



State of art soft computing based simulation models for bearing capacity of pile foundation: a comparative study of hybrid ANNs and conventional models

Manish Kumar¹ · Vinay Kumar² · Balaji Ganesh Rajagopal³ · Pijush Samui⁴ · Avijit Burman⁴

Received: 10 October 2022 / Accepted: 7 December 2022 / Published online: 23 December 2022
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2022

Abstract

Safety has been always challenging in geotechnical engineering owing to the inherently variable nature of the soil. In pile foundations, conducting field tests is highly expensive and time-consuming, and thus soft-computing based simulation models analysis is a realistic and useful alternative. This study presented a comparative analysis of artificial neural network (ANN)-based hybrid models and conventional soft computing techniques to estimate the probability of failure of pile foundation. With this respect, dynamic pile load test data of pile foundations were used to construct ANN-based models. Five widely used meta-heuristic optimization algorithms, namely particle swarm optimization, grasshopper optimization algorithm, artificial bee colony, ant colony optimization, and ant lion optimizer, were employed for this purpose. In addition, three widely used conventional soft computing techniques; including genetic programming (GP), multivariate adaptive regression splines (MARS), and group method of data handling (GMDH) were utilized for comparison purposes. The performances of all the developed models were assessed using various statistical performance indices. Experimental results show that the ANN-PSO (hybrid model of ANN and particle swarm optimization) and GP estimate the probability of failure of pile foundation accurately both in training and testing phases. However, a detailed review of results reveals that the ANN-PSO ($R^2 = 0.9773$, RMSE = 0.0439) and GP ($R^2 = 0.9859$, RMSE = 0.0353) showed comparatively better performance in the testing phase. The result of the ANN-PSO and GP models is significantly better than those obtained from other benchmark methods. Based on the results, the developed ANN-PSO and GP models can be used to estimate the probability of failure of pile foundation in the design phase of civil engineering projects.

Keywords Pile foundations · Dynamic pile load tests · Meta-heuristic optimization · Rank analysis

Introduction

With the desperate shortage of land and the growing tendency for high-rise structures in recent years, pile foundation is attracting massive attention owing to its ability to support high loads in weaker soils. Having soft soils underneath

our foundations will cause high total settlements, differential settlement, and bearing-capacity problems (Charlie et al. 2009; Gabrielaitis et al. 2013; Wang et al. 2016). Dynamic pile load test is increasingly used to evaluate the load capacity of the pile since it is cheaper, easier to handle that allows conducting many tests, compared to static load

✉ Vinay Kumar
vinaykumarnitp@gmail.com

Manish Kumar
mkumarod@gmail.com

Balaji Ganesh Rajagopal
balajiganesh.r@ist.srmtrichy.edu.in

Pijush Samui
pijush@nitp.ac.in

Avijit Burman
avijit@nitp.ac.in

¹ Department of Civil Engineering, SRM Institute of Science and Technology (SRMIST) Tiruchirappalli, Tiruchirappalli, TN 621105, India

² Department of Civil Engineering, Government Engineering College Nawada, Bihar 805122, India

³ Department of Computer Science and Engineering, SRM Institute of Science and Technology (SRMIST) Tiruchirappalli Campus, Tiruchirappalli, TN, India

⁴ Department of Civil Engineering, National Institute of Technology (NIT) Patna, Patna, Bihar 800005, India

tests (Rajagopal et al. 2012). More importantly, the results obtained by dynamic testing are quite similar to that of static tests (Nayak et al. 2000; Rausche et al. 2004; Bradshaw and Baxter 2006; Long 2007; Basarkar 2011; Sakr 2013; Liu et al. 2020). Therefore, contractors are encouraged to choose dynamic pile load tests as an alternative for pile testing if the code allows.

Machine learning (ML) is an area of research that allows computers to learn from observed data without being specifically programmed (Asteris et al. 2021a; Bardhan et al. 2021; Kardani et al. 2021a; Kumar et al. 2022a). Moreover, geophysical design parameters are not always collected directly from field or laboratory tests but are often approximated by developing regression fitting to datasets. Artificial neural network (ANN) as one of the commonly used ML methods has been used in estimating the bearing capacity of piles (Asteris et al. 2021b; Benali et al. 2017; Che et al. 2003; Goh and Goh 2007; Goh 2000; Moayedi 2018; Lee and Lee 1996; Jiang et al. 2016; Jiang and Zhang 2018; Kiefa 1998; Low et al. 2001; Moayedi and Hayati 2019; Pal and Deswal 2008; Pradeep et al. 2021; Shahin et al. 2009). Kiefa (1998) developed general regression neural network (GRNN)-based model to predict the bearing capacity of piles in cohesionless soils. Che et al. (2003) used the data collected from dynamic wave tests to develop a back-propagation neural network-based model to predict the bearing capacity of piles, where a feed-forward neural network of one layer and 10 neurons was built. ANN models have become increasingly popular and successfully used in various fields of geotechnical engineering (Shahin et al. 2001). Recently, Alzo'ubi and Ibrahim (2019) used backpropagation neural network and generalized regression neural network to predict accurately the pile static load test curves.

The most popular one among the various ANN models is the back-propagation (BP) algorithm. However, in the BP algorithm, the trial-and-error approach to ascertain the optimal number of hidden neurons makes it very time-consuming. To improve the simulation performance of ANN, integrating ANN with metaheuristic optimization techniques becomes preferred (Kumar et al. 2022b). The optimization techniques are used to optimize various parameters like weight and bias of the neural network to improve its performance (Kardani et al. 2021b). Benali et al. (2017) presented ANN and Principal component analysis (PCA)-based ANN to predict the axial load capacity of piles, and concluded that the results obtained by the PCA-based models were in good agreement with those of standard penetration test (SPT)-based analysis. Nguyen et al. (2020) applied hybrid ANN-based prediction of column deflection exposed to seismic conditions. Particle swarm optimization (PSO)-based model gives satisfactory results and outperforms the traditional ANN model. Murlidhar et al. (2020) applied hybrid ANN models: genetic algorithm (GA)-based ANN and particle

swarm optimization (PSO)-based ANN (ANN-PSO) in predicting pile bearing capacity. Chen et al. (2020) compared the performances of genetic programming (GP) and ANN in predicting the load capacity of piles where 50 datasets of concrete piles were collected from the literature and found that the GP model outperformed ANN, GA-ANN (hybrid model of ANN and GA) and ICA-ANN (hybrid model of ANN and imperialist competitive algorithms). Liu (2020b) compared the performance of ANN, Adaptive neuro-fuzzy inference system (ANFIS), and GA-ANN in reliability analysis of vertical settlement of Pile raft foundation. GA-ANN was proved to outperform ANN and ANFIS models.

Over the past three decades, researchers and academics have shown a growing interest in meta-heuristic optimization, leading to the regular proposal of novel meta-heuristics for the solution of complicated and real-world issues in many fields. Single-based algorithms and population-based algorithms are the two primary categories of meta-heuristics. The foundation of single-based meta-heuristic algorithms, also called trajectory algorithms, is the generation of a single solution at each iteration. This solution is made more efficient by the neighbourhood mechanism. Population-based meta-heuristic algorithms, in contrast to their single-based counterparts, produce a set of multiple solutions (population) on each iteration. Population-based meta-heuristics can be broken down into four distinct types: those based on evolution, swarm intelligence, events, and the physical sciences. Based on the principles of natural evolution, Evolutionary Algorithms (EA) use the three operators of selection, recombination, and mutation to achieve their goals. Swarm Intelligence (SI) is one example of the second category of techniques, which draws its inspiration from the study of collective behaviour in the natural world. Insects, birds, mammals, reptiles, fish, etc. The third category, which includes activities such as instructing a learning-based algorithm, is inspired less by the wonders of nature than by the actions of humans.

The study presents a comparative analysis of five hybrid ANN models namely ANN-PSO (hybrid model of ANN and particle swarm optimization), ANN-GOA (hybrid model of ANN and grasshopper optimization algorithm), ANN-ABC (hybrid model of ANN and artificial bee colony), ANN-ACO (hybrid modes of ANN and ant colony optimization), ANN-ALO (hybrid model of ANN and ant lion optimizer) and three traditional soft computing models including multivariate adaptive regression splines (MARS), GP and group method of data handling (GMDH) for estimating the probability of failure of piles. These methods have not been explored in foundation engineering earlier but found robust in literature (Alizadeh et al. 2019; Moayedi et al. 2020; Seifi et al. 2020). PSO is a widely used optimization technique of swarm intelligence family imitating bird swarm behaviours (Armaghani et al. 2020b; Kashani et al. 2020; Ray et al.

2021). GOA is based on the herding behaviour of grasshoppers. ABC algorithm follows the social cooperation behaviour of honey bees in AI (Bui et al. 2020; Huang et al. 2020; Wang et al. 2020). ACO is based on ant’s behaviour of forage and found to be very reliable in literature (Moayedi et al. 2019b; Xu et al. 2019; Zhang et al. 2020a). ALO follows the way ant lion chases the prey. Moayedi et al. (2019a) demonstrated the robust prediction of ALO-ANN and its superiority over conventional models. MARS, GP and GMDH are popular models used successfully in various geotechnical problems (Ardakani and Kordnaeij 2019; Hassanlourad et al. 2017; Kardani et al. 2021a; Mola-Abasi and Eslami 2019; Samui et al. 2019; Yin et al. 2020; Zhang et al. 2020b; Zhang and Goh 2013). In this study, based on the results of dynamic tests on piles, five hybrid ANN models and three traditional models will be thoroughly investigated for the prediction of bearing capacity of pile foundations.

materials of far lesser stiffness (Salgado 2008). Deploying a pair of accelerometer and strain transducer on top of the pile evaluates the complex monitoring of piles the reported data is transmitted through a cable to the PDA which is converted and recorded as force and speed. In the next stage, using the CAPWAP program, the bearing potential of the pile is estimated. To assess soil resistance and its distribution along with the pile, CAPWAP integrates the calculated force and velocity with wave equation analysis. It uses the iterative curve-fitting technique which matches the response of the model pile, subjected to wave analysis, to the pile under investigation for a single hammer strike (FHWA 2006). Susilo (2006) suggested some guidelines for monitoring the criteria like impact factor and hammer weight. Minimum hammer weight should be 1% of the required ultimate load capacity and enhanced to 2% of the required load capacity for piles anticipated to have high-end bearing capacity.

Methodology and theoretical background

High strain dynamic testing of piles

Dynamic testing of piles (PDA test) is an innovative method to determine the load capacity of piles (Fellenius 1999; Rausche et al. 1985, 2004; Smith 2002). One-dimension wave propagation theory is feasible to be extended to piles in the PDA test since the strike of the hammer leads to the downward propagation of waves. The uniform cross-section of the piles is postulated to be a slender element enclosed by

Details of models and meta-heuristic optimization algorithms

Artificial neural network (ANN)

ANN is a popular approximation tool to simulate and predict the output and it is developed by emulating the neural system of the human body. It comprises three parallel layers connected via weights and biases: input layer, hidden layer, and output layer as shown in Fig. 1 (Moayedi and Rezaei 2019). Backpropagation (BP) is the most popular learning tool applied in feedforward ANN models. It uses the gradient

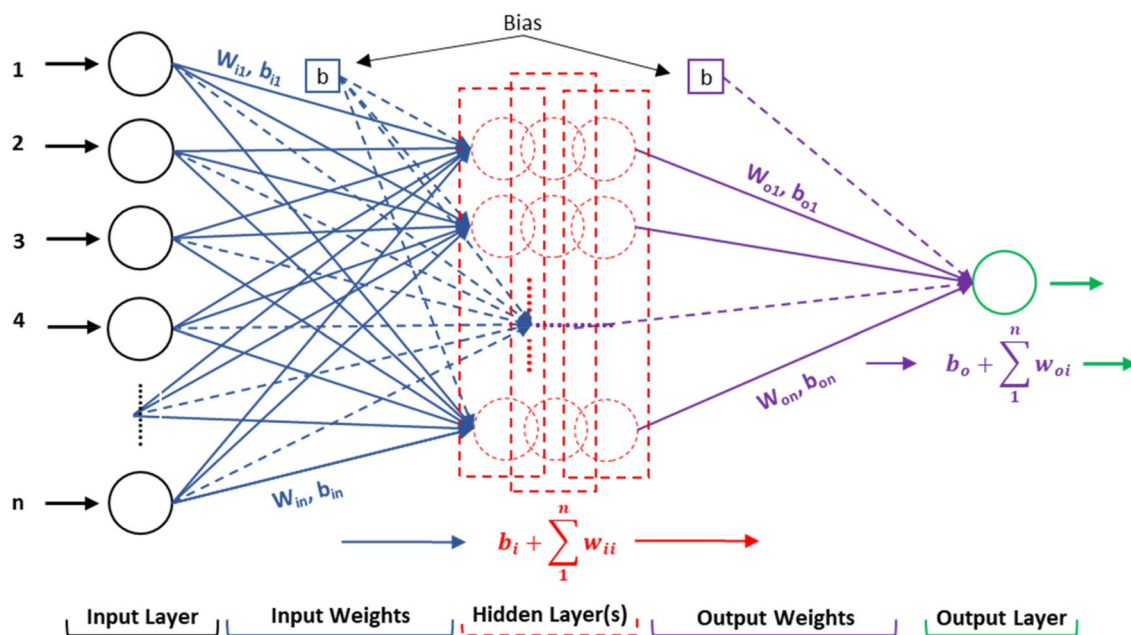


Fig. 1 A basic structure of an ANN

descendent optimization technique. Powered by high neuronal interconnections, the ANN can handle complex and non-linear correlations between input and output variables. The number of neurons in the hidden layer can be adjusted by the users to obtain the best performance. Through the harnessing of such a structure, ANNs have been employed as effective soft computing techniques for different purposes such as function approximation and pattern recognition in many engineering disciplines.

Particle swarm optimization (PSO)

PSO is a widely used optimization technique that belongs to the swarm intelligence family, proposed by Kennedy and Eberhart (1995). The principal origin of impulse for the PSO algorithm is to gather and school patterns among birds and fish. So that the central goal of this algorithm is to provide a universal best resolution in multidimensional space. PSO performs the search of the optimal solution through particles, whose trajectories are adjusted by a stochastic and a deterministic component. Each particle is influenced by its ‘best’ achieved position and the group ‘best’ position but tends to move randomly. In PSO, the population P is represented by:

$$P = (p_1, p_2, p_3 \dots p_n) \quad (1)$$

The velocities of the individual particle are denoted by:

$$u = (u_1, u_2, u_3 \dots u_n) \quad (2)$$

Previously visited best location (l_{best}) is shown as:

$$l = (l_1, l_2, l_3 \dots w_n) \quad (3)$$

The swarm is updated as follows {for ($i = 1, 2, \dots, n$) and k being current iteration}:

$$u_i^{k+1} = w^k u_i^k + d_1 k_1^k (l_i^k - w_i^k) + d_2 k_2^k (l_g^k - w_i^k) \quad (4)$$

$$w_i^{k+1} = w_i^k + v_i^{k+1} \quad (5)$$

where n is the total dimension, l_g denotes the best particle, and superscripts is used for the number of iterations. w is weight and d_1, d_2 are two learning factors called cognitive and social parameters, respectively (position constants). The best performance of the model requires proper tuning of the two position constants. k_1 and k_2 are uniformly distributed random numbers in the range of 0–1. Unlike evolutionary algorithms, PSO does not use Darwinian principles of ‘survival of fittest’ or genetic operators. In POS, the sociometric principle of exchange of information between the experience of the individual swarm and best performer is adopted as its working principle (Gaitonde and Karnik 2012).

Grasshopper optimization algorithm (GOA)

Based on the harm they cause to agriculture; grasshoppers are acknowledged as pests. The grasshopper optimization algorithm (GOA) imitates the action of the swarm of grasshopper finding a food source in nature. Grasshoppers do not act as an individual but form some of the largest swarms of all living organisms. Swarm motion is influenced by the interactions of individuals in a swarm, wind, gravity, and food sources, etc. Like large rolling cylinders, millions of grasshoppers jump and proceed. Saremi et al. (2017) proposed a mathematical model for this action given by:

$$P_i = R_i + G + W \quad (6)$$

where P_i represents the position of the i th grasshopper, R_i represents social interaction, G is the gravity force on i th grasshopper and W is the wind direction. The advanced formulation of the expression can be given by:

$$P_i = \sum_{\substack{j=1 \\ j \neq i}}^N r \left(|P_j^k - P_i^k| \right) \frac{P_j^k - P_i^k}{d_{ij}} - g \hat{e}_g + w' \hat{e}_w \quad (7)$$

where r is a function that simulates the effect of social interactions of N individual grasshopper which can be expressed as:

$$r(p) = f e^{-\frac{p}{i}} - e^{-p} \quad (8)$$

where d_{ij} is the distance between two grasshoppers (say at points i and j) and given by:

$$d_{ij} = |P_j^k - P_i^k| \quad (9)$$

If g is the gravitational constant and \hat{e}_g represents a unit vector towards the centre of the earth, gravitational force G is given by:

$$G = -g \hat{e}_g \quad (10)$$

If w' is the wind drift constant and \hat{e}_w represents a unit vector towards the direction of the wind, the wind drift effect W is given by:

$$W = w' \hat{e}_w \quad (11)$$

The effects of wind and gravity are much weaker than the relationship between grasshoppers since grasshoppers easily find secure zones and show low convergence. Thus, the model’s modified version of Eq. (6) can be re-written as:

$$P_i = c^k \left(\sum_{\substack{j=1 \\ j \neq i}}^N c^k \frac{P_i^{pub} - P_i^{lb}}{2} r \left(|P_j^k - P_i^k| \right) \frac{P_j^k - P_i^k}{d_{ij}} \right) + \widehat{T}_d \tag{12}$$

where *ub* and *lb* are the upper and lower bound, respectively, associated with the variables. *c* is a decreasing coefficient described in Eq. 13. \widehat{T}_d is the value of the variable in the target (best solution obtained so far).

$$c^k = c_{max} - (c_{max} - c_{min}) \frac{k}{k_{max}} \tag{13}$$

where *c_{max}* and *c_{min}* are 1 and 0.00001 respectively in the present work. The higher the value of *c*, the more is the swarm exploration.

Artificial bee colony (ABC)

ABC algorithm (Karaboga 2005; Tereshko and Loengarov 2005) is a metaheuristics optimization approach that shadows the behaviour of honey-swarm bees of social cooperation into machine learning. It divides honey bees into two types: employed and unemployed. Unemployed bees are further sub-divided into onlooker and scout bees. Employed bees first attack the food sources and search for other food sources in the neighbourhood, which represents possible solutions. There are as many employed bees as the number of food sources. The onlooker bees observe the motion of the employed bees and based on the information generated about the amount of nectar i.e., the fitness value of the solutions, and selects the food sources to be exploited. It memorises the best solutions and abandons the poor ones. When food sources get finished, employed bees become scout bees and search for further food sources to replenish the abandoned food sources.

Let *u_i* (*i* = 1; 2; 3; ...; *m*) be the food source. Neighbourhood food sources or possible solutions are expressed by:

$$V_i = x_i + \mu_i(x_i - x_j) \tag{14}$$

where μ_i is a random number between -1 and 1 , *x_j* is chosen randomly, (*j* = 1, 2, ...*n*; *j* ≠ *i*). Based on the information, onlooker bee chooses food sources based on the following probability:

$$P_i = \frac{f_i}{\sum_{n=1}^t f_n} \tag{15}$$

where *t* is the total number of sources *f_i* is the amount of nectar or fitness value of the *i*th source and it is calculated using an objective function $\phi(x_i)$:

$$f_i = \begin{cases} \frac{1}{1+\phi(x_i)}, & \phi(x_i) \geq 0 \\ 1 + |\phi(x_i)|, & \phi(x_i) < 0 \end{cases} \tag{16}$$

Scout bees replenish the abandoned food sources with new ones by the following expression:

$$x_i = x_{min} \max_{x_{min}} \tag{17}$$

where *x_{max}* is the upper bound and *x_{min}* is the lower bound of *x_i*. The iteration is repeated till the termination condition is met.

Ant colony optimization (ACO)

ACO simulates the food-searching behaviours of ants (Dorigo et al. n.d.; Dorigo and Blum 2005; Dorigo and Socha 2007). Artificial ants search for the best solutions in the parameter space. The journey of the ants to the food source from the nest and returning to the nest is modelled as one iteration in the algorithm. While on the journey, ants release pheromones which guides the later ants toward the possible solutions instead of a random search. The shorter path has the highest pheromones concentration as it is traversed by a maximum number of ants and also since pheromones have nature of evaporating with time which reduces their concentration towards the minimum traversed paths as well as longer paths which takes longer time in reaching and coming forth leading to evaporation of pheromones. The algorithm consists of three phases: ant-based solution construction, pheromone evaporation, and iteration. In the first phase, artificial ants explore possible solutions and build paths by recording the positions and quality of the solutions. In the later simulation, more ants follow the path, and records of the longer path get evaporated. Simulated ants probabilistically pick a trail that is based on the pheromone density and objective function value etc. heuristic values.

If *i* and *j* be the beginning and end notes of the path, *d_{ij}* is the distance between them, *t_{ij}* is the pheromones density then the probability of choosing path *i* to *j* for *n* number of nodes is:

$$P_{ij} = \frac{\tau_{ij}^\alpha d_{ij}^\beta}{\sum_{i,j=1}^n \tau_{ij}^\alpha d_{ij}^\beta} \tag{18}$$

Pheromone’s concentration decreases exponentially with time due to evaporation between time *t* and *t + 1*:

$$\tau_{ij}(t + 1) = \rho \tau_{ij}(t) + \Delta \tau_{ij} \tag{19}$$

where $0 < \rho < 1$ is the constant of evaporation and $\Delta \tau_{ij}^k(t)$ is the increment of pheromone. For *m* number of ants, additional pheromone laid by *k*th ant at *t*th iteration is:

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k(t) \tag{20}$$

Each ant has individual pheromone contribution of:

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{h_k} & \text{if the } k\text{th ant passes } (i,j) \text{ in current tour} \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

where Q is a constant and L_k = length of path traversed by k th ant.

Ant lion optimizer (ALO)

ALO is a metaheuristic algorithm based on the hunting behaviour of ant lions (Mirjalili 2015). Ant lions catch their prey, ants, by digging sharp cone-shaped curves. Ant lions position themselves at the bottom of the pit, waiting for the ants to fall. No sooner than the ants fall in the trap, ant lions start throwing sands to catch the prey which is trying to escape. When the ants fall at the bottom, ant lions consume it and further create another bigger cone-shaped trap. The matrices M_{ant} and $M_{antlion}$ give the position of ants and ant lions, respectively and M_{oa} and M_{oal} are the objective functions for m number of both parameters.

$$M_{ant} = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1r} \\ A_{21} & A_{22} & \dots & A_{2r} \\ \dots & \dots & \dots & \dots \\ A_{m1} & A_{m2} & \dots & A_{mr} \end{bmatrix}; M_{antlion} = \begin{bmatrix} L_{11} & L_{12} & \dots & L_{1r} \\ L_{21} & L_{22} & \dots & L_{2r} \\ \dots & \dots & \dots & \dots \\ L_{m1} & L_{m2} & \dots & L_{mr} \end{bmatrix} \tag{22}$$

$$M_{oa} = \begin{bmatrix} f([A_{11}, A_{12}, \dots, A_{1d}]) \\ f([A_{21}, A_{22}, \dots, A_{2d}]) \\ \dots \\ f([A_{m1}, A_{m2}, \dots, A_{md}]) \end{bmatrix}; M_{oal} = \begin{bmatrix} f([L_{11}, L_{12}, \dots, L_{1d}]) \\ f([L_{21}, L_{22}, \dots, L_{2d}]) \\ \dots \\ f([L_{m1}, L_{m2}, \dots, L_{md}]) \end{bmatrix} \tag{23}$$

The random walk of the ants is modelled as:

$$X_i = [0, cumsum(2\psi - 1)] \tag{24}$$

where $cumsum$ represents the cumulative sum for the maximum number of iterations. If $rand$ is the random number with uniform distribution in the range $[0, 1]$, the stochastic function is defined as:

$$\psi = \begin{cases} 1 & \text{if } rand > 0.5 \\ 0 & \text{if } rand \leq 0.5 \end{cases} \tag{25}$$

The normalized ant position is given by the following equation:

$$X_i^{itr} = \frac{(X_i^{itr} - \alpha_i) \times (\lambda_i^{itr} - v_i^{itr})}{(\phi_i - \alpha_i)} + v_i^{itr} \tag{26}$$

where l_i^{itr} and n_i^{itr} are the maximum and minimum values of i th dimension the particular iteration, respectively. f_i and a_i the maximum and minimum values of the random walk of the i th variable. The lower and upper bounds of i th dimension are calculated as follows if $Antlion_j^{itr}$ denotes position of j th antlion at particular iteration.

$$v_i^{itr} = Antlion_j^{itr} + v_i^{itr-1} \tag{27}$$

$$\lambda_i^{itr} = Antlion_j^{itr} + \lambda_i^{itr-1} \tag{28}$$

To model the phenomenon of ants falling to the bottom of the pit, their random walk is reduced by a factor:

$$v_i^{itr} = \frac{v_i^{itr}}{K}; \lambda_i^{itr} = \frac{\lambda_i^{itr}}{K} \tag{29}$$

K is a constant such that,

$$K = 10^\mu \times \frac{\text{current iteration}}{\text{maximum number of iterations}} \tag{30}$$

Taking $itrmax$ as the maximum number of iterations, μ is calculated as follows:

$$\mu = \begin{cases} 2 & \text{if } itr > 0.1itr \text{ max} \\ 3 & \text{if } itr > 0.5itr \text{ max} \\ 4 & \text{if } itr > 0.75itr \text{ max} \\ 5 & \text{if } itr > 0.9itr \text{ max} \\ 6 & \text{if } itr > 0.95itr \text{ max} \\ \text{else} & \text{if } itr \end{cases} \tag{31}$$

Antlion catches the ant and consumes it by dragging it inside the sand and moves to a new position:

$$Antlion_j^{itr} = Ant_i^{itr} \text{ iff } (Ant_i^{itr}) > Antlion_j^{itr} \tag{32}$$

Further, in the optimization method, elitism is applied. Elitism is the method of choosing the best ant lion as an elite which being the fittest impacts the movement of every single ant in the iteration. The randomness of ants is given by the Roulette wheel approach:

$$Ant_i^{itr} = \frac{W_{al}^{itr} + W_{elite}^{itr}}{2} \tag{33}$$

where W_{al}^{itr} is the random walk of ant lion and W_{elite}^{itr} random walk of elite ant lion.

Multivariate adaptive regression splines (MARS)

MARS introduced by Friedman (1991) is a non-parametric regression method that uses basis functions to define the correlation between input parameters and output variable

and piecewise linear splines, called basis functions (BFs) to establish this correlation. The MARS methodology initially advances as a forward stepwise function (constructive phase) and then as a backward stepwise function (pruning phase). With the initially existing constant BF, the forward stepwise function starts. The basic functions are split at each step, satisfying the “lack of fit criterion.” The model becomes over-fitted and then the pruning stage begins. Finally, the optimum model is developed in the third step. More details about the method can be studied in the literature (Samui and Kim 2013; Zhang et al. 2020c, 2021; Zhang and Goh 2016).

Genetic programming (GP)

GP (Koza 1992) is a symbolic machine learning technique that uses the Darwinian concept of natural selection and genetic recombination. It evolves from GA (Holland 1975) and uses tree-structure seeming computer programs instead of a string of numbers. The model initializes by the creation of a random population and is followed by the reproduction of individuals and the creation of new by processes of mutation and crossover. In traditional GP, symbolic regression is typically performed to generate a population of trees, which in turn encodes a mathematical expression. The generated expression predicts the desired output ($m \times 1$) using the given inputs ($n \times m$), where n and m are the number of input variables and the number of observations respectively. On the other hand, multi-gene GP (MGGP) is a weighted linear combination of GP trees. For each SR model, the linear weights are derived from the training dataset, which is used further to predict the new outputs. It is understood from the literature that the MGGP regression technique is computationally more efficient than the traditional GP. However, to obtain higher predictive accuracy, the hyper-parameters, such as population size, tournament size, the maximum number of generations, the maximum number of genes, crossover and mutation probability, and functions should be designed properly.

Group method of data handling (GMDH)

GMDH is a self-organized neural network. In this feed-forward method, the elementary unit is a quadratic equation of two variables. The coefficients of the function are calculated using regression analysis (Armaghani et al. 2020a). It simulates datasets having several inputs (u_1, \dots, u_n) and one output (V):

$$V = f(u_1, \dots, u_n) \tag{34}$$

A simplified example of polynomial comprising two variables, u_i and u_k :

$$U_i = \alpha_0 + \alpha_1 u_i + \alpha_2 u_k + \alpha_3 u_i^2 + \alpha_4 u_k^2 + \alpha_5 u_i u_k \tag{35}$$

The GMDH model (shown in Fig. 2) can be described by neuron layers, each with several data points related by quadratic polynomials to each other, and new neurons are created in the process. Here, the filter results are denoted by $U_1, U_2 \dots U_n$. The best outputs are passed through the selection layer ($U^1, U^2 \dots U^r$). The performance Z_1, Z_2, Z_p is polynomial to a higher degree than the previous one. Selected few are transported through the selection layer. The output is Z^1, Z^2, Z^q . The process is carried out till the desired outcome is achieved.

Hybridization of ANN and metaheuristic algorithms

Shortcomings of ANN involves extensive calculation time and trial-and-error approach to discover the appropriate number of hidden neurons. There is a growing initiative to combine ANN with metaheuristic optimization strategies to boost ANN's simulation performance. Several neural network parameters, such as weight and bias, are optimised using optimization methods to boost performance. Recently, many studies are being conducted in engineering applications to augment the capability of ANN models by

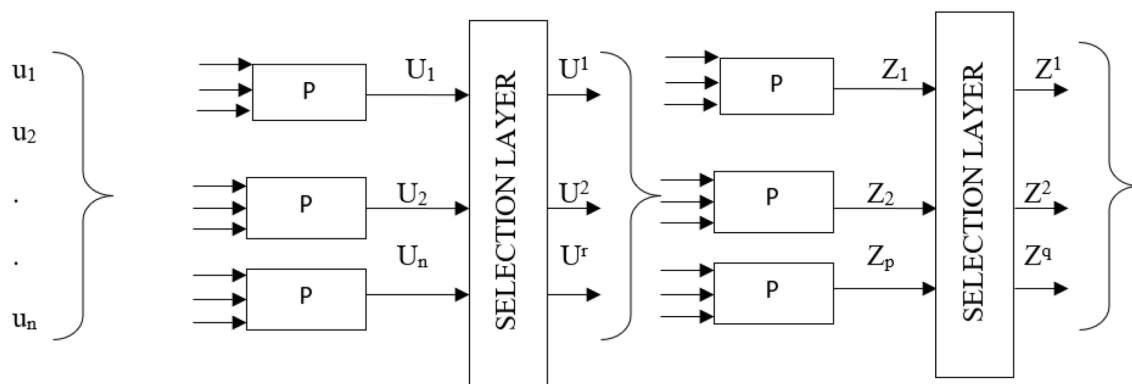


Fig. 2 A basic structure of GMDH algorithm

optimization algorithms (OAs) such as ABC, ACO, ALO, PSO, GOA (Adnan et al. 2019; Armaghani et al. 2014; Malekpour and Mohammad Rezapour Tabari 2020; Moayedi et al. 2019b; Ozturk and Karaboga 2011; Rahgoshay et al. 2019; Taheri et al. 2017; Umar et al. 2019; Xu et al. 2019). ANN models may lead to unwanted outcomes since Back Propagation (BP) lacks in finding the exact global minimum. Moreover, ANN models are more vulnerable to be caught in local minima. OAs have been found to eradicate this problem of ANN by assigning weights and biases. In the study, PSO, GOA, ABC, ACO, and ALO were used to optimize the learning parameters (weights and biases) of ANN and five hybrid models, namely ANN-PSO, ANN-GOA, ANN-ABC, ANN-ACO, and ANN-ALO were constructed to predict the

bearing capacity of the pile foundation. The steps of developing hybrid ANN models are shown in Fig. 3 in the form of a flow chart.

Data processing and analysis

Descriptive statistics of the datasets

To simulate the soft computing models, 50 PDA datasets were collected from the study of Momeni et al. (2015). 36 PDA tests were conducted at the various project sites in Indonesia. Note that, these tests were conducted as per the guidelines of ASTM (D4945-08 in cohesion-less soils

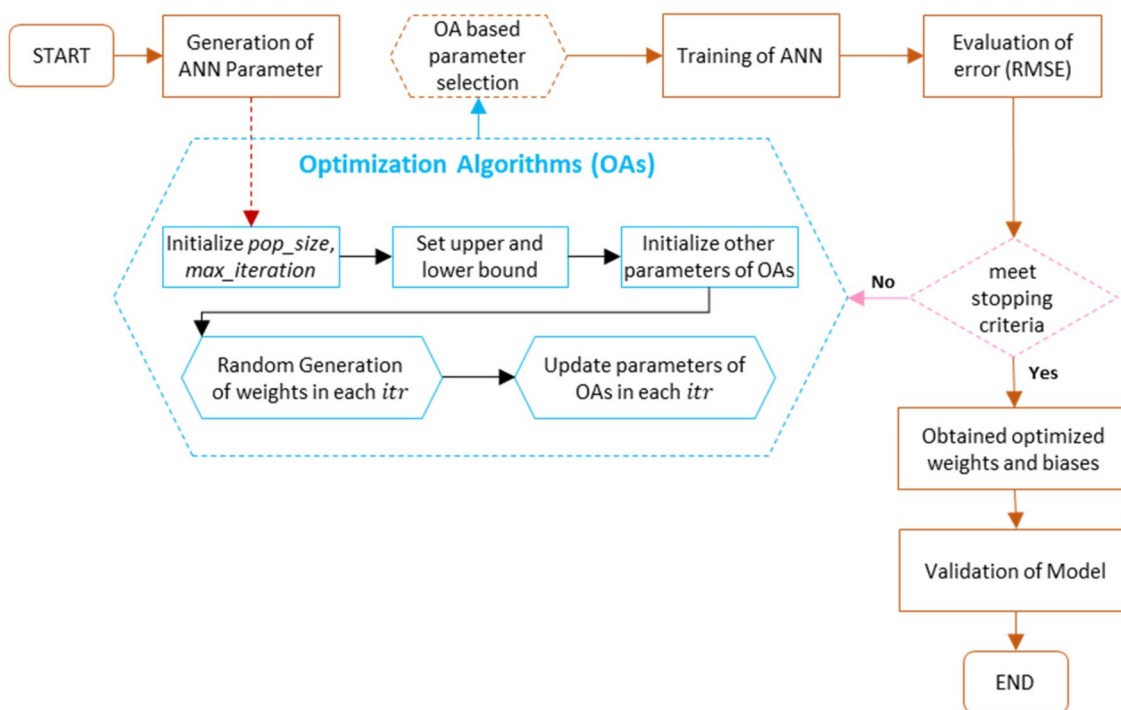


Fig. 3 A flow chart showing the steps of developing hybrid ANN models

Table 1 Descriptive statistics of the input and output variables

Particulars	w (kN)	H (m)	A (cm ²)	L (m)	S (mm)	Q _U (kN)
Minimum	12.00	0.50	226.00	3.40	0.10	410.00
Mean	23.10	1.34	311.64	13.31	3.96	1284.10
Median	13.00	1.00	282.00	10.40	4.00	884.00
Mode	13.00	1.00	282.00	8.00	3.00	1480.00
Maximum	63.00	4.00	500.00	33.60	8.00	3692.00
Standard error	2.57	0.13	11.15	1.17	0.35	129.32
Standard deviation	18.18	0.90	78.83	8.28	2.49	914.40
Sample variance	330.62	0.81	6213.58	68.63	6.22	836,134.34
Kurtosis	0.65	4.44	0.06	1.28	-0.88	1.04
Skewness	1.56	2.39	1.05	1.54	-0.17	1.52

(American Society for Testing and Materials 2010). Table 1 presents the descriptive statistics of the parameters of the collected dataset. The dataset comprises five parameters, namely weight of the hammer (w in kN), the height of fall of the hammer (H in m), cross-sectional area (A in cm^2), length of the pile (L in m), pile set value (S in mm), and ultimate bearing capacity (Q_U) of the pile, among which first five parameters were used as the input parameters to predict the Q_U , the output parameter. As can be seen, the sample variances are scattered in the range of 0.81 to 836,134.34, which indicates that the present dataset has a wide range of input and output parameters. On the other hand, Fig. 4 represents the frequency histogram of the input and output variables. In addition, the values of standard error (scattered in the range of 0.35–129.32) confirm that the present database consists of a wide range of variables, and hence useful for soft computing modelling.

Data processing and computation of models

In soft computing field, to enhance model accuracy, it is important to normalize the inputs and output variables with a predefined range. The normalization aims to adjust the numeric data values to a standard scale, without ambiguous variations in the value ranges. The process is not essential

for all machine learning datasets, but only if the parameters have different ranges. All the variables have been normalized from 0 to 1 in this dataset using the expression given by:

$$x_{\text{Normalized}} = \frac{(x_{\text{Actual}} - x_{\text{min}})}{(x_{\text{max}} - x_{\text{min}})} \tag{36}$$

where x_{Actual} is the actual value of the particular parameter, x_{min} is the minimum value of the parameter in the dataset and x_{max} is the maximum value of the parameter in the dataset. Post-normalization, the dataset is randomly divided into training (70% of the total dataset) and testing (30% of the total dataset) subsets. Amongst them, the training subset was used to train the model. In the training phase, the model learns the correlation between input and output variables and constructs a predictive model. Then, the testing dataset was used to test the prediction of the trained model. The performance of the models is further ascertained by using various statistical parameters, described in detail in later sections. The entire methodology is depicted in Fig. 5. The results of the employed models are compared with those of the traditional FOSM model and the robustness of the model was determined

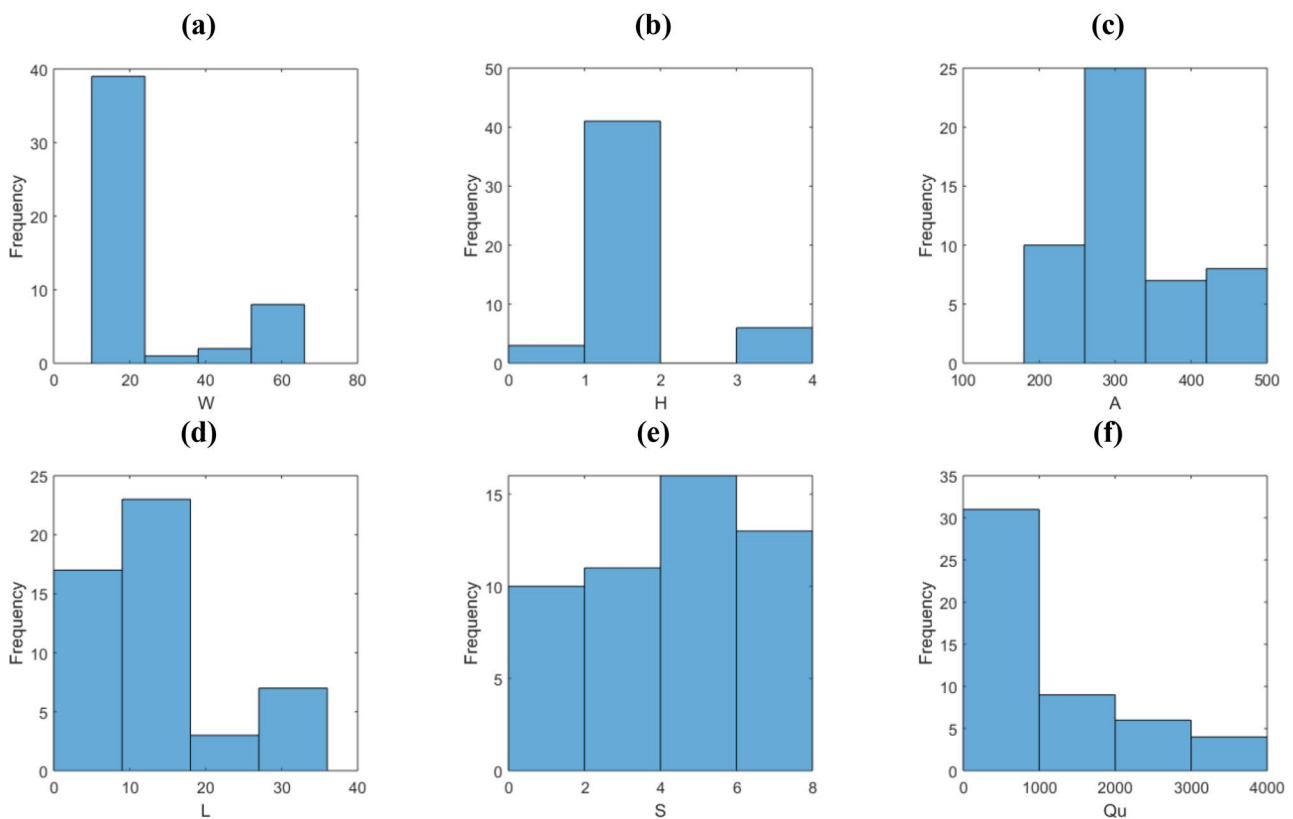


Fig. 4 Frequency histogram of the input and output variables

Fig. 5 Research methodology of AI-based models

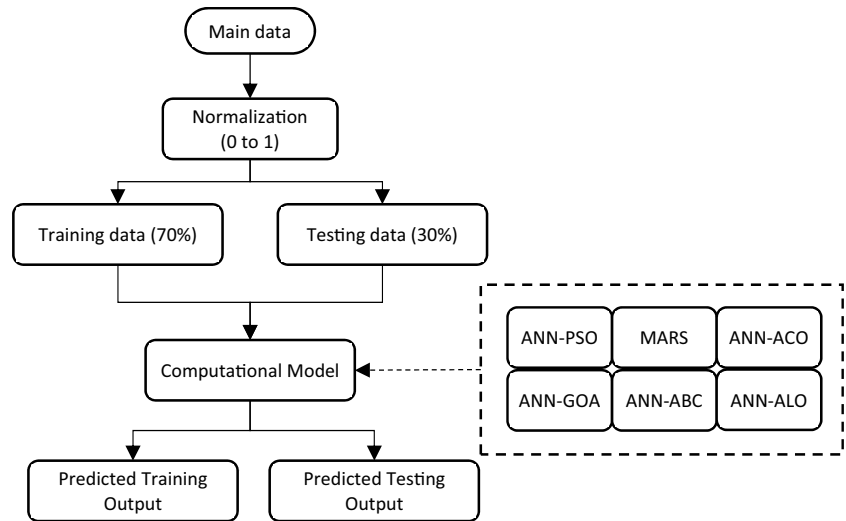


Fig. 6 Convergence curves of the developed hybrid ANN models

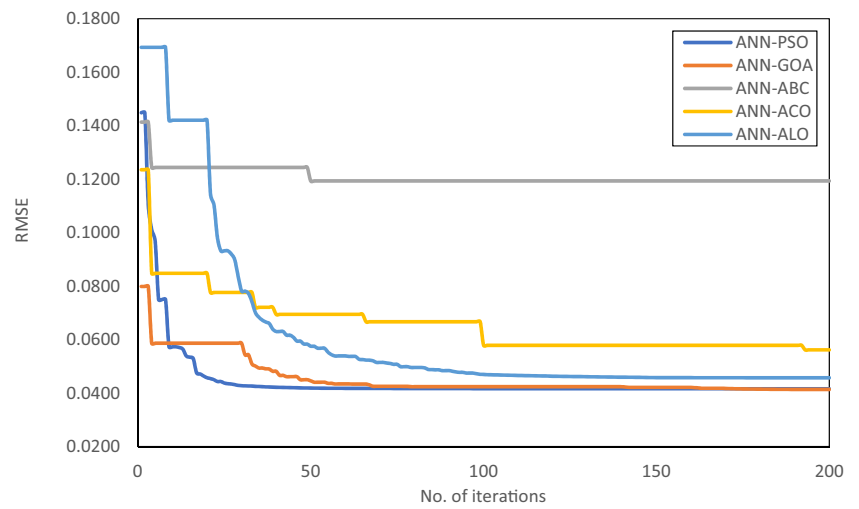


Table 2 Ideal values of performance indices

Indices	R^2	PI	NS	WI	VAF	RMSE	MAE	MAPE	RSR	WMAPE
Ideal value	1	2	1	1	100	0	0	0	0	0

Table 3 Configuration of optimum hybrid ANN models

Parameters	ANN-PSO	ANN-GOA	ANN-ABC	ANN-ACO	ANN-ALO
n_s	50	50	50	50	50
N_h	6	7	6	7	7
k	200	200	200	200	200
lb	-1	-1	-1	-1	-1
ub	+1	+1	+1	+1	+1
Cost function	RMSE	RMSE	RMSE	RMSE	RMSE

Table 4 Optimal values of effective parameters of MARS model

Parameters	MARS-L
GCV penalty per knot	0
Cubic modelling	0 (no)
Self-interactions	1 (no)
Maximum interactions	1
Prune	1 (true)
No. of F_b in the final model	15

Results and discussion

Performance parameters

To estimate the performance of the developed models, several widely used statistical indices were determined (Behar et al. 2015; Despotovic et al. 2015; Kumar et al. 2021; Kumar and Samui 2020; Legates and McCabe 2013; Stone 1993). These are coefficient of determination (R^2), performance index (PI), Nash–Sutcliffe efficiency (NS), Willmott’s index of agreement (WI), variance account for (VAF), root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), root mean square error to observation’s standard deviation ratio (RSR), and weighted mean absolute percentage error (WMAPE). The expressions for these parameters are given below:

$$R^2 = \frac{\sum_{i=1}^N (d_i - d_{\text{mean}})^2 - \sum_{i=1}^N (d_i - y_i)^2}{\sum_{i=1}^N (d_i - d_{\text{mean}})^2} \tag{37}$$

$$PI = adj.R^2 + (0.01 \times VAF) - RMSE \tag{38}$$

$$NS = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - y_{\text{mean}})^2} \tag{39}$$

Table 6 Parametric configuration and terminating criteria of the optimum GP model

Parameters/terminating criteria	Values
Population size	400
Number of generations	200
Tournament size	20
Number of genes	6
Maximum tree depth	4
Mutation probability	0.95
Functions set	$\times, -, +, \tanh, \sin, \cos, \exp$

$$WI = 1 - \left[\frac{\sum_{i=1}^N (d_i - y_i)^2}{\sum_{i=1}^N \{|y_i - d_{\text{mean}}| + |d_i - d_{\text{mean}}|\}^2} \right] \tag{40}$$

$$VAF = \left(1 - \frac{\text{var}(d_i - y_i)}{\text{var}(d_i)} \right) \times 100 \tag{41}$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i - y_i)^2} \tag{42}$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - d_i| \tag{43}$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{d_i - y_i}{d_i} \right| \times 100 \tag{44}$$

$$RSR = \frac{RMSE}{\sqrt{\frac{1}{N} \sum_{i=1}^N (d_i - d_{\text{mean}})^2}} \tag{45}$$

Table 5 Equations of the basis functions in MARS model

BFs	Expression	BFs	Expression
BF1	$\max(0, 0.058824 - x1)$	BF9	$\max(0, 0.2649 - x4) \times \max(0, x5 - 0.62025)$
BF2	$\max(0, x2 - 0.14286)$	BF10	$\max(0, 0.2649 - x4) \times \max(0, 0.62025 - x5)$
BF3	$\max(0, 0.14286 - x2)$	BF11	$\max(0, x5 - 0.49367)$
BF4	$BF1 \times \max(0, x4 - 0.15563)$	BF12	$BF1 \times \max(0, x5 - 0.74684)$
BF5	$\max(0, x1 - 0.058824) \times \max(0, x2 - 0.14286)$	BF13	$BF1 \times \max(0, 0.74684 - x5)$
BF6	$\max(0, x1 - 0.058824) \times \max(0, 0.14286 - x2)$	BF14	$\max(0, x3 - 0.20438)$
BF7	$\max(0, x4 - 0.2649)$	BF15	$\max(0, 0.2649 - x4) \times \max(0, x3 - 0.20438)$
BF8	$\max(0, 0.2649 - x4) \times \max(0, 0.49367 - x5)$		

Table 7 Details of performance parameters for the training dataset

Indices	ANN-PSO	ANN-GOA	ANN-ABC	ANN-ACO	ANN-ALO	MARS	GP	GMDH
R ²	0.9762	0.9764	0.8300	0.9573	0.9712	0.9967	0.9914	0.9833
PI	1.9066	1.9074	1.5107	1.8511	1.8917	1.9773	1.9563	1.9289
NS	0.9761	0.9764	0.8036	0.9565	0.9712	0.9967	0.9914	0.9833
WI	0.9939	0.9940	0.9462	0.9889	0.9926	0.9992	0.9978	0.9958
VAF	97.6151	97.6408	82.9317	95.7272	97.1186	99.6695	99.1398	98.3302
RMSE	0.0416	0.0414	0.1193	0.0562	0.0457	0.0155	0.0250	0.0348
MAE	0.0312	0.0308	0.0930	0.0415	0.0345	0.0100	0.0186	0.0253
MAPE	24.9522	23.9677	75.7937	31.8801	27.2426	5.9399	15.5431	26.3014
RSR	0.1545	0.1536	0.4431	0.2085	0.1698	0.0575	0.0927	0.1293
WMAPE	0.1217	0.1187	0.3629	0.1612	0.1315	0.0391	0.0716	0.0944

Table 8 Details of performance parameters for the testing dataset

Indices	ANN-PSO	ANN-GOA	ANN-ABC	ANN-ACO	ANN-ALO	MARS	GP	GMDH
R ²	0.9773	0.9378	0.7735	0.9736	0.9738	0.9406	0.9859	0.7959
PI	1.8981	1.7666	1.2190	1.8828	1.8858	1.7765	1.9283	1.3158
NS	0.9765	0.9325	0.7151	0.9707	0.9731	0.9388	0.9849	0.7495
WI	0.9940	0.9827	0.9325	0.9927	0.9930	0.9845	0.9961	0.9382
VAF	97.7285	93.7722	72.4181	97.2914	97.3514	93.9870	98.5536	77.6615
RMSE	0.0439	0.0744	0.1529	0.0491	0.0470	0.0709	0.0353	0.1433
MAE	0.0307	0.0458	0.1110	0.0363	0.0394	0.0520	0.0289	0.0767
MAPE	17.7220	28.8963	56.4098	17.1733	23.0294	34.4244	22.2236	53.7030
RSR	0.1532	0.2598	0.5337	0.1713	0.1640	0.2474	0.1231	0.5005
WMAPE	0.1001	0.1493	0.3620	0.1183	0.1285	0.1695	0.0943	0.2501

$$WMAPE = \frac{\sum_{i=1}^N \left| \frac{d_i - y_i}{d_i} \right| \times d_i}{\sum_{i=1}^N d_i} \quad (46)$$

where d_i is the observed i th value, y_i is the predicted i th value, d_{mean} is the average of observed value, N is the number of the data sample. Note that, for an ideal model, the values of these indices should be equal to their ideal values, the details of which are presented in Table 2.

Configuration of the developed hybrid ANN models

As mentioned earlier, to optimize the learning parameters of ANN, five OAs were used. In ANN, these learning parameters are the input weights, biases of hidden neurons, output weights, and output bias. After the initialization of ANN, OAs were used to optimize the learning parameters, i.e., the weights and biases of the models. For this purpose, OAs were set up before optimizing the learning parameters of ANNs, including the population size (n_s), the maximum number of iterations (k), lowerbound (lb), upper bound (ub), and other parameters besides the number of hidden neurons (Nh) of ANNs. Then, the weight and biases of ANN were optimized by OAs based on the training dataset. The

optimized values of weight and biases were determined by setting RMSE as the cost function. It is pertinent to mention here that, although n_s , k , lb , and ub were kept the same during the optimization process, however, the optimized value of learning parameters is different in each case.

Following the above-mentioned procedure and using the same training dataset, the Nh was varied in the range of 5–20, and the most appropriate value obtained was 6 for ANN-PSO, 7 for ANN-GOA, 6 for ANN-ABC, 7 for ANN-ACO, and 7 for ANN-ALO. The values of other parameters were set as $n_s = 50$, $k = 200$, $lb = -1$, and $ub = +1$. Therefore, the optimum number of optimized weight and biases are 43 ($5 \times 6 + 6 + 6 + 1$) for ANN-PSO, 50 ($5 \times 7 + 7 + 7 + 1$) for ANN-GOA, 43 ($5 \times 6 + 6 + 6 + 1$) for ANN-ABC, 50 ($5 \times 7 + 7 + 7 + 1$) for ANN-ACO, and 50 ($5 \times 7 + 7 + 7 + 1$) for ANN-ALO, and note that values of these optimized weights and biases are different from one another. The detailed configuration of the developed hybrid ANN models is presented in Table 3. Furthermore, the convergence behaviour of the developed hybrid ANN models is presented in Fig. 6, from which the converging ability of the hybrid models in finding the global minimum can be assessed.

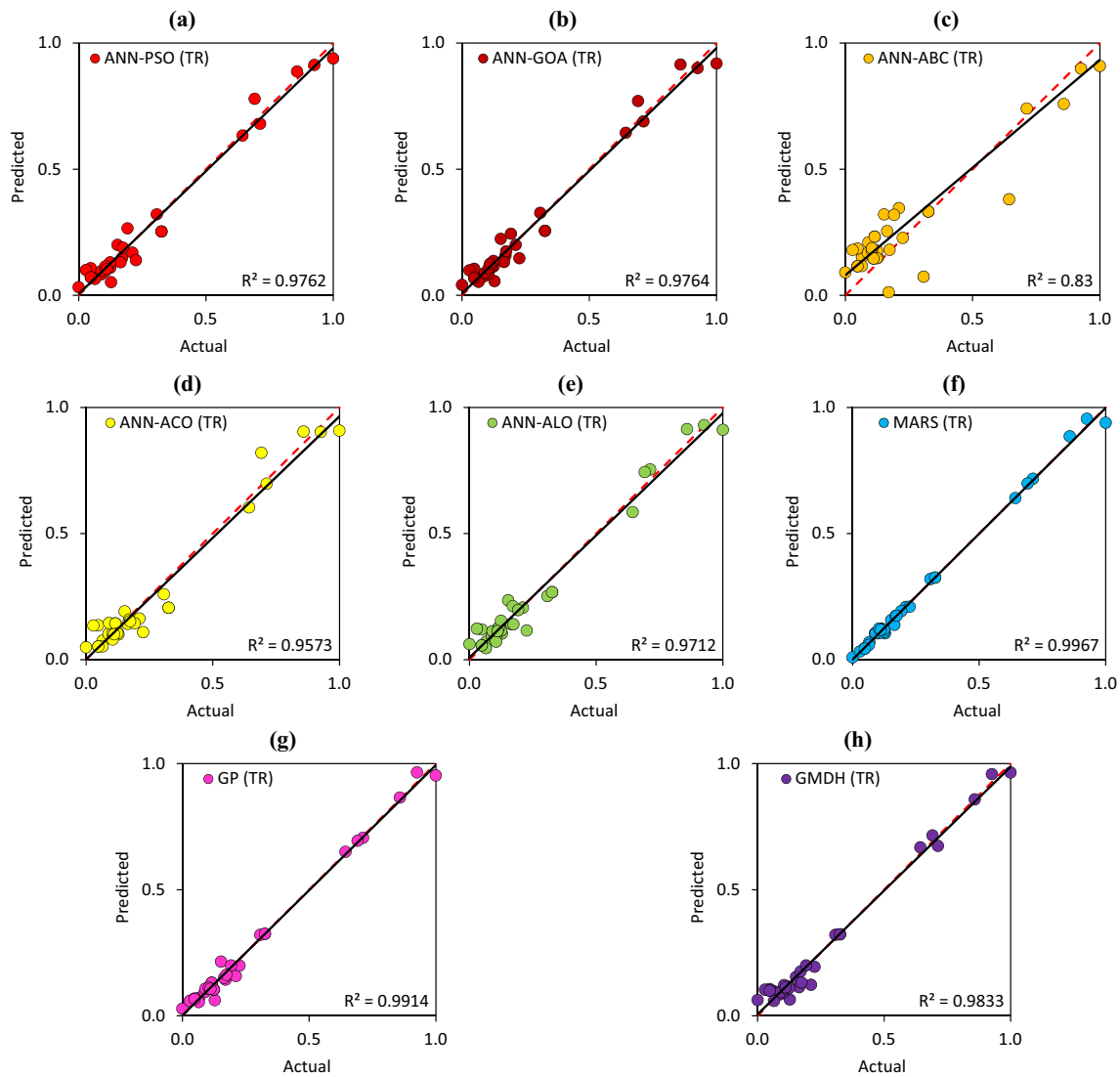


Fig. 7 Illustration of actual vs. predicted values of the developed models for the training (TR) dataset

Configuration of the employed MARS, GP, and GMDH models

With the same training dataset, the MARS, GP, and GMDH models were constructed and accordingly evaluated. To design the MARS model, a piecewise linear regression variant of MARS was considered in the present study. The hyper-parameters, such as the number of BFs, GCV, self-interaction, maximum interactions, threshold value, and pruning option were designed using trial-and-error runs, the details of which of the designed MARS model are presented in Table 4. In addition, the details of each BF are presented in Table 5, and the expression of the designed MARS model in Eq. (47). The expression is given in Eq. (47) can readily be used to estimate the bearing capacity of piles.

Analogous to the MARS model, the parameters of GP and GMDH models were designed based on trial-and-error approaches. The most effective choices of different GP parameters and terminating criteria (population size, number of generations, tournament size, maximum number of genes, maximum tree depth, mutation probability, and functions set) are presented in Table 6, and the final GP model for predicting the bearing capacity of the pile is given in Eq. (48) which can also be used as a readymade formula to estimate the probability of failure using relativity index. On the other hand, the most suitable structure of GMDH consists of 4 hidden layers with 10 neurons in each layer. The best performance was achieved when the number of hidden layers was set to 3.

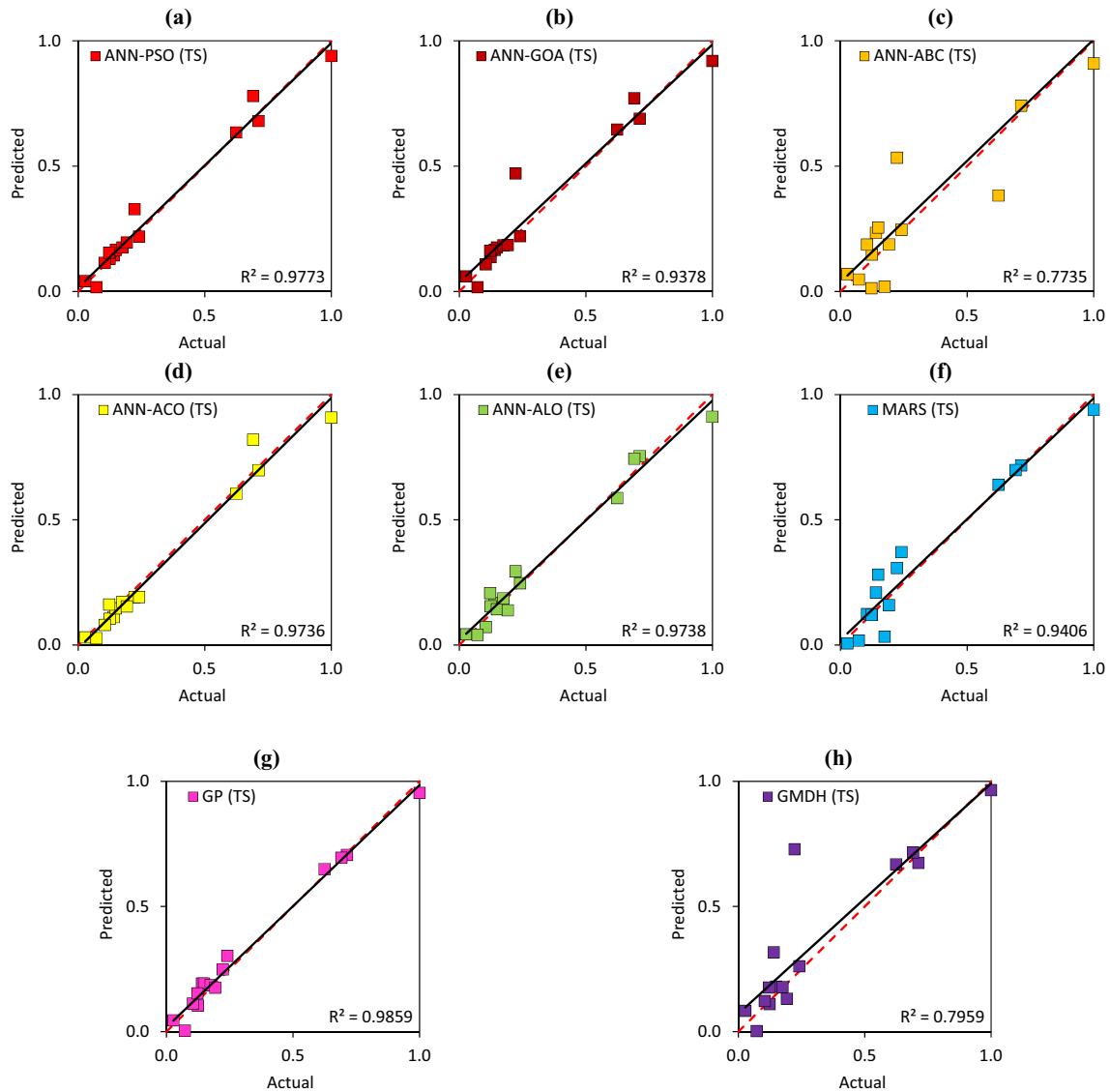


Fig. 8 Illustration of actual vs. predicted values of the developed models for the testing (TS) dataset

$$\begin{aligned}
 y_{\text{MARS}} = & 0.28444 - 1.1447 * BF1 + 12.706 * BF2 - 1.6146 * BF3 \\
 & + 8.3353 * BF4 - 12.524 * BF5 + 7.3737 * BF6 + 0.12589 * BF7 \\
 & + 12.732 * BF8 - 1.4795 * BF9 - 4.6314 * BF10 - 0.59395 * BF11 \\
 & + 14.463 * BF12 - 12.111 * BF13 - 0.32972 * BF14 + 8.811 * BF15
 \end{aligned}
 \tag{47}$$

$$\begin{aligned}
 y_{\text{GP}} = & 4.488x1 + 1.122x2 - \exp(2x1)0.5154 + 1.122 \tanh(\exp(x1)) \\
 & + 0.1091 \cos(\sin(\text{square}(x5 + 4.662))) + \tanh(1.533x4)0.28 \\
 & - 0.28 \cos(x5) - 0.28 \tanh(x2 - 1.315) + \square(\cos(2x2 + \square(x5)))0.1233 \\
 & - \tanh(\text{square}(-2.482x1) + x1x5\exp(x5))1.456 - 0.6767
 \end{aligned}
 \tag{48}$$

Statistical details of results

This sub-section describes the outcomes of all the performance parameters of the models. The output parameter values for all nine models are presented in Tables 7 and 8, respectively, for the training and testing datasets. Note that, only one or two parameters are never enough because every parameter has its advantages as well as limitations. Therefore, to determine the efficiency of the developed models, ten performance indices were determined and assessed in detail. As can be seen, all models have captured the correlations in estimating the pile bearing capacity. However, based on the experimental results with the R^2 criteria, it can be seen that the R^2 values of the top two performing models are 0.9967 (MARS) and 0.9914 (GP) in the training phase. These facts

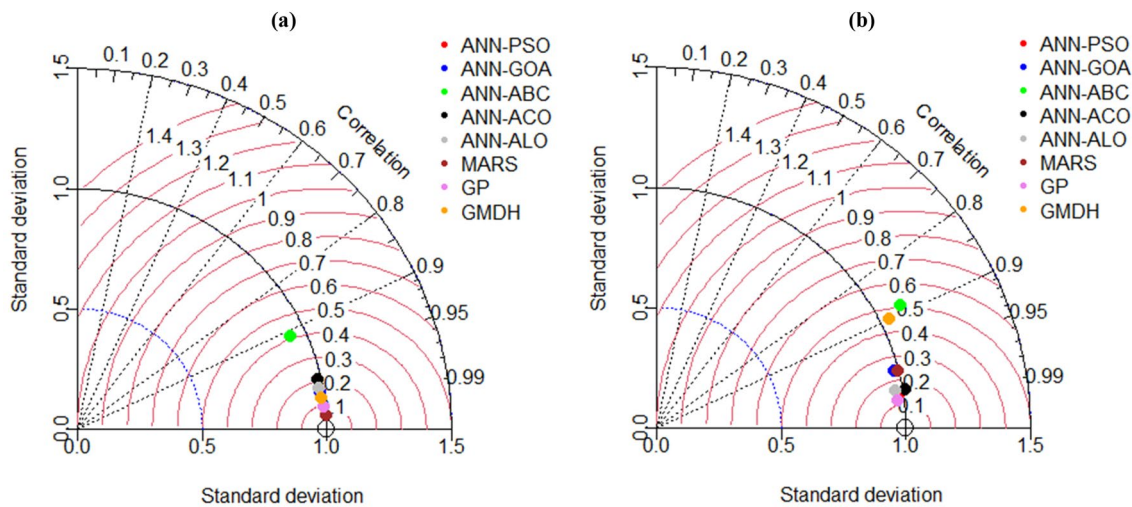


Fig. 9 Taylor diagrams: **a** training phase and **b** testing phase

	(a)									(b)								
	ANN-PSO	ANN-GOA	ANN-ABC	ANN-ACO	ANN-ALO	MARS	GP	GMDH		ANN-PSO	ANN-GOA	ANN-ABC	ANN-ACO	ANN-ALO	MARS	GP	GMDH	
R^2	98%	98%	83%	96%	97%	100%	99%	98%	100%	98%	94%	77%	97%	97%	94%	99%	80%	100%
PI	95%	95%	76%	93%	95%	99%	98%	96%	91%	95%	88%	61%	94%	94%	89%	96%	66%	87%
NS	98%	98%	80%	96%	97%	100%	99%	98%		98%	93%	72%	97%	97%	94%	98%	75%	
WI	99%	99%	95%	99%	99%	100%	100%	100%		99%	98%	93%	99%	99%	98%	100%	94%	
VAF	98%	98%	83%	96%	97%	100%	99%	98%		98%	94%	72%	97%	97%	94%	99%	78%	
RMSE	96%	96%	88%	94%	95%	98%	98%	97%		96%	93%	85%	95%	95%	93%	96%	86%	
MAE	97%	97%	91%	96%	97%	99%	98%	97%		97%	95%	89%	96%	96%	95%	97%	92%	
MAPE	75%	76%	24%	68%	73%	94%	84%	74%		82%	71%	44%	83%	77%	66%	78%	46%	
RSR	85%	85%	56%	79%	83%	94%	91%	87%		85%	74%	47%	83%	84%	75%	88%	50%	
WMAPE	88%	88%	64%	84%	87%	96%	93%	91%	24%	90%	85%	64%	88%	87%	83%	91%	75%	44%

Fig. 10 Accuracy matrix: **a** for training results and **b** for testing results

demonstrate that the conventional soft computing models have attained the most accurate prediction in the training phase. While based on the R^2 and RMSE criteria ANN-GOA attained the best prediction performance among the ANN-based models. On the other hand, in the testing phase, GP outperformed all other models by far, with $R^2=0.9859$ and $RMSE=0.0353$, while ANN-PS was found to be the second-best model ($R^2=0.9773$ and $RMSE=0.0439$) in estimating the bearing capacity of pile foundation. Tables 7 and 8 report the prediction performances of all the models using 10 performance metrics, respectively, for the training and testing phases. It is observed that MARS and GP have achieved the best outcomes in all metrics in the training and testing

phases, respectively. However, for the ANN-based hybrid models, ANN-PSO has achieved second place, followed by ANN-ALO, ANN-ACO, ANN-GOA, and ANN-ABC. Figures 7 and 8 depict the comparison of the actual values with the predicted values of all the employed models for the case of training and testing phases, respectively.

Furthermore, to visualise the results, the Taylor diagram and accuracy matrix are presented. It may be noted that Taylor diagram (Taylor 2001) is a simple visual representation of how well a model performs compared to the other used models. It plots correlation, standard deviation, and RMSE on a 2-dimensional graph. In Taylor diagrams, the radial distance from the origin, azimuthal angle on the graph denotes

standard deviation and correlation coefficient, respectively. RMSE error is plotted as the distance between observed and simulated fields, related in identical units to standard deviation. On the other hand, an accuracy matrix, recently proposed by Kardani et al. (2021a) was used to analyse the accuracy level of the developed models in the form of a heat map matrix. One can estimate the overall status of the developed models based on the colour variation of performance parameters. Figures 9a and b and 10a, b represent the Taylor diagram and accuracy matrixes, respectively, for the developed models.

Discussion

In the above sub-sections, performance of applied machine learning models in terms of bearing capacity of pile foundation are analyzed and presented. For this purpose, 50 dynamic pile load test data of concrete piles were collected from literature and utilized. Five hybrid ANN models and three conventional soft computing models were employed to estimate the bearing capacity of piles first. The employed models are evaluated on the ground of various statistical parameters. Based on the experimental presented in the above sub-sections, it is seen that the MARS model attained the highest prediction with $R^2 = 0.9967$, RMSE = 0.0155, in the training phase, while ANN-PSO ($R^2 = 0.9773$, RMSE = 0.0439) and GP ($R^2 = 0.9859$, RMSE = 0.0353) attained the most accurate results in the analysis of piles. Note that, all the models were developed in MATLAB environment with MATLAB 2015a version and version with i3-8130U CPU @ 2.20 GHz, 12.00 GB RAM. The computational cost of the top two best-performing models was noted as 69.316796s (ANN-PSO) and 14.253398s (GP). It is pertinent to mention here that, a prediction model with higher prediction accuracy attained in the testing phase should be accepted with more conviction. Therefore, the ANN-PSO and GP models can be considered as robust models in analysis of piles.

Conclusion

Soft computing have transformed all the sectors of engineering and civil engineering is not an exception. Soft-computing models can potentially be used as an alternative to expensive and time-intensive field tests and inefficient numerical methods. A comparative assessment of five hybrid ANN models and three conventional soft computing models in estimating bearing capacity of piles are presented in this study. For this purpose, 50 sets of dynamic pile testing data were collected from the available literature. The values of statistical performance parameters, regression curves, Taylor diagrams and accuracy matrixes recommends that the piles considered in the analysis

could be considered safe against bearing capacity failure. Experimental results point out that ANN-PSO and GP can estimate the bearing capacity of pile accurately both in the training and testing phases. However, a detailed review of results reveals that the ANN-PSO ($R^2 = 0.9773$, RMSE = 0.0439) and GP ($R^2 = 0.9859$, RMSE = 0.0353) showed comparatively better performance in the testing phase. The unique advantages of the proposed ANN-PSO model are higher prediction accuracy, ease of implementation with the existing datasets, and high generalization capability. On the other hand, the predicting expression of GP can be used as a user-friendly equation to determine the bearing capacity of pile. Furthermore, the ANN-PSO and GP models proposed in this study would be used to analyze other civil engineering structures once the corresponding database is prepared for the purpose.

Declarations

Conflict of interest There is no conflict of interest to declare.

References

- Adnan RM, Malik A, Kumar A, Parmar KS, Kisi O (2019) Pan evaporation modeling by three different neuro-fuzzy intelligent systems using climatic inputs. Arab J Geosci. <https://doi.org/10.1007/s12517-019-4781-6>
- Alizadeh Z, Yazdi J, Mohammadiun S, Hewage K, Sadiq R (2019) Evaluation of data driven models for pipe burst prediction in urban water distribution systems. Urban Water J. <https://doi.org/10.1080/1573062X.2019.1637004>
- Alzoubi AK, Ibrahim F (2019) Predicting loading-unloading pile static load test curves by using artificial neural networks. Geotech Geol Eng. <https://doi.org/10.1007/s10706-018-0687-4>
- American Society for Testing and Materials (2010) Standard test method for high-strain dynamic testing of deep foundations, D 4945-08
- Ardakani A, Kordnaeij A (2019) Soil compaction parameters prediction using GMDH-type neural network and genetic algorithm. Eur J Environ Civ Eng. <https://doi.org/10.1080/19648189.2017.1304269>
- Armaghani DJ, Hajihassani M, Mohamad ET, Marto A, Noorani SA (2014) Blasting-induced flyrock and ground vibration prediction through an expert artificial neural network based on particle swarm optimization. Arab J Geosci. <https://doi.org/10.1007/s12517-013-1174-0>
- Armaghani DJ, Hasanipanah M, Amnieh HB, Bui DT, Mehrabi P, Khorami M (2020a) Development of a novel hybrid intelligent model for solving engineering problems using GS-GMDH algorithm. Eng Comput 36(4):1379–1391. <https://doi.org/10.1007/s00366-019-00769-2>
- Armaghani DJ, Mirzaei F, Shariati M, Trung NT, Shariati M, Trnava D (2020b) Hybrid ANN-based techniques in predicting cohesion of sandy-soil combined with fiber. Geomech Eng. <https://doi.org/10.12989/gae.2020.20.3.191>
- Asteris PG, Skentou AD, Bardhan A, Samui P, Lourenço PB (2021a) Soft computing techniques for the prediction of concrete

- compressive strength using non-destructive tests. *Constr Build Mater* 303:124450. <https://doi.org/10.1016/J.CONBUILDMAT.2021.124450>
- Asteris PG, Skentou AD, Bardhan A, Samui P, Pilakoutas K (2021b) Predicting concrete compressive strength using hybrid ensemble of surrogate machine learning models. *Cem Concr Res* 145:106449. <https://doi.org/10.1016/j.cemconres.2021.106449>
- Bardhan A, Gokceoglu C, Burman A, Samui P, Asteris PG (2021) Efficient computational techniques for predicting the California bearing ratio of soil in soaked conditions. *Eng Geol* 291:106239. <https://doi.org/10.1016/J.ENGGEOL.2021.106239>
- Basarkar SS (2011) High strain dynamic pile testing practices in india-favourable situations and correlation studies. In: Proceedings of Indian Geotech. Conf. Kochi (paper no. Q-303)
- Behar O, Khellaf A, Mohammedi K (2015) Comparison of solar radiation models and their validation under Algerian climate—the case of direct irradiance. *Energy Convers Manag* 98:236–251. <https://doi.org/10.1016/j.enconman.2015.03.067>
- Benali A, Boukhatem B, Hussien MN, Nechnech A, Karray M (2017) Prediction of axial capacity of piles driven in non-cohesive soils based on neural networks approach. *J Civ Eng Manag* 23(3):393–408. <https://doi.org/10.3846/13923730.2016.1144643>
- Bradshaw AS, Baxter CDP (2006) Design and construction of driven pile foundations—lessons learned on the central artery/tunnel project. Publ. No. FWA-HRT-05-159
- Bui XN, Nguyen H, Choi Y, Nguyen-Thoi T, Zhou J, Dou J (2020) Prediction of slope failure in open-pit mines using a novel hybrid artificial intelligence model based on decision tree and evolution algorithm. *Sci Rep*. <https://doi.org/10.1038/s41598-020-66904-y>
- Charlie W, Allard D, Doehring O (2009) Pile settlement and uplift in liquefying sand deposit. *Geotech Test J* 32(2):147–156
- Che WF, Lok TMH, Tam SC, Novais-Ferreira H (2003) Axial capacity prediction for driven piles at Macao using artificial neural network. AA Balkema Publishers, Amsterdam
- Chen W, Sarir P, Bui XN, Nguyen H, Tahir MM, Jahed Armaghani D (2020) Neuro-genetic, neuro-imperialism and genetic programming models in predicting ultimate bearing capacity of pile. *Eng Comput*. <https://doi.org/10.1007/s00366-019-00752-x>
- Despotovic M, Nedjic V, Despotovic D, Cvetanovic S (2015) Review and statistical analysis of different global solar radiation sunshine models. *Renew Sustain Energy Rev* 52:1869–1880
- Dorigo M, Blum C (2005) Ant colony optimization theory: a survey. *Theor Comput Sci*. <https://doi.org/10.1016/j.tcs.2005.05.020>
- Dorigo M, Socha K (2007) Ant colony optimization. *Handb. approx. algorithms metaheuristics*
- Dorigo M, Stützle T. n.d. *Ant colony optimization*. Cambridge
- Fellenius BH (1999) Using the pile driving analyzer. *Pile Driv. Contract. Assoc. PDCA, Annu. Meet. San Diego, Febr. 19–20, 1999*
- FHWA (2006) Design and construction of driven pile foundations—lessons learned on the central artery/tunnel project. Washington, DC
- Friedman JH (1991) Multivariate adaptive regression splines. *Ann Stat*. <https://doi.org/10.1214/aos/1176347963>
- Gabrielaitis L, Papinigis V, Žaržojū G (2013) Estimation of settlements of bored piles foundation. *Struct Tech*, 287–293
- Gaitonde VN, Karnik SR (2012) Minimizing burr size in drilling using artificial neural network (ANN)-particle swarm optimization (PSO) approach. *J Intell Manuf* 23(5):1783–1793. <https://doi.org/10.1007/s10845-010-0481-5>
- Goh ATC (2000) Search for critical slip circle using genetic algorithms. *Civ Eng Environ Syst* 17(3):181–211. <https://doi.org/10.1080/02630250008970282>
- Goh A, Goh S (2007) Support vector machines: their use in geotechnical engineering as illustrated using seismic liquefaction data. *Comput Geotech* 34(5):410–421
- Hassanlourad M, Ardakani A, Kordnaeij A, Mola-Abasi H (2017) Dry unit weight of compacted soils prediction using GMDH-type neural network. *Eur Phys J plus*. <https://doi.org/10.1140/epjp/i2017-11623-5>
- Holland JH (1975) *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor
- Huang J, Koopialipoor M, Armaghani DJ (2020) A combination of fuzzy Delphi method and hybrid ANN-based systems to forecast ground vibration resulting from blasting. *Sci Rep*. <https://doi.org/10.1038/s41598-020-76569-2>
- Jiang T, Zhang C (2018) Application of grey wolf optimization for solving combinatorial problems: job shop and flexible job shop scheduling cases. *IEEE Access* 6(c):26231–26240. <https://doi.org/10.1109/ACCESS.2018.2833552>
- Jiang C, Li TB, Zhou KP, Chen Z, Chen L, Zhou ZL, Liu L, Sha C (2016) Reliability analysis of piles constructed on slopes under laterally loading. *Trans Nonferrous Met Soc China (english Ed)* 26(7):1955–1964. [https://doi.org/10.1016/S1003-6326\(16\)64306-6](https://doi.org/10.1016/S1003-6326(16)64306-6)
- Karaboga D (2005) An idea based on honey bee swarm for numerical optimization. *Tech. Rep. TR06, Erciyes Univ. citeulike-article-id:6592152*
- Kardani N, Bardhan A, Kim D, Samui P, Zhou A (2021a) Modelling the energy performance of residential buildings using advanced computational frameworks based on RVM, GMDH, ANFIS-BBO and ANFIS-IPSO. *J Build Eng*. <https://doi.org/10.1016/j.jobee.2020.102105>
- Kardani N, Bardhan A, Samui P, Nazem M, Zhou A, Armaghani DJ (2021b) A novel technique based on the improved firefly algorithm coupled with extreme learning machine (ELM-IFF) for predicting the thermal conductivity of soil. *Eng Comput*. <https://doi.org/10.1007/s00366-021-01329-3>
- Kashani AR, Chiong R, Mirjalili S, Gandomi AH (2020) Particle swarm optimization variants for solving geotechnical problems: review and comparative analysis. *Arch Comput Methods Eng*. <https://doi.org/10.1007/s11831-020-09442-0>
- Kennedy J, Eberhart (1995) Particle swarm optimization. In: *IEEE int conf neural networks—conf proc.*
- Kiefa MAA (1998) General regression neural networks for driven piles in cohesionless soils. *J Geotech Geoenviron Eng* 124(12):1177–1185. [https://doi.org/10.1061/\(ASCE\)1090-0241\(1998\)124:12\(1177\)](https://doi.org/10.1061/(ASCE)1090-0241(1998)124:12(1177))
- Koza J (1992) Genetic programming: on the programming of computers by means of natural selection. *Biosystems*.
- Kumar M, Samui P (2020) Reliability analysis of settlement of pile group in clay using LSSVM, GMDH, GPR. *Geotech Geol Eng*. <https://doi.org/10.1007/s10706-020-01464-6>
- 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. <https://doi.org/10.3390/pr9030486>
- Kumar DR, Samui P, Burman A (2022a) Prediction of probability of liquefaction using soft computing techniques. *J Inst Eng Ser A* 103(4):1195–1208. <https://doi.org/10.1007/S40030-022-00683-9>
- Kumar DR, Samui P, Burman A (2022b) Prediction of probability of liquefaction using hybrid ANN with optimization techniques. *Arab J Geosci* 15(20):1–21. <https://doi.org/10.1007/S12517-022-10855-3>
- Lee IM, Lee JH (1996) Prediction of pile bearing capacity using artificial neural networks. *Comput Geotech* 18:189–200
- Legates DR, McCabe GJ (2013) A refined index of model performance: a rejoinder. *Int J Climatol*. <https://doi.org/10.1002/joc.3487>
- Liu L, Moayedi H, Rashid ASA, Rahman SSA, Nguyen H (2020) Optimizing an ANN model with genetic algorithm (GA) predicting load-settlement behaviours of eco-friendly raft-pile

- foundation (ERP) system. *Eng Comput.* <https://doi.org/10.1007/s00366-019-00767-4>
- Long M (2007) Comparing dynamic and static test results of bored piles. *Proc Inst Civ Eng Geotech Eng* 160(1):43–49. <https://doi.org/10.1680/geng.2007.160.1.43>
- Low BK, The CI, Tang WH (2001) Stochastic nonlinear p-y analysis of laterally loaded piles. *Struct Saf* 1–8
- Malekpour MM, Mohammad Rezapour Tabari M (2020) Implementation of supervised intelligence committee machine method for monthly water level prediction. *Arab J Geosci.* <https://doi.org/10.1007/s12517-020-06034-x>
- Mirjalili S (2015) The ant lion optimizer. *Adv Eng Softw.* <https://doi.org/10.1016/j.advengsoft.2015.01.010>
- Moayedi SH (2018) Applicability of a CPT-based neural network solution in predicting load-settlement responses of bored pile. *Int J Geomech* 18:06018009
- Moayedi H, Hayati S (2019) Artificial intelligence design charts for predicting friction capacity of driven pile in clay. *Neural Comput Appl* 31(11):7429–7445. <https://doi.org/10.1007/s00521-018-3555-5>
- Moayedi H, Rezaei A (2019) An artificial neural network approach for under-reamed piles subjected to uplift forces in dry sand. *Neural Comput Appl.* <https://doi.org/10.1007/s00521-017-2990-z>
- Moayedi H, Bui DT, Anastasios D, Kalantar B (2019a) Spotted hyena optimizer and ant lion optimization in predicting the shear strength of soil. *Appl Sci.* <https://doi.org/10.3390/app9224738>
- Moayedi H, Bui DT, Ngo PTT (2019b) Neural computing improvement using four metaheuristic optimizers in bearing capacity analysis of footings settled on two-layer soils. *Appl Sci.* <https://doi.org/10.3390/app9235264>
- Moayedi H, Gör M, Lyu Z, Bui DT (2020) Herding Behaviors of grasshopper and Harris hawk for hybridizing the neural network in predicting the soil compression coefficient. *Mea J Int Meas Confed.* <https://doi.org/10.1016/j.measurement.2019.107389>
- Mola-Abasi H, Eslami A (2019) Prediction of drained soil shear strength parameters of marine deposit from CPTu data using GMDH-type neural network. *Mar Georesour Geotechnol* 37(2):180–189. <https://doi.org/10.1080/1064119X.2017.1415400>
- Momeni E, Nazir R, Armaghani DJ, Maizir H (2015) Application of artificial neural network for predicting shaft and tip resistances of concrete piles. *Earth Sci Res J.* <https://doi.org/10.15446/esrj.v19n1.38712>
- Murlidhar BR, Sinha RK, Mohamad ET, Sonkar R, Khorami M (2020) The effects of particle swarm optimisation and genetic algorithm on ANN results in predicting pile bearing capacity. *Int J Hydro-mechatron.* <https://doi.org/10.1504/ijhm.2020.105484>
- Nayak NV, Kanhere DK, Vaidya R (2000) Static and high strain dynamic test co-relation studies on cast-in-situ concrete bored piles. In: *Proc. 25th annu. members' conf. 8th int. conf. expo. deep found. institute*
- Nguyen H, Moayedi H, Foong LK, Al Najjar HAH, Jusoh WAW, Rashid ASA, Jamali J (2020) Optimizing ANN models with PSO for predicting short building seismic response. *Eng Comput.* <https://doi.org/10.1007/s00366-019-00733-0>
- Ozturk C, Karaboga D (2011) Hybrid artificial bee colony algorithm for neural network training. In: *2011 IEEE Congr. evol. comput. CEC 2011*
- Pal M, Deswal S (2008) Modeling pile capacity using support vector machines and generalized regression neural network. *J Geotech Geoenviron Eng* 134(7):1021–1024. [https://doi.org/10.1061/\(ASCE\)1090-0241\(2008\)134:7\(1021\)](https://doi.org/10.1061/(ASCE)1090-0241(2008)134:7(1021))
- Pradeep T, Bardhan A, Burman A, Samui P (2021) Rock strain prediction using deep neural network and hybrid models of ANFIS and meta-heuristic optimization algorithms. *Infrastructures* 6(9):129. <https://doi.org/10.3390/infrastructures6090129>
- Rahgoshay M, Feiznia S, Arian M, Hashemi SAA (2019) Simulation of daily suspended sediment load using an improved model of support vector machine and genetic algorithms and particle swarm. *Arab J Geosci.* <https://doi.org/10.1007/s12517-019-4444-7>
- Rajagopal C, Solanki CH, Tandel YK (2012) Comparison of static and dynamic load test of pile. *Electron J Geotech Eng* 17M:1905–1914
- Rausche F, Goble GG, Likins GE (1985) Dynamic determination of pile capacity. *J Geotech Eng* 111(3):367–383. [https://doi.org/10.1061/\(ASCE\)0733-9410\(1985\)111:3\(367\)](https://doi.org/10.1061/(ASCE)0733-9410(1985)111:3(367))
- Rausche F, Goble GG, Likins GE Jr (2004) Dynamic determination of pile capacity. In: *Curr. pract. futur. trends deep found.* American Society of Civil Engineers, Reston, p 398–417
- Ray R, Kumar D, Samui P, Roy LB, Goh ATC, Zhang W (2021) Application of soft computing techniques for shallow foundation reliability in geotechnical engineering. *Geosci Front* 12(1):375–383. <https://doi.org/10.1016/j.gsf.2020.05.003>
- Sakr M (2013) Comparison between high strain dynamic and static load tests of helical piles in cohesive soils. *Soil Dyn Earthq Eng* 54:20–30. <https://doi.org/10.1016/j.soildyn.2013.07.010>
- Salgado R (2008) *The engineering of foundations*
- Samui P, Kim D (2013) Least square support vector machine and multivariate adaptive regression spline for modeling lateral load capacity of piles. *Neural Comput Appl* 23(3–4):1123–1127. <https://doi.org/10.1007/s00521-012-1043-x>
- Samui P, Kumar R, Yadav U, Kumari S, Bui DT (2019) Reliability analysis of slope safety factor by using GPR and GP. *Geotech Geol Eng.* <https://doi.org/10.1007/s10706-018-0697-2>
- Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. *Adv Eng Softw.* <https://doi.org/10.1016/j.advengsoft.2017.01.004>
- Seifi A, Ehteram M, Singh VP, Mosavi A (2020) Modeling and uncertainty analysis of groundwater level using six evolutionary optimization algorithms hybridized with ANFIS, SVM, and ANN. *Sustainability.* <https://doi.org/10.3390/SU12104023>
- Shahin MA, Jaksa MB, Maier HR (2009) Recent advances and future challenges for artificial neural systems in geotechnical engineering applications. *Adv Artif Neural Syst* 2009:1–9. <https://doi.org/10.1155/2009/308239>
- Shahin MA, Jaksa MB, Maier HR (2001) Artificial neural network applications in geotechnical engineering. *Aust Geomech J*
- Smith EAL (2002) Pile-driving analysis by the wave equation. *Geotech Spec Publ*
- Stone RJ (1993) Improved statistical procedure for the evaluation of solar radiation estimation models. *Sol Energy.* [https://doi.org/10.1016/0038-092X\(93\)90124-7](https://doi.org/10.1016/0038-092X(93)90124-7)
- Susilo S (2006) Distribusi Gesekan Tanah pada Pondasi Tiang Bor Dalam. *Pertem. Ilmiah Tahunan-X HATTI*, p 107–109
- Taheri K, Hasanipanah M, Golzar SB, Majid MZA (2017) A hybrid artificial bee colony algorithm-artificial neural network for forecasting the blast-produced ground vibration. *Eng Comput.* <https://doi.org/10.1007/s00366-016-0497-3>
- Taylor KE (2001) Summarizing multiple aspects of model performance in a single diagram. *J Geophys Res Atmos* 106:7183–7192
- Tereshko V, Loengarov A (2005) Collective decision-making in honey bee foraging dynamics. *Comput Inf Syst J*
- Umar BU, Muazu MB, Kolo JG, Agajo J, Matthew ID (2019) Epilepsy detection using artificial neural network and grasshopper optimization algorithm (GOA). In: *2019 15th international conf. electron. comput. ICECCO 2019*
- Wang C, Zhou S, Wang B, Guo P, Su H (2016) Settlement behavior and controlling effectiveness of two types of rigid pile structure embankments in high-speed railways. *Geomech Eng* 11:847–865

- Wang X, Lu H, Wei X, Wei G, Behbahani SS, Iseley T (2020) Application of artificial neural network in tunnel engineering: a systematic review. *IEEE Access*. 8:119527–119543
- Xu C, Gordan B, Koopialipoor M, Armaghani DJ, Tahir MM, Zhang X (2019) Improving performance of retaining walls under dynamic conditions developing an optimized ANN based on ant colony optimization technique. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2019.2927632>
- Yin ZY, Jin YF, Liu ZQ (2020) Practice of artificial intelligence in geotechnical engineering. *J Zhejiang Univ Sci A* 21:407–411
- Zhang WG, Goh ATC (2013) Multivariate adaptive regression splines for analysis of geotechnical engineering systems. *Comput Geotech* 48:82–95. <https://doi.org/10.1016/j.compgeo.2012.09.016>
- Zhang W, Goh ATC (2016) Multivariate adaptive regression splines and neural network models for prediction of pile drivability. *Geosci Front* 7(1):45–52. <https://doi.org/10.1016/j.gsf.2014.10.003>
- Zhang H, Nguyen H, Bui XN, Nguyen-Thoi T, Bui TT, Nguyen N, Vu DA, Mahesh V, Moayedi H (2020a) Developing a novel artificial intelligence model to estimate the capital cost of mining projects using deep neural network-based ant colony optimization algorithm. *Resour Policy*. <https://doi.org/10.1016/j.resourpol.2020.101604>
- Zhang Q, Barri K, Jiao P, Salehi H, Alavi AH (2020b) Genetic programming in civil engineering: advent, applications and future trends. *Artif Intell Rev*. <https://doi.org/10.1007/s10462-020-09894-7>
- Zhang W, Zhang R, Wu C, Goh ATC, Lacasse S, Liu Z, Liu H (2020c) State-of-the-art review of soft computing applications in underground excavations. *Geosci Front* 11(4):1095–1106. <https://doi.org/10.1016/j.gsf.2019.12.003>
- Zhang W, Wu C, Li Y, Wang L, Samui P (2021) Assessment of pile drivability using random forest regression and multivariate adaptive regression splines. *Georisk Assess Manag Risk Eng Syst Geohazards* 15(1):27–40. <https://doi.org/10.1080/17499518.2019.1674340>

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.