Engineered Science
DOI: https://dx.doi.org/10.30919/es929

Bearing Capacity of Eccentrically Loaded Footings on Rock Masses Using Soft Computing Techniques

Divesh Ranjan Kumar,[1] Pijush Samui,[2] Warit Wipulanusat,[3,*] Suraparb Keawsawasvong,[4] Kongtawan Sangjinda[5] and Wittaya Jitchaijaroen[6]

Abstract

A crucial characteristic of real-world engineering operations is a strip footing's bearing capacity on a rock mass subjected to incline and eccentric loading conditions. Many scientists have attempted to establish and implement artificial intelligence (AI) models for estimating strip footings' bearing capacity. In this study, four data-driven models, namely, extreme gradient boosting (XGBoost), random forest (RF), deep neural network (DNN), and long short-term memory (LSTM), are developed and compared to calculate the strip footing's bearing capacity. The strip footing's bearing capacity is obtained numerically by performing a lower bound (LB) and upper bound (UB) finite element limit analysis (FELA) for the purpose of training machine learning models. A total of 5120 FELA solutions with six dimensionless input parameters, namely, the geological strength index (*GSI*), the yield parameter (m_i), the dimensionless strength ($\gamma B/\sigma_{ci}$), inclination angle (β), the dimensionless eccentricity (e/B), and the adhesion factor (a), and one output parameter, the bearing capacity factor ($P/\sigma_{ci}B$), were utilized in the analysis. The results show that the efficiency of all the proposed models is sufficient for bearing capacity factor determination, with coefficient of determination (R^2) values ranging from 0.87 to 0.997 in the training phase and 0.975 to 0.999 in the testing phase. The proposed XGBoost model outperforms other models, such as RF, DNN, and LSTM, and can be used accurately for estimating a strip footing's bearing capacity on rock mass subjected to incline and eccentric loading loads.

Keywords: Bearing capacity; Hoek-Brown; Strip footing; Soft computing.
Received: 05 July 2023; Revised: 18 July 2023; Accepted: 19 July 2023.
Article type: Research article.

1. Introduction

The issue of eccentrically loaded footings is one of the classic works in the discipline of geotechnical engineering. It is well known that an eccentric length and inclined load can yield a reduction in the bearing capacity of footings. However, in many circumstances, the load induced by wind or suspended slab floors may cause the footing to be subjected to horizontal force. Additionally, in some situations, the load is not exactly on the center of the footings, yielding an eccentric length that causes a reduction in the footing capacity. In the past, several researchers have considered this problem of eccentrically loaded footings on soils, such as Hansen,[1] Meyerhof,[2,3] and Chen,[4] by introducing factors called "modifying Terzaghi's bearing capacity factor". Taiebat and Carter[5] and Loukidis *et al*.[6] also investigated the same problem, but they employed finite element analysis (FEA) to determine the numerical solutions of eccentrically loaded footings on soils. Moreover, Hjiaj *et al*.[7] and Krabbenhoft *et al*.[8,9] carried out finite element limit analysis (FELA) to obtain upper bound (UB) and lower bound (LB) solutions of the same problem. Nevertheless, none of these aromanticness works consider the cases of eccentrically loaded footings on rock masses.

To capture the failure behaviors of rock masses, Hoek and Brown[10] and Hoek *et al*.[11] proposed a model that is currently

[1] *Department of Civil Engineering, National Institute of Technology, Patna, India.*
[2] *Department of Civil Engineering, National Institute of Technology, Patna, India.*
[3] *Department of Civil Engineering, Faculty of Engineering, Thammasat School of Engineering, Thammasat University, Pathumthani, Thailand.*
[4] *Department of Civil Engineering, Faculty of Engineering, Thammasat School of Engineering, Thammasat University, Pathumthani, Thailand.*
[5] *Department of Civil Engineering, Faculty of Engineering, Thammasat School of Engineering, Thammasat University, Pathumthani, Thailand.*
[6] *Department of Civil Engineering, Faculty of Engineering, Thammasat School of Engineering, Thammasat University, Pathumthani, Thailand.*
Email: wwarit@engr.tu.ac.th (W. Wipulanusat)

known as the Hoek-Brown (HB) failure criterion. This model was developed as a non-linear function with the terms of major and minor effective principal stresses based on the triaxial data of tested intact rocks. For footing problems, various researchers have adopted this failure criterion to study the bearing capacity of footings on rock masses.[12–23]

Recent work was conducted by Lai *et al*..[24] This paper employed FELA to obtain 5,120 solutions of the strip footings' bearing capacity on rock masses under inclined and eccentric loads. The problem definition of Lai's study is shown in Fig. 1, where the footing has a width of B. The footing is under the ultimate load (P), where the inclination angle (β) and an eccentric length of (e) are also considered. The underlying rock masses have four material parameters: 1) the strength of the intact rock mass obtained from a uniaxial compression test (σ_{ci}); 2) the geological strength index (GSI); 3) the yield parameter (m_i); and 4) the rock unit weight (γ). In addition, the adhesion factor (a) for the contact roughness between the rock and footing is also considered in Lai *et al*..[24] Understanding how each input parameter affects the bearing capacity factor is helpful for real-world engineering applications due to the intricate coupling effects among multiple design elements. Kumar *et al*.[25] predicted the buckling response of CNT based hybrid FG plates using various machine learning techniques such as extreme gradient boosting (XGBoost), random forest (RF), and deep neural network (DNN).

In this paper, four advanced machine learning algorithms, namely, extreme gradient boosting (XGBoost), random forest (RF), deep neural network (DNN), and long short-term memory (LSTM) were constructed for calculating the strip footings' bearing capacity subjected to incline and eccentric loading. The comparative assessment of the result obtained through the FELA solution and model predicted result was presented. The study aims to accomplish the following main points: (a) utilization of an advanced data-driven regression model for estimating the bearing capacity of strip footing and (b) simplified prediction models for estimating the bearing capacity factor of strip footing using six input parameters, namely, the geological strength index (GSI), the yield parameter (m_i), the dimensionless strength ($\gamma\beta/\sigma_{ci}$), the inclination angle (β), the dimensionless eccentricity (e/B), and the adhesion factor (a). Using the numerical model, geotechnical engineers require considerable time and resources to determine the strip footing's bearing capacity. Therefore, the models proposed in this study have a very fast computational speed and are helpful for geotechnical engineers because they employ software and specific computer software knowledge.

2. Data collection

The dimensionless technique was carried out by Lai *et al*.[24] to reduce their input parameters. Note that there was only one dimensionless output, the bearing capacity factor ($P/\sigma_{ci}B$), for the problem of strip footings on rock masses under inclined and eccentric loads. In contrast, according to the following Eq. (1), six dimensionless inputs influenced the output.

$$\frac{P}{\sigma_{ci}B} \propto f\left(GSI, m_i, \beta, \frac{e}{B}, \alpha, \frac{\gamma B}{\sigma_{ci}}\right) \quad (1)$$

where e/B is the dimensionless eccentricity and $\gamma B/\sigma_{ci}$ is the dimensionless strength. Other parameters are mentioned in the Introduction. The selected values of these sixdimensionless inputs are shown in Table 1, according to Lai *et al*.[24] Notably, Lai *et al*.[24] utilized FELA techniques[26] to obtain the LB and UB solutions of the strip footing's bearing capacity on rock masses, where the HB model was used in their study. The FELA software, namely, OptumG2, was used to optimize the

Table 1. List of parametric values used in this study.

Parameters	Selected values
e/B	0.1, 0.2, 0.3, 0.4
b	45°, 60°, 75°, 90°
gB/s_{ci}	0, 0.001, 0.002, 0.01
m_i	5, 10, 20, 35
GSI	30, 50, 80, 100
a	0.25, 0.5, 0.75, 1

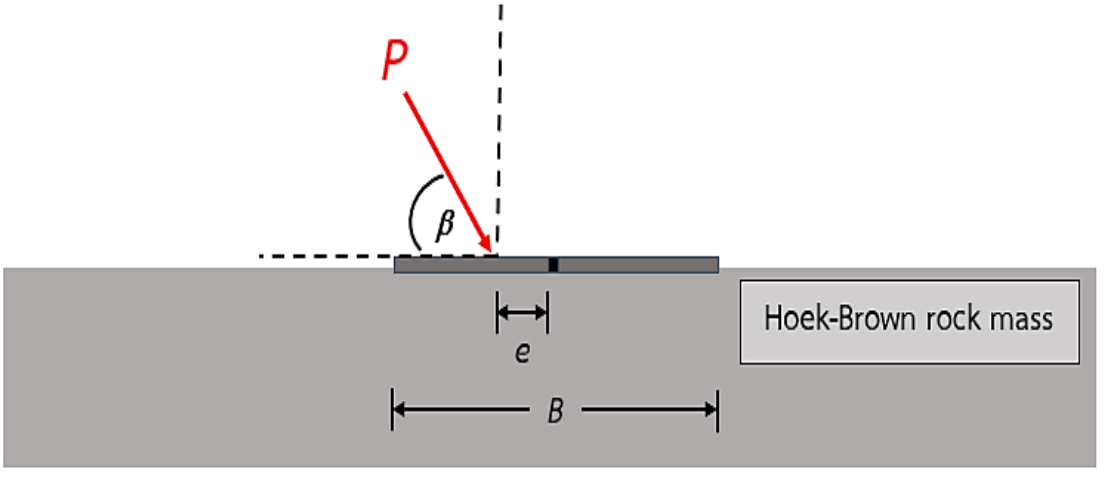


Fig. 1 Problem definition of bearing on the rock.

limit load of foundations (P). An example of a numerical model of a strip footing on a rock mass in OptumG2 is shown in Fig. 2, where the inclination angle (β) and an eccentric length of (e) are also considered. When the value of β is less than 45°, the bearing capacity factor becomes very small (less than 0.01). This is because the strip footing collapses in the sliding mode along the horizontal surface (footing-soil interface), especially when rock masses are soft (small GSI and m_i). As a result, the footing with small β fails in this sliding mode of failure rather than in ultimate shear failure. Note that the adaptive meshing approach, which is an effective feature in OptumG2,[9] was used in Lai *et al*.[24] to increase the accuracy of the limit analysis solutions. With three iteration steps with the initial mesh of 3,000 to the final mesh of 5,000 elements, the variance between UB and LB solutions is less than 5%, which is acceptable for FELA. Fig. 3 shows an example of the FELA model with the adaptive meshing approach. The mesh number is dramatically enhanced in the sensitive zones with large plastic shear strains. This can be considered the failure mechanics of this footing problem showing slip line failure. In addition, more details regarding the dimensions and boundary conditions of the models can be found in Lai *et al*..[24]

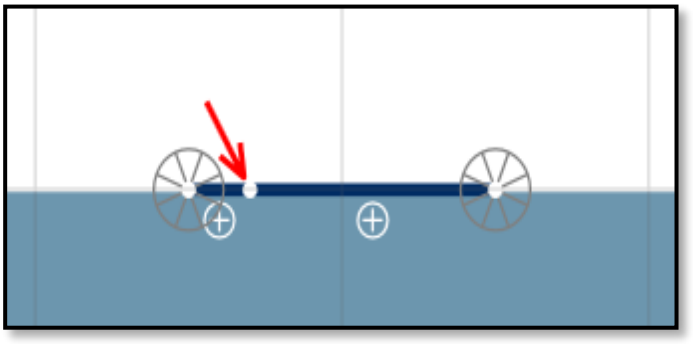


Fig. 2 Numerical model used for FELA analysis of bearing on the rock with b = 60º and e/B = 0.3.

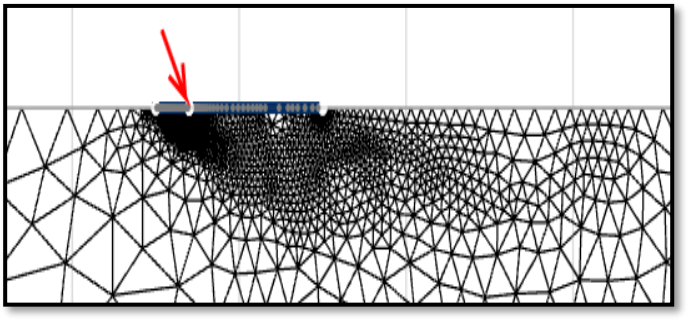


Fig. 3 Final adaptive meshes of bearing on the rock with b = 60º and e/B = 0.3.

3. Methodology of soft computing techniques

3.1 Extreme gradient boosting (XGBoost)

Chen and Guestrian[27] proposed XGBoost as an emerging machine learning technique that is frequently used in many engineering studies. The XGBoost method produces the most accurate results on a wide variety of datasets, and it performs particularly well on large and diversified datasets.[28–32] The XGBoost algorithm was created by combining the cause-based decision tree (CBDT) and gradient boosting machine (GBM) to increase the performance of the model. The XGBoost algorithm is becoming more popular due to its high execution speed and model performance. Generally, the XGBoost algorithm is faster than the gradient boosting machine algorithm. The following mathematical Eq. (2) governs the accumulation of each tree's results for the initial productive estimate of the XGBoost algorithm.

$$\hat{y}_i = \sum_{n=1}^{n} f_n(d_i) \tag{2}$$

where f_n is defined as the outputs of the regression tree belonging to the regression tree's space, and $\hat{y}_i$ represents the target prediction of the XGBoost algorithm.

The XGBoost loss function combines the regular term (Ω) and loss function (l), virtually identical to those of other ML models, to reduce the discrepancy between the actual and forecasted values. Model complexity and accuracy are controlled by the regular term and loss function, respectively. To improve the model's precision, the residual of each iteration of XGBoost is calibrated with the prior prediction. XGBoost incorporates regularization parameters in this algorithm to prevent model complexity and minimize overfitting. The mathematical expression used for the objective function is represented below in Eq. (3), and the regular term is expressed in Eq. (4).

$$Obj = L(\varphi) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{n=1}^{n} \Omega(f_n) \tag{3}$$

$$\Omega(f_n) = yT + \frac{1}{2}\lambda|\omega|^2 \tag{4}$$

where φ represents the skilled parameters obtained from input data, y represents the leaf complexity, T is the total number of leaves, λ is the penalty parameter, and ω represents the vector score on the leaf.

Using Taylor series expansion, the approximate solution for the loss function is as follows Eq. (5).

$$L^{(t)} \approx \sum_{i=1}^{T}\left[g_i\omega_j + \frac{1}{2}(h_i\omega_j^2\right] + \gamma T + \frac{1}{2}\lambda\sum_{j=1}^{T}\omega_j^2 = \gamma T + \sum_{j=1}^{t}\left[\left(\sum_{i\epsilon I_j} g_i\right)\omega_j + \frac{1}{2}\left(\sum_{i\epsilon I_j} h_i + \lambda\right)\omega_j^2\right] \tag{5}$$

where $g_i$and h_i indicate the first and second derivatives of the loss function, respectively, which are calculated using the mathematical equation $G_j = \sum_{i\epsilon i_j} g_i$, $H_j = \sum_{i\epsilon i_j} h_i$, I_j is the total dataset of leaf nodes, and the model is trained by considering y to be the estimate of the i^{th} instance at the t^{th} iteration. T is the total number of leaves, and n represents the number of decision trees. t is the t^{th} iteration.

The following greedy mathematical Eq. (6) is used to compare how the objective function changes for each feature at each node before and after splitting.

$$L_{split} = \frac{1}{2}\left[\frac{\left(\sum_{i\epsilon I_L} g_i\right)^2}{\sum_{i\epsilon I_L} h_i + \lambda} + \frac{\left(\sum_{i\epsilon I_R} g_i\right)^2}{\sum_{i\epsilon I_R} h_i + \lambda} + \frac{\left(\sum_{i\epsilon I_i} g_i\right)^2}{\sum_{i\epsilon I_i} h_i + \lambda}\right] - \lambda \tag{6}$$

where I_L represents the instance set of the left leaf node after the split, I_R represents the instance set of the right leaf after the split, and I represents the total leaf. Fig. 4 illustrates the flow chart of the XGBoost algorithm. Y_1, Y_2, ….Y_n represent the output of individual trees.

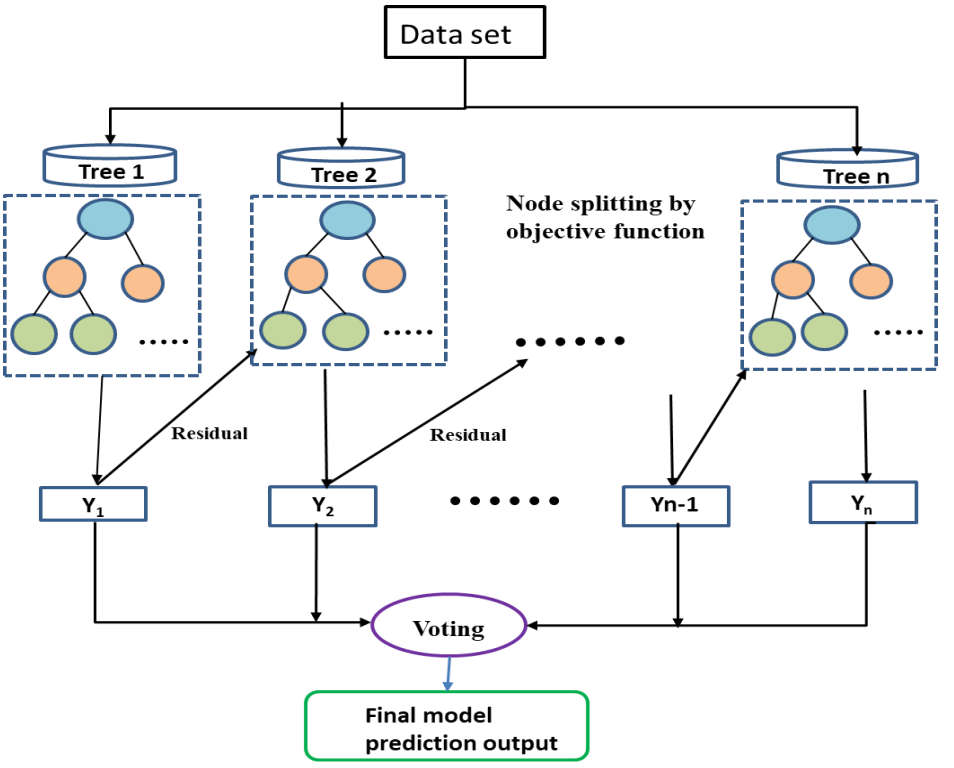


Fig. 4 Flow chart of XGBoost algorithm.

The following hyperparameters have been used in the XGBoost algorithm presented in Table 2.

Table 2. Description of hyperparameters for the XGBoost model.

Parameter Name	Description	Value
seed	Random number seed to be used	123
Learning rate	Learning rate	0.3
max depth	Maximum depth or number of split	10
no of estimators	-	100
Num Iterations	Number of iterations to be performed	10
nround	The total number of boosting rounds	12

3.2 Random forest (RF) algorithm

Breiman[33] proposed the concept of the random forest (RF) algorithm, which is an integrated machine learning model consisting of multiple decision trees that are popularly used for solving classification and regression problems. To produce a classifier with strong generalization abilities, the RF algorithm combines the output obtained from each decision tree using a particular combination technique. The model randomness is identified in two aspects. First, to generate each decision tree (DT) throughout the training of the model, the input features involved are selected randomly, but all features are not involved in the training process. Second, in training sample selection, the bootstrap sample for model training is also based on random selection. The construction procedure of every decision tree is independent of each other, so parallel computation can be performed during the construction of the model to enhance the performance. The construction procedure of the model is executed in three phases, as discussed below.

(1) Randomly select "n" number of features to avoid overfitting from the total set of "M" input variables.
(2) Select the "m" number of best sample variables for training purposes.
(3) A total of "m" number of DTs were constructed and trained, and the error of the developed model was computed using the out-of-bag data of each tree. Finally, the result of each DT is averaged to obtain the final RF model result.

The model was developed utilizing machine learning and statistical modelling using the well-known Python library scikit-learn. The subsequent hyperparameters must be established when building the RF model: (1) Maximum depth: The maximum depth parameters control the overfitting and underfitting tendency of the model. The model tends to overfit if the value of max depth is taken too high, whereas a smaller value results in poor confidence in the model and underfitting. (2) Number of trees or estimators: A larger number of trees will yield a more accurate prediction. However, computational efforts have grown due to the increased number of trees used. When using the scikit-learn library, n estimators are set to 100 by default. (3) Max features: When looking for the optimal split at each node, this parameter determines the size of random subsets of features. Increasing this parameter's value enhances performance on the training set but also introduces a correlation between the trees, which we want to avoid. This option is named max features in the scikit-learn library, and its default value is $\sqrt{n\_features}$. Where n_features indicate the total number of input features. (4) The size of each bag in proportion to the size of the training set. (5) The maximum number of leaf nodes. All other random forest

hyperparameters and settings have been left at the default values provided by the Python library. The hyperparameters used in the random forest algorithm are shown in Table 3.

Table 3. Description of hyperparameters for the RF model.

Parameter Name	Description	Value
seed	Random number seed to be used	120
Num Execution slots	Maximum number of leaf nodes	20
Bag Size percent	Size of each bag in proportion to the size of the training set	70
Num Iterations	Number of iterations to be performed	5
Max Depth	Maximum depth of the tree, 0 for unlimited	1000

3.3 Deep neural network (DNN)

Deep neural networks (DNNs) are a special type of artificial neural network (ANN) that mimics the structure and function of the human nervous system by stacking numerous layers of computation between the input and output layers. This research predicts the bearing capacity factor ($P/\sigma_{ci}B$) for strip footings on rock masses under inclined and eccentric loads using a DNN model trained using a back-propagation learning algorithm. This algorithm allows for the simultaneous transmission of training signals forward and the propagation of training errors backward. A series of tests are conducted using models with varying numbers of hidden layers and numbers of neurons in each layer to guarantee that the DNN model can achieve adequate precision. It has been discovered that DNN models with fewer than four hidden layers and fewer than 10 neurons in each layer provide 'underfitting' prediction results, which concludes that in the training process, the DNN model does not learn properly. It has also been discovered that DNN models with more than four hidden layers and more than ten neurons cause 'over fitting' problems due to their powerful nonlinear mapping capacities in the training phase. Due to overfitting problems, they might not provide accurate predictions during testing. As shown in Fig. 5, the best nonlinear mapping skills are displayed by the DNN regression model with back-propagation, which consists of hidden layers with an identical number of 10 neurons in each layer. The rectified linear unit (ReLU) activation function is used as the activation function between the input layer and the first hidden layer, as well as between any two hidden layers as presented in Eq. (7).

$$f(x) = \begin{cases} 0, & if\, x \le 0 \\ x, & x > 0 \end{cases} \tag{7}$$

The mathematical equation of the sigmoid function is presented in Eq. (8), which is used between the hidden layer and output layer.

$$(x) = \frac{1}{1+e^{-x}} \tag{8}$$

The Keras function from the TensorFlow package in Python is used to implement the created DNN predictive model.[34,35]

The DNN algorithm uses the following hyperparameters to obtain the optimum result, as shown in Table 4. The loss function is used as mean absolute error (MAE), which is defined as the average of the absolute differences between the actual and predicted values.

Table 4. Description of hypermeters for DNN model.

Model	Parameter name	Value
DNN	Number of layers	4
	Number of hidden neurons	1 to 10 neurons in each hidden layer
	Activation function	ReLU
	Optimizer	Adam
	Learning rate	0.3
	Epochs	1000
	Loss function	MAE

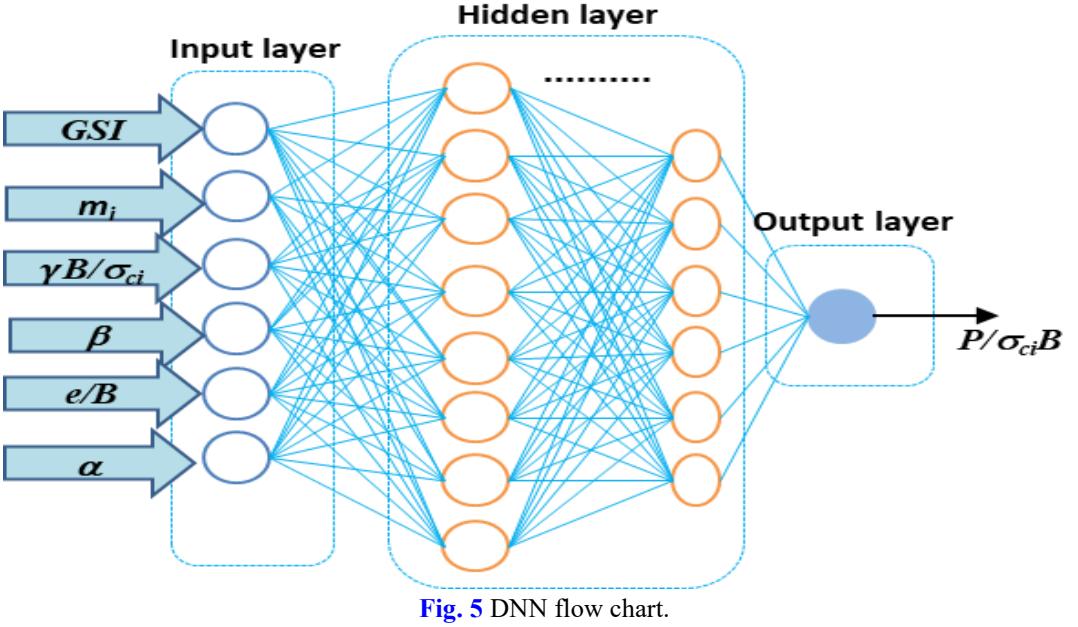


Fig. 5 DNN flow chart.

3.4 Long short-term memory (LSTM)

The long short-term memory (LSTM) neural network, first proposed by Schimidhuber,[36] is an advancement on the recursive neural network. It has several benefits in data modelling because it not only takes on the properties of a recursive neural network but also avoids the gradient explosion problem that occurs during the reverse error training of such a network. A long short-term memory (LSTM) unit is built from a neuron, and three gates, including forget gates, input gates, and output gates, govern the detection and addition of new memory cells within an LSTM. The basic structural unit of the LSTM model is represented in Fig. 6.

The forget gate, input gate, and output gate are all linked to the covered state from the preceding time step. The neuron stores data for the network in the form of a weighted sum of activation, and gates are a mechanism to make that data available selectively. Each threshold must be trained to determine when it is necessary to conceal the state in the egressing storage space.

Through the forgetting gate, the forward transfer layer describes some previous state and then acquires the necessary previous state information $f(t)$ from the input unit, which is shown in Eq. (9).

$$f(t) = \sigma(\mathrm{W_f} \times [h_{t-1}, X_t] + b_f) \quad (9)$$

where $f(t)$ represents the forget gate. $\mathrm{W_f}$ and b_f represent the weight matrix and bias matrix of the forget gate, $\sigma = (1 + e^{-x})^{-1}$ is known as the sigmoid function, X_t represents the input layer at time step t, and h_{t-1} represents the output at time step $t-1$.

After that, a new candidate value $\widetilde{C_t}$ is added to the cell state of the current time as presented in Eq. (11), and the past cell state C_{t-1} is updated based on the above steps using Eq. (12) with the help of Eqs. (10) and (11)

$$i_t = \sigma(\mathrm{W_i} \times [h_{t-1}, X_t] + b_i) \quad (10)$$

$$\widetilde{C_t} = \tanh(W_C \times [h_{t-1}, X_t] + b_C) \quad (11)$$

$$\mathrm{C_t} = \mathrm{f_t} \times C_{t-1} + i_t \times \widetilde{C_t} \quad (12)$$

where i_t represents the input gate, $\mathrm{W_i}$ represents the weight of the input gate, b_i represents the offset matrix of the input gate, $\widetilde{C_t}$ represents the vector of the new candidate value, W_C represents the weight matrix at time t, b_C represents the bias matrix at time t, and $\mathrm{C_t}$ and C_{t-1} represent the new state and old state respectively.

In the last step, the final output is obtained by multiplying the output value by the state value of the transformed function using Eq. (14).

$$\mathrm{h_t} = \mathrm{O_t} \times \tanh(\mathrm{C_t}) \quad (13)$$

$$\mathrm{O_t} = \sigma(\mathrm{W_o} \times [h_{t-1}, X_t] + b_o) \quad (14)$$

where $\mathrm{W_o}$ and b_o represent the weight and bias of the output gate, respectively, and $\mathrm{h_t}$ represents the current value of the output gate. The term $\mathrm{O_t}$ is presented in Eq. (13) represents the output gate at time *t*. The hyperparameters used in the LSTM algorithm are presented in Table 5.

Table 5. Description of hyperparameters for the LSTM model.

Parameter Name	Description	Value
Seed	Random number seed to be used	120
Hidden neurons	number of nodes and hidden layers	5
Dense layer	number of units in a dense layer	65
Dropout	dropout	0.3
Activation function	Type of activation function	sigmoid
Decay rate	Weight decay can be added in the weight update	0.96
Learning rate	It defines how quickly the network updates its parameters	0.1

4. Statistical analysis of the dataset

4.1 Data analysis and model processing

In this study, the dataset consisted of six input variables, namely, the geological strength index (*GSI*), the yield parameter (m_i), the dimensionless strength ($\gamma B/\sigma_{ci}$), the inclination angle (β), the dimensionless eccentricity (e/B), and the adhesion factor (a). The statistical description details of the dataset are presented in Table 6. The geological strength index (*GSI*) ranges from 30 to 100, the yield parameter (m_i) ranges from 5 to 35, the dimensionless strength $\gamma B/\sigma_{ci}$ ranges from 0

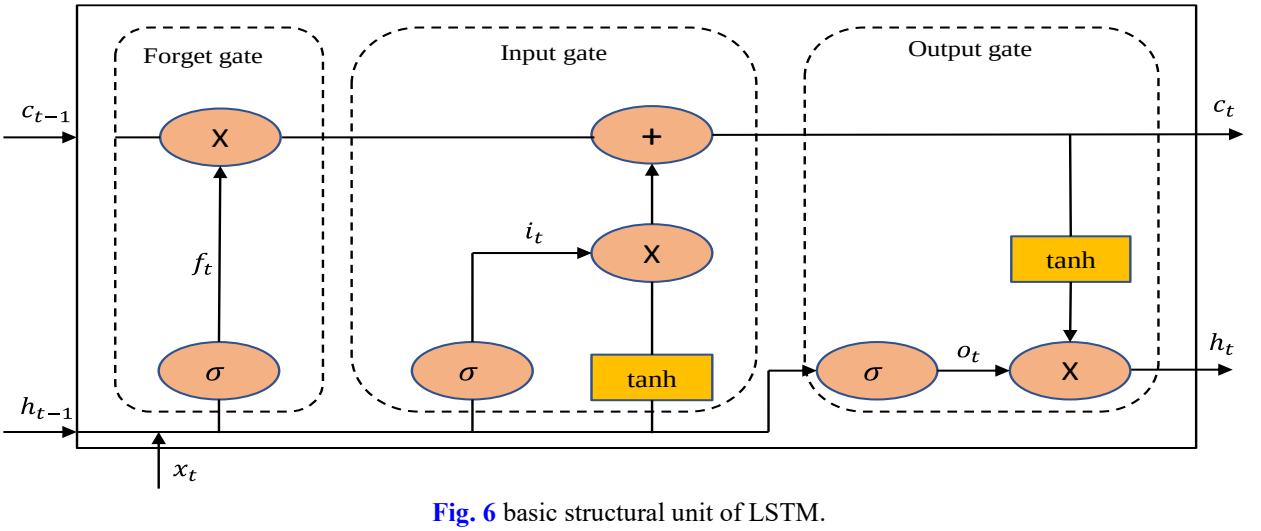


Fig. 6 basic structural unit of LSTM.

to 0.01, the inclination angle (β) ranges from 45 to 90 degrees, the dimensionless eccentricity (e/B) ranges from 0 to 0.4, the adhesion factor (a) ranges from 0.25 to 1, and the bearing capacity factor ($P/\sigma_{ci}B$) ranges from 0.001 to 20.622. Other statistical particulars are presented in Table 2.

In addition, the correlation heatmap matrix is plotted for all the input and output variables presented in Fig. 7, which concludes that GSI and β have the highest degree of correlation with the bearing capacity factor ($P/\sigma_{ci}B$), while the other input variables have a significant impact on the bearing capacity factor ($P/\sigma_{ci}B$) except for the dimensionless strength ($\gamma B/\sigma_{ci}$).

This analysis considers a set of data including 5,120 observations, with six input parameters and one output parameter. All input and output variables are normalized between 0 and 1 using the min-max technique to make the numeric columns in the dataset uniform in scale.[37] The mathematical Eq. (15) used for the normalization process is given below.

$$X_N = \frac{X_{act} - X_{min}}{X_{max} - X_{min}} \quad (15)$$

Where X_{act} is the actual value of the parameters, X_{max} and X_{min} are the maximum and minimum values of the parameters, respectively, and X_N is the normalized value of the parameters. After normalization, the 5120 dataset is randomly divided into two phases: training and testing. Seventy percent (*i.e.*, 3,584 dataset) of the entire dataset is used in the training of the model, whereas thirty percent (*i.e.*, 1,536 dataset) is used in the testing phase of the model. Finally, for the calculation of bearing capacity, factor estimates were converted back to their original scale so that the effectiveness of the used models could be assessed.

Four advanced regression models, *i.e.*, XGBoost, RF, DNN, and LSTM, were employed and accessed. It is important to note that the best regression models were developed following a trial-and-error approach. For the XGBoost model, the most effective hyperparameters, such as the learning rate, maximum depth, number of estimators, number of iterations and total number of boosting rounds with their corresponding values, are listed in Table 2. With the same training dataset, the RF, DNN, and LSTM models were constructed and accordingly evaluated. For the RF model, the most effective choice of hyperparameters is presented in Table 3. For the DNN model, the most effective choices of hyperparameters are presented in Table 4. For the DNN model, a widely used rectified linear unit (ReLU) activation function is used and the mean absolute error (MAE) is used as the loss function. Similarly, the LSTM model is also developed following a trial-and-error approach. The most effective hyperparameters with their corresponding values are listed in Table 5. For the LSTM model, the sigmoid function is used as the activation function, and the learning rate of the model is chosen as 0.1. The methodology is presented as a flow chart in Fig. 8.

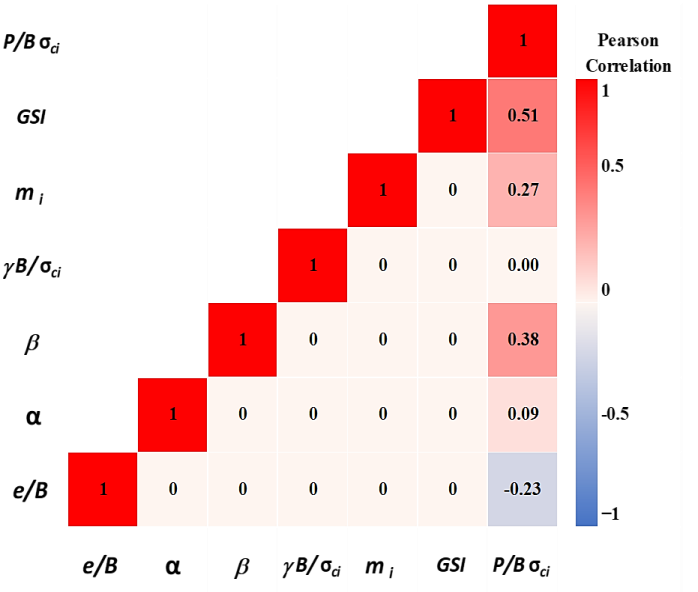


Fig. 7 Correlation heatmap matrix.

5. Results and discussion

5.1 Comparing the effectiveness of constructed models

Several performance indices, such as the coefficient of determination (R^2), weighted mean absolute percentage error ($WMAPE$), Nash-Sutcliffe efficiency (NS), root mean square error ($RMSE$), variance account factor (VAF), performance index (PI), RMSE to observation's standard deviation ratio (RSR), Willmott`s index of agreement (WI), mean absolute error (MAE), and mean bias error (MBE), were used to evaluate the effectiveness of the utilized models.[38–41] The mathematical expressions and their ideal values are presented in Table 7.

Table 6. Statistical analysis of data used in this study.

Statistics	GSI	m_i	gB/s_{ci}	β	e/B	α	$P/\sigma_{ci}B$
Maximum	100	35	0.01	90	0.4	1	20.622
Minimum	30	5	0	45	0	0.25	0.004
Standard deviation	26.93	11.46	0.00	16.77	0.14	0.28	2.67
Mean	65	17.5	0.0032	67.5	0.2	0.625	1.548
Median	65	15	0.0015	67.5	0.2	0.625	0.507
Variance	725.14	131.28	0.00	281.30	0.02	0.08	7.11
Skewness	0.00	0.50	1.05	0.00	0.00	0.00	3.38
Kurtosis	-1.52	-1.24	-0.75	-1.36	-1.30	-1.36	14.51

Table 7. Mathematical equation and ideal value of performance indices.

Performance indices	Ideal value
$R^2 = \frac{\sum_{i=1}^{n}(f_i - f_{mean})^2 - \sum_{i=1}^{n}(f_i - y_i)^2}{\sum_{i=1}^{n}(f_i - f_{mean})^2}$	1
$WMAPE = \frac{\sum_{i=1}^{n}\left\|\frac{f_i - y_i}{y_i}\right\| \times f_i}{\sum_{i=1}^{n} f_i}$	0
$NS = 1 - \frac{\sum_{i=1}^{n}(f_i - y_i)^2}{\sum_{i=1}^{n}(f_i - f_{mean})^2}$	1
$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_i - y_i)^2}$	0
$VAF = \left(1 - \frac{var(f_i - y_i)}{var(f_i)}\right) \times 100$	1
$PI = adj.R^2 + (0.01 \times VAF) - RMSE$	2
$RSR = \frac{RMSE}{\sqrt{\frac{1}{n} * \sum_{i=1}^{n}(f_i - f_{mean})^2}}$	0
$MAE = \frac{1}{n}\sum_{i=1}^{n}\|(y_i - f_i)\|$	0
$MBE = \frac{1}{n}\sum_{i=1}^{n}(y_i - f_i)$	0
$WI = 1 - \left[\frac{\sum_{i=1}^{n}(f_i - y_i)^2}{\sum_{i=1}^{n}\{\|y_i - f_{avg}\| + \|f_i - f_{mean}\|\}^2}\right]$	1

where f_i represents the i^{th} measured bearing capacity factor, y_i represents the i^{th} model predicted bearing capacity factor, and f_{mean} represents the average value of the measured bearing capacity factor. *n* represents the total number of datasets under consideration.

The performance of the developed models is summarized in Table 8 for both the training and testing phases. From the results presented in Table 8, it can be concluded that the proposed XGBoost model attained the highest value of the coefficient of determination $R^2 = 0.9994$ and lowest value of *RMSE* = 0.0032, followed by the DNN ($R^2 = 0.9994$ and RMSE=0.0082), LSTM ($R^2 = 0.9571$ and RMSE=0.0327) and RF ($R^2 = 0.9327$ and RMSE=0.0428) models in the testing phase. However, in the training phase, the DNN model achieves $R^2 = 0.9986$ and *RMSE* = 0.0064, followed by the XGBoost ($R^2 = 0.9967$ and RMSE=0.0075), LSTM ($R^2 = 0.9595$ and RMSE=0.0323), and RF ($R^2 = 0.9351$ and RMSE=0.0408) models. The constructed XGBoost models achieved the best predictive performance in both the training and testing phases, as shown by the details of the indices presented in Table 8.

5.2 Scatter plot of measured and model-predicted data

This section draws a scatter plot between the measured bearing capacity factor and the predicted bearing capacity factor along the line (y=x). A perfect prediction of the model's performance is represented by a point lying on the line (x = y), and a prediction that is closer to the line (x = y) indicates a more accurate model. The coefficient of determination (R^2) and the following regression equation are also presented for each model separately for the training and testing data shown in Fig. 9.

5.3 Taylor diagram

In this section, a Taylor diagram is plotted to assess the performance of the proposed model using the measured bearing capacity factor and predicted bearing capacity factor for both the training and testing phases separately. Taylor diagrams are 2-dimensional mathematical diagrams used to display the root-mean-square error (*RMSE*), coefficient of correlation (*R*), and standard deviation simultaneously to provide a summarizing assessment of the model's performance. Figs. 10 and 11 represent the Taylor diagrams for the training and testing phases, respectively. Figs. 10 and

Table 8. Results of various performance indices for the training and testing phases.

	XGBoost		RF		DNN		LSTM	
	TR	TS	TR	TS	TR	TS	TR	TS
R^2	0.9967	0.9994	0.9351	0.9327	0.9986	0.9975	0.9595	0.9571
WMAPE	0.0499	0.0240	0.3164	0.3214	0.0563	0.0631	0.2706	0.2631
NS	0.9967	0.9994	0.9012	0.8914	0.9975	0.9960	0.9381	0.9365
RMSE	0.0075	0.0032	0.0408	0.0428	0.0064	0.0082	0.0323	0.0327
VAF	99.67	99.94	91.88	90.89	99.82	99.68	93.85	93.56
PI	1.986	1.996	1.813	1.799	1.990	1.986	1.866	1.860
RSR	0.0577	0.0244	0.3143	0.3296	0.0495	0.0628	0.2487	0.2520
WI	0.9992	0.9999	0.9705	0.9659	0.9994	0.9990	0.9814	0.9809
MAE	0.0037	0.0018	0.0234	0.0247	0.0042	0.0049	0.0200	0.0202
MBE	0.0002	0.0000	-0.0172	-0.0181	-0.0034	-0.0038	-0.0026	-0.0030

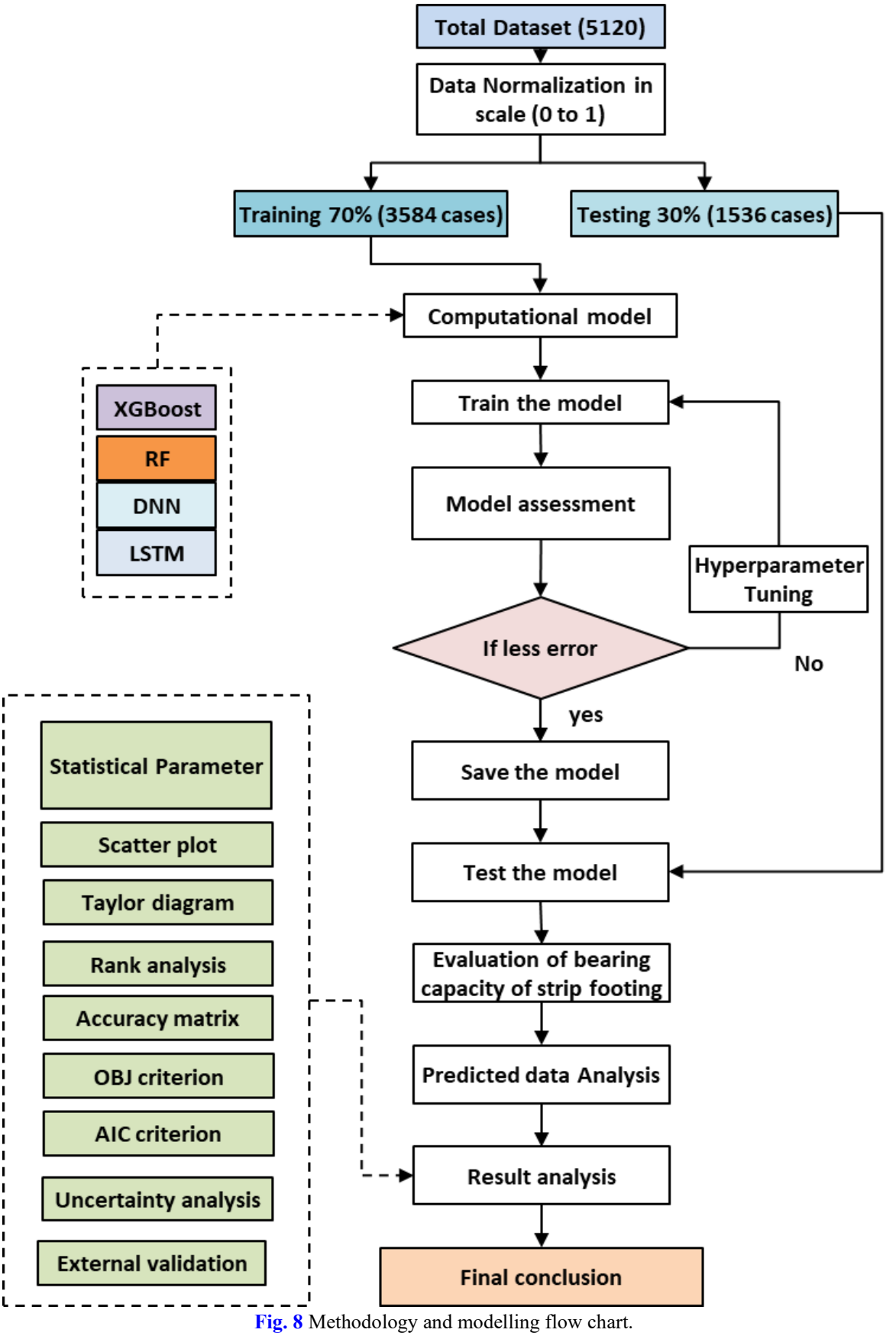
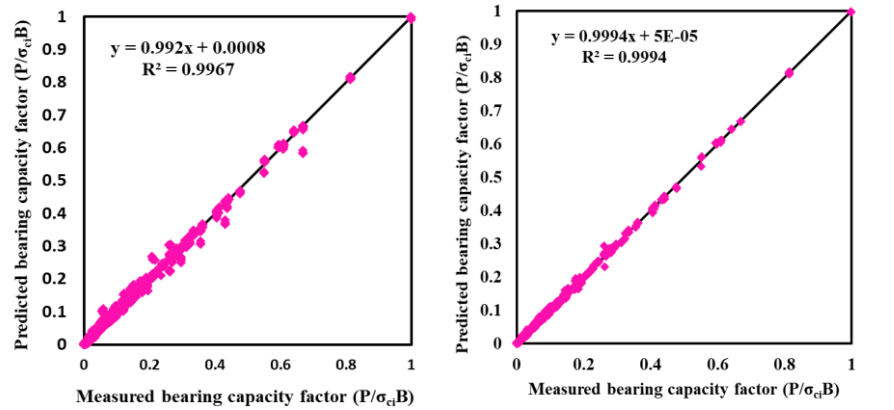


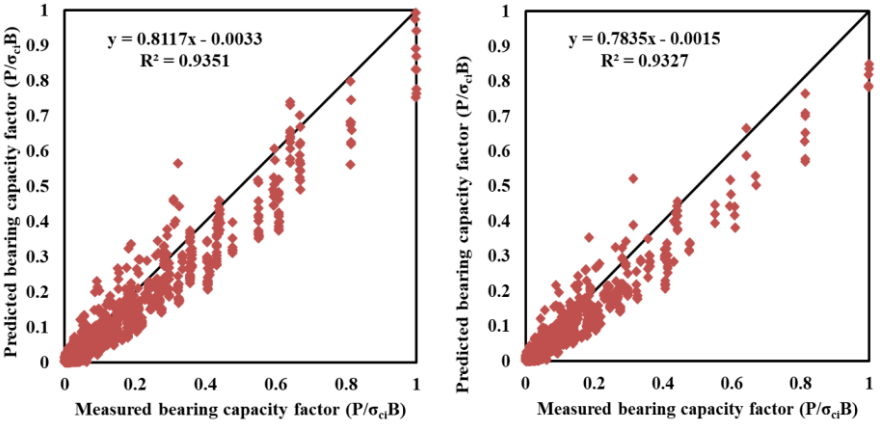
Fig. 8 Methodology and modelling flow chart.

11 show that the XGBoost model performed best, followed by the DNN, LSTM, and RF models in both the training and testing phases, with a low root mean square error and a high correlation between the actual and predicted bearing capacity ratios.



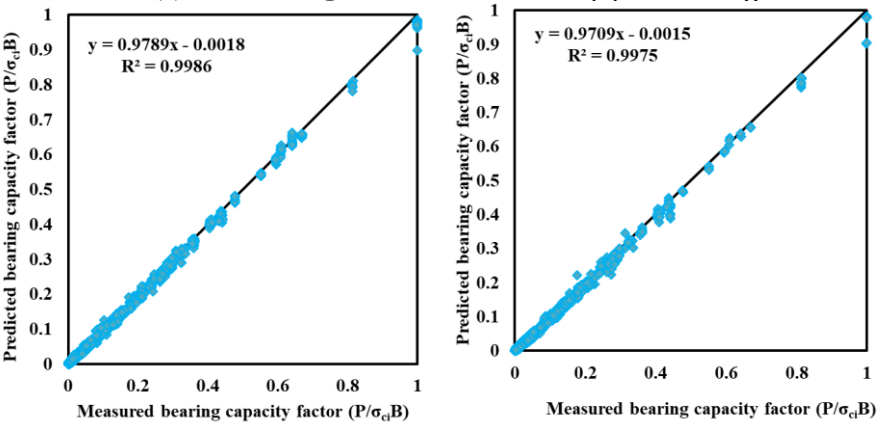
(a) XGBoost Training

(b) XGBoost Testing



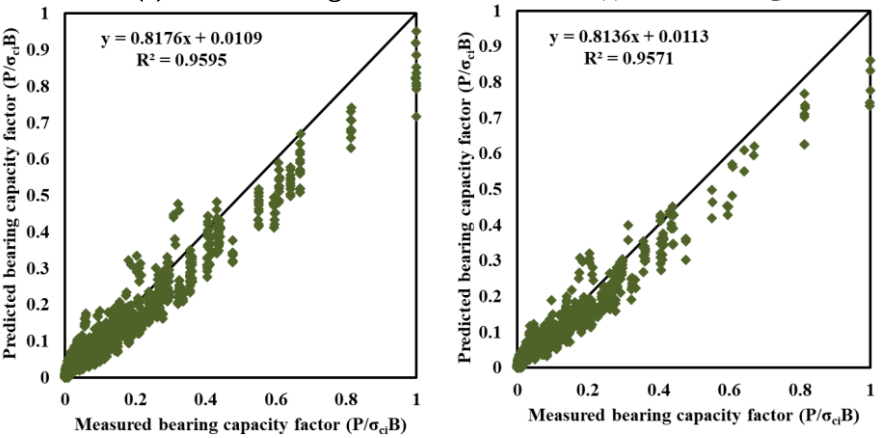
(c) RF Training

(d) RF Testing



(e) DNN Training

(f) DNN Testing



(g) LSTM Training

(h) LSTM Testing

Fig. 9 Scatter plot of measured and predicted bearing capacity factor.

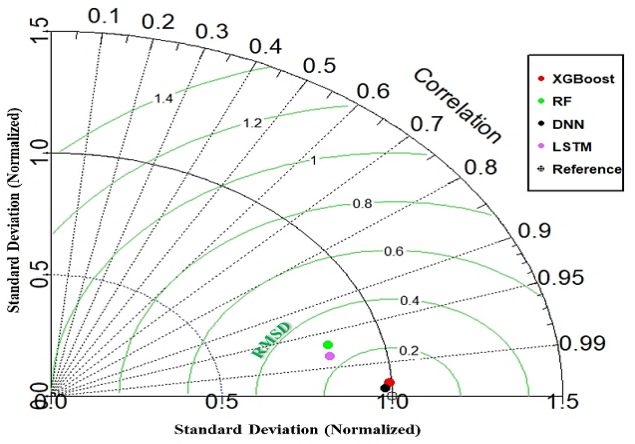


Fig. 10 Taylor diagram for training phase.

5.4 Rank analysis

A straightforward method for evaluating the models' performance is rank analysis, performed based on the performance index values. In this approach, a score of "n" (in the current study, n = 4, *i.e.*, the number of computational models) is designated to the proposed computational model that has achieved the best value for each performance parameter and a score of 1 (one) is given to the model with the worst value for the same performance parameter, individually for training and testing results. For error parameters such as *WMAPE*, *RMSE*, *RSR*, *MAE*, and *MBE*, the highest score (*i.e.*, n=4) is designated to the model that attains the lowest value of error parameters for each training and testing phase separately. Next, the individual scores for each model are added together to determine the models' total score. By adding the scores from the training and testing phases together, the final score of the model is calculated. From best to worst, the models are ranked based on their total scores; the model with the highest score is considered the best overall. The details of the score analysis are presented in Table 9, where it is evident that the XGBoost

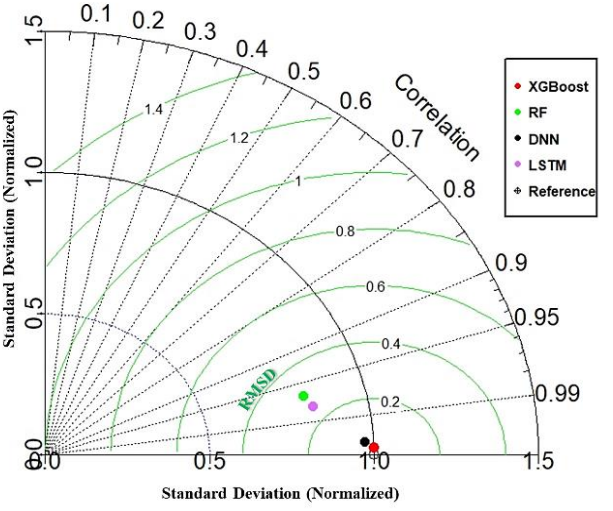


Fig. 11 Taylor diagram for testing phase.

Table 9. Results of rank analysis based on performance indices.

Parameters	XGBoost		RF		DNN		LSTM	
	TR	TS	TR	TS	TR	TS	TR	TS
R^2	0.9967	0.9994	0.9351	0.9327	0.9986	0.9975	0.9595	0.9571
Score	3	4	1	1	4	3	2	2
WMAPE	0.0499	0.0240	0.3164	0.3214	0.0563	0.0631	0.2706	0.2631
Score	4	4	1	1	3	3	2	2
NS	0.9967	0.9994	0.9012	0.8914	0.9975	0.9960	0.9381	0.9365
Score	3	4	1	1	4	3	2	2
RMSE	0.0075	0.0032	0.0408	0.0428	0.0064	0.0082	0.0323	0.0327
Score	3	4	1	1	4	3	2	2
VAF	99.67	99.94	91.88	90.89	99.82	99.68	93.85	93.56
Score	3	4	1	1	4	3	2	2
PI	1.9859	1.9956	1.8131	1.7986	1.9904	1.9862	1.8657	1.8599
Score	3	4	1	1	4	3	2	2
RSR	0.0577	0.0244	0.3143	0.3296	0.0495	0.0628	0.2487	0.2520
Score	3	4	1	1	4	3	2	2
WI	0.9992	0.9999	0.9705	0.9659	0.9994	0.9990	0.9814	0.9809
Score	3	4	1	1	4	3	2	2
MAE	0.0037	0.0018	0.0234	0.0247	0.0042	0.0049	0.0200	0.0202
Score	4	4	1	1	3	3	2	2
MBE	0.0002	0.0000	-0.0172	-0.0181	-0.0034	-0.0038	-0.0026	-0.0030
Score	4	4	1	1	2	2	3	3
Sub Total	33	40	10	10	36	29	21	21
Total Score	73		20		65		42	
Rank	1		4		2		3	

model achieved the highest total score (total score = 73), followed by DNN (total score = 65), LSTM (total score = 42), and RF (total score = 20).

5.5 Accuracy matrix

A model's level of accuracy is evaluated using the accuracy matrix, a recently developed heat map matrix, based on the statistical performance metrics shown in Fig. 12. To assess a model's correctness from numerous perspectives, a number of performance indices must be established; however, interpreting results by looking at the values of each performance parameter requires not only extensive time-investigation but also close attention to detail. Therefore, this heat map matrix is also quite helpful for a quick examination. Various statistical parameters such as R^2, *WMAPE, NS, RMSE, VAF, PI, RSR, WI, MAE* and *MBE* were included in the matrix to evaluate how accurately the model predicted results for the training (TR) and testing (TS) datasets. As per the result of the accuracy analysis, the XGBoost model accuracy is greater than that of the DNN, LSTM, and RF models in predicting the bearing capacity ratio. This heat map matrix shows that the proposed model has an overall performance of between 67% and 100%, with an average accuracy of 95%.

5.6 Objective function (OBJ) criterion

The proposed models' performance was evaluated by using the objective function (OBJ) criteria proposed by Gandomi *et al*.[42] for both the training and testing phases. The OBJ was computed using the statistical parameters of the coefficient of determination (R^2) and mean-absolute error (MAE). The following mathematical equation was applied when determining OBJ using Eq. (16).

$$OBJ = \left(\frac{N_{TR}-N_{TS}}{N_{AB}}\right) \times \left(\frac{MAE_{TR}}{R^2_{TR}}\right) + \left(\frac{2N_{TS}}{N_{AB}}\right) \times \left(\frac{MAE_{TS}}{R^2_{TS}}\right) \quad (16)$$

In the above Eq. (16), the notation N_{TR}, N_{TS}, and N_{AB} denotes the number of training data, testing data and total data, respectively. MAE_{TR} and MAE_{TS} denote the mean absolute error for the training data and testing data respectively. The coefficients of determination for the training and validation sets are denoted by R^2_{TR} and R^2_{TS}.

The model with the smallest OBJ value attained is the best. In Table 10 and Fig. 13, we can see that the OBJ value for the XGBoost (OBJ = 0.0026) model is smaller than those of the DNN (OBJ = 0.0046), LSTM (OBJ = 0.0210), and RF (OBJ = 0.0259) models. As a result, we can conclude that XGBoost outperforms the other models tested in terms of accuracy (such as DNN, LSTM, and RF).

Table 10. OBJ values for the proposed models.

Models	OBJ value
XGBoost	0.0026
RF	0.0259
DNN	0.0046
LSTM	0.0210

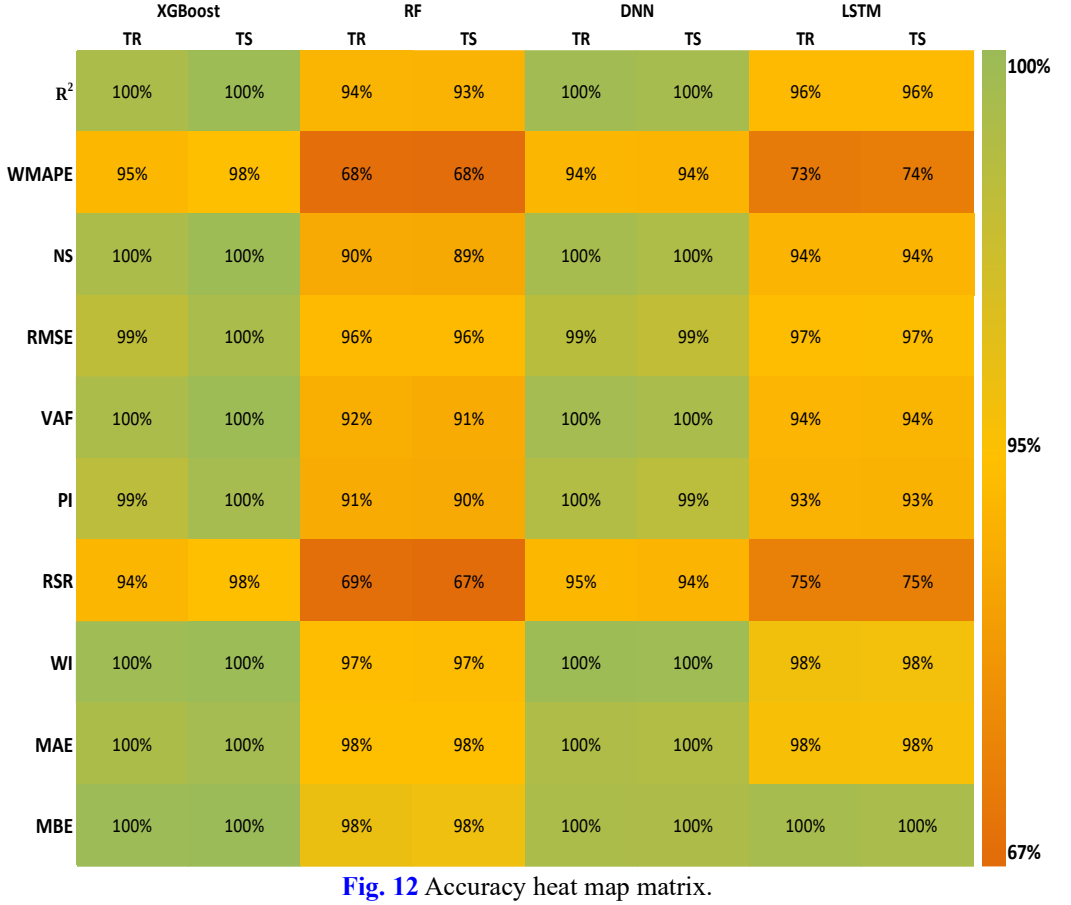


Fig. 12 Accuracy heat map matrix.

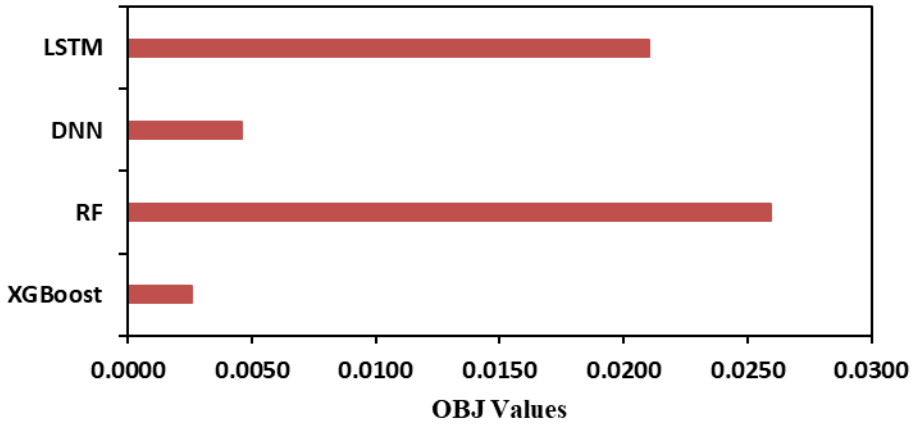


Fig. 13 Illustration of OBJ value for proposed models.

5.7 Akaike information criterion (AIC)

The Akaike information criterion (AIC) was found by Akaike[43] as a means of comparing the generalization abilities of various suggested models. The main objective of this analysis is to verify the robustness of the data-driven models that have been constructed. The AIC criteria technique has been used to evaluate the impact of models that have been built for a variety of engineering issues. The mathematical equation used for the AIC criteria analysis is presented in Eq. (17).

$$AIC = N \times \ln((RMSE)^2) + 2p \quad (17)$$

where N denotes the number of datasets used in the training phase or testing phase, and p denotes the total number of input variables used in this study. The smallest value of AIC for a particular model is considered the best one. Table 11 presents the AIC value of all the proposed models for both the training and testing phases.

Figure 14 shows the comparison of the AIC values for both the training and testing phases. In the training phase, the DNN model achieved the lowest AIC value, followed by the XGBoost, LSTM, and RF models, and in the testing phase, the XGBoost model achieved the lowest AIC value among all models. As a result, both the XGBoost and DNN models have good generalization potential compared to the RF and LSTM models.

5.8 Uncertainty analysis

The ability of the proposed machine learning models to accurately predict the outputs is demonstrated through the use of uncertainty analysis to assess model uncertainty. The uncertainty analysis is performed using the total training dataset and testing dataset separately. The standard deviation (σ) of the predicted error and mean error ($\hat{e}$) are used to perform the analysis. The mathematical equations used for

Table 11. AIC values for the training and testing phases.

Model	Training	Testing
XGBoost	-35076.4	-17670.36
RF	-22920.7	-9671.11
DNN	-36164.8	-14761.69
LSTM	-24597	-10495.49

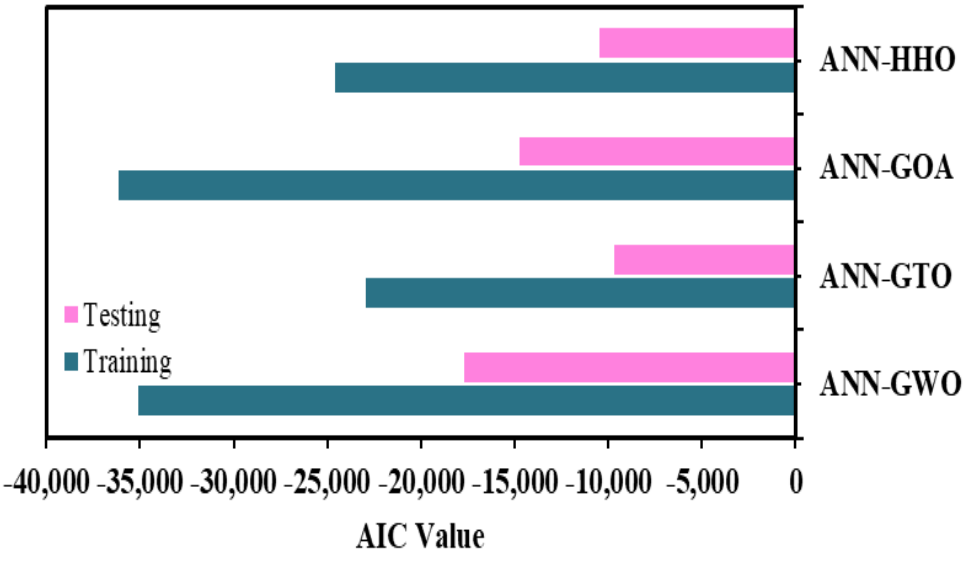


Fig. 14 Comparison of AIC value of training and testing phase.

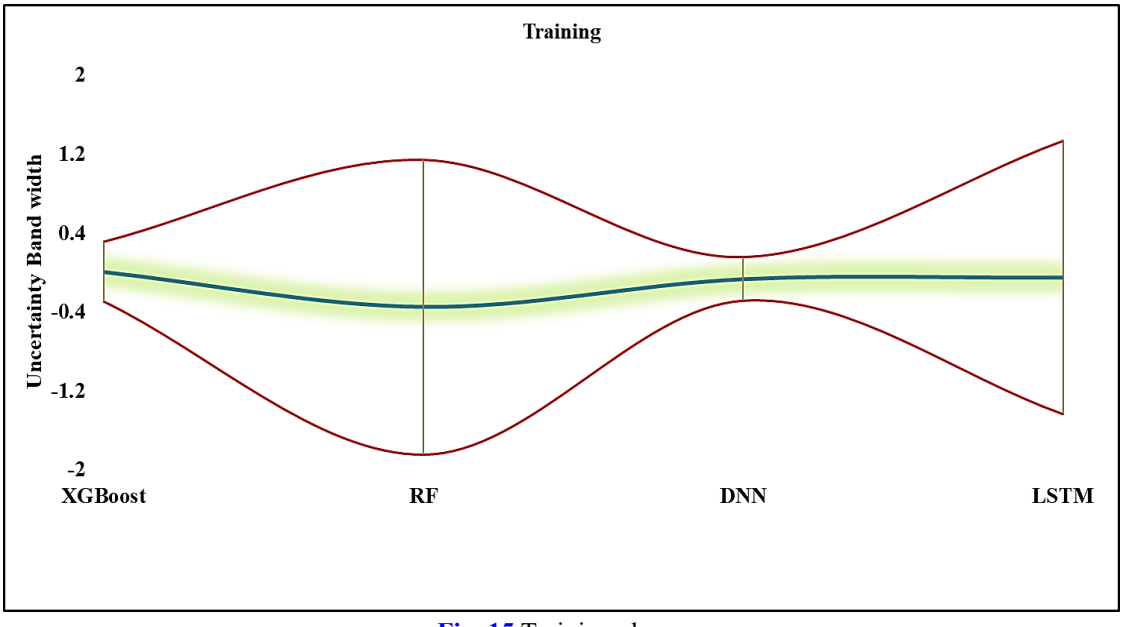


Fig. 15 Training phase.

calculating the standard deviation (σ) of the predicted error and mean error ($\hat{e}$) are presented below in Eqs. (18) and (19).

$$\sigma = \frac{\sqrt{\sum_{i=0}^{n}(e_i - \hat{e})}}{n-1} \quad (18)$$

$$\hat{e} = \frac{\sum_{i=1}^{n} e_i}{n} \quad (19)$$

where the parameters e_i represent the prediction error of the ith individual data point calculated by the difference between the predicted and actual bearing capacity factor values. A model is thought to be too conservative if the calculated mean error ($\hat{e}$) is less than zero and considered an overestimate if the calculated mean error ($\hat{e}$) is greater than zero. With the help of the Wilson score technique, the confidence interval for the predicted error is calculated with the help of the standard deviation (σ) and mean error ($\hat{e}$). Table 8 represents the uncertainty analysis results, such as the mean error standard deviation, band width, and 95% prediction interval error of the proposed models. The XGBoost model attained the lowest value of mean error, followed by the DNN, LSTM, and RF models. Furthermore, the XGBoost and DNN models have a narrower uncertainty bandwidth than the RF and LSTM models, which is shown in Figs. 15 and 16 for both the training and testing phases, respectively.

5.9 External validation

Golbraikh and Tropsha[44] and Pradeep and Samui[45] both used external validation to assess the efficiency of the developed AI model. Through the use of external validation, a more trustworthy and consistent model can be chosen. The robustness of the presented models is measured by employing external validation. The values and relationships of the external validation and conditions are presented in Table 12. Table 12 displays the results of the training and testing phases of the proposed models, demonstrating that all validity requirements have been satisfied. While all models perform

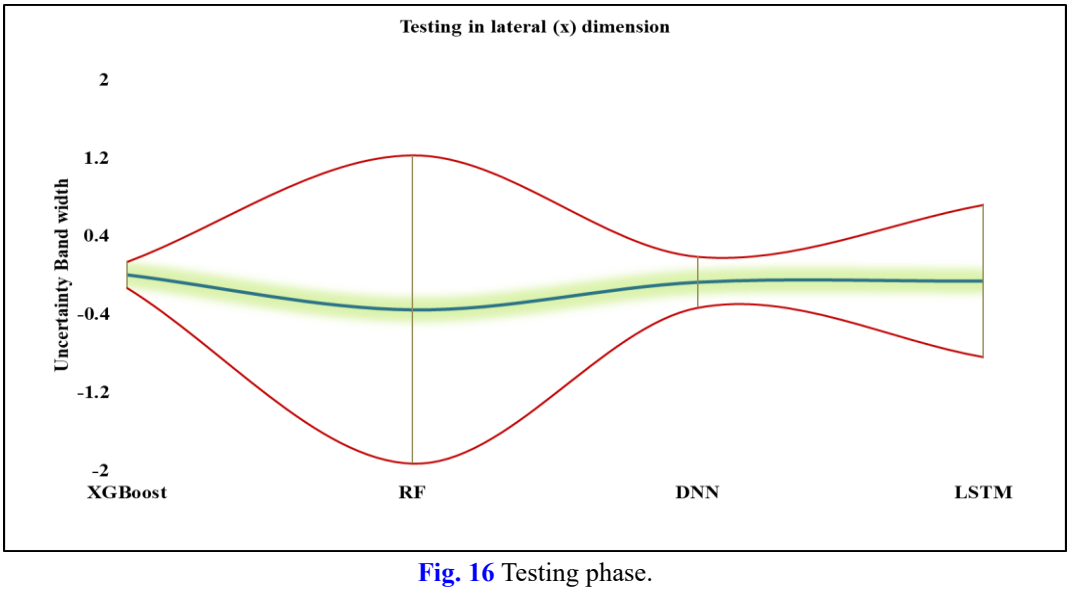


Fig. 16 Testing phase.

Table 12. External validation parameters and their conditions.

Validation Parameters	Condition	XGBoost		RF		DNN		LSTM	
		TR	TS	TR	TS	TR	TS	TR	TS
k	$0.85 < k < 1.15$	1.00	1.00	1.18	1.22	1.03	1.03	1.13	1.13
k'	$0.85 < k' < 1.15$	0.99	1.00	0.80	0.78	0.97	0.97	0.85	0.85
R_o^2	Close to 1	1.00	1.00	0.96	0.94	1.00	1.00	0.97	0.97
R'^2_o	Close to 1	1.00	1.00	0.95	0.93	1.00	1.00	0.97	0.97
R_m	$R_m > 0.5$	0.94	0.97	0.80	0.87	0.98	0.97	0.84	0.83
$\|m\|$	$\|m\| < 0.1$	0.00	0.00	-0.02	0.00	0.00	0.00	-0.02	-0.02
$\|n\|$	$\|n\| < 0.1$	0.00	0.00	-0.01	0.00	0.00	0.00	-0.01	-0.01

admirably in regard to predicting the bearing capacity factor, the XGBoost model performs the best overall. The mathematical equations used for the external validation using the various parameters are presented in Eqs. (20) to (26).

$$k = \frac{\sum_{i=0}^{n}(f_i \times y_i)}{\sum_{i=0}^{n} y_i^2} \tag{20}$$

$$k' = \frac{\sum_{i=0}^{n}(f_i \times y_i)}{\sum_{i=0}^{n} f_i^2} \tag{21}$$

$$R_o^2 = 1 - \frac{\sum_{i=1}^{n} y_i^2 (1-k)^2}{\sum_{i=1}^{n}(y_i - \bar{y})} \tag{22}$$

$$R'^2_o = 1 - \frac{\sum_{i=1}^{n} f_i^2 (1-k\prime)^2}{\sum_{i=1}^{n}(f_i - \bar{f})} \tag{23}$$

$$R_m = R^2 \times \left(1 - \sqrt{|R^2 - R_o^2|}\right) \tag{24}$$

$$m = \frac{R^2 - R_o^2}{R^2} \tag{25}$$

$$n = \frac{R^2 - R'^2_o}{R^2} \tag{26}$$

where f_i and y_i represent the measured and predicted bearing capacity factors, respectively, and k and k' represent the slopes of the regression lines (predicted versus actual bearing capacity factor, and actual versus predicted bearing capacity factor) through the origin. R_o^2 represents the coefficients of determination of the predicted versus actual bearing capacity factor and R'^2_o represents the coefficients of determination of the actual versus predicted bearing capacity factor. m and n are the conditions for estimating the predictive power of the proposed models.

6. Conclusions

In this paper, the authors present the finite element analysis (FELA) solutions for the bearing capacity factors of strip footings on Hoek-Brown rock masses under incline and eccentric loading. For practical engineers, calculating the bearing capacity factor using FELA and the Hoek-Brown failure criterion is time-consuming. Therefore, this study estimated the bearing capacity factor of HB rock masses under incline and eccentric loading using XGBoost, RF, DNN, and LSTM models. The models were developed using 5120 finite element analysis solutions with six dimensionless parameters, namely, the geological strength index (*GSI*), the yield parameter (m_i), the dimensionless strength ($\gamma B/\sigma_{ci}$) inclination angle (β), the dimensionless eccentricity (e/B), and the adhesion factor (a). To determine the most effective model for estimating the bearing capacity factor, the proposed models' performances were compared and evaluated using a wide range of performance indices, scatter plots, Taylor diagrams, rank analyses, accuracy matrices, uncertainty analyses, OBJ criteria, AIC criteria, and external validation. All of the models have the ability to estimate the bearing capacity factors of HB rock masses. Based on the accuracy of their predictions throughout the training and testing phases, the models' performances can be rated as follows: XGBoost, DNN, LSTM, and RF models. Moreover, the training and testing results for each of the proposed models were compared, and the constructed XGBoost model was determined to perform better, followed by the DNN, RF and LSTM models. As a result, the proposed XGBoost model achieved better results than other models and provides a reliable means of estimating the bearing capacity factors of strip footing. However, all the proposed ML models are capable of predicting the bearing capacity factors of strip footing subjected to inclined and eccentric loading and resting on rock masses by practical engineers. The designed XGBoost, DNN, LSTM, and RF model offers many benefits, including improved predictive accuracy during both the training and testing phases, much less computing effort, and elimination of the iteration technique. This research has some limitations, one of which is the relatively small number of datasets used. Thus, the next steps for this research should consist of (a) constructing a model using a large number of datasets, (b) comparing the results of the proposed models to those of other regression-based machine learning techniques, such as support vector machines, relevance vector machines, and hybrid models based on artificial neural networks (ANNs), and (c) conducting a thorough evaluation of the precision of other machine learning models using actual data from a wide range of engineering disciplines.

Acknowledgment

This work was supported by the Thailand Science Research and Innovation Fundamental Fund fiscal year 2023 and Thammasat University Research Unit in Data Science and Digital Transformation.

Conflict of Interest

There is no conflict of interest.

Supporting Information

Not applicable.

References

[1] J. B. Hansen, A revised and extended formula for bearing capacity, *Danish Geotechnical Institute Bulletin*, 1970, **28**, 5-11, doi: 10.3208/sandf1972.33.169.

[2] G. G. Meyerhof, The Bearing Capacity of Foundations under Eccentric and Inclined Loads, *In Proceedings of the Proceedings of the 3rd International Conference on Soil Mechanics and Foundation Engineering (ICSMFE)*, 1953, **1**, 440-445.

[3] G. G. Meyerhof, Some recent research on the bearing capacity of foundations, *Canadian Geotechnical Journal*, 1963, **1**, 16-26, doi: 10.1139/t63-003.

[4] W.-F. Chen, W. O. McCarron, Bearing capacity of shallow foundations, Foundation Engineering Handbook, Boston, MA: Springer US, 1991: 144-165, doi: 10.1007/978-1-4615-3928-5_4.

[5] H. A. Taiebat, J. P. Carter, Bearing capacity of strip and circular foundations on undrained clay subjected to eccentric loads, *Géotechnique*, 2002, **52**, 61-64, doi: 10.1680/geot.52.1.61.40828.

[6] D. Loukidis, T. Chakraborty, R. Salgado, Bearing capacity of strip footings on purely frictional soil under eccentric and inclined loads, *Canadian Geotechnical Journal*, 2008, **45**, 768-787, doi: 10.1139/t08-015.

[7] M. Hjiaj, A. V. Lyamin, S. W. Sloan, Bearing capacity of a cohesive-frictional soil under non-eccentric inclined loading, *Computers and Geotechnics*, 2004, **31**, 491-516, doi: 10.1016/j.compgeo.2004.06.001.

[8] S. Krabbenhoft, L. Damkilde, K. Krabbenhoft, Lower-bound calculations of the bearing capacity of eccentrically loaded footings in cohesionless soil, *Canadian Geotechnical Journal*, 2012, **49**, 298-310, doi: 10.1139/t11-103.

[9] S. Krabbenhoft, L. Damkilde, K. Krabbenhoft, Bearing capacity of strip footings in cohesionless soil subject to eccentric and inclined loads, *International Journal of Geomechanics*, 2014, **14**, 4014003, doi: 10.1061/(asce)gm.1943-5622.0000332.

[10] E. Hoek, E. T. Brown, Empirical strength criterion for rock masses, *Journal of the Geotechnical Engineering Division*, 1980, **106**, 1013-1035, doi: 10.1061/ajgeb6.0001029.

[11] E. Hoek, C. Carranza-Torres, B. Corkum, Hoek-Brown Failure Criterion-2002 Edition. Proc. NARMS-Tac 2002, **1**, 267-273.

[12] K. Birid, D. Choudhury, Bearing capacity of ring foundations over a rock mass using numerical analysis, *Geomechanics and Geoengineering*, 2022, **17**, 2013-2039, doi: 10.1080/17486025.2021.1975050.

[13] V. B. Chauhan, P. Kumar, S. Keawsawasvong, Limit analysis solution for ultimate bearing capacity of footing resting on the rock mass with a circular void subjected to line loading, *Indian Geotechnical Journal*, 2023, **53**, 334-347, doi: 10.1007/s40098-022-00676-2.

[14] X.-L. Yang, J.-H. Yin, Upper bound solution for ultimate bearing capacity with a modified Hoek-Brown failure criterion, *International Journal of Rock Mechanics and Mining Sciences*, 2005, **42**, 550-560, doi: 10.1016/j.ijrmms.2005.03.002.

[15] W. Yodsomjai, S. Keawsawasvong, V. Q. Lai, Limit analysis solutions for bearing capacity of ring foundations on rocks using hoek-brown failure criterion, *International Journal of Geosynthetics and Ground Engineering*, 2021, **7**, 1-10, doi: 10.1007/s40891-021-00281-y.

[16] S. Jaiswal, V. B. Chauhan, Ultimate bearing capacity of strip footing resting on rock mass using adaptive finite element method, *Journal of King Saud University - Engineering Sciences*, 2021, doi: 10.1016/j.jksues.2021.09.004.

[17] S. Keawsawasvong, C. Thongchom, S. Likitlersuang, Bearing capacity of strip footing on hoek-brown rock mass subjected to eccentric and inclined loading, *Transportation Infrastructure Geotechnology*, 2021, **8**, 189-202, doi: 10.1007/s40515-020-00133-8.

[18] S. Keawsawasvong, Bearing capacity of conical footings on Hoek-Brown rock masses using finite element limit analysis, *International Journal of Computational Materials Science and Engineering*, 2021, **10**, 2150015, doi: 10.1142/s2047684121500159.

[19] S. Keawsawasvong, J. Shiau, K. Limpanawannakul, S. Panomchaivath, Stability charts for closely spaced strip footings on hoek-brown rock mass, *Geotechnical and Geological Engineering*, 2022, **40**, 3051-3066, doi: 10.1007/s10706-022-02077-x.

[20] P. Kumar, V. B. Chauhan, Bearing capacity of strip footing resting above a circular void in the rock mass using adaptive finite element method, *Innovative Infrastructure Solutions*, 2021, **7**, 1-14, doi: 10.1007/s41062-021-00666-y.

[21] A. Parimi, S. Keawsawasvong, J. T. Chavda, Numerical evaluation of bearing capacity of strip footing on rockmass slope, *Transportation Infrastructure Geotechnology*, 2022, 1-17, doi: 10.1007/s40515-022-00255-1.

[22] N. Roy, S. Koul, Effect of embedment depth on the seismic bearing capacity of strip footing in rock mass, *International Journal of Geomechanics*, 2022, **22**, 6022010, doi: 10.1061/(asce)gm.1943-5622.0002463.

[23] Z. Saada, S. Maghous, D. Garnier, Bearing capacity of shallow foundations on rocks obeying a modified Hoek-Brown failure criterion, *Computers and Geotechnics*, 2008, **35**, 144-154,

doi: 10.1016/j.compgeo.2007.06.003.

[24] V. Q. Lai, K. Sangjinda, S. Keawsawasvong, A. Eskandarinejad, V. B. Chauhan, W. Sae-Long, S. Limkatanyu, A machine learning regression approach for predicting the bearing capacity of a strip footing on rock mass under inclined and eccentric load, *Frontiers in Built Environment*, 2022, **8**, 962331, doi: 10.3389/fbuil.2022.962331.

[25] R. Kumar, A. Kumar, D. Ranjan Kumar, Buckling response of CNT based hybrid FG plates using finite element method and machine learning method, *Composite Structures*, 2023, **319**, 117204, doi: 10.1016/j.compstruct.2023.117204.

[26] S. W. Sloan, Geotechnical stability analysis, *Géotechnique*, 2013, **63**, 531-571, doi: 10.1680/geot.12.rl.001.

[27] G. X. Gu, C.-T. Chen, M. J. Buehler, De novo composite design based on machine learning algorithm, *Extreme Mechanics Letters*, 2018, **18**, 19-28, doi: 10.1016/j.eml.2017.10.001.

[28] D. R. Kumar, P. Samui, A. Burman, Prediction of probability of liquefaction using soft computing techniques, *Journal of the Institution of Engineers (India): Series A*, 2022, **103**, 1195-1208, doi: 10.1007/s40030-022-00683-9.

[29] V.-Q. Nguyen, V.-L. Tran, D.-D. Nguyen, S. Sadiq, D. Park, Novel hybrid MFO-XGBoost model for predicting the racking ratio of the rectangular tunnels subjected to seismic loading, *Transportation Geotechnics*, 2022, **37**, 100878, doi: 10.1016/j.trgeo.2022.100878.

[30] H. D. Nguyen, G. T. Truong, M. Shin, Development of extreme gradient boosting model for prediction of punching shear resistance of r/c interior slabs, *Engineering Structures*, 2021, **235**, 112067, doi: 10.1016/j.engstruct.2021.112067.

[31] N. V. Nguyen, H. D. Nguyen, N. D. Dao, Machine learning models for predicting maximum displacement of triple pendulum isolation systems, *Structures*, 2022, **36**, 404-415, doi: 10.1016/j.istruc.2021.12.024.

[32] D. Hoang, Nguyen, Prediction of seismic drift responses of planar steel moment frames using artificial neural network and extreme gradient boosting, *Engineering Structures*, 2021, **242**, 112518, doi: 10.1016/j.engstruct.2021.112518.

[33] L. Breiman, Random forests, Machine Learning, 2001, **45**, 5-32, doi: 10.1023/A: 1010933404324.

[34] A. Gulli, S. Pal, Deep Learning with Keras; *Packt Publishing Ltd*, 2017; ISBN 1787129039.

[35] H. Langtangen, A primer on scientific programming with python, *Springer*, 2011.

[36] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Computation*, 1997, **9**, 1735-1780, doi: 10.1162/neco.1997.9.8.1735.

[37] D. R. Kumar, P. Samui, W. Wipulanusat, S. Keawsawasvong, K. Sangjinda, W. Jitchaijaroen, Soft-computing techniques for predicting seismic bearing capacity of strip footings in slopes, *Buildings*, 2023, **13**, 1371, doi: 10.3390/buildings13061371.

[38] D. R. Kumar, P. Samui, A. Burman, Prediction of probability of liquefaction using hybrid ANN with optimization techniques, *Arabian Journal of Geosciences*, 2022, **15**, 1-21, doi: 10.1007/s12517-022-10855-3.

[39] Wen-Chuan, Wang, A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series, *Journal of Hydrology*, 2009, **374**, 294-306, doi: 10.1016/j.jhydrol.2009.06.019.

[40] D. R. Kumar, P. Samui, W. Wipulanusat, S. Keawsawasvong, K. Sangjinda, W. Jitchaijaroen, Soft computing techniques for predicting penetration and uplift resistances of dual pipelines in cohesive soils, *Engineered Science*, 2023, **24**, 897, doi: 10.30919/es897.

[41] M. Kumar, R. Biswas, D. R. Kumar, T. Pradeep, P. Samui, Metaheuristic models for the prediction of bearing capacity of pile foundation, 2022, **31**, 129-147, doi: 10.12989/gae.2022.31.2.129.

[42] A. H. Gandomi, A. H. Alavi, M. G. Sahab, P. Arjmandi, Formulation of elastic modulus of concrete using linear genetic programming, *Journal of Mechanical Science and Technology*, 2010, **24**, 1273-1278, doi: 10.1007/s12206-010-0330-7.

[43] H. Akaike, Information theory and an extension of the maximum likelihood principle, *Springer Series in Statistics. New York, NY: Springer New York*, 1998: 199-213, doi: 10.1007/978-1-4612-1694-0_15.

[44] A. Golbraikh, A. Tropsha, Beware of q2!, *Journal of Molecular Graphics and Modelling*, 2002, **20**, 269-276, doi: 10.1016/s1093-3263(01)00123-1.

[45] T. Pradeep, P. Samui, Prediction of rock strain using hybrid approach of ann and optimization algorithms, *Geotechnical and Geological Engineering*, 2022, **40**, 4617-4643, doi: 10.1007/s10706-022-02174-x.