silica fume concrete using hybrid artificial neural network with multi-objective grey wolves, Journal of Cleaner Production. 202 (2018) 54–64. https://doi.org/10.1016/j.jclepro.2018.08.065.
[54] S. Dreiseitl, L. Ohno-Machado, Logistic regression and artificial neural network classification models: A methodology review, Journal of Biomedical Informatics. 35 (2002) 352–359. https://doi.org/10.1016/S1532-0464(03)00034-0.
[55] S. Dipanjan, B. Raghav, S. Tushar, Practical Machine Learning with Python: A Problem-Solver's Guide to Building Real-World Intelligent Systems, Apress, Berkeley, CA, 2018. https://doi.org/10.1007/978-1-4842-6222-1.