Journal of Engineering and Applied Science

RESEARCH

Open Access

Optimized systems of multi-layer perceptron predictive model for estimating pile-bearing capacity

Yuanke Shen[1*]

*Correspondence:
10801@zbvc.edu.cn

[1] School Enterprise Cooperation and Employment Guidance Center, Zibo Vocational Institute, Zibo 255300, Shandong, China

Abstract

The primary goal of this research is to leverage the advancements in machine learning techniques to forecast the bearing capacity of piles effectively. Accurately predicting load-bearing capability is an indispensable aspect in the field of substructure engineering. It is worth noting that determining load-bearing capability via in-place burden tests is a resource-intensive and labor-intensive process. This study presents a pragmatic soft computing methodology to tackle the aforementioned challenge, employing a multi-layer perceptron (MLP) for the estimation of load-bearing capacity. The dataset employed in this research encompasses a multitude of field-based pile load tests, with a meticulous selection of the most impactful factors influencing pile-bearing capacity as input variables. For a comprehensive comparative analysis, genetic algorithm-based optimizers (Crystal Structure Algorithm (CSA) and Fox Optimization (FOX)) were incorporated with MLP, leading to the development of hybrid models referred to as MLFO and MLSC, both structured with three layers. The performance of these models was rigorously evaluated using five key performance indices. The findings indicated a consistent superiority of MLFO over MLSC across all three layers. Remarkably, MLFO exhibited exceptional performance in the second layer (MLFO (2)), boasting an impressive R^2 value of 0.992, an RMSE of 33.470, and a minimal SI value of 0.031. On the other hand, MLCS (1) registered the lowest accuracy in predicting the process with the least R^2 value related to the validation phase of the model with 0.953. Taken together, these results affirm that the optimized MLP model stands as a valuable and practical tool for accurately estimating pile-bearing capacity in civil engineering applications.

Keywords: Pile-bearing capacity, Multi-layer perceptron, Crystal structure algorithm, Fox optimization

Introduction

The expense associated with foundation work typically constitutes a substantial portion of the overall construction expenditure [1]. Consequently, selecting an appropriate foundation structure solution and determining the foundation's load-bearing capacity is significant in cost reduction within construction projects [2]. Pile foundations are among the prevalent foundation solutions employed today as one of the most favored

deep foundation options due to their inherent advantages. Piles represent a foundation type known for their elevated bearing capacity, broad applicability, and extensive historical use. As the field of infrastructure construction continues to evolve, piles find widespread application in various domains, including high-rise buildings, ports, and bridge engineering. The ultimate bearing capacity (P_u) of a pile holds paramount importance in pile design, given its direct implications for the safety and cost-efficiency of engineering projects [3]. Particularly noteworthy is the pile foundation's ability to transmit loads into deeper soil layers [4, 5]. Accurately determining the P_u of a pile facilitates the determination of the appropriate foundation dimensions and pile depth, aiding in the selection of the most suitable foundation solution. Various methods are available to assess P_u, including (PDA). and immobile pile burden examinations [6–11]. Furthermore, several conventional formulas have been proposed, primarily founded on in situ soil testing results, such as cone penetration tests (CPT) and standard penetration tests (SPT) [6–8, 10, 11]. Additionally, certain studies have utilized the limited component approach to assess the relationship between pile displacement and pile load [9, 12, 13].

In specific circumstances, the approaches mentioned above exhibit several advantages; however, it is crucial to acknowledge that numerous issues necessitate careful consideration before widespread implementation in construction practices. For instance, the practical interactions between piles and soil are often oversimplified and assumed within theoretical analyses and numerical simulations. To illustrate, a study by Jesswein et al. [14] highlighted the unreliability of pile load capacity calculations based on the Standard Penetration Test (SPT) despite its cost-effectiveness and simplicity. Similarly, analytical methods are considered unfeasible due to their reliance on numerous assumptions and simplifications [4]. On an alternate note, Abu-Farsakh and Titi [15] argued that empirical and static analyses of piles are costly and offer limited accuracy due to the extensive use of safety factors. Regarding pile load testing, although it boasts a high level of reliability, it is a laborious and expensive process, often involving cumbersome equipment [16]. The dynamic approach heavily relies on pile characteristics, the impact hammer, and pile positioning to predict P_u of pile, largely overlooking soil effects [12–17]. Finally, it is essential to note that numerical simulation methods, predominantly based on finite elements, remain essentially approximate, with results significantly contingent on the modeling process [18].

In recent research endeavors and academic investigations, researchers have increasingly embraced a pioneering approach when addressing concerns related to building foundation issues. This innovative strategy harnesses the capabilities of (AI). As the field of computing knowledge has advanced, AI has consistently demonstrated its remarkable effectiveness across diverse domains, spanning construction [19], transportation [20], security [21], and medicine [22]. AI algorithms, fundamentally grounded in the fusion of mathematical principles, algorithms, and creative problem-solving, endow AI with the capacity to address complex challenges, particularly uncertainties. Consequently, AI finds a well-suited application for addressing intricate issues within the domain of geotechnical engineering [23, 24].

In their research, Kumar et al. [25] introduced AI techniques for predicting shallow foundation-bearing capacity, comparing ELM-EO and ELM-PSO hybrid models with traditional ELM and MARS models. ELM-EO demonstrated remarkable robustness,

outperforming others with an R^2 value of 0.995 and an impressively low RMSE of 0.01. In a related study, Kumar and Samui [26] focused on risk and reliability in geotechnical structures, proposing an efficient AI-based method for predicting pile-bearing capacity using MARS, GMDH, and GP models. Analysis of dynamic test data from Indonesian sites revealed GP and MARS as robust models for accurate bearing capacity estimation, while GMDH exhibited comparatively less satisfactory performance. These studies underscore the efficacy of AI-based techniques, particularly showcasing the superiority of specific hybrid models in predicting foundation and pile-bearing capacities with high accuracy and reliability.

In the context of predicting P_u, researchers have extensively investigated a diverse range of AI algorithms, underscoring the adaptability of artificial intelligence in geotechnical applications. Prominent methodologies encompass artificial neural networks (ANN) [27–29], deep neural networks (DNN) [30–32], adaptive neuro-fuzzy inference systems (ANFIS) [33, 34], and random forests (RF) [28]. For example, Shahin et al. [27, 35–37] utilized an artificial neural network (ANN) model to forecast P_u in both driven piles and drilled shafts. Their approach involved integrating data from in-place burden tests and CPT outcomes, enriching their dataset with valuable information. Similarly, Nawari et al. [38] developed a specialized ANN algorithm designed to predict settlement patterns in drilled shafts. This model utilized inputs derived from SPT data and various parameters related to shaft geometry, showcasing the flexibility of AI in assimilating diverse information sources. Taking a different route, Pham et al. [28] incorporated an ANN algorithm alongside the random forest (RF) method to anticipate axial pile-bearing capacity. This hybrid strategy leveraged the strengths of both algorithms, potentially enhancing the precision and reliability of predictions. Furthermore, Suman et al. [39] conducted a thorough assessment of the friction resistance of driven piles in clay using multivariate adaptive regression splines (MARS) and functional networks (FN). Their study not only outperformed existing models but also underscored the inherent predictive capabilities of these AI-based models, affirming their potential to advance geotechnical engineering analyses.

A comprehensive overview of prior research endeavors within the same field (prediction of P_u) is concisely presented in Table 1. This table encapsulates and summarizes the key findings and insights from earlier studies, offering a consolidated reference point for understanding the breadth and scope of the existing body of knowledge in the subject area.

Table 1 Briefly review published articles regarding the employment of ML in the prediction of P_u

Article	Model	Data size	Models' performance	
			R^2	RMSE
Gnananandarao et al. [40]	Sigmoid Symmetric	409	0.94	0.60
Onyelowe et al. [41]	SVMRBF	121	0.99	0.043
Onyelowe et al. [42]	ANN	121	0.99	0.14
Kumar et al. [43]	ENN	212	1	0
Kumar et al. [44]	ELM-PSO	300	0.88	0.08

In prior research, ML-based models were employed for diverse predictive tasks, such as determining the ultimate bearing capacity of soil, assessing the compressive strength of concrete, and predicting various engineering-related outcomes. Notably, researchers have emphasized the beneficial utilization of the multi-layer perceptron (MLP) model, augmented by the integration of Gray Wolf Optimization (GWO), for approximating the ultimate load-carrying capacity of driven posts, as detailed in [45]. Despite this knowledge, the combination of the MLP model with two distinct optimization systems for predicting pile-bearing capacity has not been explored. In light of this, the current study introduces an innovative design model suitable for comparative analysis. This is accomplished by optimizing MLP models using two alternative optimization algorithms: the Crystal Structure Algorithm (CSA) and Fox Optimization (FOX). CSA and FOX are recognized for their effectiveness in fine-tuning model parameters. When integrated with MLP, these optimization techniques aim to improve the performance of predictive models, ultimately contributing to more accurate estimates of Pu. The performance of the developed models has been evaluated using statistical metrics, and the optimal model has been identified.

Significance of the present study

This research stands out due to its creative application of machine learning, particularly the MLP technique, to address the intricate challenge of predicting pile-bearing capacity using field trial data. The study introduces a novel hybrid approach by integrating the FOX and CSA methodologies, resulting in enhanced prediction accuracy. The models undergo thorough training and validation using a meticulously curated dataset compiled from diverse literature sources, establishing a solid foundation. The research methodology yields highly precise results, highlighting the effectiveness of the proposed models.

Methods

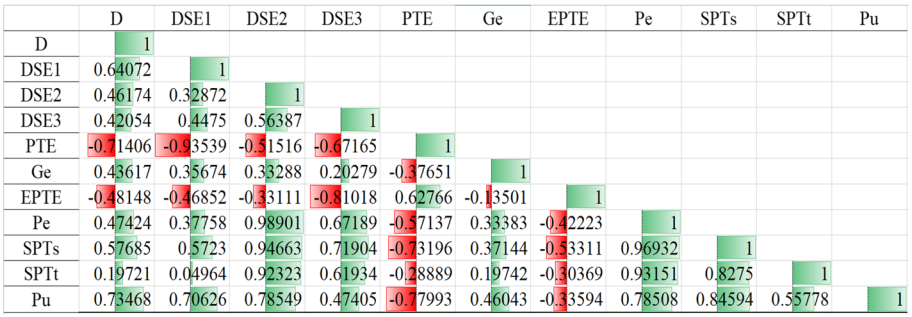
Dataset description

In this study, a dataset comprising P_u test results for reinforced concrete piles were utilized to train and evaluate predictive models. Initially, a comprehensive consideration of all pertinent factors influencing P_u was undertaken, guided by previous research [11, 30], which indicated that a multitude of parameters influences P_u. These parameters encompass pile diameter (D), depths of soil layers (DES1, DES2, ADES3), pile top and tip elevations (PTE, Pe), ground elevation (Ge), additional pile top elevation (EPTE), and SPT (Standard Penetration Test) blow counts at both the pile shaft and tip (SPTs, SPTt). These key variables were employed in the development of the proposed models, with the dataset partitioned into training (70%), validation (15%), and testing (15%) subsets. The statistical analysis results, encompassing minimum, average, maximum, and standard deviation values for both input and output variables, are briefly summarized in Table 2.

The correlation plot in Fig. 1 visualizes the relationship between input and output variables. It provides valuable insights into the interdependencies and potential associations among the various parameters under consideration. Through the correlation plot, patterns, trends, and the strength of relationships between inputs and the P_u of piles can be discerned. This analysis not only aids in identifying the most influential factors and informs the model-building process by highlighting variables that may

Table 2 The statistic properties of the input variable of Pu

Variables	Indicators				
	Category	Min	Max	Avg	St. Dev
D (mm)	Input	300	400	376.5	42.51
DES1 (m)	Input	3.4	5.4	3.961	0.486
DES2 (m)	Input	1.72	8	6.698	1.756
ADES3 (m)	Input	0	1.18	0.339	0.448
PTE (m)	Input	1.95	3.4	2.66	0.575
Ge (m)	Input	3.27	3.72	3.517	0.072
EPTE (m)	Input	1.06	4.45	2.906	0.593
Pe (m)	Input	8.52	15.58	13.66	1.893
SPTs	Input	5.82	14.7	11.01	2.287
SPTt	Input	4.57	7.73	7.012	0.756
Pu (KN)	Output	407.2	1551	1069.4	360.6



	D	DSE1	DSE2	DSE3	PTE	Ge	EPTE	Pe	SPTs	SPTt	Pu
D	1										
DSE1	0.64072	1									
DSE2	0.46174	0.32872	1								
DSE3	0.42054	0.4475	0.56387	1							
PTE	-0.71406	-0.93539	-0.51516	-0.67165	1						
Ge	0.43617	0.35674	0.33288	0.20279	-0.37651	1					
EPTE	-0.48148	-0.46852	-0.33111	-0.81018	0.62766	-0.13501	1				
Pe	0.47424	0.37758	0.98901	0.67189	-0.57137	0.33383	-0.42223	1			
SPTs	0.57685	0.5723	0.94663	0.71904	-0.73196	0.37144	-0.53311	0.96932	1		
SPTt	0.19721	0.04964	0.92323	0.61934	-0.28889	0.19742	-0.30369	0.93151	0.8275	1	
Pu	0.73468	0.70626	0.78549	0.47405	-0.77993	0.46043	-0.33594	0.78508	0.84594	0.55778	1

Fig. 1 The correlation plot between input and output

require special attention or feature selection. It was observed that SPTs and EPTE exerted the most significant and minimal influences on P_u outcomes, respectively. Conversely, Pe and DSE2 emerged as the primary contributors to variations in SPTs values.

Multi-layer perceptron (MLP)

The multi-layer perceptron (MLP) is a vast, widely utilized neural network strategy generally trained with the backpropagation algorithm. The MLP is called the assessment and training art because it is developed for asset processes and learning derivation. MLP neural networks are also called tools for nonlinear processes and modeling complicated and happening in the real world because of their conformable approximation capabilities [46]. The anatomy of MLP is separated into three attached layers: output, input, and hidden. Some nodes in the input layer show the predictor variables' number.

In addition, a single hidden layer of MLP can suitably model involved functions with concealed neurons. A small number of neurons causes poor neural network function.

Against that, MLP neural nets are challenging to train but also inclined to overfitting. The output layer nodes are linked to the number of modeled variables.

For the nonlinear function (h) generalization, the function modeling task with one prophesier uses an MLP neural net as $X \in R^D \rightarrow Y \in R^1$. X and Y are the input and output parameters, respectively. The function (h) is represented in Eq. (1):

$$Y = h(X) = s_2 + M_2 \times (k_a(s_1 + M_1 \times X)) \tag{1}$$

Here

M_2 and M_1 displays the output and hidden layers' weight matrixes alternatively.

s_2 and s_1 are the output and hidden layers' bias vectors, respectively.

k_a is the function of activation.

The log-sigmoid and tan-sigmoid activation functions are widely used. Their equations have been denoted, respectively, in the Eqs. (2) and (3):

$$h_b(T) = \frac{1}{1 + \exp(-T)} \tag{2}$$

$$h_b(T) = \frac{\exp(T) - \exp(-T)}{\exp(T) + \exp(-T)} \tag{3}$$

where T shows the input activation function

Crystal Structure Algorithm (CSA)

Solid minerals contain molecules, atoms, and origins that have crystallographic forms named crystals. Kepler in 1619, Hooke in 1665, and Hogens in 1690 discovered the particles inside the crystals [47]. Lattice is the underlying element of a crystal that shows a cyclical queue of atoms in preplanned spaces. Only the overall figure of the crystal is specified by the lattice so that different geometrical figures can be composed in the light of infinite geometrical figures discovered in nature. An intermittent structure of the crystal is determined taking into account a boundless grid figure where any grid point exists related to the position of its grid spot using a course like this [48]:

$$r = \sum m_i d_i \tag{4}$$

where m_i is a whole number, d_i represents the briefest course along the primary crystalline axes, and i denotes the count of quartz vertices.

Within this part, the measured representation of CSA exists as given, wherein the main notions of crystalline structures are employed within essential alterations. Crystals numbers are random numbers for initialization.

$$Cr = \begin{bmatrix} Cr_1 \\ Cr_2 \\ \vdots \\ Cr_i \\ \vdots \\ Cr_n \end{bmatrix} = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^j & \cdots & x_1^d \\ x_2^1 & x_2^2 & \cdots & x_2^j & \cdots & x_2^d \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^1 & x_i^2 & \cdots & x_i^j & \cdots & x_i^d \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^1 & x_n^2 & \cdots & x_n^j & \cdots & x_n^d \end{bmatrix} \tag{5}$$

$$\begin{cases} i = 1, 2, 3, \ldots, n \\ j = 1, 2, 3, \ldots, j \end{cases}$$

$$x_i^j(0) = x_{i,min}^j + \xi\left(x_{i,max}^j - x_{i,min}^j\right), \begin{cases} i = 1, 2, 3, \ldots, n \\ j = 1, 2, 3, \ldots, d \end{cases} \tag{6}$$

Here, n and d are the number of crystals and the dimension of the problem, respectively, And $x_i^j(0)$ determines the primary position of the crystals; $x_{i,max}^j$ and $x_{i,min}^j$ are the maximum and minimum allowable numerical amounts, correspondingly, for the j^{th} choice parameter of the i^{th} potential resolution; and ξ represents an arbitrary number within the range of $[0, 1]$.

Due to the crystallography and the notion of 'foundation' in it, the main crystallines, Cr_m, are all the crystals at the angles, and the primary crystallines are considered haphazardly from the primary-formed crystallines. By dropping the current Cr, the haphazard choice procedure for each phase is established F_c is the mean values of randomly selected crystals and Cr_b is the crystal with the *best* configuration.

Basic lattice principles are considered for updating the candidate solutions in four sorts of improving processes are specified as follows:

Cubicles;

Simple:

$$Cr_n = Cr_o + aCr_m \tag{7}$$

With the best crystals:

$$Cr_n = Cr_o + a_1 Cr_m + a_2 Cr_b \tag{8}$$

With the mean crystals:

$$Cr_n = Cr_o + a_1 Cr_m + a_2 f_c \tag{9}$$

With the best and mean crystals:

$$Cr_n = Cr_o + a_1 Cr_m + a_2 Cr_b + a_3 F_c \tag{10}$$

In the four equations above, $\mathrm{Cr_n}$ and $\mathrm{Cr_o}$ denotes the new position and the old position, respectively, also a, a_1, a_2 and a_3 are random numbers.

Exploitation and exploration from metaheuristics, as two crucial attributes, have been used in this procedure via the cubicle Eqs. (7) to (10). The maximum number of iterations is the terminating criterion, and the enhancement procedure is ended following a predetermined count of cycles. To address the resolution parameters x_i^j transgressing the border situation of the parameters, a measured indicator is specified in which for the x_i^j beyond the range of the parameters, the indicator prompts an adjustment to the limits for the transgressing parameters.

The pseudo-code of the CSA is as follows:

Procedure *CryStAl*

Produce unpredictable amounts for main locations (x_i^j) *of main CRs* ($\boldsymbol{Cr_i}$)

Evaluate the values of fitness for every CR

while ($t <$ *iterations highest number)*

for $i = 1$ *: initial crystals number*

Produce $\boldsymbol{Cr_m}$

Produce new crystals using Eq.(7)

Produce $\boldsymbol{Cr_b}$

Produce novel CRs using Eq.(8)

Produce $\boldsymbol{F_c}$

Produce novel CRs using Eq.(9)

Produce novel CRs using Eq.(10)

if *novel CRs conditions violate boundary*

Command the location restraints for novel CRs and revise them.

end if

Assess the fitness amounts for novel CRs

Refresh (GB) if a superior resolution is discovered

end for

$t = t + 1$

end while

Revert GB

End procedure

Fox Optimization (FOX)

One of the new optimization algorithms, the Red Fox Optimization Algorithm (FOX), originated from the hunting lifestyle of the red fox. FOX has two sections: exploitation and exploration. The exploitation section of the model happens by getting the fox close to the victim to attack it, and the exploration section depends on the distance between the fox and the victim. The population of a constant number of foxes is represented below [49]:

$$\overline{x} = (x_0, x_1, \ldots, x_{n-1}) \tag{11}$$

For recognizing each fox $\overline{x}^t$ in repetition, the notation $\left(\overline{\mathcal{X}}_j^i\right)^t$ is introduced, i represents the number of the foxes, and as per the measurements of the resolution area j denotes coordinates. The notation $(\overline{x})^{(i)} = \left[(x_0)^{(i)}, (x_1)^{(i)}, (x_2)^{(i)}, \ldots, (x_{n-1})^{(i)}\right]$ represents every fact in the solution planetary $< a, b >^n$ and $a, b \in \mathbb{R}$, also according to the solution space

functions, let $f \in \mathbb{R}^n$ be the standard function of n variables. If function $f\left((\overline{x})^{(i)}\right)$ amount is a worldwide maximum and minimum on $< a, b >$, then $\left((\overline{x})^{(i)}\right)$ is the optimal solution.

When the foxes cannot find prey to hunt, members of a family travel in search of food. They send the location to others when they find a better area. By the cost amount, the population is provided. Euclidean distance square is used for this goal:

$$d\left(\left(\overline{x}^i\right)^t, \left(\overline{x}^b\right)^t\right) = \sqrt{\|\left(\overline{x}^i\right)^t - \left(\overline{x}^b\right)^t\|}, \tag{12}$$

Here $\left(\overline{x}^b\right)$ means $\left(\overline{x}^{best}\right)$, and individuals in the population move toward the best one:

$$\left(\overline{x}^i\right)^t = \left(\overline{x}^i\right)^t + \alpha * sign * \left(\left(\overline{x}^b\right)^t - \left(\overline{x}^i\right)^t\right) \tag{13}$$

Here $\alpha \in \left(0, d\left(\left(\overline{x}^i\right)^t, \left(\overline{x}^b\right)^t\right)\right)$ is randomly selected. The random value $\beta \in < 0, 1 >$ is implemented, set once in the repetition for all individuals in the population, which describes the action of the fox as:

$$\begin{cases} Stay\ and\ masquerade\ if\ \beta\ \leq 0.75 \\ Move\ closer\ if\ \beta\ > 0.75 \end{cases}$$

An advanced Cochleoid equation is used to visualize the action of each individual if β displays to move the population in this repetition. The fox radius is represented by two items: to model the fox observation angle, $\phi_0 \in < 0, 2\pi >$ is chosen for all individuals at the inception of the algorithm, and $\alpha \in < 0, 0.2 >$ is a grading variable group previously in the repetition for all members in the populace to simulate altering proximity randomly away from the victim throughout dodger getting closer.

$$r = \begin{cases} a\frac{sin\phi_0}{\phi_0} if\, \phi_0 \neq 0 \\ \delta if\, \phi_0 = 0 \end{cases} \tag{14}$$

Here $\delta \in < 0, 1 >$, and it is a random value established previously at the inception of the procedure, which is dependent on the conditions of weather. The movement model for the population of individuals is as follows:

$$\begin{cases} x_0^{new} = ar * cos(\phi_1) + x_0^{ac} \\ x_1^{new} = ar * sin(\phi_1) + ar * cos(\phi_2) + x_1^{ac} \\ x_2^{new} = ar * sin(\phi_1) + ar * sin(\phi_2) + ar * cos(\phi_3) + x_2^{ac} \\ \cdots \\ x_{n-2}^{new} = ar * \sum_{q=1}^{n-2} sin\left(\phi_q\right) + ar * cos(\phi_{n-1}) + x_{n-2}^{ac} \\ x_{n-1}^{new} = ar * sin(\phi_1) + ar * cos(\phi_2) + \cdots + ar * sin(\phi_{n-1}) + x_{n-1}^{ac} \end{cases} \tag{15}$$

ac in x_0^{ac} represented actual, and $\phi_1, \phi_2, \phi_3, \ldots, \phi_{n-1} \in < 0, 2\pi >$.

For modeling this action in each repetition, 5% of the worst applicants are selected in line with the amount of function of criterion. This value is utilized as a personal presumption for simulating minor variations among the group. In iteration t, for an alpha couple, the two best individuals are selected:

$\left(\overline{x}^{(1)}\right)^t$ and $\left(\overline{x}^{(2)}\right)^t$, and the center of the habitat is calculated as following equation, and the habitat is the square of the Euclidean distance between the couple, respectively:

$$\left(Hab^{cntr}\right)^t = \frac{\left(\overline{x}^{(1)}\right)^t + \left(\overline{x}^{(2)}\right)^t}{2} \tag{16}$$

$$\left(Hab^{diamtr}\right)^t = \sqrt{\| \left(\overline{x}^{(1)}\right)^t - \left(\overline{x}^{(2)}\right)^t \|} \tag{17}$$

A random parameter $q \in < 0, 1 >$ is taken for each iteration, which specifies replacements in the repetition following:

$$\begin{cases} Reproduction\ Of\ The\ Alpha\ Couple\ if\ q\ < 0.45 \\ New\ Nomadic\ Individual\ if\ q\ \geq\ 0.45 \end{cases} \tag{18}$$

Two best candidates $\left(\overline{x}^{(1)}\right)^t$ and $\left(\overline{x}^{(2)}\right)^t$ combined with a new candidate $\left(\overline{x}^{(rep)}\right)^t$, *rep* means reproduced, as:

$$\left(\overline{x}^{(rep)}\right)^t = q\frac{\left(\overline{x}^{(1)}\right)^t + \left(\overline{x}^{(2)}\right)^t}{2} \tag{19}$$

The pseudo-code of the FOX optimization algorithm is as follows:

Start,
Define parameters of the algorithm: fitness function $f(0)$, number of repetitions T, fox observation angle ϕ_0, the maximum size of the population n, weather conditions θ, size of search space solution < a,b >
Generate a population consisting of n foxes at random within the search space,
$t := 0$
while $t \leq T$ do
Define coefficients for iteration: fox approaching change a, scaling parameter α,
for every fox within the current populace, do
Arrange members based on the function of fitness,
Choose $(\bar{x}^b)^t$
Calculate the reallocation of individuals according to Eq. (12),
if reassignment is superior to the former location, then
Relocate the fox,
else
Revert the fox to its former location,
end if
select variable β value to define noticing the hunting fox,
if the fox is overlooked, then
Compute fox observation radius r according to Eq. (14),
Calculate reallocation according to Eq. (15),
else
Fox remains in its location to hide,
end if
end for
Organize the populace based on the function of fitness,
Worst foxes leave the herd or get killed by hunters,
New foxes are replaced in the population using Eq. (18) as nomadic foxes outside the habitat or are reproduced from the alpha couple inside the herd Eq. (19),
$t + 1$,
end while
Revert the fittest fox $(\bar{x})^b$,
Stop.

Application of CSA and FOX algorithms in the training of MLP model

In this section, the application of the FOX and CSA optimizers is illuminated in the training of the MLP. These optimization algorithms are important in fine-tuning the MLP's parameters, leading to enhanced predictive performance. The discussion encompasses the specific adaptation of FOX and CSA to the MLP architecture, detailing their impact on weight and bias adjustments, convergence behavior, and overall model optimization.

Fox Optimization Algorithm in MLP training

The integration of the FOX algorithm into the training process of the MLP allows for capitalization on its unique optimization principles. Emphasis is placed on how the weights and biases of the MLP are dynamically adjusted by FOX, fostering efficient convergence. Detailed insights into the convergence curves and the adaptive nature of the algorithm during the training iterations are provided. The thorough examination focuses on the FOX optimizer's influence on the MLP's capability to capture intricate patterns within the dataset.

Crystal Structure Algorithm in MLP training

Similarly, the employment of the CSA contributes to the refinement of the MLP's parameters, thereby enhancing the model's adaptability and predictive accuracy. In this section, the specific application of CSA in MLP training is delved into, with an emphasis on its role in guiding the optimization process. The interaction between CSA and MLP is elucidated, shedding light on how CSA optimally configures the MLP's architecture to achieve superior performance. Detailed discussions on convergence behaviors and the impact on the MLP's generalization capabilities are included.

Performance evaluation metrics

Various metrics are utilized to assess the predictive ability of the developed model in a quantitative manner. The determination coefficient (R^2) gauges the strength of the linear association among the observed and predicted outcomes. The root mean squared error (RMSE) measures the magnitude of the differences between the predicted and the observed values, and MSE is the mean square error. SI is the Scatter Index, and WAPE represents weighted absolute percentage error.

$$R^2 = 1 - \frac{\sum_{i=r}^{r}(P_i - T_i)^2}{\sum_{i=r}^{r}(T_i - \overline{T})^2} \tag{20}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=r}^{r}(P_i - T_i)^2}{r}} \tag{21}$$

$$\mathrm{MSE} = \frac{1}{r}\sum_{i=r}^{r}(P_i - T_i)^2 \tag{22}$$

$$\mathrm{SI} = \frac{\mathrm{RMSE}}{T_i} \tag{23}$$

$$\text{WAPE} = \frac{1}{r}\sum_{i=r}^{r}\frac{|T_i - P_i|}{P_i} \tag{24}$$

where P_i and T_i represent predicted and tested values, respectively. $\overline{T}$ is the average of all the tested results, while r represents the number of samples in the analyzed dataset.

K-fold cross-validation

In the process of applying *k*-fold cross-validation, the dataset is first divided into '*k*' equal folds or subsets. Subsequently, the model undergoes training '*k*' times, where each iteration utilizes a different fold as the test set while the remaining folds are used for training. This iterative process persists until each of the '*k*' folds has been employed as the test data precisely once. In this particular study, a fivefold cross-validation approach was employed, leading to the dataset's segmentation into five subsets. The model experiences five training sessions, with each session utilizing four folds for training and one fold for testing. This meticulous approach guarantees a thorough evaluation, assessing the model on each segment of the data and providing a robust appraisal of its performance. Table 3 and Fig. 2 present the results of k-fold validation for three primary metrics (R^2, RMSE, and MAE). From these findings, the outcomes of the third fold are recognized as the optimal choice, yielding values of 0.953 for R^2 and 77.851 for RMSE.

Research methodology

The approach to research methodology can be outlined as follows:

Introduction

This study introduces the examination of a pivotal issue, emphasizing the necessity for improved performance in the MLP model. The emphasis is on advancing the domain of machine learning, specifically in practical implementations within geotechnical engineering projects. The urgent requirement for heightened efficiency in the MLP model is discussed, making a valuable contribution to the broader realm of machine learning and its practical application to real-world issues in geotechnical engineering.

Hybridization procedure

This study introduces a novel machine learning methodology, incorporating the fusion of two sophisticated optimization techniques. The intricate details outline the amalgamation of optimization approaches employed to improve the effectiveness of MLP models. By strategically integrating these advanced optimization techniques, a

Table 3 The result of the developed K-fold

Model	Indicator	Number of K-fold				
		K1	*K2*	*K3*	*K4*	*K5*
MLP	R^2	0.920	0.893	0.953	0.798	0.888
	RMSE	101.769	117.467	77.851	161.844	120.727
	MAE	79.230	92.269	58.887	115.415	85.891

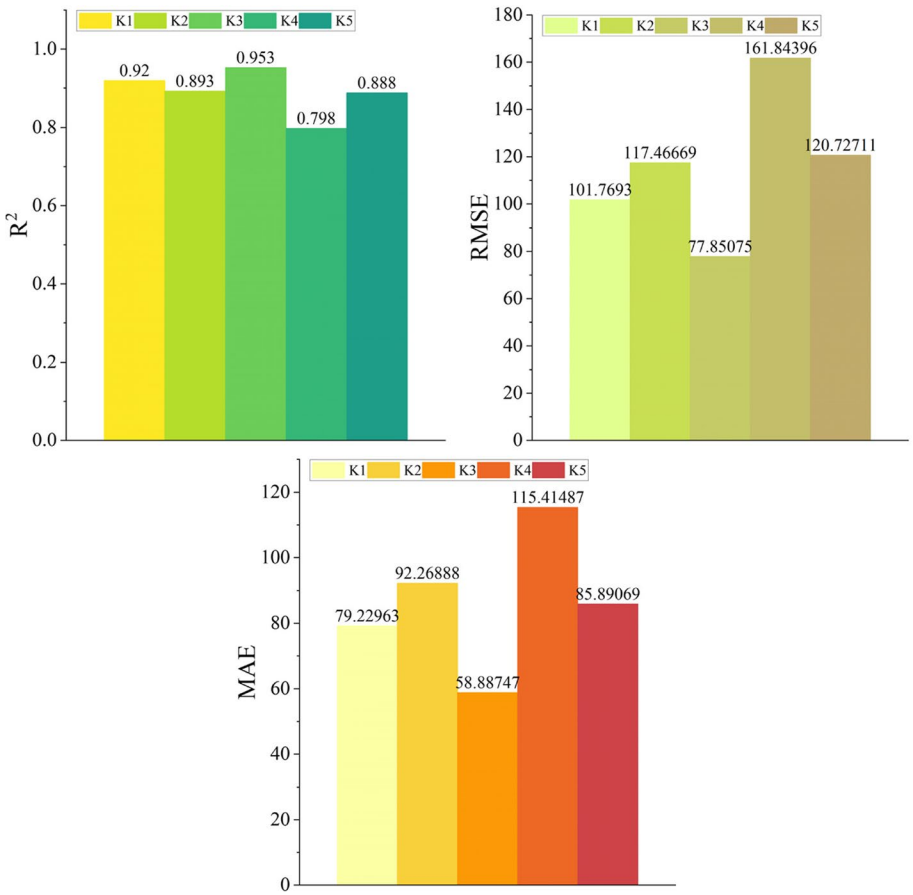


Fig. 2 Obtained fivefold result

pioneering perspective is introduced to the field of machine learning, with the primary aim of boosting the efficiency of MLP models.

Optimizers utilized

This study presents a thorough introduction and detailed explanation of two distinct optimizers utilized in the hybridization method: the CSA and the FOX. The unique strengths of each optimizer and the reasoning behind their inclusion in the hybrid model are comprehensively elucidated. This contributes to a comprehensive understanding of the strategic integration of these optimizers within the research framework.

Assessment of models

This study conducts a thorough assessment of both conventional and hybridized MLP models, employing established performance metrics like R^2 and RMSE. The selection of these metrics is justified to ensure an unbiased evaluation of model performance, thereby enhancing the reliability and objectivity of the assessment process.

Comparing the applicability of predictive models

This study meticulously contrasts the performance of hybridized models with conventional MLP counterparts, underscoring the superiority of the proposed methodology. The incorporation of rigorous statistical analyses or visual representations of results enhances the credibility and precision of the comparative evaluation between these two model types.

Result, discussion, and conclusion

This section encapsulates a concise summary of the research's significant findings and their implications, providing a brief overview of the study's outcomes. Furthermore, it explores the study's limitations and proposes potential avenues for future research, aiming to stimulate further exploration in related domains.

Figure 3 offers a visual representation that illustrates the procedural steps taken in this study. This graphical depiction complements and improves the understanding of the textual insights.

Results and discussion

Hyperparameter and convergence

In the field of machine learning, external configurations called hyperparameters, including factors like learning rates and regularization strengths, exert influence on

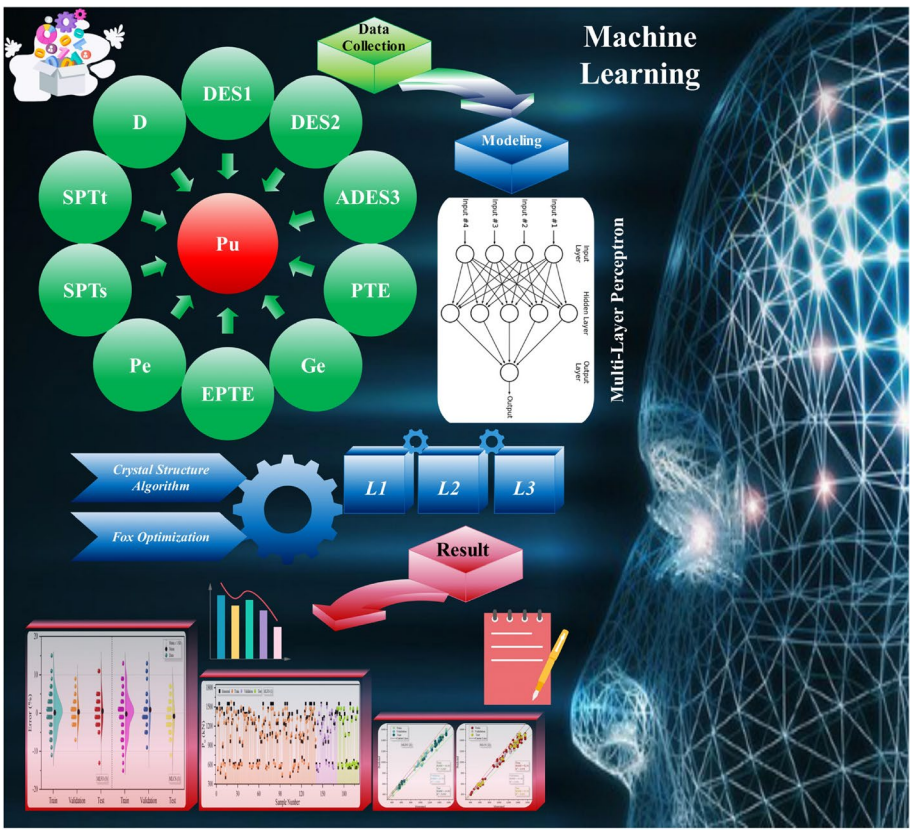


Fig. 3 Flowchart related to research methodology

Table 4 The results of hyperparameters for MLP

Models	Hyperparameter		
	Neuron	Neuron	Neuron
MLFO (1)	3	–	–
MLCS (1)	13	–	–
MLFO (2)	25	19	–
MLCS (2)	15	19	–
MLFO (3)	19	25	27
MLCS (3)	16	20	20

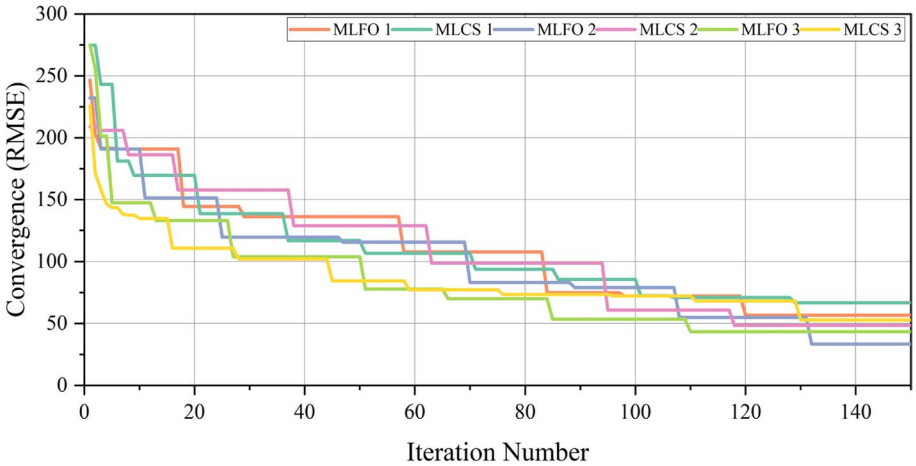


Fig. 4 Convergence of developed hybrid models

the behavior of a model. Unlike parameters, hyperparameters are predetermined and are not learned directly from the data. Importantly, the optimization of model performance relies on the crucial step of tuning hyperparameters, which necessitates experimentation and the application of optimization techniques [50–52]. Table 4 meticulously outlines the hyperparameter values associated with MLFO and MLCS models within the three layers of the MLP. This detailed presentation significantly enhances the transparency and replicability of models in the field of machine learning research, providing crucial insights for a deeper understanding and accurate reproduction of model configurations.

Figure 4 presents a graph illustrating the progression of RMSE during iterations. The x-axis denotes the iteration number, while the *y*-axis represents RMSE. The line graph commences with a high RMSE, gradually decreasing with each iteration and ultimately converging to a low RMSE after about 150 iterations. Among all the models, the MLFO model, resulting from the integration of the FOX optimizer into the MLP model's second layer, demonstrated the most favorable performance in the convergence process. It initiated with an RMSE of 230 in the first iteration and reached the optimal RMSE value of approximately 40 after 150 iterations.

Comparison of models' performance

In the current research study, MLP comprising three layers is augmented by integrating CSA and FOX optimizers to create two distinct hybrid models, MLCS and MLFO. These hybrid models serve the purpose of comparing experimentally measured results with predicted values of P_u. The dataset employed in constructing these hybrid models is partitioned into three phases: training, validation, and testing, constituting 70%, 15%, and 15% of the overall model data, respectively. The outcomes of the comparative analysis between the MLP single model and the two hybrid models across the three layers of MLP are succinctly summarized in Table 5. This analysis involves a meticulous layer-by-layer evaluation of the models, with a focused examination of the contributions and characteristics unique to each layer.

MLP single model

During the testing phase, the MLP model demonstrated its highest R^2 value at 0.973, underscoring the effectiveness of the training process. However, this peak R^2 value was accompanied by error-based metrics that revealed certain limitations. The recorded

Table 5 The result of developed models for MLP

Model	Phase	Index values				
		RMSE	R^2	MSE	SI	WAPE
MLP	Train	73.947	0.958	5467.267	0.068	0.052
	Validation	102.375	0.908	10480.576	0.095	0.072
	Test	66.336	0.973	4400.528	0.066	0.055
	All	77.851	0.953	6059.253	0.073	0.055
MLFO(1)	Train	60.497	0.971	3660.633	0.056	0.049
	Validation	43.777	0.983	1916.407	0.041	0.036
	Test	49.634	0.984	2463.503	0.050	0.042
	All	56.737	0.975	3219.429	0.053	0.046
MLCS(1)	Train	69.496	0.962	4830.148	0.064	0.049
	Validation	73.419	0.953	5390.290	0.068	0.046
	Test	43.980	0.988	1934.221	0.044	0.035
	All	66.932	0.965	4479.780	0.063	0.047
MLFO(2)	Train	36.665	0.990	1344.348	0.034	0.028
	Validation	23.992	0.995	575.592	0.022	0.015
	Test	24.884	0.996	619.231	0.025	0.020
	All	33.470	0.992	1120.267	0.031	0.025
MLCS(2)	Train	52.308	0.978	2736.144	0.048	0.041
	Validation	41.486	0.985	1721.120	0.038	0.029
	Test	35.395	0.992	1252.771	0.035	0.030
	All	48.594	0.982	2361.385	0.045	0.038
MLFO(3)	Train	46.222	0.983	2136.444	0.043	0.033
	Validation	29.642	0.993	878.622	0.027	0.023
	Test	41.627	0.989	1732.803	0.042	0.030
	All	43.442	0.986	1887.225	0.041	0.031
MLCS(3)	Train	56.381	0.975	3178.861	0.052	0.037
	Validation	46.237	0.981	2137.902	0.043	0.031
	Test	40.957	0.991	1677.480	0.041	0.031
	All	52.891	0.978	2797.510	0.049	0.036

values include 102.375 for RMSE, the highest among all seven models (comprising both the single model and hybrid models), along with 10480.576 for MSE, 0.095 for SI, and 0.072 for WAPE. When collectively considering these metrics, the MLP single model occupies the lowest position in the superiority ranking among the evaluated models. This suggests that, despite achieving a notable R^2 value, the MLP single model faces challenges in terms of error-based performance metrics compared to the other models in the study.

First layer

The maximum R^2 value of 0.975 occurred in this layer for MLFO, indicating that this model fits the data well and that the selected input variables are good predictors of the expected output. Minimum error values of 56.737, 3219.429, and 0.046 for RMSE, MSE, and WAPE confirm the accuracy of MLFO in P_u prediction. SI offers insights into data spread, outlier identification, and overall dataset consistency. Minimum $SI = 0.053$ represents low data variability and high accuracy of MLFO.

Second layer

For MLFO, it is noteworthy that this specific layer exhibited exceptional performance, with the maximum R^2 value reaching an impressive 0.992 and the lowest RMSE recorded at 33.470. MLFO demonstrated a noteworthy advantage when considering SI values, showcasing approximately a 31% reduction in SI compared to its counterpart, MLCS. This substantial reduction in SI suggests that MLFO offers superior predictive accuracy and minimized data variability in its estimations.

Third layer

MLFO exhibited superior efficiency compared to MLCS. When comparing this specific layer to the two preceding layers, it becomes evident that MLFO(3) displayed a commendable performance, characterized by an R^2 value of 0.986, an RMSE of 43.442, and a SI of 0.041. It is worth noting that MLFO(3) outperformed MLFO(1) but fell short of matching the performance of MLFO(2).

Figure 5 indicates scattered representations of the correlation between predicted and measured values of P_u. The reported numbers are related to their two evaluation sets of RMSE and R^2. Generally, the RMSE functions as a distributed controller, so the lower the amount of this evaluator, the higher the density. In addition, the R^2 evaluator moves the testing and training points near the centerline. The figure contains several other variables; for instance, the centerline at coordinate $Y=X$ and two lines are drawn below and above the centerline for 10% underestimation and 10% overestimation. This figure comprises a total of seven scatter plots designed to facilitate a comparison between measured and predicted values of Pu. Each plot corresponds to a specific model: one for the MLP single model and six additional models created by integrating the MLP method with two optimizers during the training, validation, and testing phases. When conducting a comprehensive comparison across all layers, it becomes evident that the R^2 values for both MLFO(2) and MLCS(2) reside in a favorable region. This is discerned by observing that the data points associated with these models are situated close to the central line and are confined within the boundaries of two threshold lines. Such a placement

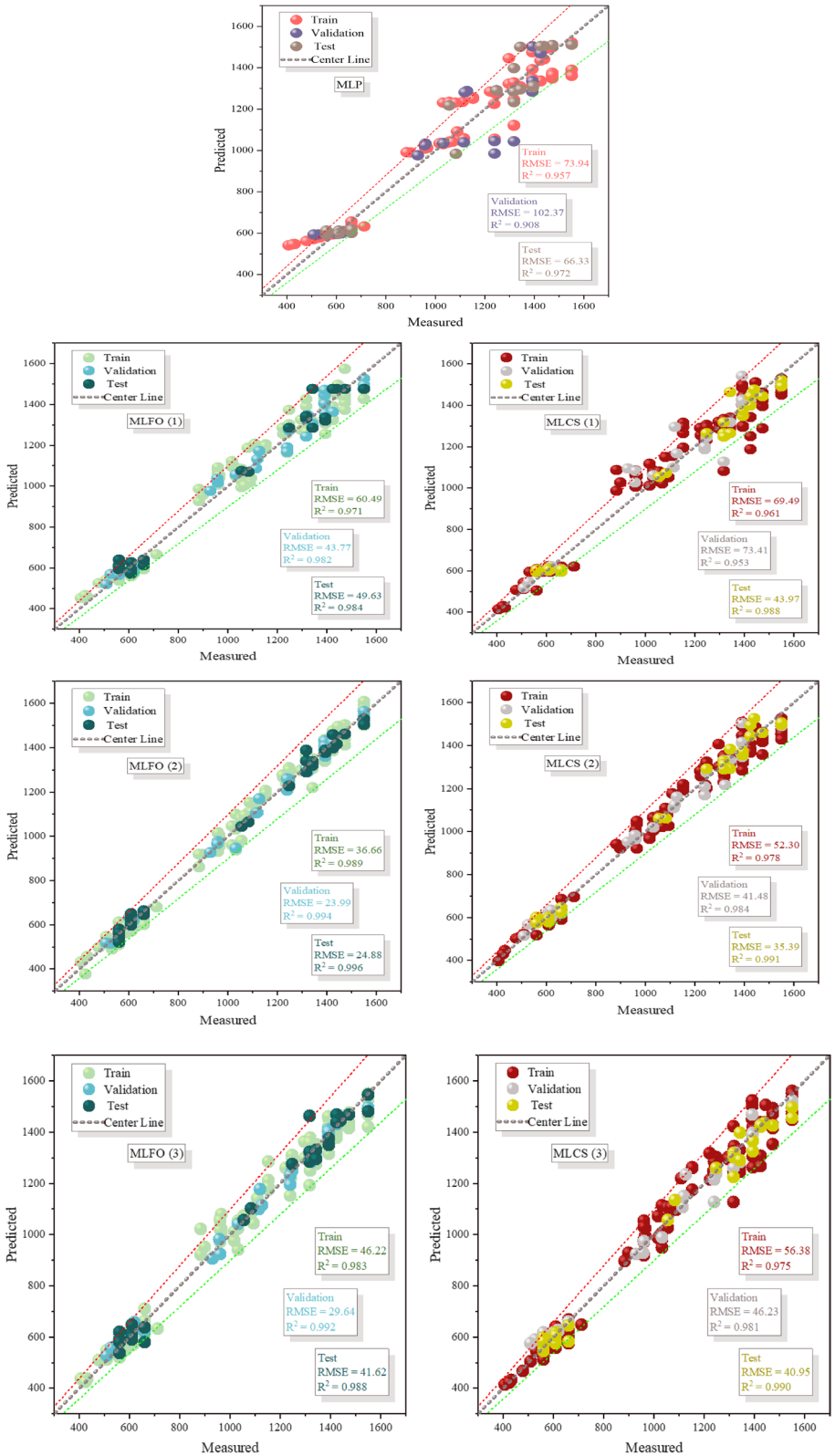


Fig. 5 A Scatter plot has points that show the relationship between three sets of data

within this region suggests that MLFO(2) and MLCS(2) models exhibit a more robust and desirable performance than other models considered in this study. It is noteworthy to highlight that the MLP single model exhibited the weakest performance among all the models, characterized by the lowest R^2 value and the highest error values. This observation underscores the relative inferiority of the MLP single model in comparison to the other models evaluated in the study.

Figure 6 evaluates the match between the predicted and measured P_u values for a single MLP model and two types of hybrid models in three layers. Each diagram is separated into training, validation, and testing models. MLFO has the most optimal performance in predicting P_u values, especially in the second layer (MLFO(2)), where the difference between the predicted and measured points was less or coincided precisely.

Upon a meticulous examination of the visual representations of error values in Fig. 7, a discernible pattern emerges. Specifically, during the training phase, the MLCS(1) model stands out for exhibiting the highest error value, surpassing 20%. In contrast, the other models demonstrate error ranges that are approximately half as large. Noteworthy is the performance of the MLFO(2) model, acknowledged for its superior accuracy, which showcases error fluctuations predominantly within the range of $[10, -10]$ percent. Remarkably, the results from the validation and testing phases consistently reveal robust performance across all model layers, emphasizing the efficacy of training the models with the provided input parameters.

Figure 8 illustrates a half-violin plot that presents error percentages for the analyzed models. In the training phase, MLFO (2) exhibited exceptional performance, showcasing an average error rate of 0% and maintaining error distribution consistently below the 5% threshold. The data depicted minimal spread, forming a tightly clustered, normally distributed pattern. Notably, the error percentage of the MLFO (2) model remained confined within the 5% limit. In contrast, the MLP model demonstrated greater dispersion and fewer close-to-zero errors, indicative of a broader range of error percentages spanning from -20 to 35%.

Sensitivity analysis

The analysis of the frequency behavior of the model output, necessary for revealing indices like the First-Order Sensitivity Index (S1) and Total-Order Sensitivity Index (ST), is conducted through the utilization of the Cosine Amplitude Method (CAM) with sinusoidal functions. Assessing the significance of parameters, aiding in model calibration, and quantifying uncertainty, these indices hold a pivotal role. Figure 9 visually captures the influence of each input parameter on predicted Pu values, revealing distinct patterns. Remarkably, the SPTs parameter has been identified as the most influential, marked by its elevated ST and S1 values. Conversely, all other inputs show negligible impacts on the Pu values.

Limitations of utilized methodology

Integrating the MLP model with CSA and FOX optimizers to predict the Pu introduces certain limitations. Firstly, the performance of the model heavily relies on the quality and representativeness of the training dataset. Inadequate or biased data may lead to suboptimal predictions. Additionally, the complexity of the MLP architecture and the interplay

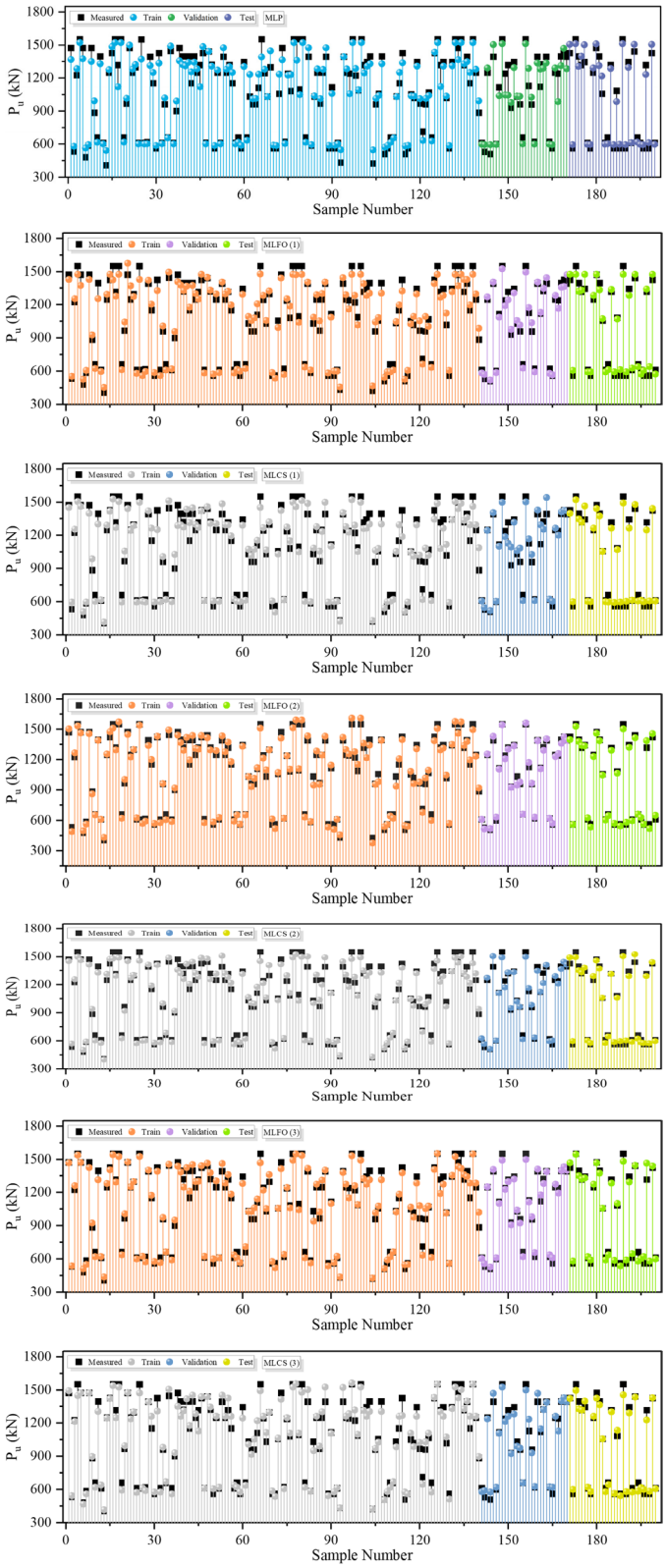


Fig. 6 The comparison of measured and predicted values

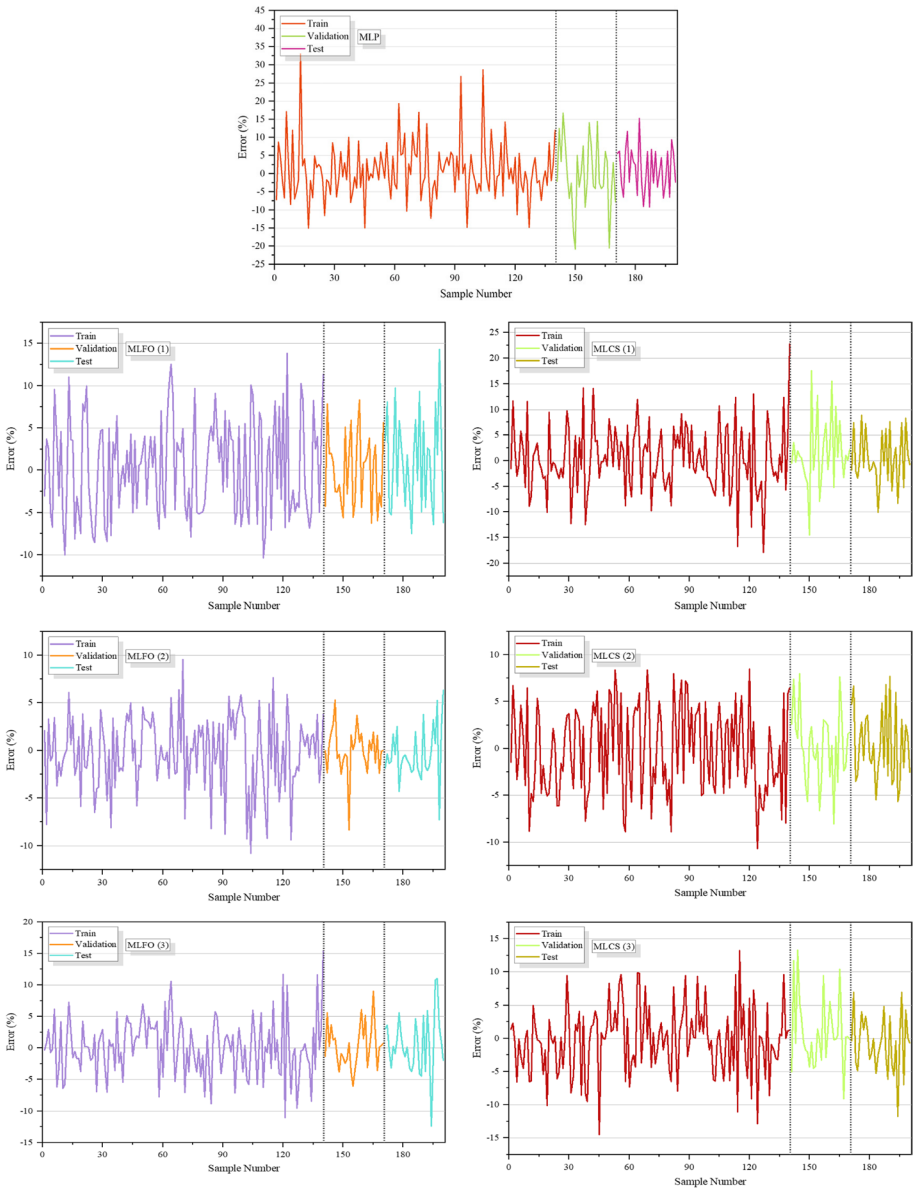


Fig. 7 The error percentage of the hybrid models is based on the line plot

with two different optimization algorithms (CSA and FOX) may result in longer training times and increased computational demands. Moreover, the effectiveness of the model could be influenced by the choice of hyperparameters and the potential need for fine-tuning, which may pose challenges in achieving optimal performance across diverse datasets.

Comparative analysis: current study vs. previous research

Table 6 presents the outcomes of previous studies in the field of Pu prediction, facilitating a comprehensive comparison with the findings of the current study. As detailed in Sect. 3.2, the investigation highlights the superior performance of the MLFO model in the second layer of MLP, achieving remarkable metrics with an R^2 value of 0.996

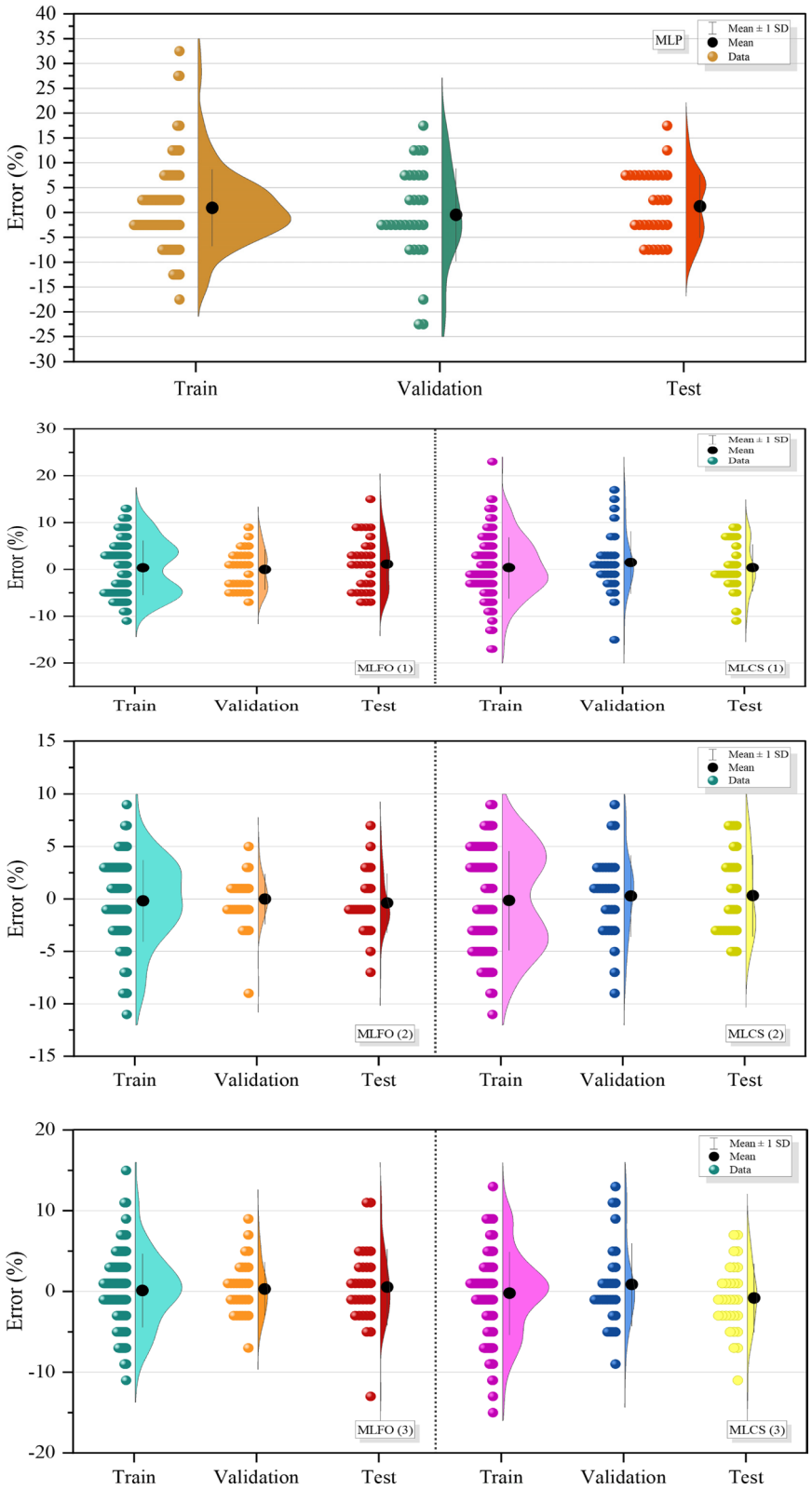


Fig. 8 The half-violin plot for the error percentage of the developed models

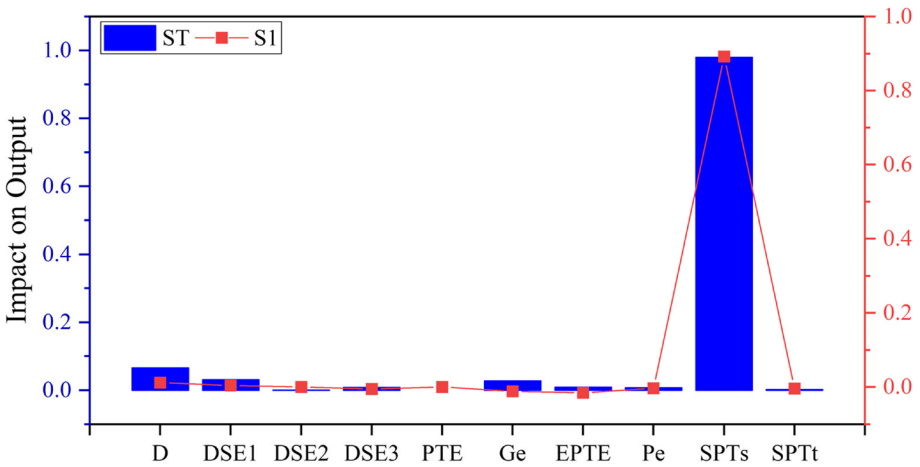


Fig. 9 The results of CAM-based Sensitivity analysis for the input parameter's impact on Pu

Table 6 Comparing the results of the present study with previous studies

Article	Model	Data size	Models' performance	
			R^2	RMSE
Gnananandarao et al. [40]	Sigmoid Symmetric	409	0.94	60
Onyelowe et al. [42]	ANN	121	0.99	14
Kumar et al. [44]	ELM-PSO	300	0.88	8
Present work	MLP-CSA-FOX	200	0.996	24.88

and an RMSE of 24.88. This simultaneous excellence in both metrics positions the MLP model in the present study as outperforming others in the comparison, emphasizing its effectiveness in Pu prediction.

Conclusions

This study introduces an innovative methodology by incorporating a multi-layer perceptron (MLP) to estimate pile-bearing capacity in the field of foundation engineering, specifically addressing the challenges posed by the resource-intensive and time-consuming nature of conventional in situ load tests. The proposed approach leverages a dataset derived from actual field-based pile load tests, providing a realistic foundation for analysis. To further elevate the prediction accuracy of the MLP model, two distinct optimizers, the Crystal Structure Algorithm (CSA) and Fox Optimization (FOX), have been deliberately chosen for integration with the MLP architecture, resulting in the creation of hybrid models, namely MLFO and MLSC. The ensuing comparative analysis, contrasting the Single MLP model against these hybrid counterparts, reveals insightful findings that can be summarized as follows:

- The MLP single model demonstrated the least effectiveness in predicting Pu, showcasing the poorest performance with the highest error values (RMSE = 102.375) and the lowest R^2 value (0.908) when compared to the hybrid models. This sub-

optimal performance underscores the necessity for an optimization process to enhance predictive accuracy.

- Among the hybrid models, the MLFO consistently outperforms MLSC in all layers, with MLFO (2) showcasing remarkable results: an R^2 value of 0.992, RMSE of 33.470, and a minimal SI of 0.031, emphasizing its superior predictive accuracy in estimating pile-bearing capacity.
- The integration of the CSA and FOX algorithms into the single MLP model yielded notable enhancements in MLP's performance, particularly evident in the improvement of R^2 values. Specifically, there was a 2.36% increase in performance when utilizing the CSA algorithm and a commendable 1.95% improvement with the FOX algorithm. This signifies the positive impact of integrating these optimization algorithms, contributing to the overall predictive capabilities of the MLP model.

Acknowledgements
I would like to take this opportunity to acknowledge that there are no individuals or organizations that require acknowledgment for their contributions.

Author's contributions
Data collection, simulation, and analysis were performed by Yuanke Shen. The first draft of the manuscript was written by Y Sh. The author read and approved the final manuscript.

Funding
This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Availability of data and materials
Data can be shared upon request.

Declarations

Competing interests
The author declares no competing interests.

Received: 9 November 2023 Accepted: 6 February 2024
Published online: 26 February 2024

References

1. Chapman T, Marcetteau A (2004) Achieving economy and reliability in piled foundation design for a building project. Structural Engineer 82(11):32–37
2. Burland JB, Broms BB, De Mello VFB (1978) Behaviour of foundations and structures
3. Kordjazi A, Nejad FP, Jaksa MB (2014) Prediction of ultimate axial load-carrying capacity of piles using a support vector machine based on CPT data. Comput Geotech 55:91–102
4. Momeni E, Nazir R, Armaghani DJ, Maizir H (2014) Prediction of pile bearing capacity using a hybrid genetic algorithm-based ANN. Measurement 57:122–131
5. Chen W, Sarir P, Bui X-N, Nguyen H, Tahir MM, JahedArmaghani D (2020) Neuro-genetic, neuro-imperialism and genetic programing models in predicting ultimate bearing capacity of pile. Eng Comput 36:1101–1115
6. Shariatmadari N, Eslami AA, Karim PFM (2008) Bearing capacity of driven piles in sands from SPT–applied to 60 case histories
7. De Kuiter J, Beringen FL (1979) Pile foundations for large North Sea structures. Mar Georesour Geotechnol 3(3):267–314
8. Bazaraa AR, Kurkur MM. N-values used to predict settlements of piles in Egypt, in Use of In Situ tests in geotechnical engineering, ASCE, 1986, pp. 462–474
9. Zhang C, Nguyen GD, Einav I (2013) The end-bearing capacity of piles penetrating into crushable soils. Géotechnique 63(5):341–354
10. Schmertmann JH (1978) Guidelines for cone penetration test: performance and design, United States. Federal Highway Administration
11. Meyerhof GG (1976) Bearing capacity and settlement of pile foundations. J Geotech Eng Div 102(3):197–228
12. Elsherbiny ZH, El Naggar MH (2013) Axial compressive capacity of helical piles from field tests and numerical study. Can Geotech J 50(12):1191–1203
13. Shooshpasha I, Hasanzadeh A, Taghavi A (2013) Prediction of the axial bearing capacity of piles by SPT-based and numerical design methods. Geomate Journal 4(8):560–564

14. Jesswein M, Liu J, Kwak M. Predicting the side resistance of piles using a genetic algorithm and SPT n-values, in Proceedings of the 71st Canadian Geotechnical Conference and the 13th Joint CGS/IAH-CNC Groundwater Conference-GeoEdmonton, 2018, pp. 1–8
15. Abu-Farsakh MY, Titi HH (2004) Assessment of direct cone penetration test methods for predicting the ultimate capacity of friction driven piles. J Geotechn Geoenvironmental Eng 130(9):935–944
16. Ozok AA. Survey design and implementation in HCI, in Human-Computer Interaction, CRC Press, 2009, pp. 269–288
17. Rausche F, Moses F, Goble GG (1972) Soil resistance predictions from pile dynamics. J Soil Mechanics Foundations Division 98(9):917–937
18. Liu P, Xing Q, Dong Y, Wang D, Oeser M, Yuan S (2017) Application of finite layer method in pavement structural analysis. Appl Sci 7(6):611
19. Dounis AI, Caraiscos C (2009) Advanced control systems engineering for energy and comfort management in a building environment—a review. Renew Sustain Energy Rev 13(6–7):1246–1261
20. Wang F-Y (2010) Parallel control and management for intelligent transportation systems: concepts, architectures, and applications. IEEE Trans Intell Transp Syst 11(3):630–638
21. Allen G, Chan T. Artificial intelligence and national security. Belfer Center for Science and International Affairs Cambridge, MA, 2017
22. Yang X, Wang Y, Byrne R, Schneider G, Yang S (2019) Concepts of artificial intelligence for computer-assisted drug discovery. Chem Rev 119(18):10520–10594
23. Akbarzadeh MR, Ghafourian H, Anvari A, Pourhanasa R, Nehdi ML (2023) Estimating compressive strength of concrete using neural electromagnetic field optimization. Materials 16(11):4200
24. Masoumi F, Najjar-Ghabel S, Safarzadeh A, Sadaghat B (2020) Automatic calibration of the groundwater simulation model with high parameter dimensionality using sequential uncertainty fitting approach. Water Supply 20(8):3487–3501. https://doi.org/10.2166/ws.2020.241
25. Kumar M et al (2022) Hybrid ELM and MARS-based prediction model for bearing capacity of shallow foundation. Processes 10(5):1013
26. M. Kumar and P. Samui, Reliability analysis of pile foundation using GMDH, GP and MARS BT - CIGOS 2021, Emerging Technologies and Applications for Green Infrastructure, C. Ha-Minh, A. M. Tang, T. Q. Bui, X. H. Vu, and D. V. K. Huynh, Eds., Singapore: Springer Nature Singapore, 2022, pp. 1151–1159
27. Shahin MA (2010) Intelligent computing for modeling axial capacity of pile foundations. Can Geotech J 47(2):230–243
28. Pham TA, Ly H-B, Tran VQ, Van Giap L, Vu H-LT, Duong H-AT (2020) Prediction of pile axial bearing capacity using artificial neural network and random forest. Appl Sci 10(5):1871
29. Shahin MA, Jaksa MB. Intelligent computing for predicting axial capacity of drilled shafts, in Contemporary Topics in In Situ Testing, Analysis, and Reliability of Foundations, 2009, pp. 26–33
30. Pham TA, Tran VQ, Vu H-LT, Ly H-B (2020) Design deep neural network architecture using a genetic algorithm for estimation of pile bearing capacity. PLoS ONE 15(12):e0243030
31. Dadhich S, Sharma JK, Madhira M (2021) Prediction of ultimate bearing capacity of aggregate pier reinforced clay using machine learning. Int J Geosynthetics Ground Eng 7:1–16
32. Fong S, Deb S, Yang X. How meta-heuristic algorithms contribute to deep learning in the hype of big data analytics, in Progress in Intelligent Computing Techniques: Theory, Practice, and Applications: Proceedings of ICACNI 2016, Volume 1, Springer, 2018, pp. 3–25
33. Ghorbani B, Sadrossadat E, BolouriBazaz J, RahimzadehOskooei P (2018) Numerical ANFIS-based formulation for prediction of the ultimate axial load bearing capacity of piles through CPT data. Geotechn Geological Eng 36:2057–2076
34. Moayedi H, Hayati S (2019) Artificial intelligence design charts for predicting friction capacity of driven pile in clay. Neural Comput Appl 31:7429–7445
35. Shahin MA, Jaksa MB (2005) Neural network prediction of pullout capacity of marquee ground anchors. Comput Geotech 32(3):153–163
36. Shahin MA (2016) State-of-the-art review of some artificial intelligence applications in pile foundations. Geosci Front 7(1):33–44
37. Shahin MA (2014) Load–settlement modeling of axially loaded steel driven piles using CPT-based recurrent neural networks. Soils Found 54(3):515–522
38. Nawari NO, Liang R, Nusairat J (1999) Artificial intelligence techniques for the design and analysis of deep foundations. Electron J Geotech Eng 4(2):1–21
39. Suman S, Das SK, Mohanty R (2016) Prediction of friction capacity of driven piles in clay using artificial intelligence techniques. Int J Geotech Eng 10(5):469–475
40. Gnananandarao T, Khatri VN, Dutta RK (2020) Bearing capacity and settlement prediction of multi-edge skirted footings resting on sand. Ingeniería e Investigación 40(3):9–21
41. Kumar M, Biswas R, Kumar DR, Pradeep T, Samui P (2022) Metaheuristic models for the prediction of bearing capacity of pile foundation. Geomechanics Engineering 31(2):129
42. Onyelowe KC, Gnananandarao T, Nwa-David C (2021) Sensitivity analysis and prediction of erodibility of treated unsaturated soil modified with nanostructured fines of quarry dust using novel artificial neural network. Nanotechnol Environ Eng 6(2):37. https://doi.org/10.1007/s41204-021-00131-2
43. Kumar M, Bardhan A, Samui P, Hu JW, Kaloop MR (2021) Reliability analysis of pile foundation using soft computing techniques: a comparative study. Processes 9(3):486
44. Onyelowe KC, Gnananandarao T, Ebid AM (2022) Estimation of the erodibility of treated unsaturated lateritic soil using support vector machine-polynomial and-radial basis function and random forest regression techniques. Cleaner Materials 3:100039
45. Dehghanbanadaki A, Khari M, Amiri ST, Armaghani DJ (2021) Estimation of ultimate bearing capacity of driven piles in c-φ soil using MLP-GWO and ANFIS-GWO models: a comparative study. Soft Comput 25:4103–4119

46. Moayedi H, Hayati S (2018) Applicability of a CPT-based neural network solution in predicting load-settlement responses of bored pile. Int J Geomech 18(6):6018009
47. Averill BA, Eldredge P. Chemistry: principles, patterns, and applications, (No Title), 2007
48. Talatahari S, Azizi M, Tolouei M, Talatahari B, Sareh P (2021) Crystal structure algorithm (CryStAl): a metaheuristic optimization method. IEEE Access 9:71244–71261
49. Połap D, Woźniak M (2021) Red fox optimization algorithm. Expert Syst Appl 166. https://doi.org/10.1016/j.eswa.2020.114107
50. Kumar M, Kumar V, Rajagopal BG, Samui P, Burman A (2023) State of art soft computing based simulation models for bearing capacity of pile foundation: a comparative study of hybrid ANNs and conventional models. Model Earth Syst Environ 9(2):2533–2551. https://doi.org/10.1007/s40808-022-01637-7
51. Kumar M, Biswas R, Kumar DR, Samui P, Kaloop MR, Eldessouki M (2023) Soft computing-based prediction models for compressive strength of concrete. Case Studies in Construction Materials 19:e02321
52. Biswas R et al (2023) A novel integrated approach of RUNge Kutta optimizer and ANN for estimating compressive strength of self-compacting concrete. Case Studies Construction Materials 18:e02163

Publisher's Note