Quest Journals
Journal of Research in Environmental and Earth Sciences
Volume 7 ~ Issue 6 (2021) pp: 37-47
ISSN(Online) :2348-2532
www.questjournals.org

Research Paper

Predicting the Effective Stress Parameter of Unsaturated Soil Using Artificial Neural Network

[1]J.A Ige and [2]M.A. Hassan
[1,2]Department of Civil Engineering, Ladoke Akintola University of Technology, Ogbomoso-Nigeria
Correspondence e-mail: jaige@lautech.edu.ng

ABSTRACT
Experimental studies on unsaturated soils are generally costly, time-consuming, and difficult to conduct. Research has justified the effectiveness of artificial neural network for future predictions.
Therefore, in this project a neural network approach is used to predict effective stress parameter of unsaturated soil. A sequential architecture was chosen for the network, that is, a multilayer perception network with feed forward capability.
The input layer consists of four neutrons and one neutron in the output layer with ten neutrons in the hidden layer for improving the network accuracy. The data used are generated with regression analysis with aid of bishop's equation.
70% of generated set were used to train the network and 30% to test the network. Neural network simulations were compared with experimental results. The comparison indicates the good performance of the proposed network for predicting the stress parameter of unsaturated soils.
KEY WORDS: Unsanturated soil, neural network, bitumen and laterite

Received 02 June, 2021; Revised: 14 June, 2021; Accepted 16 June, 2021

I. INTRODUCTION

Artificial neural networks (ANNs) are a form of artificial intelligence which attempt to mimic the function of the human brain and nervous system. ANNs learn from data examples presented to them in order to capture the subtle functional relationships among the data even if the underlying relationships are unknown or the physical meaning is difficult to explain. This is in contrast to most traditional empirical and statistical methods, which need prior knowledge about the nature of the relationships among the data. ANNs are thus well suited to modeling the complex behavior of most geotechnical engineering materials which, by their very nature, exhibit extreme variability. This modeling capability, as well as the ability to learn from experience, has given ANNs superiority over most traditional modeling methods since there is no need for making assumptions about what the underlying rules that govern the problem in hand could be.

The conventional method in geotechnical engineering for calculating the effective stress of soil as postulated by Terzaghi (i.e. $\sigma' = \sigma - u$) is found to be restricted to saturated soil alone. Subsequent developments on this equation brought about bishop's equation (i.e. $\sigma' = (\sigma - \mu a) + \chi(\sigma - \mu w)$) which is applicable to unsaturated soil. The bishop's equation consists of net stress, suction change and the effective stress parameter. In some extreme scenario, the prediction of effective stress parameter of unsaturated soil seems to be complex due to varied parameters within the equation. The use of artificial intelligent techniques for instance, neural networks have shown great potential in this field. With the involvement of soft computing, the pattern matching, classification and detection of algorithms which have direct applications in many geotechnical problems have become much easier to be implemented and predict the effective stress parameter. Hence, this project tries to find out how artificial neural network can be applied in geotechnical engineering to predict effective stress parameter of unsaturated soil.

Overview of Artificial Neural Networks

ANNs consist of a number of artificial neurons variously known as „processing elements" (PEs), „nodes" or „units". For multilayer perceptrons (MLPs), which is the most commonly used ANNs in geotechnical engineering, processing elements in are usually arranged in layers: an input layer, an output layer and one or more intermediate layers called hidden layers.

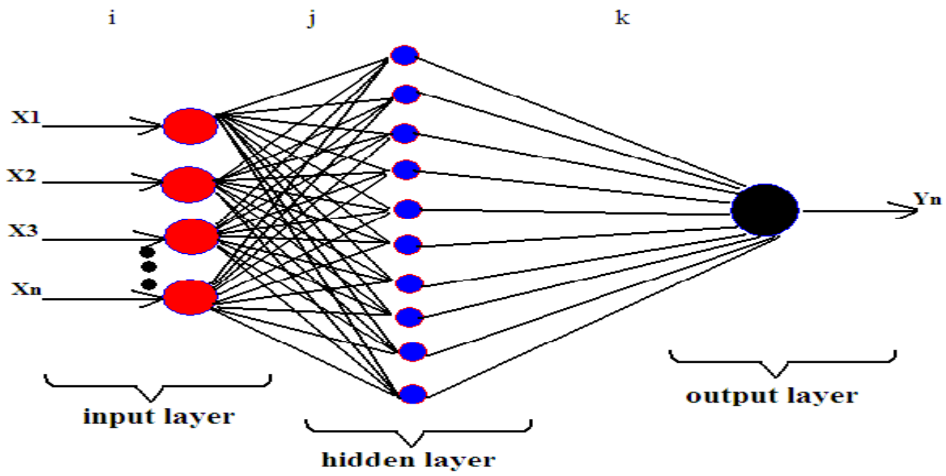

Figure 1 Structure of artificial neural network

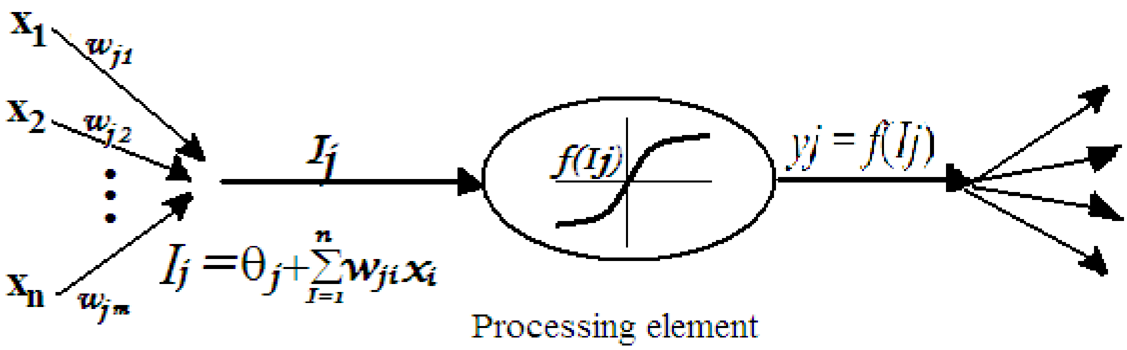

Figure 2 Typical operation of ANNs

Each processing element in a specific layer is fully or partially connected to many other processing elements via weighted connections. The scalar weights determine the strength of the connections between interconnected neurons. A zero weight refers to no connection between two neurons and a negative weight refers to a prohibitive relationship. From many other processing elements, an individual processing element receives its weighted inputs, which are summed and a bias unit or threshold is added or subtracted. The bias unit is used to scale the input to a useful range to improve the convergence properties of the neural network. The result of this combined summation is passed through a transfer function (e.g. logistic sigmoid or hyperbolic tangent) to produce the output of the processing element. For node *j*, this process is summarized in Equations 1 and 2 and illustrated in Figures 1 and 2

$$I_j = \theta_j + \sum_{i=1}^{n} w_{ji} x_i \qquad \text{summation}$$

$$y_j = f(I_j) \qquad \text{transfer}$$

where
Ij= the activation level of node *j*;
wji= the connection weight between nodes *j* and *i*;
xi= the input from node *i*, *i* = 0, 1, …, *n*;
øj= the bias or threshold for node *j*;
yj= the output of node *j*; and
f (.) = the transfer function.

The propagation of information in MLPs starts at the input layer where the input data are presented. The inputs are weighted and received by each node in the next layer. The weighted inputs are then summed and passed through a transfer function to produce the nodal output, which is weighted and passed to processing elements in the next layer. The network adjusts its weights on presentation of a set of training data and uses a learning rule until it can find a set of weights that will produce the input-output mapping that has the smallest possible error. The above process is known as „learning" or „training".

Learning Phase

Basically, there are two ways of learning in artificial neural network which are **supervised** and **unsupervised learning**.

Supervised Learning: The network is presented with a historical set of model inputs and the corresponding (desired) outputs. The actual output of the network is compared with the desired output and an error is calculated. This error is used to adjust the connection weights between the model inputs and outputs to reduce the error between the historical outputs and those predicted by the ANN.

Unsupervised Learning: the network is only presented with the input stimuli and there are no desired outputs. The network itself adjusts the connection weights according to the input values. The idea of training in unsupervised networks is to cluster the input records into classes of similar features.

ANNs can be categorized on the basis of two major criteria: **(i)** the learning rule used and **(ii)** the connections between processing elements. Based on learning rules, ANNs can be divided into supervised and unsupervised networks. But based on connections between processing elements, ANNs can be divided into **feed-forward and feedback networks**.

In feed forward networks, the connections between the processing elements are in the forward direction only, whereas, **in feedback networks**, connections between processing elements are in both the forward and backward directions.

The ANN modeling philosophy is similar to a number of conventional statistical models in the sense that both are attempting to capture the relationship between a historical set of model inputs and corresponding outputs. For example, suppose a set of x-values and corresponding y values in 2 dimensional space, where $y = f(x)$. The objective is to find the unknown function f, which relates the input variable x to the output variable y. In a linear regression model, the function f can be obtained by changing the slope tanφ and intercept β of the straight line in Fig. 2(a), so that the error between the actual outputs and outputs of the straight line is minimized. The same principle is used in ANN models. ANNs can form the simple linear regression model by having one input, one output, no hidden layer nodes and a linear transfer function (Fig. 2(b)). The connection weight w in the ANN model is equivalent to the slope tanφ and the threshold θ is equivalent to the intercept β, in the linear regression model. ANNs adjust their weights by repeatedly presenting examples of the model inputs and outputs in order to minimize an error function between the historical outputs and the outputs predicted by the ANN model.

Effective Stress Review

The principle of effective stress is one of the most important concepts of modern soil mechanics. It has been found useful as a basis for the understanding of stress and strain characteristics of soils has become increasingly important in practical engineering problems. Terzaghi (1925) appears to have been the first to recognize the importance of effective stresses within soil masses. Bishop (1960) has summarized the historical development of the concept of effective stresses in soil masses and has considered the theoretical aspects of the principle in detail.

According of the principle of effective stress the strength and compressibility properties of a soil depend not on the total stress applied to the soil mass, but rather on the difference between the total stress and the stress carried by the pore fluid. The difference is termed the effective stress and is given for a saturated soil, by

$\sigma = \sigma - u_w$

Recent studies have shown effective stress as stated by the above equation to be inadequate to account for the behavior of partially saturated soils. Modifications of this equation have been made by bishop (1960), Jennings (1960), Aitchison (1960), and Croney and Cole man 1953. The expression suggested by Bishop is the most general in that it accounts for pore water and pore pressures. (Abu-Kiefa !998, Agrawal and Bourdeau 1997).

His expression is $\sigma' = (\sigma - \mu a) + \chi(\sigma - \mu w)$

Review of Stress Frameworks for Unsaturated Soils

Even though the long-time debate on the most appropriate stress framework for unsaturated soils has eventually come to an end, evidencing the need for complete hydro-mechanical stress framework, several forms of adequate stress variables are still possible. Those different unified stresses, now used for advanced constitutive modeling of unsaturated soils, present several levels of complexity and can be considered as inherited either from the Bishop single effective stress or from the independent stress variable approach. The common feature of the new stress conceptions is nevertheless the use of suction, or a modified version of it, as a second stress variable to build complete hydro-mechanical frameworks. It is proposed hereafter to overview the arguments at the origin of these advanced unified stresses, by retracing the chronological evolution of effective, independent and combined stresses for unsaturated soil modeling

II. METHODS

2.1 Methods

Basically, two main phases are employed in this project which is theoretical phase and practical phase. Theoretical phase involves the process of reading and understanding, reviewing of theories, studying, surveying the neural network and relating it to effective stress parameter concept.

On the other hand, the practical phase involves data processing, network model coding, simulation, assumptions, adaptation and training of the Neurons by using MATLAB R2007b.

2.1 Theoretical Phase

There are many stages involved in this model which starts from the data generation, Input to output. Below are the sequential operations in orderly form.(Agrawal et al 1994)

1. Determination of adequate data input: this is done by generating data input for 100 experiments with the aid of the formula: $\sigma' = (\sigma - \mu a) + \chi(\sigma - \mu w)$ (bishop 1960) using linear regression analysis.

Where

σ'= Effective stress of soil

μw = Pore water pressure of the soil

μa = Pore air pressure of the soil

σ = Total pressure of the soil

χ = Effective stress parameter of unsaturated soil which ranges from 0% (for dry soil) to 100% (for fully saturated soil).

And

$\sigma = Hsoil\gamma soil$ ………………… (1)

$\mu w = Hw\gamma w$ ………………… (2)

$\mu a = Ha\gamma a$ ……………………..… (3)

$\gamma = \rho g$ (g= 9.8m/s^2)…………..….. (4)

NB

H= depth of soil, water and air.

γ = unit weight of density and air.

g= acceleration due to gravity (g= 9.8m/s^2).

ρ = density

2. To ensure that the model generalize within the range of the data used for prediction, the generated data is divided into two subsets; a training set 70% of generated data, to construct the neural network model, and an independent testing set 30% of generated data to estimate the model performance.(Mathworks 2007).

Table 1: Training Set

s/n	INPUT DATA				OUTPUT DATA
	σ . (KN/m^2)	μw (KN/m^2)	μa (KN/m^2)	σ (KN/m^2)	χ
1	177	97.5	58.52	248.39	0.33
2	173.1	101.43	60.86	250.99	0.42
3	169.3	105.23	63.14	250.96	0.44
4	165.6	108.93	65.36	254.92	0.55
5	162	112.53	67.52	256.08	0.59
6	158.5	116.03	69.62	256.89	0.62
7	155.1	119.43	71.66	259.72	0.69
8	151.8	122.73	73.64	261.28	0.73
9	148.6	125.93	75.56	269.99	0.91
10	145.5	129.03	77.42	261.66	0.77
11	142.5	132.03	79.22	268.19	0.88
12	139.6	134.93	80.96	264.28	0.81
13	136.8	137.73	82.64	231.00	0.21
14	134.1	140.43	84.26	237.46	0.34
15	131.5	143.03	85.82	224.19	0.12
16	129	145.53	87.32	244.26	0.48
17	126.6	147.93	88.76	241.39	0.44
18	124.3	150.23	90.14	241.48	0.45

19	122.1	152.43	91.46	253.80	0.66
20	120.0	154.53	92.72	244.24	0.51
21	118.0	156.53	93.92	246.98	0.56
22	116.10	158.43	95.06	225.10	0.22
23	114.3	160.23	96.14	229.02	0.29
24	112.6	161.93	97.16	241.49	0.49
25	111.0	163.53	98.12	238.55	0.45
26	109.5	165.03	99.02	243.51	0.53
27	108.1	166.43	99.86	244.57	0.55
28	106.8	167.73	100.64	229.58	0.33
29	105.60	168.93	101.36	273.85	0.99
30	104.50	170.03	102.02	274.53	1.00
31	99.79	174.74	104.84	285.02	1.15
32	103.69	158.43	95.06	283.03	1.33
33	107.49	154.53	92.72	337.43	2.22
34	111.19	150.73	90.44	261.92	1.00
35	114.79	147.03	88.22	231.24	0.48
36	118.29	143.43	86.06	210.67	0.11
37	121.69	139.93	83.96	236.98	0.56
38	124.99	136.53	81.92	224.93	0.33
39	128.99	133.23	79.94	219.85	0.22
40	131.29	130.05	78.03	234.29	0.48
41	134.29	126.97	76.18	239.42	0.57
42	137.19	123.98	74.39	240.34	0.58
43	139.99	121.10	72.66	241.23	0.59
44	142.69	118.33	71.00	243.51	0.63
45	145.29	115.65	69.39	245.67	0.67
46	147.79	113.08	67.85	232.38	0.37
47	150.19	110.61	66.37	236.03	0.44
48	152.49	108.25	64.95	236.93	0.45
49	154.69	105.98	63.59	270.00	1.22
50	156.79	103.82	62.29	274.31	1.33
51	158.79	101.77	61.06	265.01	1.11
52	160.69	99.81	59.89	233.75	0.33
53	162.49	97.96	58.78	256.92	0.91
54	162.49	97.96	58.78	239.68	0.47
55	164.19	96.16	57.69	246.89	0.65
56	165.79	94.51	56.71	247.45	0.66
57	167.29	92.97	55.78	234.97	0.33
58	168.69	91.53	54.92	244.11	0.56
59	170.00	90.18	54.11	246.47	0.62
60	171.19	89.57	53.74	247.50	0.63
61	202.00	153.70	92.22	362.46	1.11
62	206.90	149.60	89.76	376.25	1.33
63	211.70	145.40	87.27	415.84	2.01
64	216.40	141.10	84.66	385.72	1.50
65	221.00	136.70	82.02	352.23	0.90
66	225.50	132.87	79.72	350.39	0.85
67	229.90	128.93	77.36	381.52	1.44
68	234.20	125.09	75.05	356.79	0.95
69	238.20	121.51	72.91	337.84	0.55
70	242.50	117.66	70.60	327.69	0.31

2.2 Data Modeling (Practical Phase)

The neural network package in MATLAB R2007b version was used in this project. The Back-propagation feed-forward algorithm was used and the scale conjugate gradient technique selected because it was found to learn the data better than other ones. The error range or goal was maintained at 1e-5 at about 70000 epochs (iterations) per set of input and output data. The network was set to show the result of the training at every 2000 iterations and a gradient of 1e-6 maintained.(Goh and Goh 2007).

During the training, the input variables and the output variables were fed into the network and the training commenced following the program written as follows:

- computation of input variables as thus

P = [σ';μw ; μa;σ];
p1=[177;97.53;58.52;248.39];
p2=[173.1;101.43;60.86;250.99];
, = ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
p70=[242.50;117.66;70.60;327.69];

- computation of target variables as thus

t=[χ]
t1=[0.33];
t2=[0.42];
, = ,,,,,,,,,,,,
, = ,,,,,,,,,,,,
t70=[0.31];

- registration of input and testing data as thus

p = {p1……………..Pn};
Tt= {t1……………..Tn};

- however, the scg (scale conjugate gradient) algorithm was used to train the above registered data as below

net.trainParam.epochs 70000
net.trainParam.show 2000
net.trainParam.goal 1e-5
net.trainParam.time
net.trainParam.min_grad 1e-6(default)
net.trainParam.max_fail 5
net.trainParam.sigma 5.0e-5(default)
net.trainParam.lambda 5.0e-7(default)
net = train (p, t);

- however to test the network after the training, the following were written in the MATLAB command window as well to identify estimated output being given by the network.

y = sim(net,p)
for i = 1: length (y) disp (y {i}), end;
p71 = [σ';μw ; μa;σ];
, = ,,,,,,,,,,,,,,,,,,,,,,,
, = ,,,,,,,,,,,,,,,,,,,,,,,,
p100=[σ';μw ; μa;σ];
y = sim (net, P71)
y = sim (net, P72)
, = ,,,,,,,,,,,,,,,,,,,,
, = ,,,,,,,,,,,,,,,,,,,,,
y = sim (net, P100)

2.3 Network Configuration

The network configuration used is 4-10-1 which means 4 neurons at the input layer, 10 neurons at the hidden layer and 1 neuron at the output layer as shown in Figure 3.(Haj-Ali et al 2001, Habibagahi and Bamded 2003)

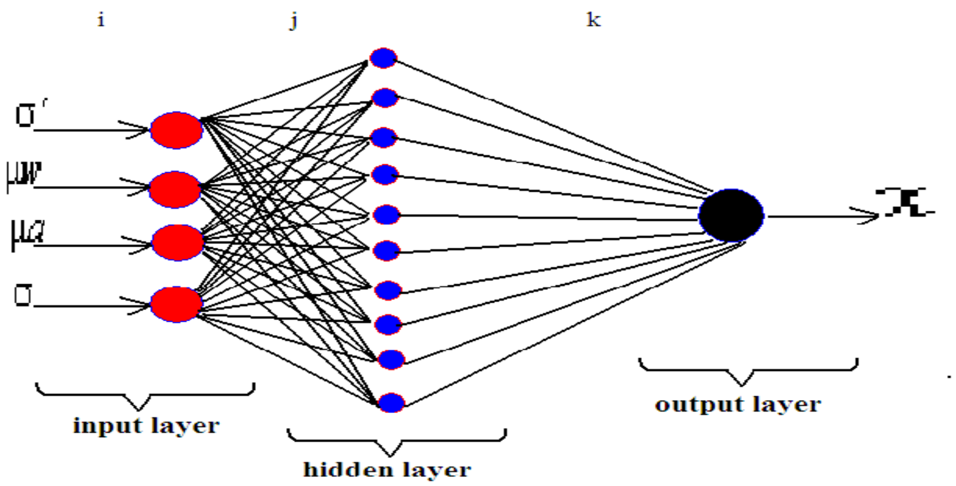

Figure 3 Showing the network configuration used

2.4 Data Preparation

Since the ANN modeling philosophy is similar to a number of conventional statistical models in the sense that both are attempting to capture the relationship between a historical set of model inputs and corresponding outputs which conform to the linear equation y=mx + c (where m & c are slope and intercept respectively). This project assumes a set of *x*-values and corresponding y values in 2 dimensional space for the generated data, where $y = f(x)$. The objective is to find the unknown function *f*, which relates the input variable *x* to the output variable *y* and knowing the pattern in which the variables are related.

III. RESULTS AND DISCUSSION

3.1 Results

Various experiments were performed and the sizes of the training and testing sets were determined by taking into consideration the classification accuracies. The data set was divided into two separate data sets – the training data set (70 subjects) and the testing data set (30 subjects). The training data set was used to train the NN, whereas the testing data set was used to verify the accuracy and the effectiveness of the trained network model for the prediction of effective stress parameter. The partition of testing data used is as shown in the table below.

Table 2 Partition of Testing Data Used

S/N	INPUT DATA				OUTPUT DATA
	σ' (KN/m^2)	μw (KN/m^2)	μa (KN/m^2)	σ (KN/m^2)	χ
71	58	111	66.6	147.24	0.51
72	61	107	64.2	146.17	0.49
73	65	103	61.8	148.64	0.53
74	69	101	60.6	151.42	0.54
75	73	97.67	58.6	157.38	0.66
76	76	95.42	57.25	141.65	0.22
77	79	93.17	55.90	152.42	0.47
78	83	90.17	54.10	154.41	0.48
79	86	87.92	52.75	141.02	0.44
80	89	85.67	51.40	155.14	0.43
81	105	130	78	217.32	0.66
82	106	128	76.8	214.54	0.62
83	108	131	78.6	213.32	0.51
84	109	131	78.6	225.33	0.72
85	111	126	75.6	236.99	0.98
86	112	122	73.2	190.57	0.11
87	114	123.16	74	193.89	0.12
88	116	121.14	72.68	199.83	0.23
89	118	119.64	71.78	202.22	0.26
90	120	117.88	70.73	213.83	0.49
91	122	116.12	69.67	212.11	0.44
92	124	114.36	68.62	243.39	1.11
93	126	112.60	67.56	253.46	1.33
94	129	109.96	65.98	258.31	1.44

95	131	108.20	64.92	282.51	2.01
96	134	105.56	63.34	244.20	0.95
97	136	103.80	62.28	233.57	0.85
98	139	101.16	60.69	218.71	0.47
99	142	98.52	59.11	222.79	0.55
100	145	95.88	57.53	215.95	0.35

It was observed in this study that there is possibility of getting different results with the same set of input parameter for the network code, but what was put into consideration was meeting the performance goal. Irrespective of the numbers of time the network was trained, the results were nearer in values and most accurate result was chosen for analysis.

3.2 Performance Analysis

The network performs best when the trainscg was selected and the calcgrad for each epoch before the performance goal was met as shown in Figure 4.(Mathworks 2007)
TRAINSCG-calcgrad, Epoch 0/70000, MSE 1.965/1e-005, Gradient 2.65738/1e-010
TRAINSCG-calcgrad, Epoch 537/70000, MSE 0.0349584/1e-005, Gradient 0.184569/1e-010
TRAINSCG-calcgrad, Epoch 653/70000, MSE 0.00340132/1e-005, Gradient 0.054656/1e-010
TRAINSCG-calcgrad, Epoch 1725/70000, MSE 0.000459539/1e-005, Gradient 0.017362/1e-010
TRAINSC-calcgrad, Epoch 20000/70000, MSE 0.00014733/1e-005, Gradient 0.0062155/1e-010
TRAINSCG-calcgrad, Epoch 53089/70000, MSE 9.6226e-005/1e-005, Gradient 0.00343/1e-010
TRAINSCG,performance goal met.

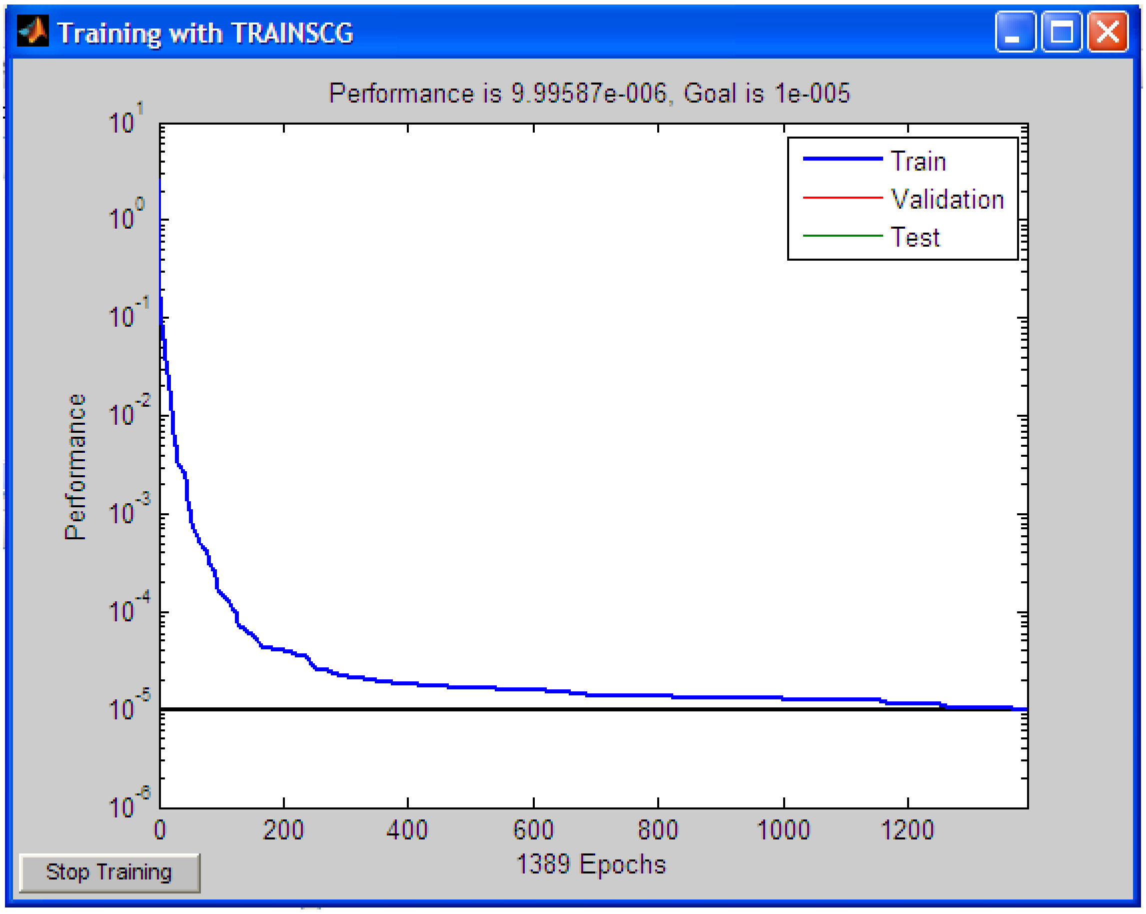

Figure 4 showing the graph of the performance of the training process

Table 3 Showing the linear relationships and equation between variables

	Data Range	Variables	Equations
INPUT	1-30	$(\boldsymbol{\sigma}',\boldsymbol{\mu w})$ $(\mu w , \mu a)$ (σ , χ)	$\mu w = -\sigma' + 274.53$ $\mu a = 0.6\mu w + 0.002$ $\chi = 0.02\sigma - 3.45$
	31-60	$(\boldsymbol{\sigma}',\boldsymbol{\mu w})$ $(\mu w , \mu a)$ (σ , χ)	$\mu w = -1.07\sigma' + 270.79$ $\mu a = 0.6\mu w + 0.001$ $\chi = 0.02\sigma - 3.66$
	61-70	$(\boldsymbol{\sigma}',\boldsymbol{\mu w})$ $(\mu w , \mu a)$ (σ , χ)	$\mu w = -0.89\sigma' + 334.55$ $\mu a = 0.6\mu w + 0.01$ $\chi = 0.02\sigma - 5.88$
OUTPUT	71-80	$(\boldsymbol{\sigma}',\boldsymbol{\mu w})$ $(\mu w , \mu a)$ (σ , χ)	$\mu w = -0.78\sigma' + 155.04$ $\mu a = 0.6\mu w + 0.012$ $\chi = 0.01\sigma - 1.37$
	81-100	$(\boldsymbol{\sigma}',\boldsymbol{\mu w})$ $(\mu w , \mu a)$ (σ , χ)	$\mu w = -0.88\sigma' + 223.89$ $\mu a = 0.6\mu w + 0.03$ $\chi = 0.02\sigma - 3.96$

3.4 Comparison of Results

Bishop's stress equation is used as the benchmark in this project for generating both the training and testing data. Results from the output of testing are compared with the target by finding their correlation coefficient and calculations are summarized the table below.

Table 4 Showing relationship between output and target

s/n	Output(x)	Target(y)	XY	X^2	Y^2
1	0.51	0.5719	0.29	0.26	0.33
2	0.49	0.5548	0.27	0.24	0.31
3	0.53	0.5948	0.31	0.28	0.35
4	0.54	0.6020	0.33	0.29	0.36
5	0.66	0.7159	0.47	0.44	0.51
6	0.22	0.2974	0.07	0.05	0.09
7	0.47	0.5316	0.05	0.22	0.28
8	0.48	0.5404	0.06	0.23	0.29
9	0.44	0.1750	0.08	0.19	0.03
10	0.43	0.4975	0.01	0.18	0.25
11	0.66	0.6641	0.44	0.44	0.44
12	0.62	0.6246	0.39	0.38	0.39
13	0.51	0.5116	0.26	0.26	0.26
14	0.72	0.7234	0.52	0.52	0.52
15	0.98	1.0072	0.99	0.96	1.01
16	0.11	0.1163	0.01	0.01	0.01
17	0.12	0.1357	0.02	0.01	0.02
18	0.23	0.2338	0.05	0.05	0.05
19	0.26	0.2635	0.07	0.07	0.07
20	0.49	0.4953	0.04	0.24	0.25
21	0.44	0.4448	0.19	0.19	0.20
22	1.11	1.1183	1.24	1.23	1.25
23	1.33	1.3340	1.77	1.77	1.78
24	1.44	1.4403	2.07	2.07	2.07
25	2.01	1.9648	3.95	4.04	3.86
26	0.95	1.1165	1.06	0.90	1.25
27	0.85	0.8568	0.73	0.72	0.73
28	0.47	0.4756	0.22	0.22	0.23
29	0.55	0.5562	0.31	0.30	0.31
30	0.35	0.3623	0.13	0.12	0.13
Σ	18.97	19.53	17.20	16.88	17.63

To find the correlation coefficient r between the both, the formula

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\left(\sqrt{n(\Sigma x^2) - (\Sigma x)^2}\right)\left(\sqrt{n(\Sigma y^2) - (\Sigma y)^2}\right)} = \frac{30\,(17.2) - (19.53)(18.97)}{\left(\sqrt{30\,(16.88) - 18.97^2}\right)\left(\sqrt{30\,(17.63) - 19.53^2}\right)}$$

$r = 0.99$

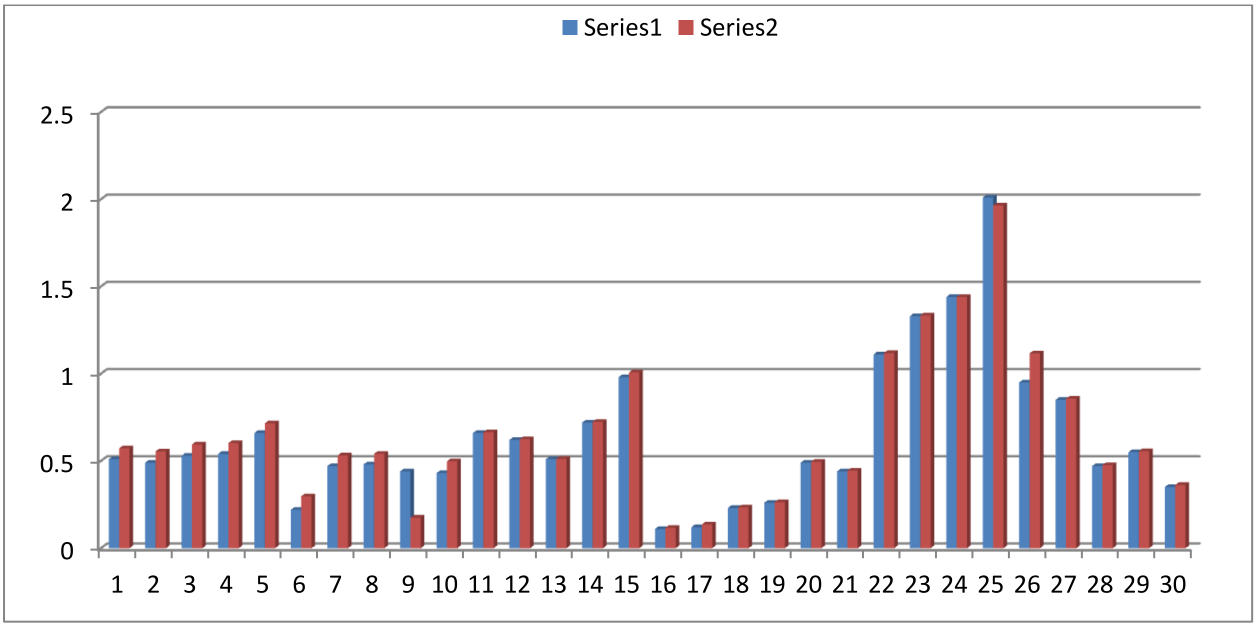

Figure 5 Comparing output and target

Since the value of r=0.99 which is close to 1 and mere looking at the chart it can be deducted that strong correlation exists betweent the two variables and ANN predicts the effective stress parameter better

IV. CONCLUSION

Prediction of effective stress parameter of unsaturated soil is an important thing in the field of geotechnical engineering. It allows for understanding of the strength of soil whether for research purpose or to check real life problems. Within the framework of this research, the following facts emerged;

- It is justified that ANN is an efficient tool in predicting stress parameter of unsaturated soil under different pore pressure conditions. This in turn, will help in reducing unnecessary funds in doing laboratory experiments.
- Neural Network technique plays an important role in dealing with uncertainty when making decisions in geotechnical problems to check whether the site specification is been met by comparing estimated data with established one.
- The neural network system is based on the learning abilities of the data fed into it and uses the prior Knowledge to predict the output.
- Experimental result indicates that the technique is workable with accuracy greater than 90%.
- Neural networks are efficient at analyzing problems where data are incomplete or fuzzy, and accurate predictions are sought more heavily than explanations.

REFERENCES

[1]. Abu-Kiefa, M. A. (1998). "General regression neural networks for driven piles in cohesionless soils." *J. Geotech. & Geoenv. Engrg.*, ASCE, 124(12), 1177-1185.

[2]. Agrawal, G., Chameau, J. A., and Bourdeau, P. L. (1997). "Assessing the liquefaction susceptibility at a site based on information from penetration testing." In: *Artificial neural networks for civil engineers: fundamentals and applications*, N. Kartam, I. Flood, and J. H. Garrett, eds., New York, 185-214.

[3]. Agrawal, G., Weeraratne, S., and Khilnani, K. (1994). "Estimating clay liner and cover permeability using computational neural networks." *Proc., First Congress on Computing in Civil Engrg.*, Washington, June 20-22.
[4]. Goh, A. T., and Goh, S. H. (2007). "Support vector machines: Their use in geotechnical engineering as illustrated using seismic liquefaction data." Computers and Geotechnics, 34(5),410-421.
[5]. Habibagahi, G., and Bamdad, A. (2003). "A neural network framework for mechanical behavior of unsaturated soils." Canadian Geotechnical Journal, 40(3), 684-693.
[6]. Haj-Ali, R., Pecknold, D. A., Ghaboussi, J., and Voyiadjis, G. Z. (2001). "Simulated micromechanical models using artificial neural networks." Journal of Engineering Mechanics, 127(7), 730-738.
[7]. MathWorks, Inc., 2007: MATLAB r2007b. The MathWorks, Inc Microsoft Encarta 2009