Front. Earth Sci. 2014, 8(3): 439–456
DOI 10.1007/s11707-014-0416-0

RESEARCH ARTICLE

Artificial Neural Network (ANN) and Regression Tree (CART) applications for the indirect estimation of unsaturated soil shear strength parameters

D.P. KANUNGO (✉), Shaifaly SHARMA, Anindya PAIN

Geotechnical Engineering Group, CSIR – Central Building Research Institute (CBRI), Roorkee 247667, India

Abstract The shear strength parameters of soil (cohesion and angle of internal friction) are quite essential in solving many civil engineering problems. In order to determine these parameters, laboratory tests are used. The main objective of this work is to evaluate the potential of Artificial Neural Network (ANN) and Regression Tree (CART) techniques for the indirect estimation of these parameters. Four different models, considering different combinations of 6 inputs, such as gravel %, sand %, silt %, clay %, dry density, and plasticity index, were investigated to evaluate the degree of their effects on the prediction of shear parameters. A performance evaluation was carried out using Correlation Coefficient and Root Mean Squared Error measures. It was observed that for the prediction of friction angle, the performance of both the techniques is about the same. However, for the prediction of cohesion, the ANN technique performs better than the CART technique. It was further observed that the model considering all of the 6 input soil parameters is the most appropriate model for the prediction of shear parameters. Also, connection weight and bias analyses of the best neural network (i.e., 6/2/2) were attempted using Connection Weight, Garson, and proposed Weight-bias approaches to characterize the influence of input variables on shear strength parameters. It was observed that the Connection Weight Approach provides the best overall methodology for accurately quantifying variable importance, and should be favored over the other approaches examined in this study.

Keywords cohesion, friction angle, Artificial Neural Network, Regression Tree, Connection Weight, Weight-bias Approach

Received May 15, 2013; accepted September 25, 2013

E-mail: debi.kanungo@gmail.com

1 Introduction

Shear strength parameters are the most important engineering properties of soil. Determinations of the shear strength properties of soil are of foremost importance in geotechnical investigation. Shear strength properties are further required for the determination of bearing capacity for foundation analysis. Understanding of the shear strength of the soil is critical for the design of road embankments, retaining walls, copper dams, etc. Shear strength is the property of the soil that enables the soil to keep its equilibrium when its surface is not level, or for that matter, under any loading situation that produces shearing stresses. Several procedures have been proposed in the literature to determine the shear strength parameters of unsaturated soil. These shear parameters can be determined either in the field, or in the laboratory, or both. The tests employed in the laboratory may include an unconfined compression test, a triaxial test, a laboratory vane shear test, a direct shear box test, and a direct simple shear test. In-situ tests are normally conducted to confirm the validity of the laboratory tests, and for design purposes. The in-situ tests include a field vane shear test, a standard penetration test, a cone penetration test, and piezocone and pressure meter readings (Jain et al., 2010). Different authors have worked considerably in the field of the prediction of shear strength parameters of unsaturated soils, using mathematical relationships such as elliptical and hyperbolic functions (Abra-mento et al., 1989; Escario and Juca, 1989; Lu, 1992; Shen and Yu, 1996; Xu, 1997).

Recently, soft computing techniques, such as Artificial Neural Network (ANN), Fuzzy System, Genetic Expression Programming, and others, have been used frequently to solve a wide variety of problems in geosciences and geotechnical engineering. These include the estimation of the probability of liquefaction (Youd and Gilstrap, 1999; Juang et al., 2000; Goh et al., 1994; Hanna et al., 2007), strength parameter modeling of different soils (Agrawal et al., 1994; Attoh-Okine, 2004; Goktepe et al., 2008; Kaya, 2009), hydraulic conductivity (Akbulut, 2005), predicting the angle of internal friction in soils using a hybrid genetic fuzzy system (Goktepe and Sezer, 2010), identification of compaction characteristics (Najjar and Basheer, 1996; Alavi et al., 2009), and the problem of slope stability (Ferentinou and Sakellariou, 2007; Cho, 2009). Kayadelen et al. (2009) studied the prediction of the angle of internal friction of soils using soft computing techniques.

Among all of these soft computing techniques, ANN has a wide range of applicability in cases of landslide susceptibility zonation (LSZ), prediction of debris flow, landslide movement monitoring, prediction of strength and deformation properties of rock, etc. Lu and Rosenbaum (2003) built models using artificial neural networks and grey systems for the prediction of slope stability. Neaupane and Achet (2004) used a back propagation neural network for monitoring a landslide in the higher Himalaya. ANN models (Arora et al., 2004; Gómez and Kavzoglu, 2005; Yesilnacar and Topal, 2005; Kanungo et al., 2006; Nefeslioglu et al., 2008) have been implemented for LSZ studies. Sonmez et al. (2006) used ANN for the determination of the deformation modulus of intact rock specimens. Das and Basudhar (2008) used artificial neural networks for the prediction of the residual friction angle of clays. Tiryaki (2008) used multivariate statistics, artificial neural networks, and regression trees to predict intact rock strength and deformation properties for mechanical excavations. Baykasoğlu et al. (2008) used genetic programming for the prediction of compressive and tensile strength of Gaziantep limestone. Maji and Sitharam (2008) used the ANN model for the prediction of elastic modulus of jointed rock mass. Çanakcı et al. (2009) predicted the compressive and tensile strength of Gaziantep basalts using neural networks and gene expression programming. Rafiai and Jafari (2011) developed a new set of rock failure criteria using the ANN approach.

In the present study, indirect estimation of shear strength parameters, such as friction angle (ϕ) and cohesion (c) of soil, under unsaturated conditions was done using ANN and Regression Tree (CART) approaches.

2 Data used for the study

Laboratory testing of soils from surface and subsurface areas is a fundamental element of the geotechnical investigation of a site before the design and practice of any civil engineering construction. These laboratory tests may vary from simple soil classification tests to complex strength and deformation tests. In this research, soil samples have been obtained from both surface and subsurface soil resources from 6 different states of India, namely, Himachal Pradesh, Uttar Pradesh, Bihar, Jharkhand, Orissa, and Andhra Pradesh. A series of laboratory tests were conducted to determine the engineering properties of these soil samples. All tests were performed according to Indian Standard 2720 (IS 2720: Parts IV, V and XIII). The tests performed included the grain size distribution, Atterberg limits, dry density, and direct shear test. Soil parameters including gravel % (GP), sand % (SP), silt % (STP), clay % (CP), dry density (DD), plasticity index (PI), c, and ϕ were measured on 115 samples. These data were taken from the unpublished reports of the Institute and utilized for this research work. The basic idea in this research is the evaluation of the capability of ANN and CART techniques to make indirect estimates of the shear strength parameters of soils under unsaturated conditions.

Statistical descriptions of the soil parameters of all 115 soil samples are given in Table 1. It can be seen from this table that the median and average values of each parameter are similar. This shows that the statistical distributions of each parameter for all of the soil samples are nearly normal. As shown in Table 1, the measured values of c range from 0.0 to 0.7 kg/cm^2, with an average value of 0.14 kg/cm^2, and a median value of 0.08 kg/cm^2. The ϕ values range from a lower value of 9 degrees to a higher value of 40.5 degrees. The mean and median values of ϕ are 25.4 and 26 degrees respectively. Amongst all of the 6 soil parameters, DD has the least spatial variation; and SP has the largest spatial variation.

A correlation matrix was produced by applying a bivariate correlation technique to the original data set, in order to analyze the strength of the linear relationships

Table 1 Basic descriptive statistics for different parameters of soil samples

Statistics	GP	SP	STP	CP	DD /(gm·cm^{-3})	PI	c/(kg·cm^{-2})	ϕ /degree
Minimum	0	5	0	0	1.24	0	0	9
Maximum	59	97	82	48	2.04	45.22	0.7	40.5
Average	4.456522	59.56957	25.84783	10.24783	1.838817	10.23383	0.139087	25.40261
Median	0	61	23	10	1.9	11	0.08	26
Standard deviation	8.730257	23.65652	17.82655	10.6948	0.166569	9.609107	0.164424	7.200542
Number of samples	115	115	115	115	115	115	115	115

between the variables considered in this case. The correlation coefficients (R) values between c and ϕ, the dependent variables, and the other 6 soil properties (GP, SP, STP, CP, DD, and PI), the independent variables, were investigated. The correlation coefficient matrix for all soil parameters, along with their significance levels, is given in Table 2. The results in Table 2 show that SP, STP, and CP have influences on c. STP and CP are significant factors contributing to c, with positive R values of 0.589 and 0.512, respectively. SP also appears to influence c, but with a negative R of –0.672. As per the significance levels of different soil parameters (refer to Table 2), GP is the only insignificant parameter with respect to c. Also SP, STP, CP and PI have influences on ϕ. SP is the most significant soil parameter contributing to ϕ of soil, with a positive R value of 0.661. STP, CP, and PI are related to ϕ with negative R values of –0.651, –0.603 and –0.631, respectively. It is also observed in Table 2 that DD is the only insignificant parameter with respect to ϕ.

3 Methodology

3.1 Artificial Neural Network (ANN)

ANN was originally devised as models of the human brain. It was hoped that ANN could reveal useful information about the structure of the brain and the processes that occur within the brain. The use of ANN as a tool for exploring brain function has become increasingly widespread within cognitive psychology and neurophysiology in recent years. However, this study is primarily interested in ANN as a tool for solving mathematical problems. In particular, ANN is used to identify unknown multivariate functions from samples of data. Aspects concerning the biological validity of ANN architecture, or of a training algorithm, are only occasionally considered.

Neural networks are non-linear statistical data modeling tools. They can be used to model complex relationships between inputs and outputs, or to find patterns in data. A set of connected units forms a neural network with the capability of nonlinear input/output approximations. If the units are grouped into layers, and all units of a layer are connected with the units of the subsequent layer, a feed forward network is developed (i.e., a static process).

Each layer in a network contains a sufficient number of neurons depending on its specific application. The neurons in a layer are connected to the neurons in the next successive layer; and each connection carries a weight (Atkinson and Tatnall, 1997). The input layer receives the data from different sources. Hence, the number of neurons in the input layer depends on the number of input data sources. The hidden and output layers actively process the data. The number of hidden layers and their neurons are often determined by trial and error (Gong, 1996). The number of neurons in output layers is fixed by the application. Each hidden neuron responds to the weighted inputs it receives from the connected neurons from the preceding input layer (Lee et al., 2004). Once the combined effect on each hidden neuron is determined,

Table 2 Correlation matrix and significance levels for the considered data set

Correlation matrix								
Parameters	GP	SP	STP	CP	DD	PI	c	ϕ
GP	1	–0.102	–0.159	–0.327	–0.252	–0.182	–0.012	0.274
SP		1	–0.890	–0.648	0.223	–0.679	–0.672	0.661
STP			1	0.433	–0.275	0.531	0.589	–0.651
CP				1	0.183	0.779	0.512	–0.603
DD					1	0.006	–0.298	0.019
PI						1	0.470	–0.631
c							1	–0.678
ϕ								1
Significance Levels (Correlation is significant at the 0.01 level)								
GP	.	0.277	0.090	0.000	0.007	0.052	0.900	0.003
SP		.	0.000	0.000	0.016	0.000	0.000	0.000
STP			.	0.000	0.003	0.000	0.000	0.000
CP				.	0.050	0.000	0.000	0.000
DD					.	0.948	0.001	0.838
PI						.	0.000	0.000
c							.	0.000
ϕ								.

the activation at this neuron is determined via a transfer function (Yesilnacar and Topal, 2005). Any differentiable nonlinear function can be used as a transfer function; but a sigmoid function is generally used, though there are many other functions (Schalkoff, 1997). The sigmoid function constrains the outputs of a network to a range between 0 and 1.

This type of feed forward network propagates the input vector X_m^n from the input-layer, through one or more hidden layers, to the output-layer, only in one direction. A dynamic process can be carried out by adding external feedback of delayed outputs; and this is referred to as external recurrent networks. The relationship between the input vector (X_m^n) and output vector (X_j^{n+1}) of this element can be described as follows:

$$X_j^{n+1} = F\left(\sum_i W_{jm}^n X_m^n\right), \quad (1)$$

where $F(x)$ is the log sigmoid function $F(x) = \frac{1}{1+\mathrm{e}^{-x}}$ or other nonlinear transfer function, e.g., tan-sigmoid function, and X_j^{n+1} is output of unit j in the n^{th} layer, and W_{jm}^n is a weight from unit m in n^{th} layer to unit j in $(m+1)^{\text{th}}$ layer, as shown in the Fig. 1. Network training is a process by which the connection weights and biases of the ANN are adapted through a continuous process of simulation by the environment in which the network is embedded. The primary goal of training is to minimize an error function, by searching for a set of connection strengths and biases that cause the ANN to produce outputs that are equal or close to targets. The network connection strengths are adjusted in the training process, which can be executed through a number of learning algorithms based on back propagation learning (Ripley, 1996; Haykin, 1998; Zhou, 1999; Lee et al., 2004; Gómez and Kavzoglu, 2005; Yesilnacar and Topal, 2005). The most widely used back propagation algorithms are gradient descent and gradient descent with momentum. These are often too slow for the solution of practical problems. The faster algorithms use standard numerical optimizers such as conjugate gradient, quasi-Newton, and Levenberg-Marquardt approaches. In this study, the Levenberg-Marquardt algorithm (implemented as TRAINLM in MATLAB software) was used for training the neural network. The details of this algorithm can be found in Hagan and Menhaj (1994) and Hagan et al. (1996). In other words, training aims at estimating the parameters' weights and biases by minimizing an error function of the output values. The total sum squared error E is averaged over all patterns in the training set, in which M_p is the target output (predicted) for the p^{th} pattern, and O_p is the actual output (measured).

$$E = \sum\nolimits_{p=1}^{p} (M_p - O_p)^2. \quad (2)$$

The process of back propagating the error is repeated iteratively until the error is minimized to an acceptable value and the adjusted weights are obtained, which are then used to determine the network outputs. The performance of the network depends on the accuracy obtained over a set of test data. If the network is trained to an acceptable level of accuracy, then the adjusted weights are used to determine the outputs of the test data set.

The R and Root Mean Square Error ($RMSE$) were considered for evaluating the ability of trained and tested networks in predicting shear parameters. The coefficient of determination is a measure of the accuracy of prediction of the trained network models. Higher R values indicate better prediction. In addition, the mean relative percentage error was also used to measure the accuracy of prediction. For each network, the training and testing errors/accuracies for the individual output shear strength parameters (i.e., either c or ϕ) were calculated using Eq. (3) and Eq. (5). Accuracies for multiple output shear strength parameters (i.e., both c and ϕ) were calculated using Eq. (4) and Eq. (6).

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{j=1}^{n} (M_j - O_j)^2}, \quad (3)$$

$$RMSE_{combined} = \sqrt{\frac{1}{2n}\left(\sum_{c=1}^{n} (M_c - O_c)^2 + \sum_{\phi=1}^{n} (M_\phi - O_\phi)^2\right)}, \quad (4)$$

$$R = \sqrt{1 - \left(\frac{\sum_j (M_j - O_j)^2}{\sum_j (O_j)^2}\right)}, \quad (5)$$

$$R_{combined} = \frac{\sum_j \sum_j \left(M_{jj} - \overline{M}\right)\left(O_{jj} - \overline{O}\right)}{\sqrt{\sum_j \sum_j \left(M_{jj} - \overline{M}\right)^2 \sum_j \sum_j \left(O_{jj} - \overline{O}\right)^2}}, \quad (6)$$

where j = number of data patterns in the independent data set; M = actual data set; O = predicted data set; $\overline{M}$ = mean

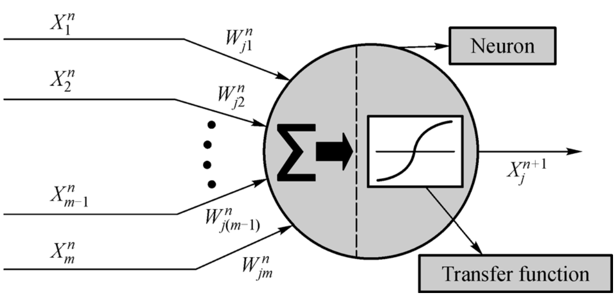


Fig. 1 A schematic description of the relationship between the input and output vectors of one neuron.

of actual data set; $\overline{O}$ = mean of predicted data set.

In this study, a feed forward back-propagation (FFBP) multi-layer ANN, with one input layer, one hidden layer, and one output layer was used to predict the shear strength parameters of the soil. The input layer contains a maximum of 6 neurons, each representing a parameter (GP, SP, STP, CP, DD, and PI) that contributes to the shear strength of the soil. The output layer contains two different neurons, representing c, and ϕ of soil under an unsaturated condition for a given set of input values. It is important to ensure that the training samples given to the network cover the range of possibilities and are representative of the frequency of occurrences of the variables. The Levenberg-Marquardt algorithm (implemented as TRAINLM in MATLAB software) was used for training the neural network; and the log-sigmoid function was applied for neurons between the input and hidden layers, and also for neurons between the hidden and output layers. By varying the number of neurons in the hidden layer, the neural networks are run several times to identify the most appropriate neural network architecture based on training and testing accuracies. The neural network processing that was implemented, using coding developed in MATLAB Software and the computation procedure of the ANN process, is given in Fig. 2. In this study, 95 of the simulated samples in a random order formed the training data; while the remaining 20 simulated samples in a random order were used for both validation and testing.

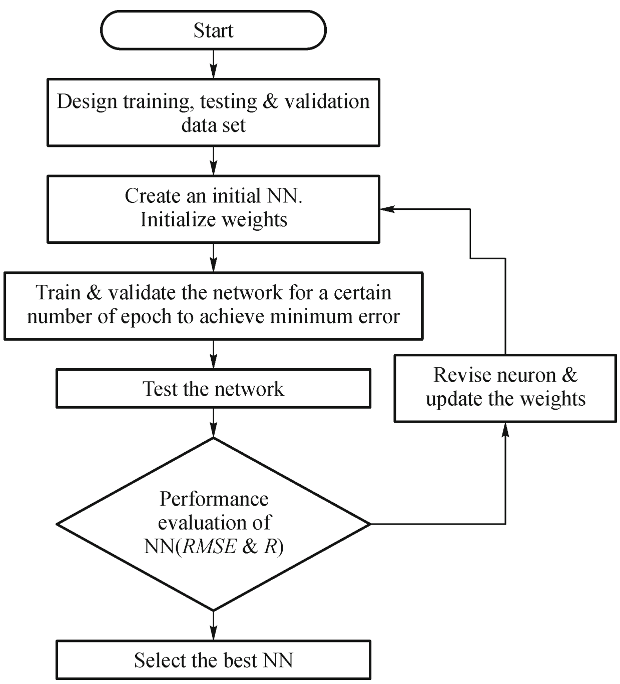


Fig. 2 Neural network processing as implemented using coding developed in MATLAB software.

3.2 Regression tree

In recent times, there has been increasing interest in the use of classification and CART analysis. CART analysis is a tree-building technique, which is different from traditional data analysis methods. Breiman et al. (1984) developed CART, which is a sophisticated program for fitting trees to data. Depending on the value of the predictor variables, a decision tree partitions the data set into regions, so that the response variable is roughly constant in that region. The attractive feature of the CART methodology is a sequence of hierarchical questions that the algorithm asks. This method is relatively simple to understand and interpret. The questions can be answered as 'yes' or 'no'; and depending on the answer, it either proceeds to another question, or arrives at a fitted response value.

Details about the mathematical treatment of regression tree analysis are discussed in Breiman et al. (1984). The unique starting point of a classification tree is called a root node and consists of the entire learning data set at the top of the tree. A node is a subset of the set of variables, and it can be a terminal or non-terminal node. A non-terminal node is a node that splits into two daughter nodes (binary split). Such a binary split is determined by a condition on a single variable, where the condition is satisfied or not satisfied by the observed value of that variable. All observations in the whole dataset that have reached a particular node and satisfy the condition for that variable drop down to one of the daughter nodes. The remaining observations at that node that do not satisfy the condition drop down to the other daughter node. A node that does not split is called a terminal node. In each terminal node, t, the fitted value, or the predicted response value $y(t)$ is constant. Suppose the data points (x_1,y_1), (x_2,y_2),…,(x_n,y_n) are all of the samples belonging to one terminal node, say node l. Then the simple model for l is given in Eq. (7), the sample mean of the dependent variable in that cell.

$$\overline{y}(l) = \frac{1}{N(t)} \sum_{x_n \in l} y_n. \tag{7}$$

Since the predictor, $\overline{y}(l)$, is constant over each terminal node, the regression tree can be thought of as a histogram estimate of the regression surface. For every node l, $\sum_{x_n \in l} (y_n - \overline{y}(l))^2$ is the within node sum of squares, that is the total squared deviations of y_n in l from their average. Summing over the entire terminal node ($l \in T$) gives the total within node sum of squares; and dividing by N gives the average (Eq. 8).

$$E = \frac{1}{N} \sum_{l \in T} \sum_{x_n \in l} (y_n - \overline{y}(l))^2. \tag{8}$$

Therefore, a particular regression tree is formed by iteratively splitting nodes to maximize the decrease in E

(Breiman et al., 1984). There are two methods to grow a decision tree: (a) stop recursive partitioning when the largest decrease in E would be less than some user defined threshold value say δ, and (b) each terminal node should not contain more or equal to P sample data points, where P is a user defined integer number (splitmin). In this study, the second method is used to grow the decision trees.

4 Implementation of different techniques for the prediction of shear strength parameters of soil

The estimation of unsaturated shear strength parameters of soils was performed using both ANN and CART techniques. Both of the techniques used different combinations of GP, SP, STP, CP, DD, and PI as the input parameters, and the unsaturated c, and ϕ of soil as the target outputs. Though GP is insignificant with respect to c, and DD is insignificant with respect to ϕ as per the correlation matrix of the original data set (refer to Table 2), neither of these two parameters (GP and DD) could be excluded from the ANN models. This is because of the fact that this study attempts to develop a single ANN model for estimating both c and ϕ, considering 6 input parameters (GP, SP, STP, CP, DD, and PI) in combination. Secondly, in order to cross verify the inference of the correlation matrix with the capability of the ANN models through the connection weight and bias analysis for input parameter importance for estimating c and ϕ, all 6 parameters in combination are considered for the ANN models.

With different combinations out of these 6 input parameters, four different models were tried to establish the efficacy of different input combinations for predicting the shear strength parameters of soil. These models are as follows: (a) Model I: GP, SP, STP, and CP; (b) Model II: GP, SP, STP, CP, and DD; (c) Model III: GP, SP, STP, CP, and PI; and (d) Model IV: GP, SP, STP, CP, DD, and PI. These techniques were implemented with MATLAB software to predict the unsaturated shear strength parameters of the soil samples.

4.1 Prediction using ANN technique

A feed forward back-propagation multi-layer ANN with one input layer, one hidden layer, and one output layer was considered in the present study. The number of neurons in the input layer equals the number of input parameters considered in case of each model (i.e., 4 inputs in the 1st case, 5 inputs in each of the 2nd and 3rd cases, and all 6 inputs in the 4th case). The data at each neuron of the input layer correspond to the normalized value of each input. The input parameters GP, SP, STP, and CP are normalized with respect to 100 as these represent the percent value. The input parameters DD and PI are normalized with respect to their maximum value of occurrence (i.e., 2.04 in the case of DD and 45.22 in the case of PI). The output layer consists of two different neurons, representing the experimentally determined value of unsaturated c, and ϕ of soil as the target outputs. The values of c and ϕ are also normalized with respect to their maximum values of occurrence (i.e., 0.7 kg/cm^2 and 40.5 degrees respectively). The number of neurons in the hidden layers is varied by running the networks several times to achieve the desired training and testing data accuracies.

One set each of training and testing data were randomly generated from the available data set using the stratified random sampling technique. The training dataset consists of data pertaining to 95 soil samples and the testing dataset consists of data pertaining to 20 soil samples out of a total of 115 samples. All of the data points in the datasets were mutually exclusive (Foody and Arora, 1997). The training dataset was used to train different network architectures, while the testing dataset was used simultaneously with the training dataset to control the overtraining of the network. The testing dataset was also used to evaluate the accuracy of the networks. The well-known back propagation Levenberg-Marquardt algorithm was used to train the neural networks. 40 neural network architectures were created by varying the number of neurons in the hidden layer, and then trained and tested. The training process was initiated with arbitrary initial connection weights, which were constantly updated until an acceptable level of accuracy was reached. The final adjusted weights of the trained network were used to derive outputs of the testing data to evaluate the performance of the network.

4.2 Prediction using the regression tree technique

A number of regression trees were calculated, respectively, for c and ϕ, using MATLAB and employing different combinations of input parameters (GP, SP, STP, CP, DD, and PI) as predictors. The original experimentally determined values of all of the parameters were used for the analysis. Determining the right size tree is not that straight forward. 50 different trees with varying splitmins were built for each individual combination of predictors. The performances of the trees developed in this study were assessed using two different statistical performance evaluation criteria. The statistical measures used were the R and $RMSE$, the same as those adopted in the ANN technique.

5 Results and discussion

5.1 Results of ANN technique

5.1.1 Performance evaluation of neural networks

The performance of the networks was evaluated by determining both the training and testing data accuracies in terms of R and $RMSE$ values (Freund, 1992). The combined accuracies in terms of R and $RMSE$ values for both of the shear parameters (c and ϕ) were calculated with MATLAB software for each of the four models, using different combinations of input parameters (as mentioned in Section 4). The training and testing accuracies for all of the 4 models for some selected networks out of a total 40 neural networks with neurons in the hidden layer varying from 1 to 40 are given in Table 3.

A variation in both training and testing data accuracies across different models can be seen as the neural network architectures change. The training and testing accuracies in terms of R values observed across different neural networks for different models were as follows: (a) Model I: –0.05 to 0.99 for training and 0.12 to 0.82 for testing, (b) Model II: 0.13 to 0.99 for training and –0.09 to 0.91 for testing, (c) Model III: 0.76 to 0.99 for training, and 0.48 to 0.81 for testing and (d) Model IV: 0.13 to 0.99 for training and 0.27 to 0.88 for testing. This implies that there an optimal network architecture for a given dataset exists.

In the present case, the network architectures 4/6/2, 5/16/2, 5/18/2, and 6/2/2 were observed to be the most appropriate ones for Models I, II, III, and IV, respectively, with the training and testing data accuracies, as highlighted

Table 3 Combined accuracies in terms of R and $RMSE$'s for both the shear parameters (c and ϕ) for all 4 models for some selected neural networks

ANN architecture	*R*		*RMSE*	
	Training	Testing	Training	Testing
Model I (GP, SP, STP and CP as inputs)				
4/1/2	0.904788	0.709213	0.128307	0.202459
4/6/2	**0.950486**	**0.820566**	**0.094369**	**0.094369**
4/7/2	0.938912	0.73586	0.103917	0.203841
4/20/2	0.983056	0.723992	0.056938	0.229616
4/24/2	0.98938	0.709713	0.045527	0.226808
4/32/2	0.984646	0.724235	0.052902	0.215475
4/40/2	0.991034	0.58296	0.040481	0.27849
Model II (GP, SP, STP, CP and DD as inputs)				
5/1/2	0.894149	0.779981	0.134873	0.177807
5/3/2	0.894153	0.894153	0.134871	0.177657
5/6/2	0.933653	0.85501	0.108095	0.15316
5/16/2	**0.978625**	**0.915253**	**0.062174**	**0.121163**
5/19/2	0.993896	0.861629	0.033275	0.16182
5/32/2	0.995472	0.886153	0.02864	0.135251
5/40/2	0.890366	0.671108	0.282153	0.37535
Model III (GP, SP, STP, CP and PI as inputs)				
5/1/2	0.887056	0.735596	0.139076	0.192906
5/7/2	0.967071	0.773272	0.076868	0.201282
5/13/2	0.98884	0.800253	0.047095	0.19212
5/18/2	**0.988724**	**0.810958**	**0.045282**	**0.180246**
5/21/2	0.986577	0.804543	0.04952	0.19896
5/29/2	0.993524	0.779988	0.034499	0.20616
5/40/2	0.996953	0.639055	0.023531	0.2481
Model IV (GP, SP, STP, CP, DD and PI as inputs)				
6/1/2	0.900495	0.801914	0.130986	0.170389
6/2/2	**0.939658**	**0.879199**	**0.103081**	**0.136468**
6/5/2	0.974072	0.87192	0.069304	0.147565
6/7/2	0.98054	0.877744	0.059174	0.156894
6/15/2	0.975545	0.861556	0.066319	0.16628
6/22/2	0.996851	0.848852	0.023911	0.169109
6/40/2	0.999205	0.761987	0.012016	0.216977

in Table 3 for the prediction of shear parameters of soil under unsaturated conditions. For the above most appropriate network architectures, the training R values across all of the models varied from 0.94 to 0.99, whereas the testing R values across these four models varied from 0.81 to 0.91. It can be inferred from these results that there exists a good correlation between training and testing results along with the good prediction capability achieved by the neural networks with the present data set.

If the results of all of the four models are compared for the prediction of c and ϕ, it can be observed from Table 3 that Model II, considering 5 input parameters such as GP, SP, STP, CP, and DD, has given the highest accuracies (5/16/2 neural network). The training and testing R values as obtained for Model II for the 5/16/2 neural network are illustrated in Fig. 3. However, in the case of Model IV, considering all of the 6 input parameters, the difference between training and testing R values (6/2/2 neural network) is the least when compared to the other three models (Kanungo et al., 2006). The training and testing R values obtained for Model IV for the 6/2/2 neural network are illustrated in Fig. 4. If several networks fit the training set equally well, then the simplest network (i.e., the network that has the smallest number of weights and biases) will, on average, give the best generalization performance (Sietsma and Dow, 1991). Hence, it can be stated from these observations that Model IV is the most appropriate model for the prediction of both the shear parameters of soil under unsaturated conditions.

In order to assess the individual accuracy of each of the shear parameters (c and ϕ) separately instead of combined (as discussed above) for all of the four models, the predicted and the target output values of both c and ϕ were analyzed in each case for all the neural networks. The training and testing accuracies in terms of R and $RMSE$ values for all of the 4 models for some selected networks are given in Table 4.

A variation in both training and testing data accuracies, for both c and ϕ across different models, can be seen as the neural network architectures change. The training and testing accuracies in terms of R values observed across different neural networks for different models in predicting c were as follows: (a) Model I: –0.60 to 0.98 for training and –0.60 to 0.75 for testing, (b) Model II: –0.31 to 0.99 for training and –0.34 to 0.92 for testing, (c) Model III: –0.09 to 0.99 for training and –0.32 to 0.82 for testing and (d) Model IV: –0.37 to 0.99 for training and –0.48 to 0.91 for testing (Fig. 5). Similarly, for ϕ, the training and testing accuracies in terms of R values observed across different neural networks for different models were as follows: (a) Model I: 0.0 to 0.97 for training and –0.41 to 0.80 for testing, (b) Model II: 0.08 to 0.99 for training and –0.42 to 0.87 for testing, (c) Model III: –0.32 to 0.99 for training and –0.32 to 0.89 for testing and (d) Model IV: –0.31 to 0.99 for training and –0.60 to 0.94 for testing (Fig. 6).

For the prediction of c, the network architectures 4/6/2, 5/16/2, 5/21/2, and 6/5/2 were observed to be the most appropriate ones for Models I, II, III, and IV, respectively, with the training and testing data accuracies as highlighted in Table 4. For these most appropriate network architectures, the training R values across all of the models varied from 0.90 to 0.97; whereas, the testing R values across these four models varied from 0.75 to 0.91. It can also be observed, from the results given in Table 4, that for the

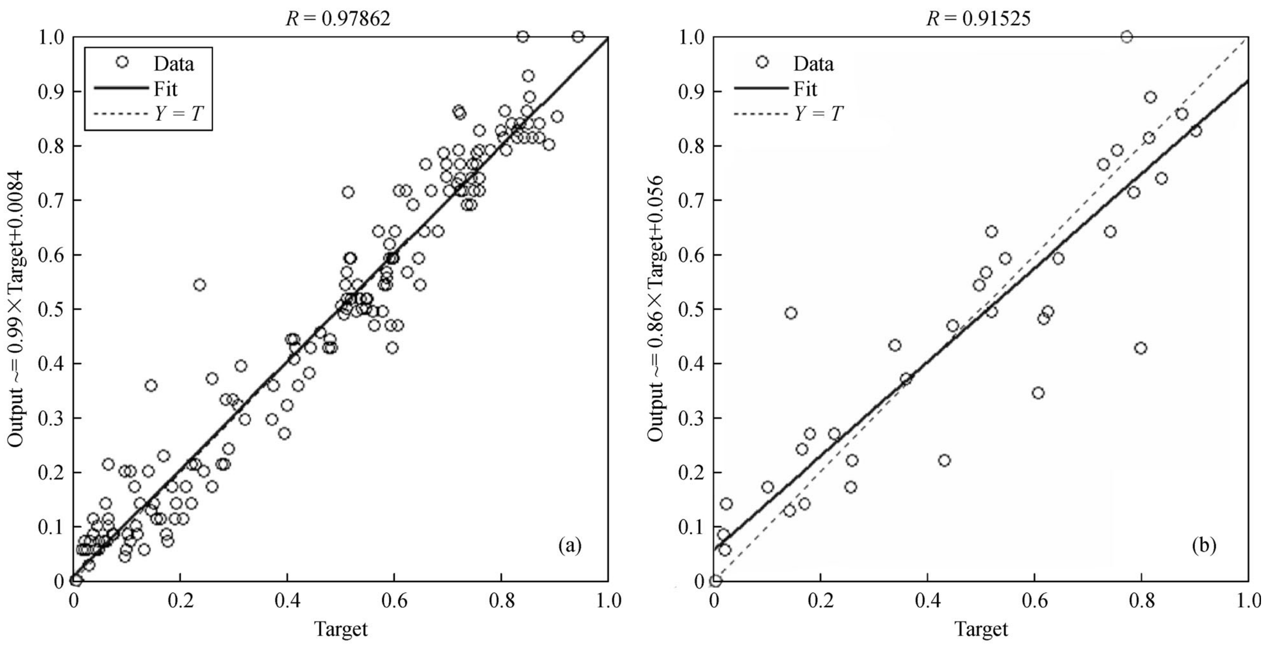


Fig. 3 Correlation coefficients as obtained for Model II for the 5/16/2 neural network: (a) training and (b) testing.

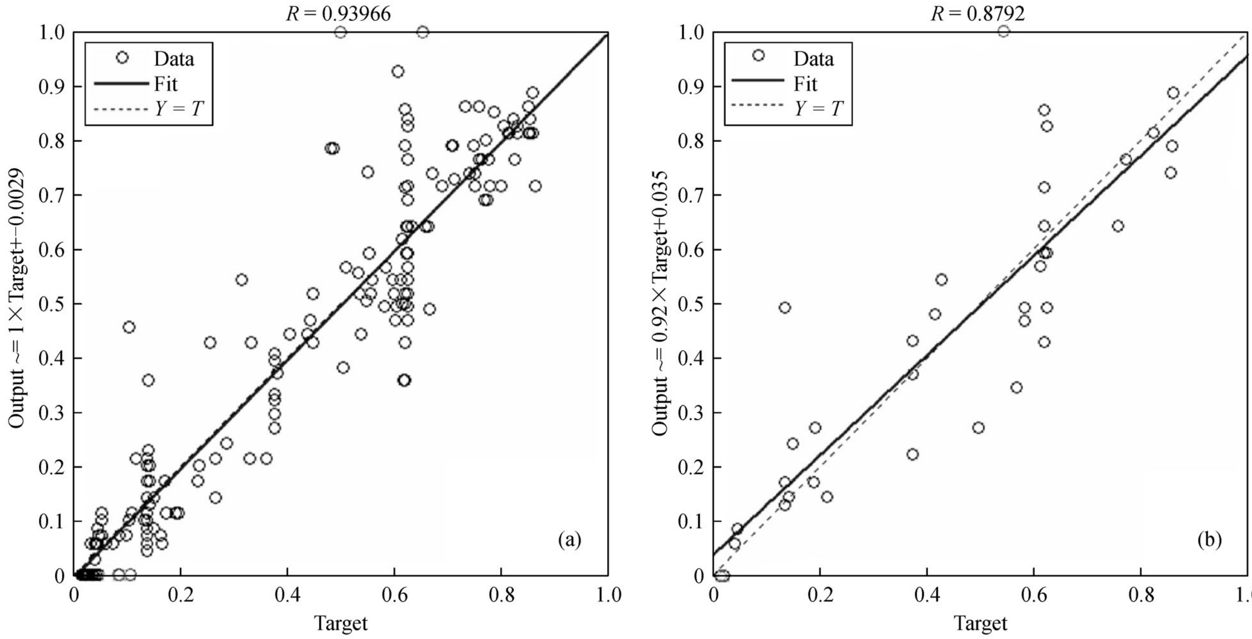


Fig. 4 Correlation coefficients as obtained for Model IV for the 6/2/2 neural network: (a) training and (b) testing.

prediction of ϕ of soil the network architectures 4/6/2, 5/16/2, 5/27/2, and 6/8/2 were the most appropriate ones for Models I, II, III, and IV, respectively. For these most appropriate network architectures, the training R values across all four models varied from 0.84 to 0.98; whereas, the testing R values across these four models varied from 0.80 to 0.92. It can be stated from these results that there exists a good correlation between training and testing results even though the individual prediction of both of the shear parameters of the soil is considered.

If the results of all of the four models are compared for the prediction of c and ϕ individually, Table 4 shows that Model IV, which considers all 6 input parameters, is the most appropriate model with the highest accuracies for the prediction of both of the shear parameters of soil under unsaturated conditions.

In summary, the overall results indicate that, whether the training and testing data accuracies for the prediction of both shear parameters for soil (c and ϕ) are analyzed combined or individually, it hardly affects the final result in terms of the most appropriate model for prediction, Model IV, which considers all 6 input parameters. But it is always easy and convenient to adopt the combined way of accuracy analysis for prediction of both c and ϕ with multiple neurons in the output layer (in this case 2 neurons for c and ϕ) to select the best single neural network architecture.

5.1.2 Connection weight analysis

The ANN connection weights (Table 5) obtained for the most appropriate neural network for the best model (6/2/2 neural network for Model IV using all 6 input parameters in the case of combined analysis) to predict both c and ϕ have been used to characterize the input data sources (GP, SP, STP, CP, DD, and PI) in terms of their influence on the shear strength parameters of soil (c and ϕ). In this process, the connection weight approach (Olden and Jackson, 2002) and Garson's Algorithm (Garson, 1991; Gevrey et al., 2003) have been used to assess the influence of each variable, in the most appropriate neural network (i.e., 6/2/2), on the prediction of c and ϕ. Both of these approaches use connection weights between input-hidden and hidden-output neurons only and not the biases. The connection weight approach calculates the product of the raw input-hidden and hidden-output connection weights between each input neuron and output neuron, and sums the products across all hidden neurons (Olden and Jackson, 2002; Olden et al., 2004). Garson's algorithm partitions hidden-output connection weights into components associated with each input neuron using the absolute values of connection weights (Garson, 1991; Gevrey et al., 2003; Olden et al., 2004). As in this case, the neural network with 6/2/2 architecture (representing 6 neurons in the input layer, 2 neurons in the hidden layer and 2 neurons in the output layer) has the connection weight matrices of 6×2, and 2×2 size. The product of 6×2 and 2×2 matrices, using the connection weight approach and Garson's approach, gives a resultant matrix of 6×2, which corresponds to the weights of the 6 input variables. The values of these weights are used to rank the input variables, meaning that the variable with maximum weight is assigned as rank 1 and the variable with the minimum weight as rank 6. The weights and ranks corresponding to all the 6 input

Table 4 Individual accuracies in terms of R and $RMSE$ values for both of the shear parameters (c and ϕ) for all the 4 models for some selected neural networks

ANN architecture	c				ϕ			
	R		***RMSE***		***R***		***RMSE***	
	Training	**Testing**	**Training**	**Testing**	**Training**	**Testing**	**Training**	**Testing**
Model I (GP, SP, STP and CP as inputs)								
4/1/2	0.710	0.564	0.336	1.181	0.808	0.693	0.079	0.516
4/6/2	**0.896**	**0.752**	**1.730**	**5.821**	**0.844**	**0.798**	**0.038**	**0.923**
4/7/2	0.833	0.639	1.148	2.083	0.864	0.716	0.320	1.102
4/24/2	0.985	0.670	1.551	0.044	0.958	0.494	0.000	0.414
4/28/2	0.958	0.742	2.006	2.188	0.906	0.377	0.036	0.558
4/32/2	0.968	0.694	0.784	1.208	0.954	0.693	0.007	0.702
4/40/2	0.981	0.555	0.645	0.029	0.975	0.236	0.040	1.592
Model II (GP, SP, STP, CP and DD as inputs)								
5/1/2	0.712	0.714	0.036	0.870	0.727	0.719	0.020	0.435
5/6/2	0.845	0.850	0.820	0.589	0.808	0.791	0.355	0.517
5/16/2	**0.950**	**0.897**	**0.740**	**0.624**	**0.943**	**0.871**	**0.118**	**0.717**
5/22/2	0.989	0.896	0.041	1.832	0.987	0.582	0.091	0.503
5/32/2	0.988	0.924	0.090	0.046	0.991	0.660	0.049	0.052
5/33/2	0.988	0.880	0.119	0.755	0.989	0.636	0.073	0.461
5/40/2	0.993	0.681	0.179	0.038	0.079	0.411	34.165	8.704
Model III (GP, SP, STP, CP and PI as inputs)								
5/1/2	0.657	0.626	0.305	0.888	0.755	0.691	0.002	0.380
5/9/2	0.851	0.510	0.304	0.971	0.888	0.792	0.231	1.025
5/13/2	0.977	0.809	1.783	0.989	0.967	0.684	0.120	0.240
5/18/2	0.975	0.808	0.587	0.180	0.968	0.603	0.076	0.698
5/21/2	**0.970**	**0.820**	**0.735**	**1.261**	0.963	0.731	0.355	0.380
5/27/2	0.989	0.680	0.422	1.735	**0.985**	**0.886**	**0.017**	**0.469**
5/40/2	0.994	0.638	0.168	0.199	0.990	0.354	0.069	0.623
Model IV (GP, SP, STP, CP, DD and PI as inputs)								
6/1/2	0.719	0.751	0.008	0.969	0.765	0.744	0.011	0.536
6/2/2	0.861	0.853	0.311	0.857	0.824	0.854	0.178	0.803
6/5/2	**0.959**	**0.908**	**1.582**	**0.658**	0.899	0.679	0.214	0.082
6/8/2	0.979	0.799	0.222	1.265	**0.916**	**0.922**	**0.045**	**0.190**
6/15/2	0.963	0.839	0.410	0.872	0.902	0.812	0.046	0.700
6/22/2	0.995	0.828	0.163	0.466	0.988	0.815	0.048	0.249
6/40/2	0.999	0.728	17.029	1.422	0.997	0.791	34.165	0.358

variables for the prediction of both c and ϕ are given in Table 6.

Ideally, the finer fractions (CP and STP), along with PI of the soil material, contribute more towards its c than coarser fractions (GP and SP) along with DD. Conversely, the coarser fractions (GP and SP), along with DD of the soil material, contribute more towards its ϕ than the finer fractions (CP and STP) along with PI. This is depicted in the correlation matrix of the original data set (refer to Table 2) used for this study.

It is observed from Table 6 that, for the prediction of c, the descending order of influence of the 6 different input variables in the case of the connection weight approach is PI, STP, SP, CP, DD, and GP. That in case of Garson's approach is CP, GP, PI, DD, SP, and STP. Similarly, from Table 6, it is also observed that, for the prediction of ϕ, the descending order of influence of the 6 input variables in the case of the connection weight approach is GP, DD, CP, SP,

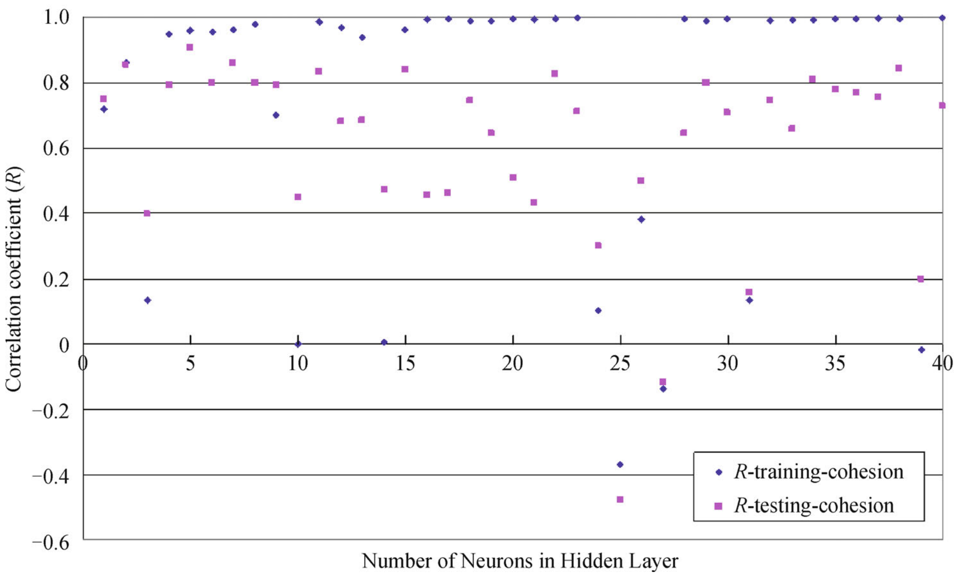


Fig. 5 ANN training and testing accuracies in terms of correlation coefficients observed across different neural networks for the case of c for Model IV.

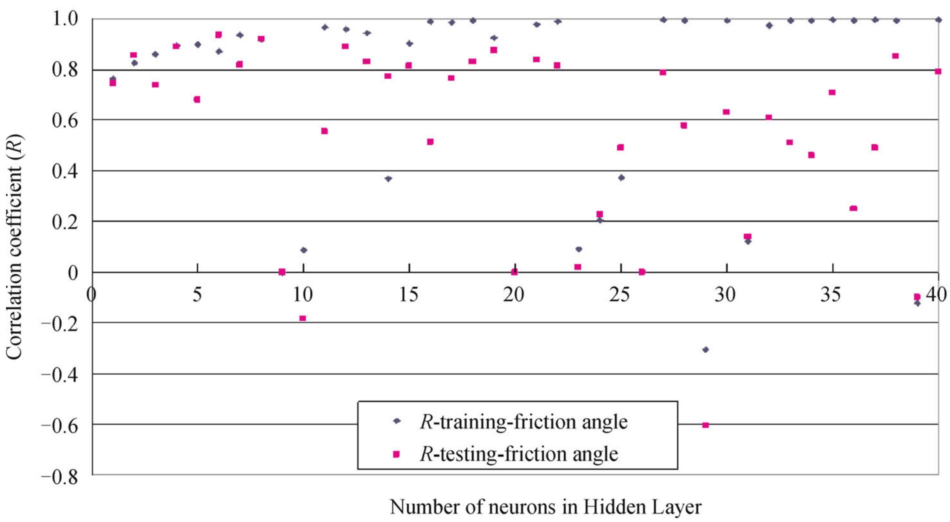


Fig. 6 ANN training and testing accuracies in terms of correlation coefficients observed across different neural networks for the case of ϕ for Model IV.

Table 5 Connection weights and biases for cohesion (c) and friction angle (ϕ) in case of a 6/2/2 neural network

Neurons	Weights (w_{ik})						Weights (v_{ko})		Biases		
	Input 1 (GP)	Input 2 (SP)	Input 3 (STP)	Input 4 (CP)	Input 5 (DD)	Input 6 (PI)	Output 1 (c)	Output 2 (ϕ)	b_k	b_o	
										(c)	(ϕ)
Hidden Neuron 1 ($k=1$)	–124.98	–5.75	–0.98	20.12	–101.27	14.18	2.34	–1.02	87.64	–4.80	2.10
Hidden Neuron 2 ($k=2$)	1.85	–4.18	2.66	–67.58	6.05	56.99	2.94	–1.59	–2.69	—	—

Table 6 Weights and ranks (in brackets) corresponding to all of the 6 input variables for the prediction of both c and ϕ as obtained using Connection weight approach, Garson's approach and Weight-bias approach

Input parameters	Weights and ranks					
	Connection weight approach		Garson's apporach		Weight-bias approach	
	c	ϕ	c	ϕ	c	ϕ
GP	–287.569167 (6)	125.21965 (1)	0.480890716 (2)	0.480890716 (2)	–41.045 (5)	12.532 (3)
SP	–25.7740852 (3)	12.54380092 (4)	0.051503799 (5)	0.051503799 (5)	–1.7203 (4)	22.123 (2)
STP	5.51500891 (2)	–3.22028038 (5)	0.02275593 (6)	0.02275593 (6)	–118.6067 (6)	83.4911 (1)
CP	–151.67199 (4)	86.88804508 (3)	0.560407326 (1)	0.560407326 (1)	200.5407 (3)	–88.731 (4)
DD	–219.64503 (5)	94.23460104 (2)	0.422294122 (4)	0.422294122 (4)	232.2052 (2)	–105.052 (5)
PI	200.9270384 (1)	–105.211826 (6)	0.462148106 (3)	0.462148106 (3)	366.723 (1)	–171.444 (6)

STP, and PI; and that in the case of Garson's approach is CP, GP, PI, DD, SP, and STP (exactly the same as in the case of c). This result illustrates, to some extent, the critical difference between the connection weight approach and Garson's approach that results in their differential ability to correctly identify variable importance in the neural network.

The connection weight product matrix (Table 6) shows that the finer fractions (CP and STP), along with PI, have comparatively more influence than a coarser fraction (GP), along with DD, on the prediction of c, as ideally the case should be. However, there exists an ambiguity in this case. SP, being a coarser fraction of soil, exhibits more influence on the prediction of its c than the finer fraction CP. Similarly, it is also demonstrated in the connection weight product matrix (Table 6) that the coarser fractions (GP), along with DD, have comparatively more influence than the finer fractions (CP and STP), along with PI, on the prediction of ϕ of the soil, as ideally the case should be. However, there exists a similar ambiguity as discussed above. On the other hand, the inability of Garson's algorithm to correctly depict the true order of variable importance can be simply because it uses absolute connection weights in its calculations, and fails to account for the contrasting influences of input neurons through different hidden neurons to output neurons. In this study, the Garson's weight product matrices for both c and ϕ are exactly same.

It is also an interesting finding that, in the case of Garson's algorithm for weight analysis, no contrast in the order of influence of input variables is depicted in the case of neural network architectures with multiple output neurons (as in this case two output neurons c and ϕ). In contrast, the connection weight approach uses raw connection weights, which accounts for the direction of the input–hidden–output relationship and results in the correct identification of variable contributions.

5.1.3 Connection weight-bias analysis

In the present study, an attempt was made to develop an approach that takes into account both the connection weights between input – hidden and hidden – output neurons, and biases at hidden and output neurons (Table 5), in order to assess the influence of input variables in the most appropriate neural network (i.e., 6/2/2) on the prediction of c and ϕ. According to Goh et al. (2005), a model equation can be established with these weights and biases as the model parameters to obtain the network output. The mathematical equation relating the input variables and the output can be written as:

$$c_n \text{ or } \phi_n = f_{\text{log-}sig}\left\{ b_0 + \sum_{k=1}^{h}\left[v_{ko} \times f_{\text{log-}sig}\left(b_k + \sum_{i=1}^{m} w_{ik} X_i \right)\right]\right\}, \tag{9}$$

where c_n or ϕ_n is the normalized (in the range of 0 to 1 in this case) value of c or ϕ depending on the weights and biases; b_0= bias at the output layer; v_{ko} connection weight between k^{th} neuron of hidden layer and the output layer (o); b_k= bias at the k^{th} neuron of hidden layer; h = number of neurons in the hidden layer; w_{ik}= connection weight between i^{th} input neuron and k^{th} neuron of hidden layer; X_i= normalized input variable i in the range [0,1]; m = total No. of input variables and $f_{\text{log-}sig}$= log-sigmoid transfer function $\left(f(x) = \dfrac{1}{1+\mathrm{e}^{-x}} \right)$.

Using the values of the weights and biases presented in

Table 5, one can easily calculate the normalized values of unsaturated c and ϕ respectively, which need to be denormalized with respect to their maximum values in the original data set.

Taking a lead from the above Eq. (9), the following equations (Eqs. (10) and (11)) are designed to calculate the resultant matrix of 6×2, which corresponds to the weights of 6 input variables for the prediction of both c and ϕ, using connection weights as well as biases (named the Weight-bias Approach).

$$\left\{\begin{array}{l}\mathrm{GP}\\ \mathrm{SP}\\ \mathrm{STP}\\ \mathrm{CP}\\ \mathrm{DD}\\ \mathrm{PI}\end{array}\right\} = \left\{\begin{array}{l} b_0(c)+v_{11}(b_1+w_{11})+v_{21}(b_2+w_{21})\\ b_0(c)+v_{11}(b_1+w_{12})+v_{21}(b_2+w_{22})\\ b_0(c)+v_{11}(b_1+w_{13})+v_{21}(b_2+w_{23})\\ b_0(c)+v_{11}(b_1+w_{14})+v_{21}(b_2+w_{24})\\ b_0(c)+v_{11}(b_1+w_{15})+v_{21}(b_2+w_{25})\\ b_0(c)+v_{11}(b_1+w_{16})+v_{21}(b_2+w_{26})\end{array}\right\}, \tag{10}$$

$$\left\{\begin{array}{l}\mathrm{GP}\\ \mathrm{SP}\\ \mathrm{STP}\\ \mathrm{CP}\\ \mathrm{DD}\\ \mathrm{PI}\end{array}\right\} = \left\{\begin{array}{l} b_0(\phi)+v_{12}(b_1+w_{11})+v_{22}(b_2+w_{21})\\ b_0(\phi)+v_{12}(b_1+w_{12})+v_{22}(b_2+w_{22})\\ b_0(\phi)+v_{12}(b_1+w_{13})+v_{22}(b_2+w_{23})\\ b_0(\phi)+v_{12}(b_1+w_{14})+v_{22}(b_2+w_{24})\\ b_0(\phi)+v_{12}(b_1+w_{15})+v_{22}(b_2+w_{25})\\ b_0(\phi)+v_{12}(b_1+w_{16})+v_{22}(b_2+w_{26})\end{array}\right\}. \tag{11}$$

The weights and ranks corresponding to all of the 6 input variables for the Prediction of both c and ϕ, as obtained using the Weight-bias Approach, are also given in Table 6. Here, it is observed that for the prediction of c, the descending order of influence of 6 input variables in this case is PI, DD, CP, SP, GP, and STP. For the prediction of ϕ, the descending order of influence of these variables is STP, SP, GP, CP, DD, and PI. The resultant matrix (Table 6) indicates that the finer fractions (CP) along with PI have comparatively more influence than the coarser fractions (SP and GP) on the prediction of c, as ideally the case should be. However, there exist ambiguities in that STP, being a finer fraction of soil, exhibits less influence on the prediction of its c than the coarser fractions GP and SP. Also, DD has more influence in predicting c, which should not be the case.

It can be observed that the coarser fractions (SP and GP) have comparatively more influence than the finer fractions (CP) along with DD of the soil on the prediction of its ϕ, as the case should be. However, there also exist ambiguities in this case; in that, STP being a finer fraction of soil exhibits more influence on the prediction of its ϕ than the coarser fractions GP and SP. Also, PI has more influence in predicting ϕ, which again should not be the case. Hence, in the case of the Weight-bias Approach, it can be stated that the results to a great extent match the correlation matrix of the original data set (refer to Table 2). This clearly reflects the use of raw connection weights and biases for generating the resultant matrix, as in the case of the Connection Weight Approach.

5.1.4 Comparison of Connection Weight and Weight-bias approaches

As the results for the Connection Weight Approach, Garson's Approach, and the proposed Weight-bias Approach are compared, it can be summarized that, for the assessment of degree of influence of input variables on the prediction of shear strength parameters of soils under unsaturated conditions, the most appropriate weight analysis approach is the Connection Weight Approach. It was found that the Connection Weight Approach provides the best overall methodology for accurately quantifying variable importance, and should be favored over the other approaches examined in this study. The proposed Weight-bias Approach performed relatively well. Garson's Approach showed both poor accuracy and precision, and showed no contrast in weights across multiple outputs. The most notable result of this study is that Garson's Algorithm is the poorest performing approach.

5.2 Results of the Regression Tree technique

As in the ANN technique, experimentally determined c and ϕ values of soils under unsaturated conditions are also used as the target outputs in the regression tree technique. Both the experimentally determined and observed values of c and ϕ are compared using R and $RMSE$ measures in the training and testing phase to assess the prediction capability of this technique. The variation in $RMSE$ and R values across different trees with varying splitmins are given in Fig. 7 for Model IV, with all 6 input parameters, to assess their prediction capability for both c and ϕ of soils. The best prediction tree in each combination case is chosen on the basis of these performance measures. Tables 7 and 8 contain the statistical performance details ($RMSE$ and R) of the best prediction trees for c and ϕ, respectively, for four different models. It can be inferred from the results given in both tables that for different combinations of input parameters the splitmin value varies for the best regression tree for prediction of both c and ϕ.

The results shown in Table 7 indicate that the best result for the prediction of c has been obtained using Model IV, which considers all 6 input parameters (GP, SP, STP, CP, DD, and PI). The training and testing R values are 0.88 and 0.73 respectively. Also, the $RMSE$ values for the training and testing datasets are quite reasonable; and the difference is less in this case than it is using the other 3 combinations

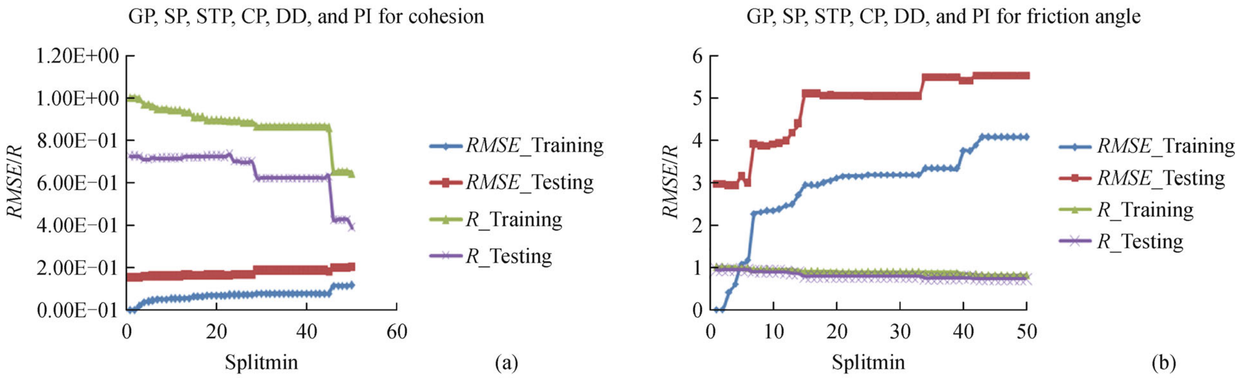


Fig. 7 Variation in *RMSE* and *R* values across different regression trees with varying splitmins for Model IV for: (a) cohesion and (b) friction angle.

Table 7 Training and testing accuracies for prediction of c using Regression Tree technique

Predictors	Splitmin	***RMSE***/(kg·cm^{-2})		***R***	
		Training	**Testing**	**Training**	**Testing**
GP,SP,STP, and CP (Model I)	19	0.080	0.192	0.84	0.45
GP,SP,STP,CP, and DD (Model II)	12	0.056	0.165	0.93	0.67
GP,SP,STP,CP, and PI (Model III)	10	0.075	0.189	0.87	0.48
GP,SP,STP,CP,DD, and PI (Model IV)	23	0.069	0.162	0.88	0.73

Table 8 Training and testing accuracies for prediction of ϕ using Regression Tree technique

Predictors	Splitmin	***RMSE***/degree		***R***	
		Training	**Testing**	**Training**	**Testing**
GP,SP,STP, and CP (Model I)	11	2.709	6.078	0.92	0.64
GP,SP,STP,CP, and DD (Model II)	9	2.361	3.749	0.94	0.89
GP,SP,STP,CP, and PI (Model III)	15	2.804	5.645	0.91	0.70
GP,SP,STP,CP,DD, and PI (Model IV)	9	2.349	3.833	0.94	0.90

of input parameters. In cases using the other 3 combinations of input parameters, it can be inferred from the results that the prediction trees are over-trained, and could not attain generalization capability because the difference between training and testing *R* is quite large.

Results in Table 8 show that, for the prediction of ϕ, again, the best result was obtained with Model IV, considering all 6 input parameters, with training and testing *R* values of 0.94 and 0.90, respectively. These values are comparatively higher than those in the case of prediction of c (refer to Table 7). In the case of Model II, with input parameters GP, SP, STP, CP, and DD, the prediction accuracies in terms of *R* values for training and testing are 0.94 and 0.89, respectively, and are of the same order as those for Model IV. Hence, it can be inferred from these results that the input parameter PI has little influence on the prediction or estimation of ϕ. In the cases of the other two combinations of input parameters, it can be stated that the regression trees were over-trained, and did not attain generalization capability because the difference between training and testing *R* values is quite large.

Hence, it can be summarized from the above results that the regression tree analysis in the case of Model IV, considering all 6 predictors (GP, SP, STP, CP, DD, and PI), gives the most appropriate result for the prediction of both c and ϕ. The most appropriate regression trees obtained for c and ϕ for Model IV, using all 6 input parameters as predictors, are given in Fig. 8. The experimentally determined and predicted c and ϕ values for the testing dataset of 20 soil samples in the case of Model IV are represented in Fig. 9. This figure clearly shows a good correlation between the experimentally determined and predicted values of both shear parameters of soil.

6 Conclusions

In this research, ANN and CART techniques for the prediction of shear strength parameters (c and ϕ) of soils under unsaturated conditions was investigated. The performance evaluation of these techniques was carried out using R and $RMSE$ measures. In this work, four different models were adopted, considering different combinations of 6 input parameters of soil such as GP, SP, STP, CP, DD, and PI. For all four models, with varied combinations of input parameters, both ANN and CART

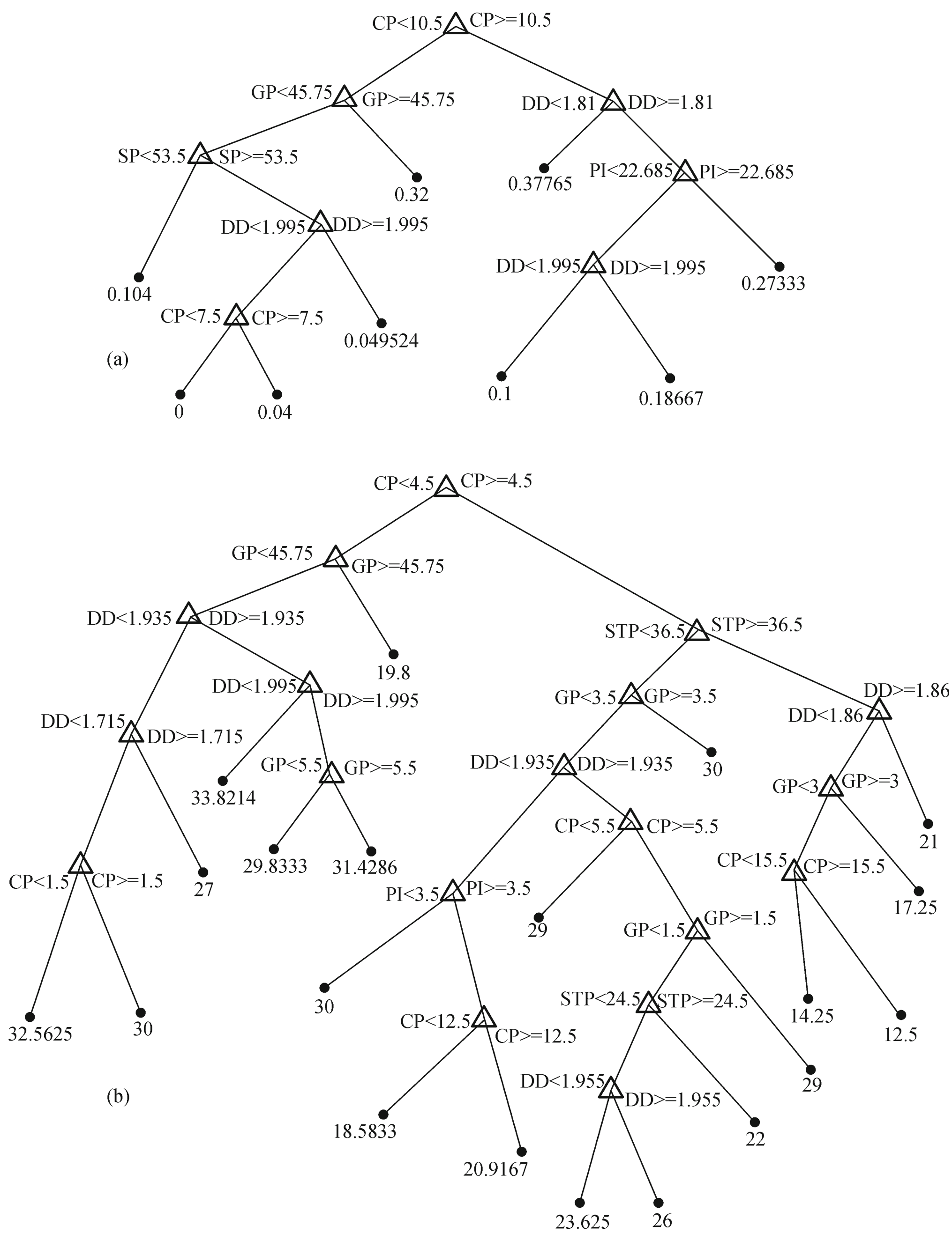
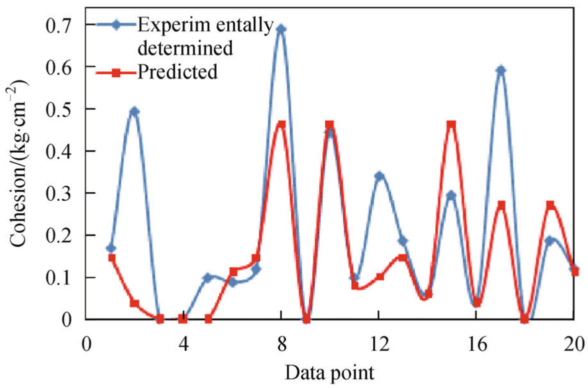
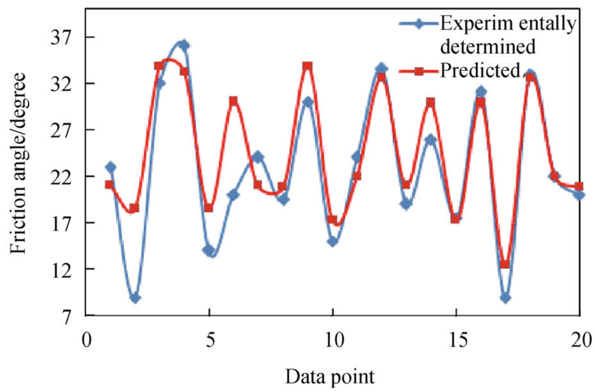


Fig. 8 Most appropriate regression tree in the case of Model IV for predicting (a) cohesion and (b) angle of internal friction.

techniques were attempted and evaluated. The results indicate that amongst these models, Model IV, considering all 6 input parameters, is the most appropriate model for the prediction of shear parameters with both ANN and CART techniques.



(a)



(b)

Fig. 9 Experimentally determined and predicted values for the testing dataset in case of Model IV using regression tree technique: (a) cohesion and (b) angle of friction.

In the case of the ANN technique, the 6/2/2 neural network architecture gave the best training and testing accuracy, with combined R values of 0.939 and 0.879, respectively; and with combined $RMSE$ values 0.103 and 0.136, respectively, for prediction of both c and ϕ. Similarly, in cases using regression tree analysis, it was found that, for Model IV, the regression tree with a splitmin of 23 obtained the best training and testing accuracy, with R values 0.88 and 0.73, respectively; and with $RMSE$ values 0.069 and 0.162, respectively, for the prediction of soil c. For the prediction of ϕ, the regression tree with a splitmin of 9 gave the best training and testing accuracy, with R values 0.94 and 0.90, respectively; and with $RMSE$ values 2.349 and 3.833, respectively. With regression trees like those generated in this study, having many branches, there is the possibility of over fitting the tree towards the training data set, hence it loses its generalization capability. Keeping this in view, simpler regression trees in this case have been tried; and results with considerably good accuracies have been obtained.

From the results obtained using ANN and CART techniques, it can be observed that for the prediction of ϕ, the performances of both the techniques are of the same order, which is indicated by the similar R values for both training and testing samples. However, for the prediction of c, the ANN technique performs better than the regression tree technique. Hence, it can be concluded from these results that the ANN technique, with the Levenberg-Marquardt learning rule, is a comparatively better approach than the CART technique for the prediction of both of the shear parameters of the soils.

Further, the performance evaluation of the ANN technique, for the indirect estimation of both of the shear parameters (c and ϕ) of soils, has also been attempted on an individual shear parameter basis, to assess and compare the combined prediction capability vis-à-vis individual prediction capability of the ANN technique. It was observed from the results that the individual or combined performance evaluation of both training and testing datasets did not affect the overall result in selecting the most appropriate model (Model IV considering all 6 input parameters in this case) for the prediction of both of the shear parameters of soil (c and ϕ). However, it does have effects in terms of the R values, and also in terms of the most appropriate neural network architecture. So, it can finally be stated that a combined performance evaluation is always convenient to use in order to select the best single neural network architecture for the prediction of both c and ϕ.

Connection weight and bias analyses of the best neural network (i.e., 6/2/2) were attempted using the Connection Weight Approach, Garson's Approach, and a proposed Weight-bias Approach to characterize the input variables in terms of their influence on the shear strength parameters of soil (c and ϕ). It can be summarized that the Connection Weight Approach appeared to be the most appropriate and accurate weight analysis approach for quantifying variables importance. This method should be favored over the other approaches examined in this study. The Connection Weight Approach uses raw connection weights, which accounts for the direction of the input-hidden-output relationship, and results in the correct identification of variable contribution. The proposed Weight-bias Approach performed relatively well; and the results were comparable with the Connection Weight Approach results. The most notable result of this analysis was that Garson's Algorithm was the poorest performing approach; and no contrast in the order of influence of the input variables was depicted in cases of neural network architecture with multiple output neurons (as in those cases with two output neurons c and ϕ).

Acknowledgements The authors are grateful to the Director, CSIR-CBRI for his kind permission to publish this work. We also thank the editors and two anonymous reviewers for their valuable comments, which helped to improve the quality of the paper.

References

Agrawal G, Weeraratne S, Khilnani K (1994). Estimating clay liner and cover permeability using computational neural networks. In: Proceedings of the 1st Congress on Computing in Civil Engineering, Washington

Akbulut S (2005). Artificial neural networks for predicting the hydraulic conductivity of coarse grained soils. Eurasian Soil Sci, 38: 392–398

Alavi A H, Gandomi A H, Gandomi M, Sadat H S S (2009). Prediction of maximum dry density and optimum moisture content of stabilized soil using RBF neural networks. IES Journal Part A: Civil & Structural Engineering, 2(2): 98–106

Arora M K, Das Gupta A S, Gupta R P (2004). An artificial neural network approach for landslide hazard zonation in the Bhagirathi (Ganga) Valley, Himalayas. Int J Remote Sens, 25(3): 559–572

Atkinson P M, Tatnall A R L (1997). Neural networks in remote sensing. Int J Remote Sens, 18(4): 699–709

Attoh-Okine N O (2004). Application of genetic-based neural network to lateritic soil strength modeling. Construct Build Mater, 18(8): 619–623

Baykasoğlu A, Güllüb H, Çanakçıb H, Özbakırc L (2008). Prediction of compressive and tensile strength of limestone via genetic programming. Expert Syst Appl, 35(1–2): 111–123

Breiman L, Friedman J H, Olshen R A, Stone C J (1984). Classification and Regression Trees. Newyork: Chapman and Hall Ltd/CRC

Çanakcı H, Baykasoglu A, Gullu H (2009). Prediction of compressive and tensile strength of Gaziantep basalts via neural networks and gene expression programming. Neural Comput Appl, 18(8): 1031–1041

Cho S E (2009). Probabilistic stability analyses of slopes using the ANN-based response surface. Comput Geotech, 36(5): 787–797

Das S K, Basudhar P K (2008). Prediction of residual friction angle of clays using artificial neural network. Eng Geol, 100(3–4): 142–145

Escario V, Juca J E T (1989). Strength and deformation of partly saturated soils. In: Proceedings 12th International Conf. Soil Mech. Foundation Engineering. Rio de Janeiro, 1: 43–46

Ferentinou M D, Sakellariou M G (2007). Computational intelligence tools for the prediction of slope performance. Comput Geotech, 34 (5): 362–384

Foody G M, Arora M K (1997). An evaluation of some factors affecting the accuracy of classification by an artificial neural network. Int J Remote Sens, 18(4): 799–810

Freund J E (1992). Mathematical Statistics (5th edition). New Delhi: Printice-Hall of India Pvt. Ltd., 658

Garson G D (1991). Interpreting neural-network connection weights. Artif Intell Expert, 6: 47–51

Gevrey M, Dimopoulos I, Lek S (2003). Review and comparison of methods to study the contribution of variables in artificial neural network models. Ecol Model, 160: 249–264

Goh A T C, Kulhawy F H, Asce F, Chua C G (1994). Seismic liquefaction potential assessed by neural networks. J Geotech Geoenviron Eng, 120(9): 1467–1480

Goh A T C, Kulhawy F H, Chua C G (2005). Bayesian neural network analysis of undrained side resistance of drilled shafts. J Geotech Geoenviron Eng, 131(1): 84–93

Goktepe A B, Altun S, Altintas G, Tan O (2008). Shear strength estimation of plastic clays with statistical and neural approaches. Build Environ, 43(5): 849–860

Goktepe A B, Sezer A (2010). Effect of particle shape on density and permeability of sands. Proceedings of institution of civil engineers geotechnical engineering, 163: 307–320

Gómez H, Kavzoglu T (2005). Assessment of shallow landslide susceptibility using artificial neural networks in Jabonosa River Basin, Venezuela. Eng Geol, 78(1–2): 11–27

Gong P (1996). Integrated analysis of spatial data from multiple sources: using evidential reasoning and artificial neural network techniques for geological mapping. Photogrammetric Engineering & Remote Sensing, 62(5): 513–523

Hagan M T, Demuth H B, Beale M H (1996). Neural Network Design. Boston: PWS Publishing, 730

Hagan M T, Menhaj M B (1994). Training feedforward networks with the Marquardt algorithm. IEEE Trans Neural Netw, 5(6): 989–993

Hanna A M, Ural D, Saygili G (2007). Neural network model for liquefaction potential in soil deposits using Turkey and Taiwan earthquake data. Soil Dyn Earthquake Eng, 27(6): 521–540

Haykin S (1998). Neural Networks: A Comprehensive Foundation. New Jersey: Prentice Hall, 842

IS 2720 (Part IV) (1985). Indian Standard for Grain Size Analysis (2nd Revision). New Delhi: Bureau of Indian Standards, 73–94

IS 2720 (Part V) (1985). Indian Standard for Determination of Liquid and Plastic Limit (2nd Revision). New Delhi: Bureau of Indian Standards, 109–114

IS 2720 (Part XIII) (1986). Indian Standard for Direct Shear Test (2nd Revision). New Delhi: Bureau of Indian Standards, 195–198

Jain V, Seung H S, Turaga S C (2010). Machines that learn to segment images: a crucial technology for connectomics. Curr Opin Neurobiol, 20(5): 653–666

Juang C H, Chen C J, Jiang T, Andrus R D (2000). Risk-based liquefaction potential evaluation using standard penetration tests. Can Geotech J, 37(6): 1195–1208

Kanungo D P, Arora M K, Sarkar S, Gupta R P (2006). A comparative study of conventional, ANN black box, fuzzy and combined neural and fuzzy weighting procedure for landslide susceptibility zonation in Darjeeling Himalayas. Eng Geol, 85(3–4): 347–366

Kaya A (2009). Residual and fully softened strength evaluation of soils using artificial neural networks. Geological and Geotechnical Engineering, 27(2): 281–288

Kayadelen C, Günaydın O, Fener M, Demir A, Özvan A (2009). Modeling of the angle of shearing resistance of soils using soft computing systems. Expert Systems with Applications, 36: 11814–11826

Lee S, Ryu J H, Won J S, Park H J (2004). Determination and application of the weights for landslide susceptibility mapping using an artificial neural network. Eng Geol, 71(3–4): 289–302

Lu P, Rosenbaum M S (2003). Artificial neural networks and grey systems for the prediction of slope stability. Nat Hazards, 30(3): 383–398

Lu Z (1992). The relationship of shear strength to swelling pressure for unsaturated soils. Chinese journal of geotechnical engineering, 14(3): 1–8 (in Chinese)

Maji V B, Sitharam T G (2008). Prediction of elastic modulus of jointed rock mass using artificial neural networks. Geotech Geol Eng, 26(4): 443–452

Najjar Y M, Basheer I A (1996). Discussion of stress-strain modeling of sands using artificial neural networks. J Geotech Eng, 122(11): 949–951

Neaupane K M, Achet S H (2004). Use of backpropagation neural network for landslide monitoring: a case study in the higher Himalaya. Eng Geol, 74(3–4): 213–226

Nefeslioglu H A, Duman T Y, Durmaz S (2008). Landslide susceptibility mapping for a part of tectonic Kelkit Valley (Eastern Black Sea region of Turkey). Geomorphology, 94(3–4): 401–418

Olden J D, Jackson D A (2002). Illuminating the "black box": understanding variable contributions in artificial neural networks. Ecol Model, 154: 135–150

Olden J D, Joy M K, Death R G (2004). An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data. Ecol Model, 178 (3–4): 389–397

Rafiai H, Jafari A (2011). Artificial neural networks as a basis for new generation of rock failure criteria. Int J Rock Mech Min Sci, 48(7): 1153–1159

Ripley B (1996). Pattern Recognition and Neural Networks. Cambridge: Cambridge University Press, 403

Schalkoff R J (1997). Artificial Neural Networks. New York: Wiley, 422

Shen Z, Yu S (1996). The problems in the present studies on mechanics of unsaturated soils. In: Proceedings of the Symposium on Geotechnical Aspects of Regional Soils. Beijing: Atomic Energy Press (in Chinese)

Sietsma J, Dow R J F (1991). Creating artificial neural networks that generalize. Neural Netw, 4(1): 67–79

Sonmez H, Gokceoglua C, Nefeslioglub H A, Kayabasi A (2006). Estimation of rock modulus: for intact rocks with an artificial neural network and for rock masses with a new empirical equation. Eng Geol, 43: 224–235

Tiryaki B (2008). Predicting intact rock strength for mechanical excavation using multivariate statistics, artificial neural networks, and regression trees. Eng Geol, 99(1–2): 51–60

Xu Y (1997). Mechanical Properties of Unsaturated Expansive Soils and Its Application to Engineering. Dissertation for Ph.D degree. Nanjing: Hohai University (in Chinese)

Yesilnacar E, Topal T (2005). Landslide susceptibility mapping: a comparison of logistic regression and neural networks methods in a medium scale study, Hendek region (Turkey). Eng Geol, 79(3–4): 251–266

Youd T L, Gilstrap S D (1999). Liquefaction and deformation of silty and fine-grained soils. In: Proceedings of the 2nd international conference on earthquake geotechnical engineering, 3: 1013–1020

Zhou W (1999). Verification of the nonparametric characteristics of back propagation neural networks for image classification. IEEE Transaction on Geoscience and remote sensing, 37: 771–779