Advances in Computational Design, *Vol. 8, No. 1 (2023) 37-59*
https://doi.org/10.12989/acd.2023.8.1.037

Ensemble techniques and hybrid intelligence algorithms for shear strength prediction of squat reinforced concrete walls

Mohammad Sadegh Barkhordari[1a] and Leonardo M. Massone*[2]

[1]*Department of Civil and Environmental Engineering, Amirkabir University of Technology, Tehran, Iran*
[2]*Department of Civil Engineering, University of Chile, Blanco Encalada 2002, Santiago, Chile*

(Received September 16, 2022, Revised November 12, 2022, Accepted November 17, 2022)

Abstract. Squat reinforced concrete (SRC) shear walls are a critical part of the structure for both office/residential buildings and nuclear structures due to their significant role in withstanding seismic loads. Despite this, empirical formulae in current design standards and published studies demonstrate a considerable disparity in predicting SRC wall shear strength. The goal of this research is to develop and evaluate hybrid and ensemble artificial neural network (ANN) models. State-of-the-art population-based algorithms are used in this research for hybrid intelligence algorithms. Six models are developed, including Honey Badger Algorithm (HBA) with ANN (HBA-ANN), Hunger Games Search with ANN (HGS-ANN), fitness-distance balance coyote optimization algorithm (FDB-COA) with ANN (FDB-COA-ANN), Averaging Ensemble (AE) neural network, Snapshot Ensemble (SE) neural network, and Stacked Generalization (SG) ensemble neural network. A total of 434 test results of SRC walls is utilized to train and assess the models. The results reveal that the SG model not only minimizes prediction variance but also produces predictions (with $R^2 = 0.99$) that are superior to other models.

Keywords: ensemble learning methods; optimization algorithm; shear strength; squat reinforced concrete wall

1. Introduction

Reinforced concrete (RC) walls with an aspect ratio of less than two are extensively employed in buildings and nuclear structures (Barkhordari *et al*. 2022). Squat, short, or low aspect ratio walls are terms used in the literature and by building-design specialists to define such walls. Squat RC (SRC) walls supply much, if not all, of a buildings' lateral stability and rigidity, allowing it to withstand earthquakes. Unlike conventional slender and moderate-aspect-ratio walls, which have been extensively investigated in recent decades (Barkhordari *et al*. 2021a), the behavior of squat walls is less well understood, particularly in terms of maximum shear strength. Kassem (2015), Ma and Li (2018), and Gondia *et al*. (2020) discovered that the mathematical relationships in building codes resulted in significant scatter when predicting the peak shear strength of squat walls, particularly those with a flanged cross-section. Also, numerical methodologies have been used in the past to quantify the shear strength of squat walls based on semi-empirical expressions

*Corresponding author, Ph.D., E-mail: m.s.barkhordari@aut.ac.ir
[a] Professor, E-mail: lmassone@ing.uchile.cl

 ISSN: 2383-8477 (Print), 2466-0523 (Online)

or using simplified physics-based approaches (e.g., expressions based on strut-and-tie (Massone and Ulloa 2014)) or more complex macro- and micro-models (e.g., (Kolozvari *et al*. 2019, Barkhordari *et al*. 2021b, Rojas *et al*. 2016, Massone *et al*. 2021)). Squat walls, on the other hand, can have a wide range of interrelated design parameters, as well as sophisticated nonlinear shear behaviors. Consequently, such mathematical models encounter meaningful inaccuracy when utilized to estimate experimental data of walls other than those utilized to calibrate models (Gondia *et al*. 2020). The design standards' expressions (e.g., ACI 318 (Committee 2019)) are mainly based on empirical fitting of a variety of experimental datasets that differ for each expression. The most commonly used fitting techniques in civil engineering works are traditional mathematical computing approaches (e.g., nonlinear regression (Ma *et al*. 2020)). Nonetheless, such methods have demonstrated apparent flaws when dealing with complicated systems (Siam *et al*. 2019).

Recently, machine learning techniques have been widely adopted (Amezquita-Sancheza *et al*. 2020, Moradi *et al*. 2020) in structural engineering for (1) predicting the response of structures (Mohammed *et al*. 2021, Guan *et al*. 2021, Barkhordari and Es-haghi 2021), (2) analyzing experimental data and developing models to estimate component response (Barkhordari and Tehranizadeh 2021, Aggarwal *et al*. 2019), and (3) structuring of walls distribution (Pizarro and Massone 2021, Pizarro *et al*. 2021). When it comes to handling nonlinear problems, machine learning techniques have a distinct advantage because the variables involved make it difficult to explain nonlinear problems in the common mathematical form. Nguyen *et al*. (2021) used an artificial neural network (ANN) for shear strength estimation of squat flanged RC walls using a dataset of 369 squat walls. They split data into testing and validating sets (12.5% of the data), and a training set (75%). The coefficient of determination of their model for the testing set was 0.97. Gondia *et al*. (2020) proposed a genetic programming-based equation for the shear strength estimation of SRC with boundary elements using 254 test results. They divided the dataset into a training (70% of the data) and a testing set (30% of the data). The coefficient of determination of their model for the testing set was 0.95. Chen *et al*. (2018) used the ANN combined with the particle swarm optimization algorithm and developed a hybrid algorithm (PSO-ANN) for the prediction of SRC walls' shear strength using 139 laboratory-tested specimens. They used 80% of data in the training phase and the remaining data (20%) was utilized for the testing phase. The coefficient of determination value given by the model for the testing set was 0.976. Feng *et al*. (2021) used the eXtreme Gradient Boosting algorithm to develop a model for estimating the SRC walls' shear strength using almost 430 laboratory-tested specimens. They split randomly the data into training and testing sets using the proportions of 70%–30%, respectively. The coefficient of determination of their model for testing set was 0.97. Supervised machine learning, on the other hand, is data-hungry, and its performance is strongly reliant on the amount of data provided. The size of a database has a lot to do with machine learning performance and the generalization of a proposed model. Due to the inherent complicated constraints and various design variables, developing a model for determining the shear strength of SRC walls is a highly non-linear and non-convex problem. Furthermore, when utilizing the traditional algorithms and methods, there is no guarantee of obtaining a global solution.

New optimization algorithms have been developed in recent years whose ability to improve the performance of predictive models has not been investigated. Moreover, neural networks are taught using a stochastic training technique, which means they are vulnerable to the training data's peculiarities and may find different weights and biases each round they are trained, resulting in different predictions. Training many models instead of a single model and combining the outputs

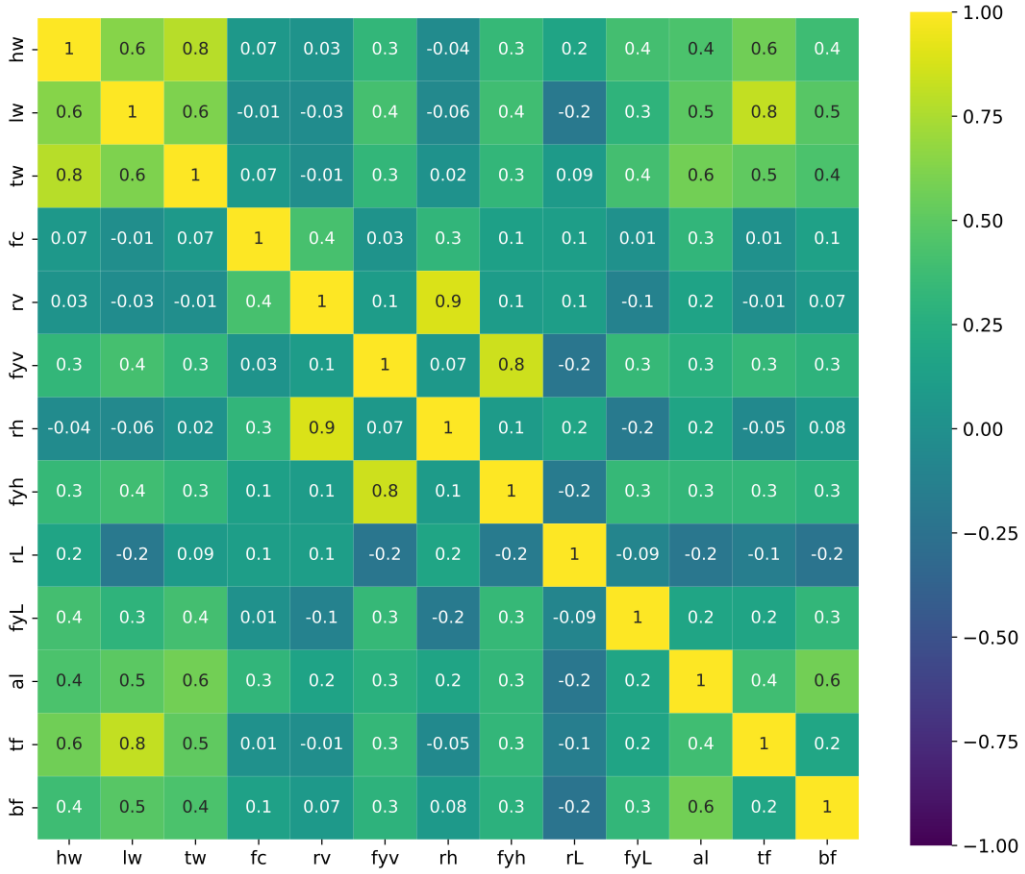


Fig. 1 Relationships between input variables

from these models is one way to reduce the dispersion of neural network outputs, as it is done by ensemble neural network models. To the best of the authors' knowledge, no research for the shear strength estimation of the SRC walls has used ensemble neural network models or new hybrid models (state-of-the-art population-based methods combined with ANN). To investigate the efficiency of these models in developing a suitable model to predict the maximum shear capacity of the wall, in this research, six models have been developed using 434 samples of SRC walls, including Honey Badger Algorithm (HBA) with ANN (HBA-ANN), Hunger Games Search with ANN (HGS-ANN), fitness-distance balance coyote optimization algorithm (FDB-COA) with ANN (FDB-COA-ANN), Averaging Ensemble (AE) neural network, Snapshot Ensemble (SE) neural network, and Stacked Generalization (SG) ensemble neural network. A detailed comparison between models is provided in terms of accuracy, standard deviations, among others.

2. Database

For calibrating their weights and biases, ANNs require a reliable database. Therefore, a total of 434 SRC wall tests from the literature are used in this study. The sets of samples acquired by

Table 1 Variables of shear strength database

Variable		Unit	Symbol	Mean	STD	Min	Max
Height		mm	hw	893.7	497.4	150	2200
Length (total)		mm	lw	1331.3	782.9	420	3960
Web thickness		mm	tw	70.4	43.1	10	160
Concrete strength		MPa	fc	26.6	12.1	12.3	104
Reinforcement ratio	vertical web	%	rv	0.66	0.57	0	3.67
	horizontal web	%	rh	0.65	0.54	0	3.67
	longitudinal	%	rL	3.25	2.05	0.4	10.58
Reinforcement strength	vertical web	MPa	fyv	357.7	119.4	0	624
	horizontal web	MPa	fyh	360.2	121.3	0	624
	longitudinal	MPa	fyL	372.8	76.2	208.9	605
Axial load		kN	al	178.1	370.0	0	2365
Flange thickness		mm	tf	123.4	76.0	30	360
Flange height		mm	bf	220.4	290.7	30	3045
Shear strength		kN	Pu	526.7	644.8	16.4	3138.1

Massone and Melo (2018), Feng *et al*. (2021), Ning and Li (2017), and Gulec (2009) are combined in this database. Duplicated specimens are eliminated. The final database includes a large number of squat wall parameters, which is expected to improve the models' prediction accuracy. Table 1 shows the range, average, and standard deviation of the parameters. The minimum, maximum, and standard deviation of variables are represented in this table by Min, Max, and STD, respectively. The relationships between selected input parameters are visualized in Fig. 1. The correlation coefficient is an indicator of how strong two variables are linked. A higher correlation coefficient number indicates a strong linear relationship between two variables, whereas a lower value indicates a poor relationship. For example, it can be seen from Fig. 1 that the vertical and horizontal web reinforcement strength also have a correlation coefficient of 0.8. The flange thickness has a correlation coefficient of 0.8 with the total length of the wall. Note that this study is aimed at establishing a relation between design variables (e.g., material and geometry properties) and shear strength regardless of the failure mode and the selection of optimal features. The data is normalized and features re-scale within the range of 1 and -1. An overview of variables distributions is presented in Fig. 2.

3. Methodology

ANN usually is a collection of interlinked neurons with a single output. Inner neurons accept numerous inputs. The output is a product of the neuron's inputs and an activation function. The activation function is the equation that applies to a weighted combination of neuron inputs plus bias. Network training is usually done using the backpropagation algorithm for training feed-forward. However, backpropagation's most significant flaw is that it can be vulnerable to noisy data. The quest for ideal values of two crucial training parameters, learning rate and momentum weight, which are established empirically in most networks, is another disadvantage of the

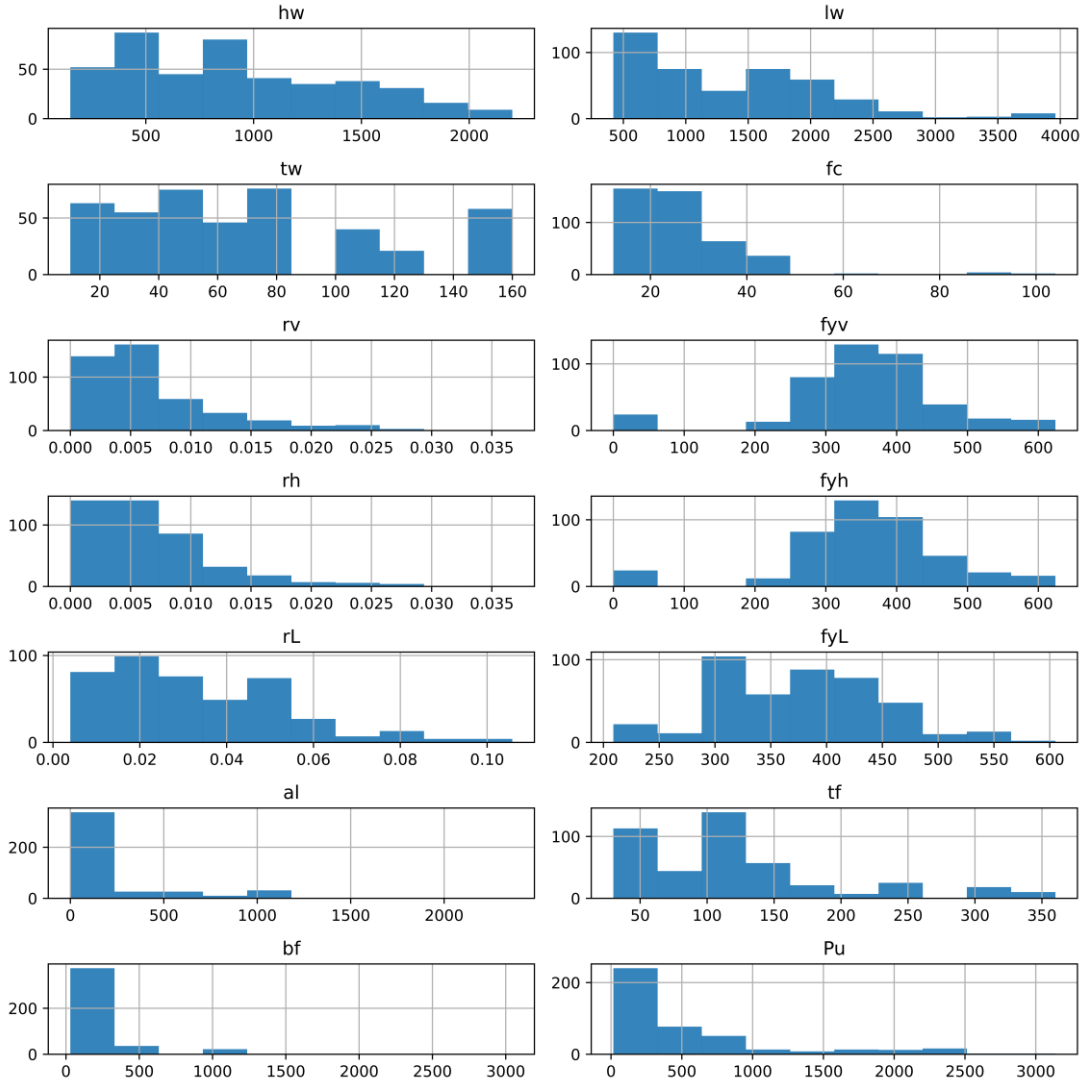


Fig. 2 Distributions of variables (histogram)

backpropagation-based optimizer (Jaeger 2020). There are other alternative methods for determining the weight and bias of shallow neural networks. Among the existing methods for determining the weights and biases of the shallow neural network (networks with one hidden layer) is the use of population-based algorithms. The purpose of these algorithms is to find the appropriate coefficients for an existing equation or relationship by reducing the value of an objective function. Therefore, the weight and bias of shallow neural networks can be considered as the unknown parameters for these algorithms and the root mean square error can be used as the objective function. However, shallow neural networks most of the time cannot learn complex information very well. Another way to deal with these limitations of neural networks is to use ensemble models. The objective of the ensemble models is to aggregate predictions from several adequate, but dissimilar deep ANNs. Also, introducing non-linearity (increasing the number of the

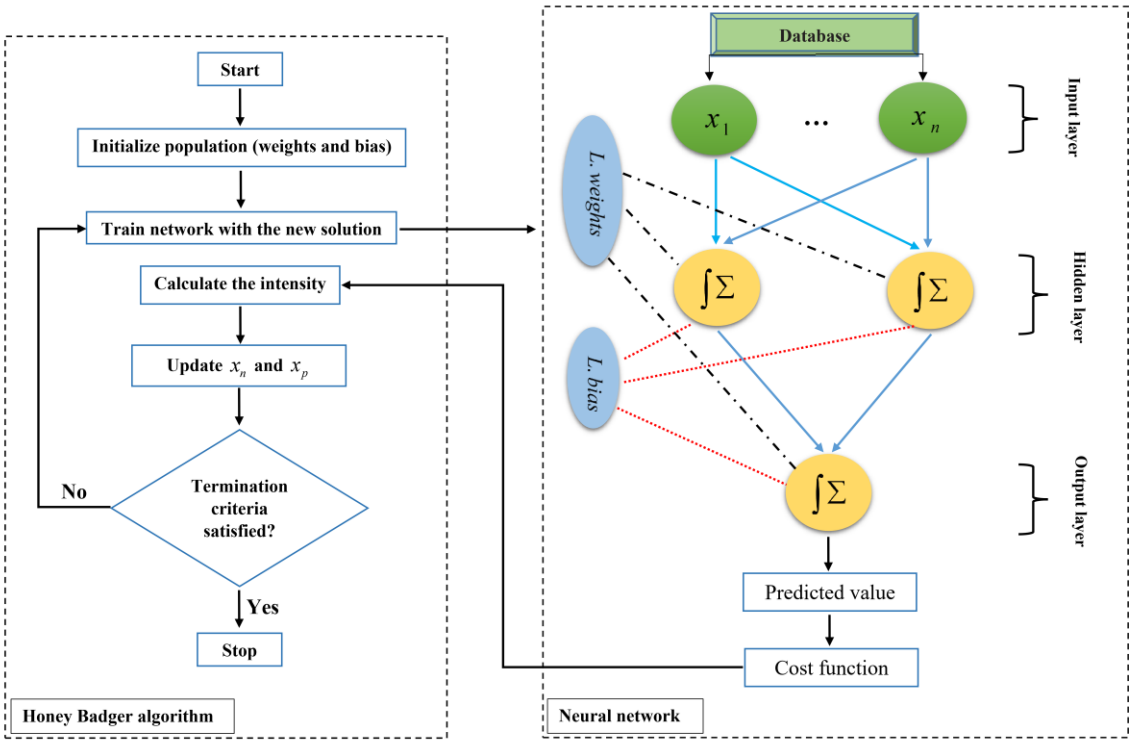


Fig. 3 Flowchart of the HBA-ANN algorithm

hidden layers) makes the neural net able to learn more complex information. In this study all of these techniques are utilized to estimate the shear strength of squat reinforced concrete walls. First, data-driven hybrid models are developed by combining state-of-the-art population-based algorithms and artificial neural networks. Then ensemble approaches are employed using deep ANNs. In the following section, the models of these methods are briefly described.

3.1 Overview of HBA-ANN

Honey Badger Algorithm (HBA) (Hashim *et al*. 2021) is an optimization algorithm that imitates the behavior of the honey badger. The honey badger either digs or tracks the honeyguide bird for food resources. It utilized its smelling skills to assess the location of the quarry in the digging phase, then maneuvered around the quarry to select the optimum spot for digging and catching the quarry whenever it arrived. The honey badger also uses the honeyguide bird as a lead to discover the beehive in the honey phase. The first circumstance is known as honey mode, and the second is known as digging mode. During the digging mode, a honey badger adopts a cardioid shape. Eq. 1 represents the mathematical model of cardioid motion. Eq. 2 is used in digging mode, when a honey badger follows a honey guide bird.

$$x_n = x_p + F \times r_7 \times \alpha \times d_i, d_i = x_p - x_i, \alpha = 2 \times exp(\frac{-t}{t_{max}}) \quad (1)$$

$$x_n = x_p + 6 \times F \times I \times x_p + F \times r_3 \times \alpha \times d_i \times |cos(2\pi r_4) \times [1 - cos(2\pi r_5)]| \quad (2)$$

$$I_i = r_2 \times \frac{S}{4\pi d_i^2}, S = (x_i - x_{i+1})^2 \quad (3)$$

$$F = \begin{cases} 1 \; if \; r_6 \leq 0.5 \\ -1 \; else \end{cases} \quad (4)$$

where $x_i$is i[th] candidate solution, $x_p$is so-far-best position, $x_n$is a new position,$r_i$is a random number in $(0,1)$, $t$is iteration number, $S$is concentration strength, t_{max} is the maximum number of iterations, and $I_i$is smell intensity of the quarry. In this study, HBA is integrated with ANN to find the weights and biases of the neural network. Fig. 3 represents the flowchart of the HBA-ANN algorithm.

3.2 Overview of HGS-ANN

The Hunger Games Search (HGS) (Yang *et al*. 2021) is based on social animals' teamwork, in which their search effort is related to their hunger levels. Hunger is a powerful motivator for activity, learning, and looking for food in any animal, according to neuroscientists (Sutton and Krashes 2020), and it functions as a pressure toward modifying the life state to a more stable condition. The common properties of social species and their food-seeking are used to build this algorithm. The core equation of the HGS method for animal cooperative communication and foraging behavior is represented by Eq. 5.

$$\vec{P}_{(t+1)} = \begin{cases} \vec{P}_{(t)} \cdot \left(1 + r \; and \; n(1)\right), & r_1 < 0.08(5-1) \\ \overrightarrow{W_1} \cdot \vec{P}_b + \vec{R} \cdot \overrightarrow{W_2} \cdot \left|\vec{P}_b - \vec{P}_{(t)}\right|, & r_1 > 0.08, r_2 > E(5-2) \\ \overrightarrow{W_1} \cdot \vec{P}_b - \vec{R} \cdot \overrightarrow{W_2} \cdot \left|\vec{P}_b - \vec{P}_{(t)}\right|, & r_1 > 0.08, r_2 < E(5-3) \end{cases} \quad (5)$$

$$E = \frac{2}{e^{|F(i)-BF|} + e^{-|F(i)-BF|}}, \vec{R} = 4 \cdot (1 - \frac{t}{tmax_{max}}) \quad (6)$$

$$\begin{aligned} \overrightarrow{W_1} &= \begin{cases} hungry(i) \cdot \dfrac{NI}{SHungry} \cdot r_4, r_3 < 0.08 \\ 1 r_3 > 0.08 \end{cases} \\ \overrightarrow{W_2} &= (1 - exp(-|hungry(i) - SHungry|)) \cdot r_5 \cdot 2 \end{aligned} \quad (7)$$

$$\begin{aligned} hungry(i) &= \begin{cases} 0, & FEICI(i) == BF \\ hungry(i) + H, & FEICI(i) \neq BF \end{cases} \\ H &= \begin{cases} 100 \cdot (1 + r) \; TH < 100 \\ THTH \geq 100 \end{cases}, \quad TH = \frac{F(i) - BF}{WF - BF} \cdot r_6 \cdot 2 \cdot (UB - LB) \end{aligned} \quad (8)$$

where r_i is a random number in $[0.1]$, $t$is iteration number, t_{max} is the maximum number of iteration, $\vec{R}$ is a ranging controller, $\overrightarrow{W_i}$ are the weights of hunger, $\vec{P}_{(t)}$ is the position of the individual, $\vec{P}_b$ is the best individual position of the current iteration, $randn$is a random number that conforms to the normal distribution, $SHungry$ is the sum of all member's hunger sensations,

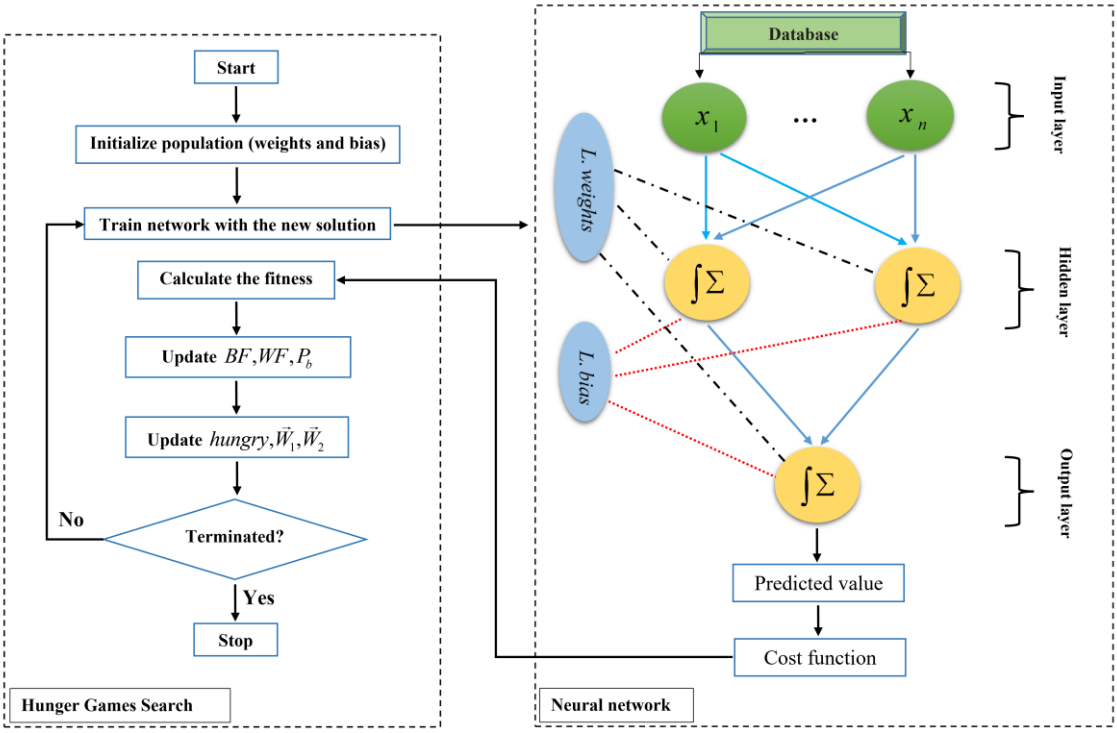


Fig. 4 Flowchart of the HGS-ANN algorithm

$FEICI(i)$ keeps track for every individual's fitness in the current loop, $BF$is the so-far best obtained-fitness, $NI$is the number of individuals, and the upper and lower boundaries of the feature set are denoted by $UB$and LB, respectively. The worst fitness gained in the current loop procedure is denoted by WF, and each individual's fitness score is represented by $F(i)$. There are two types of search directions available in this algorithm. The first search direction (Eq. 5-1) replicates an individual who lacks a sense of collaboration, is uninterested in cooperating, and only wants to hunt for food. The variables $\vec{W}_1$, $\vec{W}_2$, and $\vec{R}$ play rule in the second search direction (Eqs. 5-2 and 5-3). The individual's position within the feature space might be evolved by refining these three elements. This strategy simulates the cooperation of multiple entities searching for food. The starvation characteristics of each sample in search space are simulated by Eqs. 7 and 8. In the current study, HGS is integrated with ANN to find the weights and biases of the neural network. Fig. 4 represents the flowchart of the HGS-ANN algorithm.

3.3 Overview of FDB-COA-ANN algorithm

Fitness-distance balance coyote optimization algorithm (FDB-COA) (Duman *et al*. 2021) is a modified version of the coyote optimization algorithm in which Levy flight has been used to improve a system that mimics the birth of new coyotes. In conventional COA, packs make up the population, which is separated into subgroups. Several coyotes and an alpha wolf make up each pack. The location of c^{th}candidate coyote in $p^{th}$the group at time t is represented by $loc_c^{p,t} =$

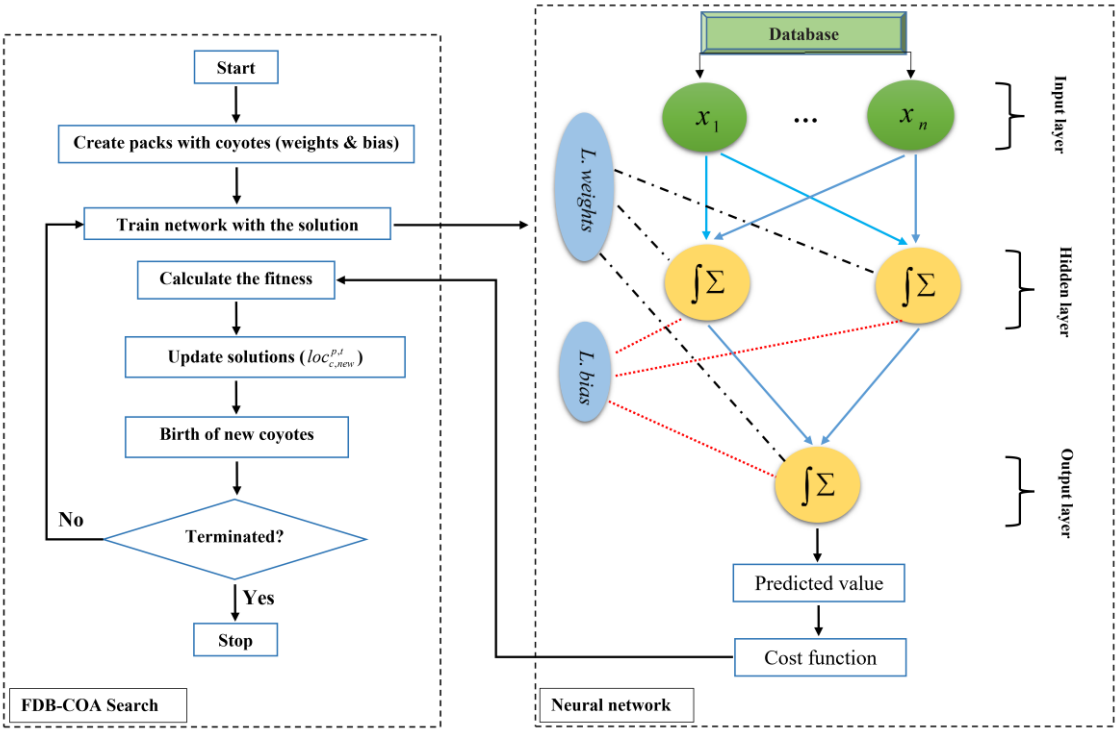


Fig. 5 Flowchart of the FDB-COA-ANN algorithm

$(x_1, x_2, \ldots x_{dim\,ension})$ where $x_i$is the design variable. Initially, some arbitrary cayotes are created as candidates in the search space. Then, the new solution $(loc_{c,new}^{p,t})$ is created as given in Eq. 9.

$$loc_{c,new}^{p,t} = loc_{c}^{p,t} + r_1 \cdot \delta_1 + r_2 \cdot \delta_2 \tag{9}$$

where $\delta_1$represents the alpha effect, δ_2 represents the pack effect, and the weight coefficient (r_i) is created at random in the range(0,1]. The coyote will just be allowed into the group if its new social position is better than previously using the result of the cost function of each coyote. Here, the cost function is the coefficient of determination. The Levy distribution is used to simulate the birth of a solution/coyote $(pup_j^{p,t})$, as seen in Eq. 10.

$$\begin{aligned} pup_j^{p,t} &= \begin{cases} loc_{cr_1}^{p,t}\ rnd_j < P_s\ or\ j = j_1 \\ loc_{cr2}^{p,t}\ rnd_j > P_s + P_a\ or\ j = j_1 \end{cases} \\ rnd_j &= \sqrt{\frac{\gamma}{2\pi}} \cdot \frac{1}{(s-\mu)^3/2}, P_s = \frac{1}{Dimension}, P_a = (1 - P_s)/2 \end{aligned} \tag{10}$$

where μ is the step size of the Levy flight distribution, cr_i is s coyote that is chosen at random, $s$is a sample of the Levy flight distribution, j_1 is randomly selected gen, and $\gamma$is control parameters of the Levy flight distribution. Fig. 5 represents the flowchart of the FDB-COA-ANN algorithm.

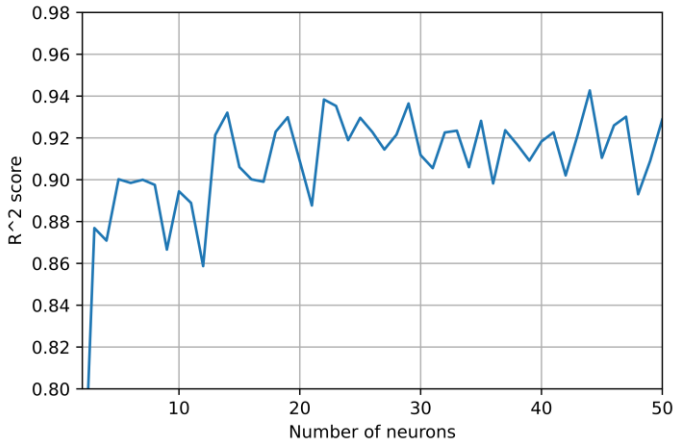


Fig. 6 Determining the minimum number of neurons

3.4 Ensemble model

Ensemble learning (Dietterich 2000, Lu and Paffenroth 2021) mixes predictions from several neural networks to decrease prediction variance, decrease generalization error, and improve model accuracy. Here three Ensemble models are considered including model averaging, snapshot, and stacked generalization. The neural network used in the model averaging and snapshot ensemble is the same as the usual neural network with a hidden layer and Adam optimizer. First, the minimum number of neurons in the neural network with the highest accuracy is determined. Fig. 6 shows the changes in the number of neurons versus the coefficient of determination of the models. The curve appears to fluctuate between 0.94 and 0.90. As shown in Fig. 6, 22 neurons can be a good choice. Therefore, the single neural network structure (base model) in the ensemble models of this study (model averaging and snapshot) consists of only one hidden layer with 22 neurons.

3.4.1 Overview of model averaging

Model averaging (Brownlee 2018) is an ensemble learning strategy for reducing the variance of the final machine learning algorithm. In this situation, the model's performance may be compromised to ensure the model's expected performance. The most straightforward method for creating a model averaging ensemble is to train many models on the same data and then average the results from each model. The number of models necessary for a group might vary based on the problem's and model's complexity. This strategy has the advantage of allowing the development of a large number of models, adding them to the collection, and evaluating the performance of the ensemble model. Fig. 7 shows the model averaging neural network procedure. Following the preparation of the models, each model is used to produce an output, and the outputs can be gathered and averaged. The point of the model averaging ensemble is that models are made using a small number of epochs. This not only makes the models differ from each other but also prevents overfitting (Brownlee 2018). Here, the models are fit for 500 training epochs (Brownlee 2018).

3.4.2 Overview of snapshot ensemble

Saving several models during a single training session and combining their predictions to

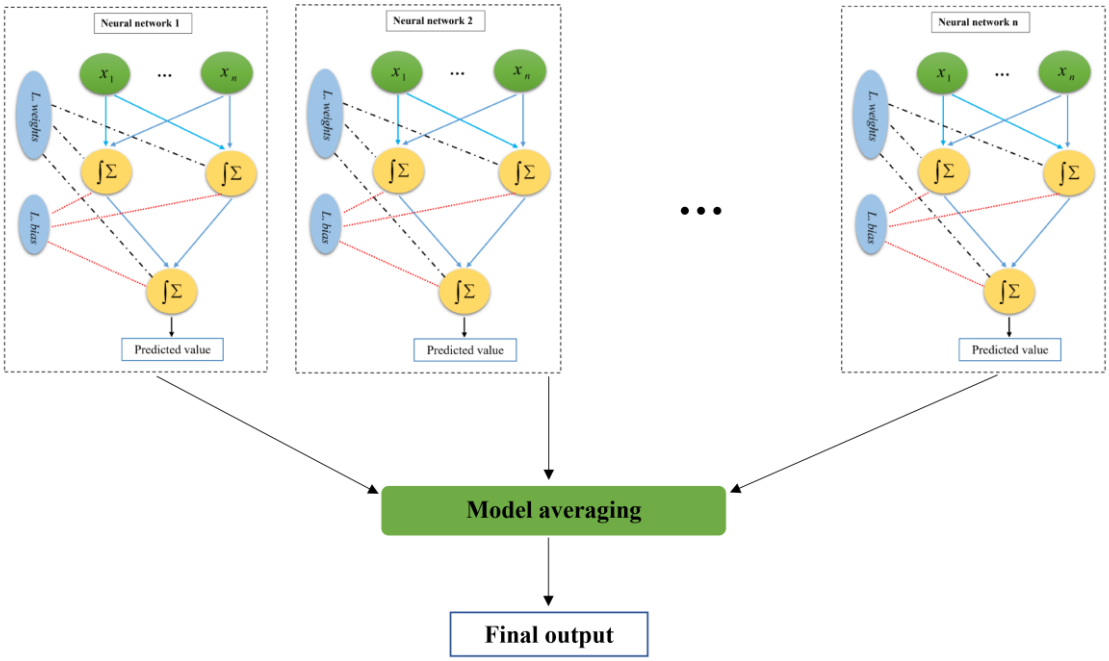


Fig. 7 Model averaging strategy for shear strength prediction

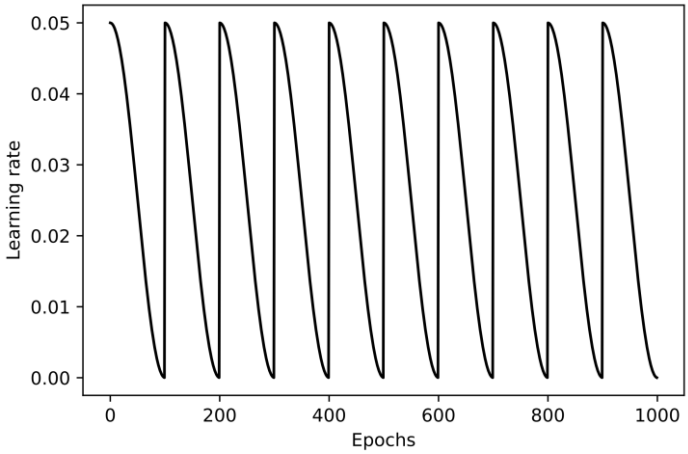


Fig. 8 Learning rate schedule

generate an ensemble prediction is an option to generate an ensemble model. Snapshot ensembles (Brownlee 2018, Huang *et al*. 2017, Loshchilov and Hutter 2016) integrate predictions from several models that are stored over a single training session. This approach also helps to control computational cost, as the model runs only once. The drawback of this strategy is that recorded models would be similar, resulting in little variability from the original models in terms of errors and predictions. A learning rate schedule (Loshchilov and Hutter 2016) is used to save a wide range of skilled ensembles throughout a single training session, forcing substantial changes in the

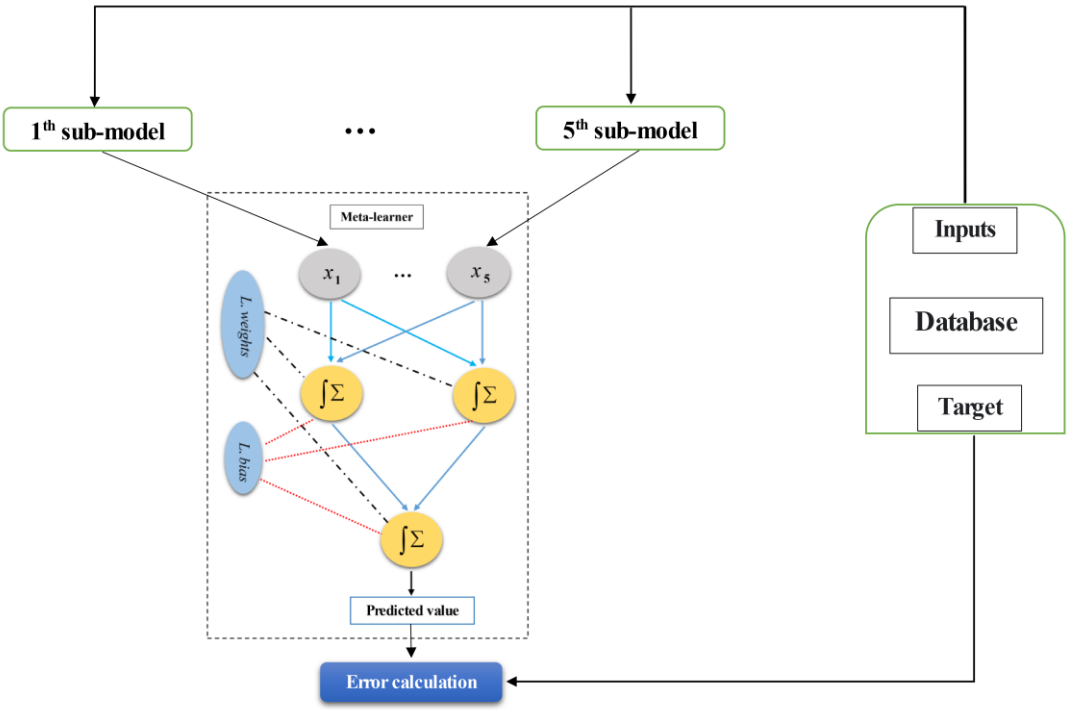


Fig. 9 Stacked generalization model's schematic diagram

model weights and, as a result, the characteristics of the model recorded at each snapshot. In this approach, the learning rate is changed during training epochs using the cosine annealing learning rate equation (Eq. 11, (Huang *et al*. 2017)). The rate of learning starts with a high value and then decreases rapidly to a small amount before increasing again. Repeating this cycle will result in good weights at the end of each cycle, providing a snapshot of the model.

$$lr(t) = \frac{lr_0}{2}\left(cos(\frac{\pi\%(t-1, \lfloor T/M \rfloor)}{\lfloor T/M \rfloor}) + 1\right) \tag{11}$$

where $lr(t)$ is the learning rate at epoch t, lr_0 is the maximum learning rate, $T$is the total epochs, M is the number of cycles, % is the modulo operation, and $\lfloor\ \rfloor$ presents a floor operation. Fig. 8 shows a line plot of the learning rate schedule used in this study where $T = 1000, M = 10, lr_0 = 0.05$.

3.4.3 Overview of stacked generalization

One of the important limitations of the averaging ensemble method is that the contribution of each sub-model to the final prediction is considered to be equal. While among the sub-models there may be a model that has performed better than the others. The results of each sub-model can be weighted to improve model averaging. This can be taken even further by training a completely new model to discover how to mix the output of each sub-model in the best possible way. This method is known as stacked generalization (SG) (Brownlee 2018, Naimi and Balzer 2018). In other words, in stacked generalization, an algorithm, so-called meta-learner, receives the

Table 2 Characteristics of sub-models

Sub-model		1	2	3	4	5
Number of neurons	Layer 1	20	20	20	20	20
	Layer 2	10	40	15	15	15
	Layer 3	-	5	20	15	40
	Layer 4	-	-	15	20	50
	Layer 5	-	-	-	15	20
	Layer 6	-	-	-	-	15
Activation		tanh	tanh	tanh	tanh	tanh
Optimizer		Adam	Adam	Adam	Adam	Adam
Learning rate		0.01	0.01	0.01	0.01	0.01
R^2 score		0.95	0.94	0.97	0.92	0.96

Table 3 Characteristics of meta-learner

Meta-learner		
Number of neurons	Layer 1	40
Activation		tanh
Optimizer		Nadam
Learning rate		0.02

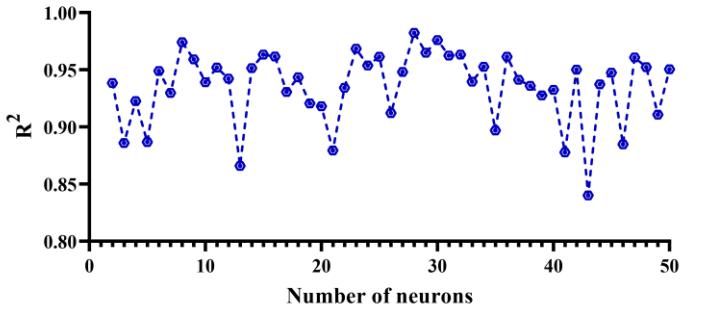
prediction of the sub-models as input and tries to figure out how to merge the predictions in the best way possible to get a superior output prediction (Fig. 9). Usually, at least five sub-models are used.

In this study, five deep neural networks with a various number of layers and neurons are used as sub-models. Here, a trial-and-error procedure (GridSearchCV) is used to identify the number of neurons in each layer, optimizer, hyper-parameters, among others, for each sub-model and the meta-learner using the training set. Tables 2 and 3 show the characteristics of the sub-models and the meta-learner, respectively. The meta-learner is a shallow neural network and has only one hidden layer. The weights and biases of the sub-models will not be updated during stacked generalization model training and only the weights and biases of the meta-learner will be updated.

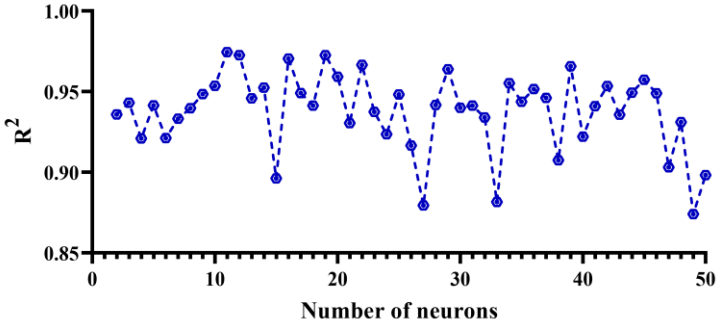
4. Result and discussion

In this section, the performance of the models described in the previous section is examined. For models developed in combination with population-based algorithms (HBA-ANN, HGS-ANN, and FDB-COA-ANN), the appropriate number of neurons must first be determined. This is usually done using a trial-and-error process. The number of neurons shows how the level of interaction between parameters affects the accuracy.

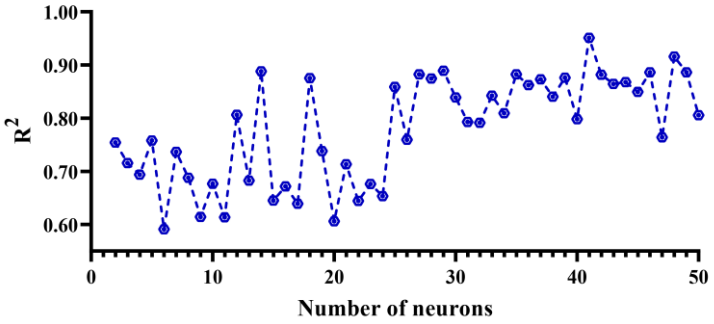
For this purpose, the number of layer neurons has been changed from 2 to 50 and the performance of the algorithms has been monitored and recorded. Fig. 10 shows the curve of coefficient of determination values versus the number of neurons. The maximum R^2 value of the



a) HBA-ANN



b) HGS-ANN



c) FDB-COA-ANN

Fig. 10 Performance curves of hybrid algorithms

HBA-ANN, HGS-ANN, and FDB-COA-ANN algorithm is 0.982, 0.974, 0.951 with 28, 11, 41 neurons, respectively.

The learning procedure of the all-ensemble model is the feed forward-back propagation procedure. As mentioned for a model averaging ensemble, multiple models (using the structure and architecture of the base model) are trained on the same dataset with 500 epochs and then their predictions are averaged. In order to determine how the number of members will impact the test accuracy, a sensitivity analysis of the member number is performed. Fig. 11 shows the changes in the number of members of a set (ensemble size) versus performance. The performance of each individual model also is shown as a red marker. A model averaging ensemble with eight models has the maximum value of $R^2 = 0.973$.

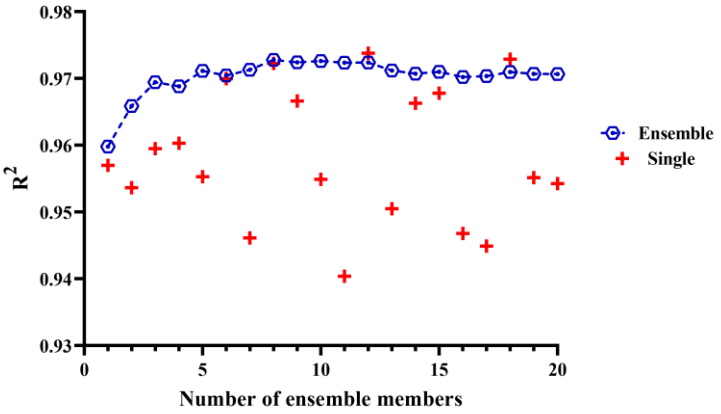


Fig. 11 Ensemble size versus model performance

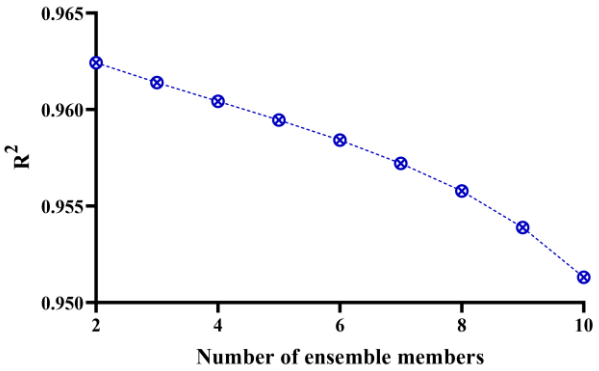


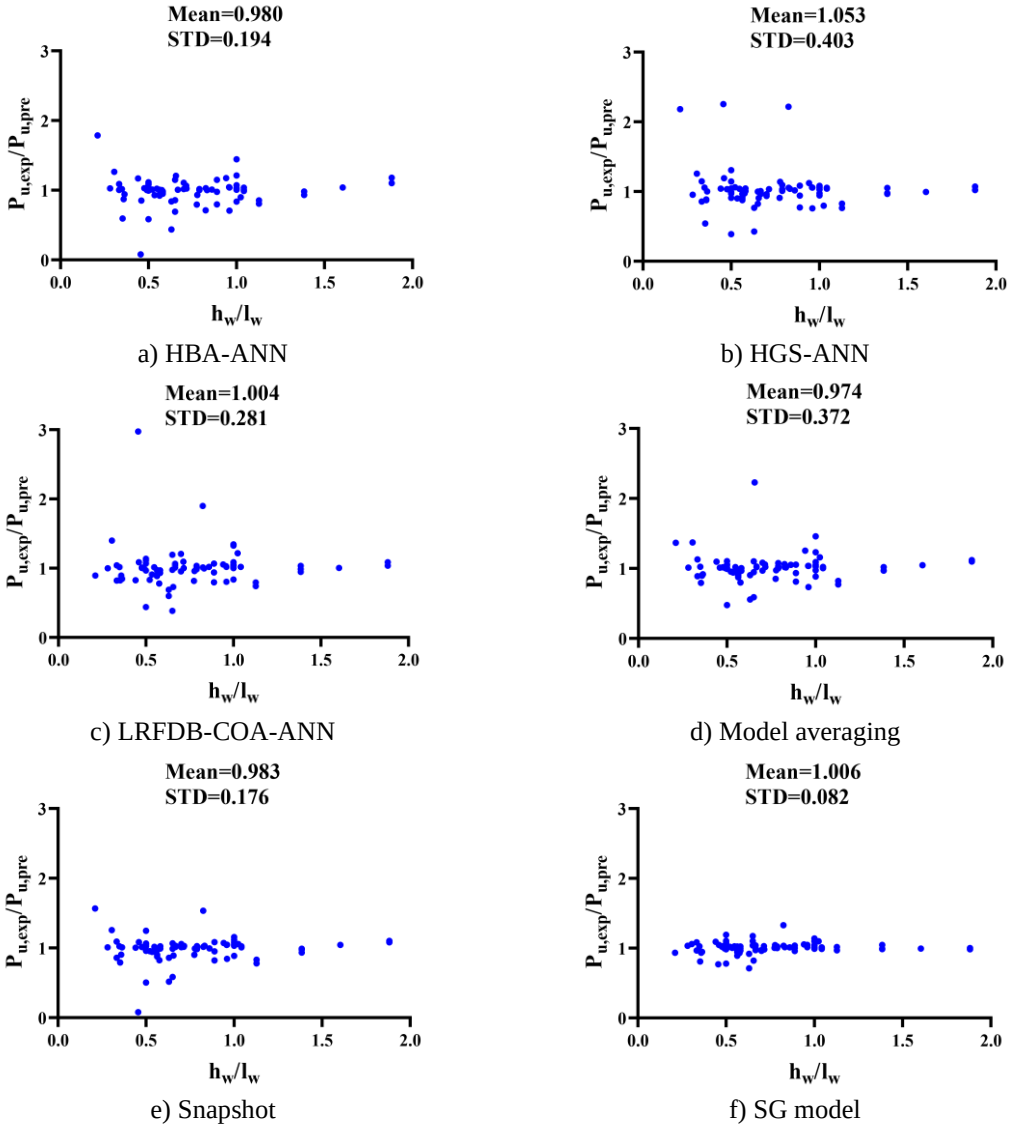
Fig. 12 Performance of individual and ensemble snapshot models

Snapshot ensemble is the second ensemble learning model that is introduced in the previous section. As mentioned, the base model is trained with an aggressive learning rate schedule and 10 models are recorded throughout the training phase to choose among them later when generating an ensemble prediction. We anticipate that models stored at the end of the session could behave better than those saved earlier in the runtime since they experience more training epochs. As a result, the list of loaded models is reordered. In other words, to create and increase the members of the ensemble model, the models of the collection are selected and added from the last to the first. In Fig. 12, the performance of the snapshot ensemble model, with increasing size (number of members) from 2 to 10, is illustrated. As can be seen from Fig. 12, the set (ensemble model) consisting of the last two single models of the training phase is the most accurate with $R^2 = 0.962$.

The third ensemble learning method is the SG model. As mentioned in the previous section, five deep neural networks are used as a sub-model and a network with only one hidden layer is utilized as the meta-leaner. Then, the defined SG neural network is fitted on the training dataset. The SG neural network is utilized to predict the test dataset. In this case, the SG neural network achieved an accuracy of about 0.99. Table 4 summarized the result of all models.

Table 4 Summary of model results

Model	HBA-ANN	HGS-ANN	FDB-COA-ANN	Averaging	Snapshot	SG model
R^2	0.982	0.974	0.951	0.973	0.962	0.99



a) HBA-ANN b) HGS-ANN

c) LRFDB-COA-ANN d) Model averaging

e) Snapshot f) SG model

Fig. 13 Models predicted values deviation

For more evaluation of the developed models' accuracy, Fig. 13 depicts the deviation distribution between real values and model predictions (experimentally-measured ($P_{u,exp}$) to the analytically-computed ratio ($P_{u,pre}$)) with respect to height (h_w)-to-length (l_w) ratios. Fig. 13 illustrates that HBA-ANN, HGS-ANN, FDB-COA-ANN, model averaging, and snapshot model

Table 5 Evaluation of the SG model and mechanics-based models

$Pu, pre_{u,exp}$			
Models	Mean	Standard deviation	Coeff. of variance
SG	1.006	0.082	0.081
ACI 318 (ACI-318 2014)	0.902	0.423	0.468
Ma *et al*. (2020)	0.785	0.366	0.466
ASCE (Committee 2005)	0.910	0.349	0.383

have a lot of variabilities when it comes to predicting the effect of the height-to-length ratio on SRC wall shear capacity, but the innovative SG model is better at predicting the SRC wall's shear strength by lowering variability to a minimum value. Since shear-type responses are anticipated for SRC walls with smaller aspect ratios, which has a lot more sophisticated mechanism to be represented, the results for SRC walls with smaller aspect ratios (e.g., < 1.0) show a significantly larger variation versus those SCR walls with greater aspect ratios.

The empirical-based equations are also considered in this study to provide a straightforward comparison with the SG model. Equations have either been highlighted in design codes or have recently been proposed by different researchers, including ASCE (Eq. 12, (Committee 2005)), Ma *et al*. (Eq. 13, (Ma *et al*. 2020)), and ACI 318 (Eq. 14, (ACI-318 2014)).

$$\begin{aligned} &V_n = [0.69\sqrt{f_c'} - 0.28\sqrt{f_c'}(\frac{h_w}{l_w} - 0.5) + \frac{P}{4 \cdot l_w \cdot t_w} + \rho_{se} f_{y,h}] \cdot d \cdot t_w \\ &\rho_{se} = A\rho_v + B\rho_h, V_n = 1.66\sqrt{f_c'}, d = 0.6l_w \\ &\quad \begin{cases} A = 1, B = 0, \ h_w/l_w < 0.5 \\ A = -h_w/l_w + 1.5, B = h_w/l_w - 0.5, 0.5 \le h_w/l_w \le 1.5 \\ A = 0, B = 1, \ h_w/l_w > 1.5 \end{cases} \end{aligned} \tag{12}$$

$$\begin{aligned} V_n = (0.32 f_{y,f} \rho_f t_f l_f + 0.18 f_{y,v} \rho_v t_w z_w + \frac{P}{2}) \frac{d_w}{h_w} + 0.54 f_{y,h} \rho_h t_w h_w \le 1.4 A_t \sqrt{f_c'} \\ d_w = l_w - t_f - 0.5 \left(\frac{0.32 f_{y,f} \rho_f t_f l_f + P}{0.59 f_c' t_w} - \frac{t_f l_f}{t_w} \right) \end{aligned} \tag{13}$$

$$\begin{aligned} &V_n = (\alpha_c \sqrt{f_c'} + \rho_h f_{y,h}) A_{cv} < 0.83 \sqrt{f_c'} A_{cw} \\ &\begin{cases} \alpha_c = 0.25, \ h_w/l_w < 1.5 \\ \alpha_c = interpolation, \ 1.5 \le h_w/l_w \le 2.0 \\ \alpha_c = 0.17, \ h_w/l_w > 2.0 \end{cases} \end{aligned} \tag{14}$$

where A_{cv} is the gross area bounded by the section length and web thickness, A_{cw} is the wall gross section, h_w is the height, l_w is the length, t_w is the web thickness, l_f is the flange height, t_f is the flange thickness, f_c' is the concrete strength, $\rho_{h/v}$ is the horizontal/vertical web reinforcement ratio, ρ_f is the flange longitudinal reinforcement ratio, $f_{y,h/v}$ is the yield strength of the horizontal/vertical reinforcements, and $P$is the axial force. Fig. 14 shows scatter plots of test results and the corresponding computed outcomes of the mechanics-based models and SG model for the test set. Statistical properties of the experimentally-measured ($P_{u,exp}$) to the

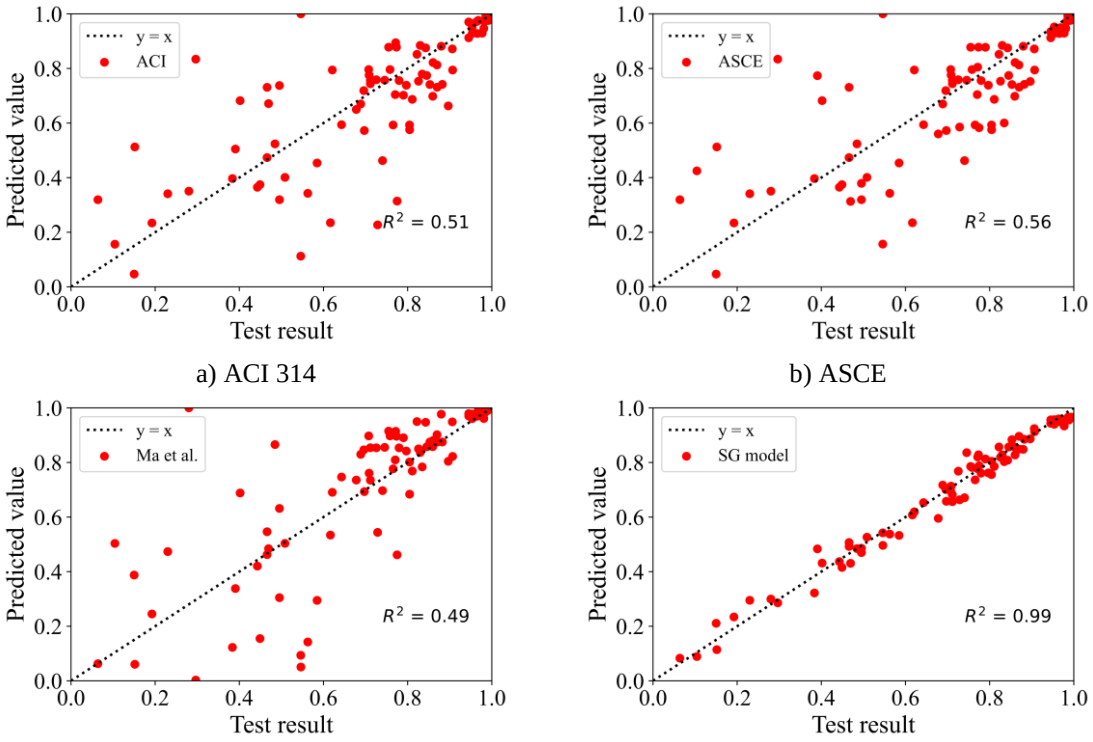


Fig. 14 Scatter diagrams of model and test results

analytically-computed ratio ($P_{u,pre}$) are presented for all equations in Table 5. The SG model has the best performance. Among mechanics-based models, the lowest coefficient of variation (0.383) is obtained by ASCE model, but the coefficient of variation by the SG model is only 0.081.

4.1 Interpretation of the SG model

Machine learning (ML) models (e.g., artificial neural networks) are usually unable to provide an explicit mapping between the predicted output and the input parameters, which is why they are often called black-box models. Practicing engineers, as well as design standards committees, are unwilling to adopt models lacking explicit predictive expressions because they do not know what is going on inside. SHAP (SHapley Additive Explanations) (Lundberg and Lee 2017) has made an important contribution to the field of ML model interpretation. The SHAP approach uses multiple methods to explain model predictions intuitively and theoretically. To put it another way, for estimating the Shapely value, surrogate models are mixed with notions from game theory. The mean marginal influence of a feature on the prediction through all conceivable subsets of features is the Shapely value of each estimation. The feature importance of the input variables, indicating the overall influence of the input parameters on the predictions, is shown in Fig. 15 (vertical axis labels are already defined in Table 1). It is calculated as the mean value of the absolute Shapley values of the whole dataset. As shown in Fig. 15, it is observed that the length of the wall, the web

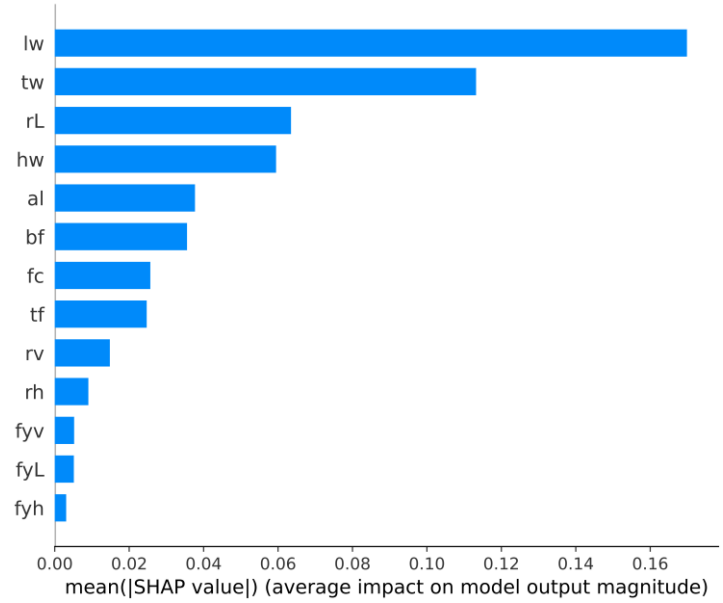


Fig. 15 Feature importance

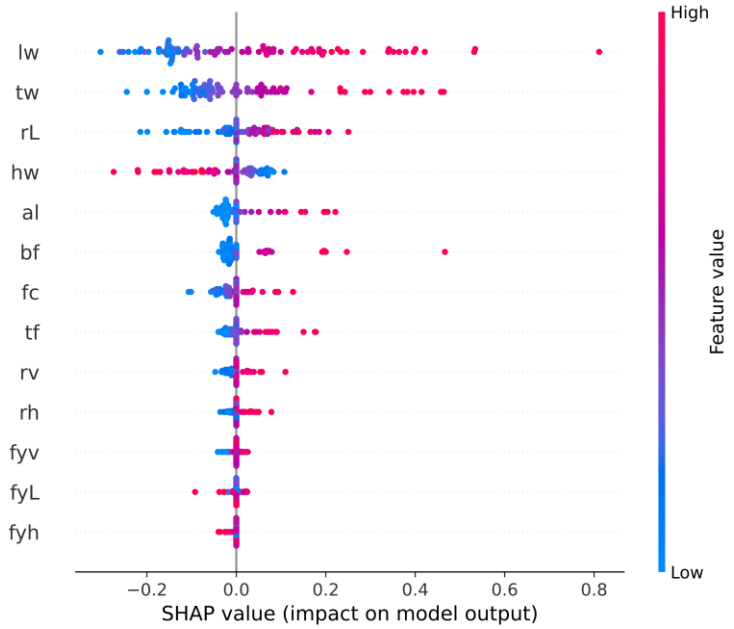


Fig. 16 Summary plots for the shear strength of SRC walls

thickness, the longitudinal reinforcement ratio, and the height of the wall have a great level of effect (most important variables) on shear strength prediction. Fig. 15 and distributions of input variables (Fig. 2) imply that geometric characteristics (such as length, web thickness, and height) have somewhat a stronger impact on shear strength than material attributes.

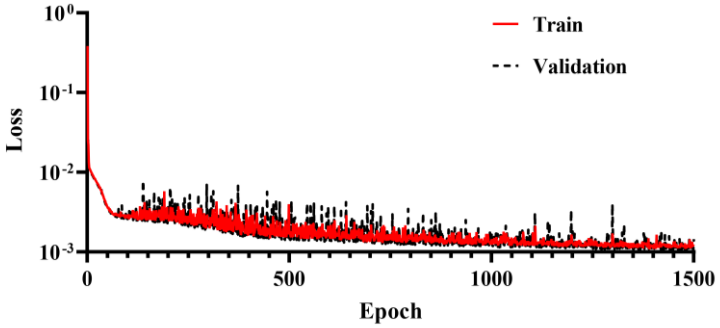


Fig. 17 Learning curve of the SG model

Fig. 16 shows the influence of each parameter, i.e., whether the estimation is changed positively or negatively. In Fig. 16, All the dots are Shapely values (the input variables). The color denotes the value of the characteristic (red high, blue low). As can be observed in Fig. 16, an increase in the length and web thickness increases the shear strength of the SRC wall. In contrast, as the values for variables like the height of the wall increase, the shear strength of the SRC wall tends to reduce. These results agree with the design equations presented earlier (Eq. 12 and 14). For example, it is revealed in Eq. 14 that decreases in the wall height and increases in wall length, result in an increase in the $\alpha_c$and the shear strength.

Another important point in interpreting the model is to examine the issue of overfitting. When a model learns the intricacies and noise in the training dataset to the point where it severely hampers the model's performance on new data, this is known as overfitting. This means that the model selects and refers to random noise or oscillations in the training dataset as concepts. The issue is that these notions do not apply to new data, resulting in limiting the model's generalization ability. The behavior of the learning curve can be used to assess the overfitting. Validation loss will reduce initially, then increase in the learning curve of a model that suffers from overfitting. A good fit model, on the other hand, is one whose curve first falls and then reaches a stable limit. Fig. 17 shows the learning curve of the SG model, which demonstrates a case of a good fit.

5. Conclusions

The use of squat reinforced concrete (SRC) shear walls in nuclear facilities and building constructions is common. Several prior investigations found significant variation in shear strength values based on empirical/numerical formulations, emphasizing the necessity for a more reliable prediction expression. In this respect, six models have been developed using 434 samples of SRC walls, including Honey Badger Algorithm (HBA) with ANN (HBA-ANN), Hunger Games Search with ANN (HGS-ANN), fitness-distance balance coyote optimization algorithm (FDB-COA) with ANN (FDB-COA-ANN), Averaging Ensemble (AE) neural network, Snapshot Ensemble (SE) neural network, and Stacked Generalization (SG) ensemble neural network. The following is a summary of the current study primary conclusions:

• SG ensemble neural network outperforms other ensemble models and hybrid intelligence algorithms.

• The coefficient of determination of the SG ensemble neural network on test data to estimate

the shear strength of the SRC walls was 0.99, implying high accuracy.

• Although some hybrid intelligence algorithms have a high coefficient of determination (such as HBA-ANN), they also have significant variance. In addition, another problem with hybrid intelligence algorithms is the large computational cost of training and finding the best model.

• The findings show that the SG model can not only make better predictions with more accuracy but can also reduce prediction variance.

• The SHAP method's results provided clear and extensive insights into how a characteristic influenced the prediction of the SRC wall shear strength. It is noted that the length of the wall is the most important feature, followed by the web thickness, the longitudinal reinforcement ratio, and the height of the wall.

Acknowledgement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

ACI-318 (2014), "Building Code Requirements for Structural Concrete (ACI 318-14): An ACI Standard; Commentary on Building Code Requirements for Structural Concrete (ACI 318R-14)", American Concrete Institute.

Aggarwal, Y., Aggarwal, P., Sihag, P., Pal, M. and Kumar, A. (2019), "Estimation of punching shear capacity of concrete slabs using data mining techniques", *Int. J. Eng.*, **32**(7), 908-914. https://doi.org/10.5829/ije.2019.32.07a.02

Amezquita-Sancheza, J., M. Valtierra-Rodriguez, and H. Adeli (2020), "Machine learning in structural engineering", *Scientia Iranica*, **27**(6), 2645-2656. https://doi.org/10.24200/sci.2020.22091

Barkhordari, M. and M. Es-haghi (2021), "Straightforward prediction for responses of the concrete shear wall buildings subject to ground motions using machine learning algorithms", *Int. J. Eng.*, **34**(7), 1586-1601. https://doi.org/10.5829/ije.2021.34.07a.04

Barkhordari, M.S. and M. Tehranizadeh (2021), "Response estimation of reinforced concrete shear walls using artificial neural network and simulated annealing algorithm", *Structures*, **34**, 1155-1168. https://doi.org/10.1016/j.istruc.2021.08.053.

Barkhordari, M.S., D.C. Feng, and M. Tehranizadeh (2022), "Efficiency of hybrid algorithms for estimating the shear strength of deep reinforced concrete beams", *Periodica Polytechnica Civil Eng.*, **66**(2), 398-410. https://doi.org/10.3311/PPci.19323

Barkhordari, M.S., M. Tehranizadeh, and M.H. Scott (2021a), "Numerical modelling strategy for predicting the response of reinforced concrete walls using Timoshenko theory", *Magazine Concr. Res.*, **73**(19), 1-23. https://doi.org/10.1680/jmacr.19.00542

Barkhordari, M.S., M. Tehranizadeh, and M.H. Scott (2021b), "Numerical modelling strategy for predicting the response of reinforced concrete walls using Timoshenko theory", *Magazine Concr. Res.*, 1-23. https://doi.org/10.1680/jmacr.19.00542.

Brownlee, J. (2018), *Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions*. Machine Learning Mastery.

Chen, X.L., Fu, J.P., Yao, J.L. and Gan, J.F. (2018), "Prediction of shear strength for squat RC walls using a hybrid ANN–PSO model", *Eng. Comput.*, **34**(2), 367-383. https://doi.org/10.1007/s00366-017-0547-5

Committee, A. (2019), "Building code requirements for structural concrete (ACI 318-19) and commentary": American Concrete Institute.

Committee, N.S. (2005), "Seismic design criteria for structures, systems, and components in nuclear facilities", *Am. Soc. Civil Eng.*, Reston, VA. https://doi.org/10.1061/9780784407622.

Dietterich, T.G. (2000), "Ensemble methods in machine learning", *Int. Workshop Multiple Classifier Syst.*, 1-15. https://doi.org/10.1007/3-540-45014-9_1.

Duman, S., Kahraman, H.T., Guvenc, U. and Aras, S. (2021), "Development of a Lévy flight and FDB-based coyote optimization algorithm for global optimization and real-world ACOPF problems", *Soft Comput.*, **25**(8), 6577-6617. https://doi.org/10.1007/s00500-021-05654-z

Feng, D.C., Wang, W.J., Mangalathu, S. and Taciroglu, E. (2021), "Interpretable XGBoost-SHAP machine-learning model for shear strength prediction of squat RC walls", *J. Struct. Eng.*, **147**(11), 04021173. https://doi.org/10.1061/(ASCE)ST.1943-541X.0003115

Gondia, A., M. Ezzeldin, and W. El-Dakhakhni (2020), "Mechanics-guided genetic programming expression for shear-strength prediction of squat reinforced concrete walls with boundary elements", *J. Struct. Eng.*, **146**(11), 04020223. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002734

Guan, X., Burton, H., Shokrabadi, M. and Yi, Z. (2021), "Seismic drift demand estimation for steel moment frame buildings: From mechanics-based to data-driven models", *J. Struct. Eng.*, **147**(6), 04021058. https://doi.org/10.1061/(ASCE)ST.1943-541X.0003004

Gulec, C.K. (2009), *Performance-Based Assessment and Design of Squat Reinforced Concrete Shear Walls*, State University of New York at Buffalo.

Hashim, F.A., Houssein, E.H., Hussain, K., Mabrouk, M.S. and Al-Atabany, W. (2021), "Honey badger algorithm: New metaheuristic algorithm for solving optimization problems", *Math. Comput. Simul.*, **192**, 84-110. https://doi.org/10.1016/j.matcom.2021.08.013

Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J.E. and Weinberger, K.Q. (2017), "Snapshot ensembles: Train 1, get m for free", *arXiv preprint arXiv:1704.00109*.

Jaeger, S. (2020), "The golden ratio of learning and momentum", *arXiv preprint arXiv:2006.04751*.

Kassem, W. (2015), "Shear strength of squat walls: A strut-and-tie model and closed-form design formula", *Eng. Struct.*, **84**, 430-438. https://doi.org/10.1016/j.engstruct.2014.11.027

Kolozvari, K., Kalbasi, K., Orakcal, K., Massone, L. M. and Wallace, J. (2019), "Shear–flexure-interaction models for planar and flanged reinforced concrete walls", *Bull. Earthq. Eng.*, **17**(12), 6391-6417. https://doi.org/10.1007/s10518-019-00658-5

Loshchilov, I. and F. Hutter (2016), "Sgdr: Stochastic gradient descent with warm restarts", *arXiv preprint arXiv:1608.03983*.

Lu, W. and R. Paffenroth (2021), "Neural network ensembles: Theory, training, and the importance of explicit diversity", *J. Machine Learning Res.* https://doi.org/10.48550/arXiv.2109.14117.

Lundberg, S.M. and S.I. Lee (2017), "A unified approach to interpreting model predictions", *Adv. Neural Inform. Proc. Syst.*, **30**.

Ma, J. and B. Li (2018), "Experimental and analytical studies on h-shaped reinforced concrete squat walls", *ACI Struct. J.*, **115**(2). https://doi.org/10.14359/51701144

Ma, J., C.L. Ning, and B. Li (2020), "Peak shear strength of flanged reinforced concrete squat walls", *J. Struct. Eng.*, **146**(4), 04020037. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002575

Massone, L.M. and F. Melo (2018), "General solution for shear strength estimate of RC elements based on panel response", *Eng. Struct.*, **172**, 239-252. https://doi.org/10.1016/j.engstruct.2018.06.038

Massone, L.M. and M.A. Ulloa (2014), "Shear response estimate for squat reinforced concrete walls via a single panel model", *Earthq. Struct.*, **7**(5), 647-665. https://doi.org/10.12989/eas.2014.7.5.647

Massone, L.M., C.N. López, and K. Kolozvari (2021), "Formulation of an efficient shear-flexure interaction model for planar reinforced concrete walls", *Eng. Struct.*, **243**, 112680. https://doi.org/10.1016/j.engstruct.2021.112680

Mohammed, S.J., H.A. Abdel-khalek, and S.M. Hafez (2021), "Predicting performance measurement of residential buildings using an artificial neural network", *Civil Eng. J.*, **7**(3), 461-476. https://doi.org/10.28991/cej-2021-03091666

Moradi, M.J., Roshani, M.M., Shabani, A. and Kioumarsi, M. (2020), "Prediction of the load-bearing behavior of SPSW with rectangular opening by RBF network", *Appl. Sci.*, **10**(3), 1185.

https://doi.org/10.3390/app10031185
Naimi, A.I. and L.B. Balzer (2018), "Stacked generalization: an introduction to super learning", *Eur. J. Epidemiol.*, **33**(5), 459-464. https://doi.org/10.1007/s10654-018-0390-z
Nguyen, D.D., Tran, V.L., Ha, D.H., Nguyen, V.Q. and Lee, T.H. (2021), "A machine learning-based formulation for predicting shear capacity of squat flanged RC walls", *Structures*, **29**, 1734-1747. https://doi.org/10.1016/j.istruc.2020.12.054.
Ning, C.L. and B. Li (2017), "Probabilistic development of shear strength model for reinforced concrete squat walls", *Earthq. Eng. Struct. Dyn.*, **46**(6), 877-897. https://doi.org/10.1002/eqe.2834
Pizarro, P.N. and L.M. Massone (2021), "Structural design of reinforced concrete buildings based on deep neural networks", *Eng. Struct.*, **241**, 112377. https://doi.org/10.1016/j.engstruct.2021.112377
Pizarro, P.N., *et al*. (2021), "Use of convolutional networks in the conceptual structural design of shear wall buildings layout", *Eng. Struct.*, **239**, 112311. https://doi.org/10.1016/j.engstruct.2021.112311
Rojas, F., J. Anderson, and L. Massone (2016), "A nonlinear quadrilateral layered membrane element with drilling degrees of freedom for the modeling of reinforced concrete walls", *Eng. Struct.*, **124**, 521-538. https://doi.org/10.1016/j.engstruct.2016.06.024
Siam, A.S., M. Ezzeldin, and W. El-Dakhakhni (2019), "Reliability of displacement capacity prediction models for reinforced concrete block shear walls", *Structures*, **20**, 385-398. https://doi.org/10.1016/j.istruc.2019.05.002.
Sutton, A.K. and M.J. Krashes (2020), "Integrating hunger with rival motivations", *Trends Endocrinol. Metabolism*, **31**(7), 495-507. https://doi.org/10.1016/j.tem.2020.04.006
Yang, Y., Chen, H., Heidari, A.A. and Gandomi, A.H. (2021), "Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts", *Expert Syst. Appl.*, **177**, 114864. https://doi.org/10.1016/j.eswa.2021.114864

TK