Scientific Research and Essays Vol. 6(16), pp. 3406-3417, 19 August, 2011
Available online at http://www.academicjournals.org/SRE
DOI: 10.5897/SRE11.311
ISSN 1992-2248 ©2011 Academic Journals

Full Length Research Paper

Artificial neural networks for mechanical strength prediction of lightweight mortar

Razavi S.V[2]*, Jumaat M. Z[1], Ahmed H. El-Shafie[3], Pegah Mohammadi[2]

[1]Civil Engineering Department, University Malaysia, Malaysia.
[2] Department of Civil Engineering, Science and Research Branch, Islamic Azad University (SRBIAU), Hesarak, Tehran, I.R. Iran.
[3]Civil Engineering Department, Universiti Kebangsaan Malaysia (UKM).

Accepted 31 May 2011

In this paper, the practical results of mechanical strength of different lightweight mortars made with 0, 5,10, 15, 20, 25, 30, 35, 40, 45, 50,55, 60, 65, 70, 75, 80, 85, 90, 95 and 100% of scoria instead of sand and 0.55 water-cement ratio and 350 kg/m^3 cement content have been used to generate artificial neural networks (ANNs). Totally, 52 feed-forward back-propagation neural networks (FFBNN) with different parameters have been investigated in the case of 80 data for training, 15 data for verifying, and 10 data for testing. The performance for producing networks was evaluated by root mean squared error (RMSE) and the correlation coefficient between data. The two selected networks, N1 (Net Architecture 2-10-2) and N2 (Net Architecture 2-10-5-2) had (0.020, 0.027) and (0.017, 0.018) as (Training, Testing) RMSE set and 0.997 and 0.982 as testing correlation coefficient.

Key word: Scoria, artificial neural networks, feed-forward back-propagation neural networks.

INTRODUCTION

Mortar is a material used in masonry for joining construction blocks together, filling the gaps between them, plastering, making partitions, and pipeline systems covering on the floor. Based on the bulk density of the mix in its dry state, mortars may be lightweight or heavy weight. It is important to make a lightweight mortar with a low density and acceptable mechanical strength to reduce the weight of buildings (Yu-Ling et al., 2008), facilitate transportation, provide thermal insulation (Marcos and García-Ruizand, 2008),etc. There are many ways to produce a lightweight mixture:

1. Increasing of mixture volume: After the adding of lime and aluminium powder to silica, water and cement, hydrogen gas will be produced that will cause an increase in the mixture volume and decrease the specific weight. For example, Siporex and Itong (Short and Kinniburgh, 1978).
2. Using natural lightweight aggregates such as Perlite (Sanahi, 1978), Vermiculite (Short and Kinniburgh, 1978), Diatoms (Short and Kinniburgh, 1978), Pumice (Shorabi and Rigi, 2005) and Scoria (Famili, 1997).
3. Using artificial aggregates, such as expanded clay (Lica) (Merikallio and Mannonen, 1996) and light residue of blast furnace (Shideler, 1975).
4. Using rice stalks for aggregate can make a lightweight mixture (Abedi, Mollahi, 2005).

A decrease in density and increase in workability (Marcos and García-Ruiz, 2009), and acceptable mechanical strength are the major issues in producing a lightweight mixture. For example, an experimental study by Gadea et al. (2010) on using rigid polyurethane foam instead of sand to make lightweight mortar shows a decrease in the density and mechanical properties and an increase in the workability.

Sari et al. (2004) have done experimental studies on lightweight concrete made of scoria, and have also experimented with the effect of plasticizer and air entrained agents to improve the workability of concrete. They used cubic samples that showed a minimum compressive strength and a density of 6.56 N/mm^2 and 1300 kg/m^3, respectively. Topcu (1996) used volcanic slag as coarse aggregate to study the properties of semi-lightweight concretes. He reported that volcanic slag can decrease the specific gravity of concrete as much as 20%

*Corresponding author. E-mail: Vahidrazavy@yahoo.com.

Table 1. The history of neural networks.

Researcher	Date	Description
McCulloch and Pitts	1942	The neural networks can calculate each kind of arithmetic and logical function.
Hebb	1949	The First Rule for Training of Biological Neuron.
Rosenblatt	1958	Persepteron a mechanic that can learn how to arrange the information by using weight comparison.
Widrow and Hoff	1960-1962	Adlain linear comparative neural network and the rule of sum of minimum square error
Minsky and Papert	1969	Expression of Persepteron theoretical limit for general computers
Kohonen	1972	Introduce of the NN that can act as saving elements.
Grossberg	1972	To explain the neural networks that can be arranged by itself.
Hapfred	1982	Using random mechanism to explain the work of an extended series
Rumelhart	1986	Introduction of back propagation networks.

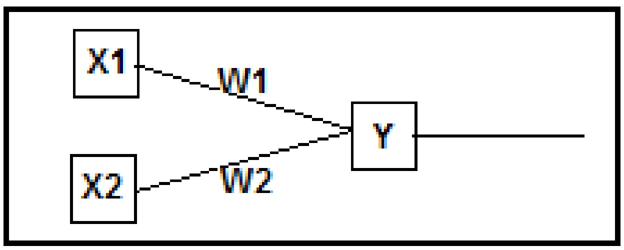

Figure 1. A simple ANN.

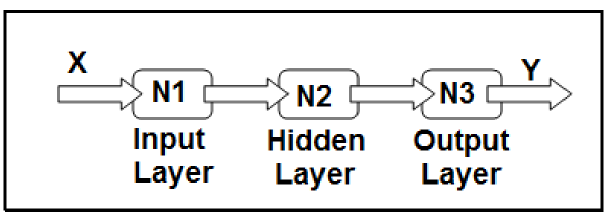

Figure 2. A three layer multi input layer.

of normal concrete. However, decreased workability and low strength were some disadvantages of using volcanic slag as aggregate in concrete. Unal et al. (2007) produced block elements with diatomite with different aggregate granulometries and cement contents. According to the result of mechanical and physical properties in this study, using diatomite in lightweight concretes can be used in construction to reduce the service load and obtain high insulation in buildings.

All of the mentioned studies need to take time to do experimental work, testing, and evaluation. Therefore, making an analytical method based on the relationship between the experimental results and that provides accurate output will be useful. In recent years, the application of artificial neural networks (ANNs) in different parts of civil engineering such as analysis, design, and concrete technology especially has been accomplished. For example, Yeh (1998), Kasperkiewics (1995), Lai (1997) and Lee (2003) applied the NN for predicting properties of conventional concrete and high performance concretes. ANNs use mathematical formulations to develop a nervous system operation that it is used to learn patterns and relationships in data. The history of neural networks is shown in Table 1. The most usual type of neural net is a single layer; the structure of a simple artificial neural net is shown in Figure 1. It shows a simple artificial neural net with two input neurons (x1, x2) and one output (y). The organized weights are given by w1 and w2. In a single layer net there is a single layer of weighted interconnections. The ANN may also be multi layer; the structure and formula of a multi input neuron is shown in Figure 2 and Equation 1, respectively.

$$n = \sum_{i=1}^{R} X_i W(1,i) + b = WX + b \tag{1a}$$

$$Y = f(WP + b) \tag{1b}$$

Where W = weight or the effect of interior connection, n= the conclusion of input layer that is defined as pure input.

Guang and Zon (2000), by using multi-layer feed-forward neural networks, proposed a method to predict the 28 day compressive strength of concrete. Ilker and Mustafa (2007) studied using ANNs to predict (ACC) concrete characteristics. In this project, 45 data results were used for neural network processing 23 data were selected for training and another 22 data were used for testing. The applied ANN had 7 input layers, 7 neurons in the first hidden layer and 8 neurons in the second hidden layer, and 4 parameters in the output layer. The results after training and testing showed only 6% difference between the ANN and experimental testing results. Dias and Pooliyadda (2001) showed that back propagation neural networks are suitable to predict the strength and slump of ready mixed concrete and high strength concrete. In this study, the compressive strength (CS) and tensile strength (TS) testing prediction of lightweight mortar using

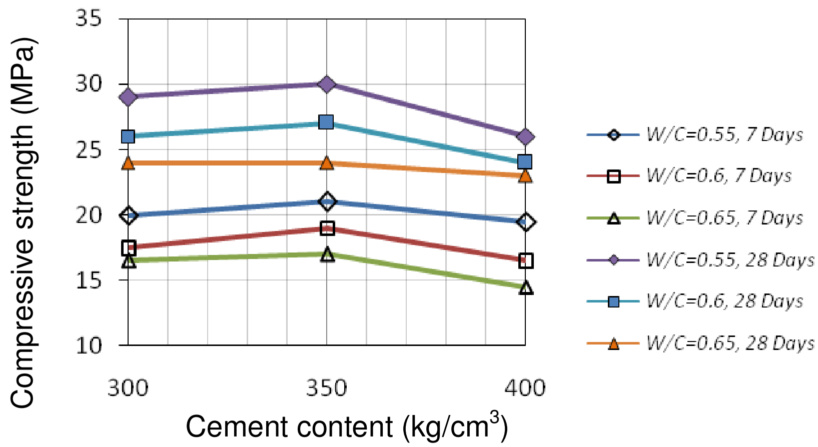

Figure 3. Relationship between cement content and compressive strength with 35~50% scoria (lightweight concrete).

Table 2. The schedule of construction of light weight mortar.

Parameter	Mortar samples
Scoria instead sand in lightweight mortar (%)	0 , 5, 10,15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95 and 100
Water-cement ratio	0.55
Cement content in compressive sample (kg/m^3)	350
Cement Content in Tensile Sample (kg/m^3)	350
Curing time (Days)	3, 7,14, 28 and 90

different percentages of scoria instead of sand will be the final results. This scoria has a white to light grey colour with irregular open and closed pores, a rough surface and angular particles. This type of scoria has been created by the accumulation of volcanic ash and slight cooling accompanied with bubbles resulting from vapour and existing gases. The specific gravity of aggregates with regard to its porosity is less than 1 g/cm^3.

STRUCTURE OF TEST

In order to study the parameters below, 210 mortar samples in the shapes of 15×15×15 (cm) cubes and 15×30 (cm) cylinders were made to determine the compressive strength and tensile strength, respectively. The water-cement ratio was 0.55 and the cement content was 350 kg/m^3. The materials mentioned above, were made according to the results of 288 lightweight concrete samples, which were made from scoria instead of a percent of sand with 0.55, 0.60 and a 0.65 water-cement ratio; and 300, 350 and 400 kg/m^3 cement content. The relationship between the compressive strength and the cement content for 35~50, 55~70 and 100% of scoria instead of sand for lightweight concrete are shown (Figures 3, 4 and 5).

As we can see, the maximum compressive strength was with a 0.55 water cement ratio and 350 kg/m^3 cement concrete. By increasing the water cement ratio and cement content, the mortar compressive strength was decreased (Joachim, 1999). Investigation of different percentages of scoria instead of sand was the main objective of the current study. The schedule of the experimental work is shown in Table 2. We consider different percentages of scoria instead of sand from 5 to 100% by increasing 5% in each step, which were studied to find the compressive and tensile strength after 3, 7, 14, 28 and 90 curing days. All samples were made of normal water and the average for the compressive strength and tensile strength was obtained from three similar samples.

In this study, the mixture of absolute volume method was used. Knowing the water and cement amounts and by using the absolute volume method (shown as Equation 2) the amount of aggregates can be extracted. It is assumed that the volume of compacted mortar is equal to the total absolute volumes of its constituents.

$$\boxed{\frac{C}{\gamma_C} + \frac{W}{\gamma_W} + \frac{A}{\gamma_A} = 1} \qquad (2)$$

In this formula C is the amount of cement, W is the amount of water and A is the amount of aggregate (sand

Table 3. The calculated amounts of sand and scoria.

Scoria (%)	0	5	10	15	20	25	30	35	40	45
Sand (Kg/m^3)	1543.2	1339.1	1167.6	1021.4	895.3	785.4	688.7	603.1	526.7	458.1
Scoria(Kg/m^3)	0.0	70.5	129.7	180.2	223.8	261.8	295.2	324.7	351.1	374.8
50	55	60	65	70	75	80	85	90	95	100
396.2	340.1	288.9	242.0	199.0	159.3	122.7	88.7	57.0	27.6	0
396.2	415.6	433.3	449.5	464.3	478.0	490.7	502.5	513.4	523.6	533.1

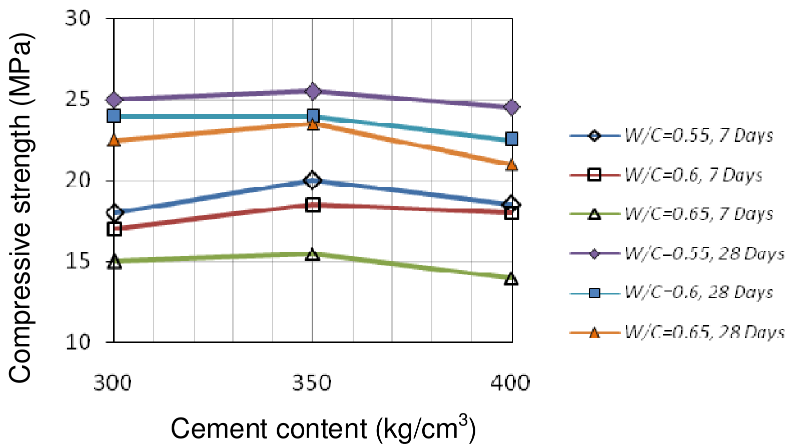

Figure 4. Relationship between cement content and compressive strength with 55~70% scoria (light weight concrete).

+ scoria) in kg/m^3 of mortar. The calculated amounts of sand and scoria in each mixture are shown in Table 3. The specific gravity of the sand, scoria, cement, and water was 2200, 760, 3300 and1000 Kg/m^3, respectively.

EXPERIMENTAL RESULTS

First, calculated amounts of sand, scoria and cement were weighed and mixed for about 60 s. Then, after one quarter of the mixture time, water was added and the mixing process continued for 180 s. Subsequently, the casting action for cubic samples 15×15×15 and cylinder samples 15×30 in 3 layers took place for later use to determine the compressive strength and tensile strength, respectively. Each layer was vibrated by shaking the table for 10 s to compact samples equally. After keeping in the lab for 24 h, the samples were opened and cured in water for 3, 7, 14, 28 and 90 days. Before testing, all samples were weighed after removal from the water tank and kept in the laboratory for a few hours for drying. The specific gravity of all the samples is indicated in Figure 6. The mortars made with ≥ 60% scoria instead of sand with specific gravity ≤1300 Kg/m^3 (BS EN, 1996) are lightweight mortar.

Increasing the percentage of scoria instead of sand from 0 to 100% caused 49% decrease in the specific gravity from 2051 to 1043 Kg/m^3. The relationship between mechanical strength and the percentage of scoria instead of sand are shown in Figures 7, 8 and 9. It is observed that by increasing the percentage of scoria instead of sand in mortar from 0 to 100 the compressive strength and tensile strength of mortar decrease. The reasons for this are the low mechanical strength of scoria compared to sand (Aydin et al., 2010) and the increase of porosity degree on aggregate (Lo and Cui, 2004; Wasserman and Bentur, 1996). The result sets for the CS and TS after 28 days curing in the concrete lab were (202, 24) Kg/cm^2 for 0% scoria instead of sand and (186.8, 20) Kg/cm^2 for 100% scoria. By replacing the sand in the mortar with scoria, the results have shown 7.52 and 16.67% decrease in the compressive and tensile strength, respectively.

ARTIFICIAL NEURAL NETWORKS

Network model

Network type

In this research, a multilayer feed-forward neural network

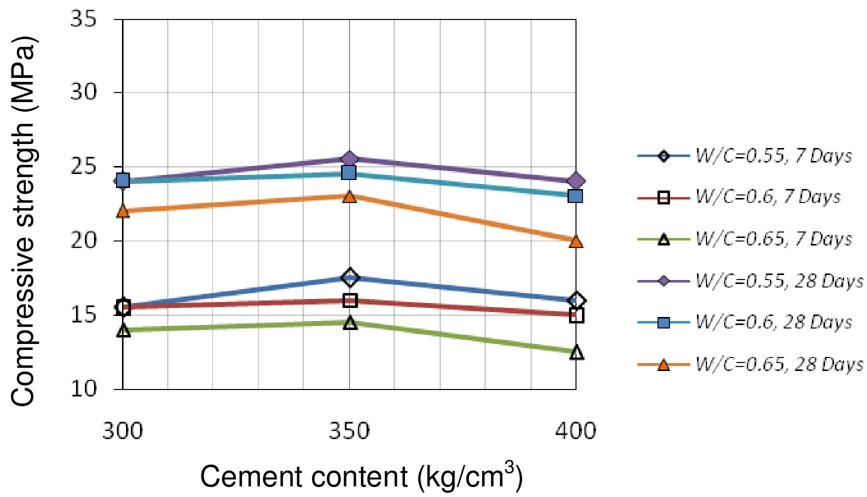

Figure 5. Relationship between cement content and compressive strength with 100% scoria (light weight concrete).

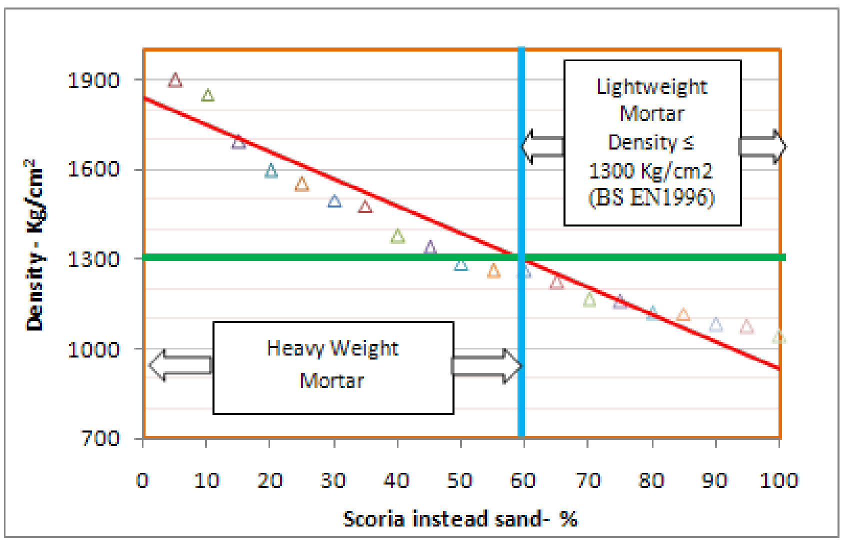

Figure 6. The limitations of lightweight mortar.

by using the back-propagation learning algorithm in training phase was adopted. The back-propagation neural networks (BPNNs) consist of an input layer, an output layer, and one or more hidden layers (Rajagopalan et al., 1973). A back-propagation network normally starts out with a casual set of weights. The weight will be changed in each process of input-output pair. Each pair involves two phases (Ayman, 2005): a forward pass and a backward pass. In the forward pass a sample input presents to the network to process, follow, and reach to the output layer. In the backward pass, the output due to the forward pass compares with the known target and computes the evaluated error for the output units. The back-propagation algorithm revises the weights in each input-output set by propagating the error back to the network using a widely used learning mechanism (Rajagopalan et al., 1973) to change the weights and biases. The Input-output pairs are used to train a network until the network can approximate a function (Haykin, 1999). After training the generated network can be tested for the new input-output pairs.

Training and learning function

Training and learning functions are numerical measures used for automatically changing the weights and biases of the system. The training function applies a comprehensive algorithm that concerns all weights and biases of a network. The learning function can be used to

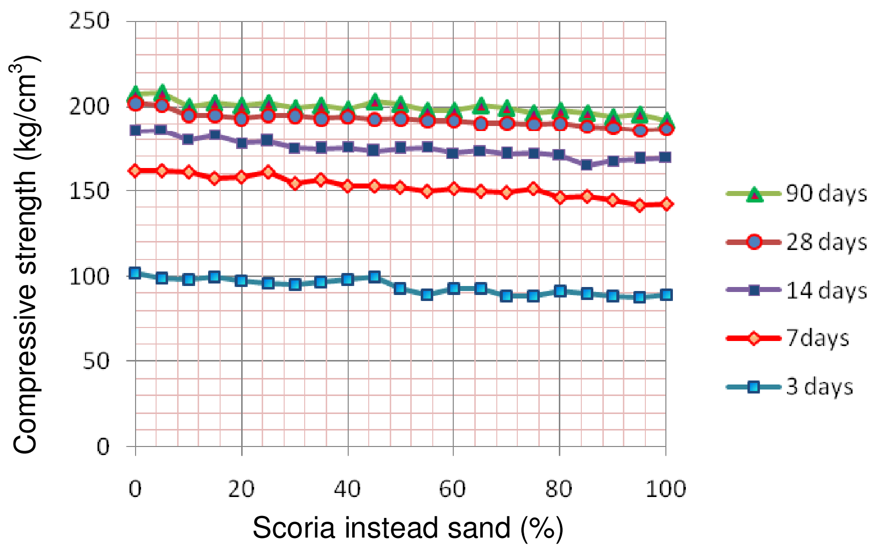

Figure 7. Relationship between compressive strength and scoria percent instead of sand

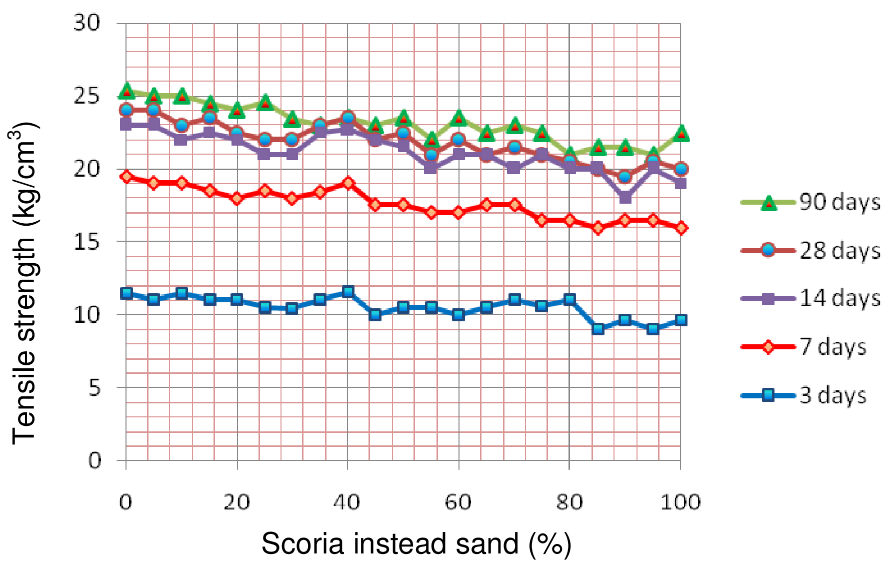

Figure 8. Relationship between tensile strength and scoria percent instead of sand.

apply individual weights and biases inside a system. In this part, different training and learning functions were tested to select the best training and learning function.

Number of hidden layers

The neuron layers linking the input and the output layers are defined as hidden layers (Reda, 2003).There are no theories to identify how many hidden layers are needed to estimate any given function. Most of the time, the linear and generalized linear model (simple network) are applicable for a wide range of purposes (McCullagh and Nelder, 1989). If the number of input layers is one, there appears to be no improvement by using more than one hidden layer in the network. However, when the number of input layers is more than one input, the case becomes much more complex. The number of hidden layers and neurons can be randomly selected and changed to find the optimum results in each try. Although, increasing the number of hidden layers improve the generalization capacity (Hornik, 1990). It has been detailed that one or two hidden layers with a randomly large number of neurons may be enough to approximate any function

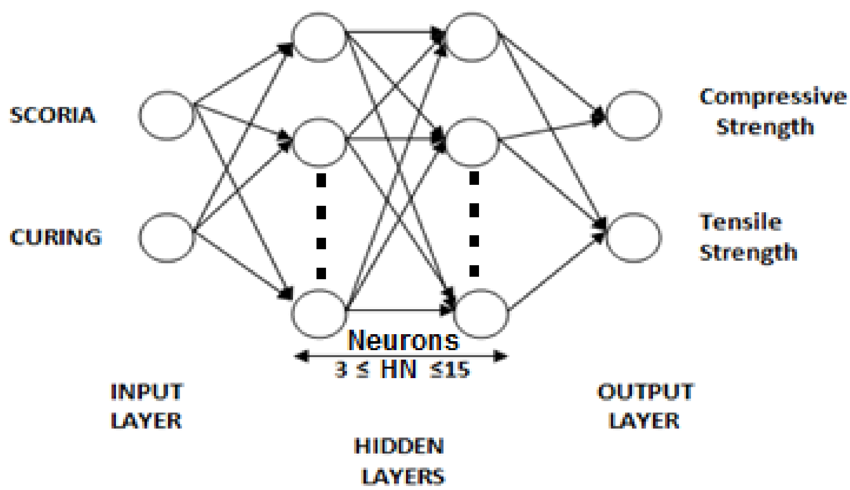

Figure 9. Network architecture.

(Haykin, 1999). In addition, it is confirmed that networks with a single hidden layer with a sufficient number of nodes can be generated for a functional relationship (Mukherjee and Deshpande, 1995). In the current research, 1 and 2 hidden layers with different neurons have been tried for generating a network.

Transfer function

The multilayer feed-forward neural networks (MFNNs) consisted of input, hidden layer, and output layer. A suitable transfer function should be chosen for numerical representation of the relation between the input and output of a system. Over the last few years, many transfer functions have been initiated by researchers using ANNs. Only three of these transfer functions are usually used (Ripley, 1996):

Linear: The output action is proportional to the total weight of output.
Threshold: The data in the output place at one of two points, depending on whether the whole input is bigger than or fewer than some entrance amount.
Sigmoid: The output differs constantly but not linearly as the changes of input data. Sigmoid units provide a better similarity to actual neurons. A log-sigmoid/Purelin unit has been used as the transfer function in the output layer.

Performance function

The mean square error or MSE has been used to estimate different values between the estimator and the actual value of the estimated amount. MSE is a risk function, corresponding to the expected value of the squared defect reduction or quadratic loss. This algorithm is an instance of supervised training, in which the learning rule has been supplied with a number of examples of the preferred network actions. The performance of an ANN depends on both the transfer function and the weights that have been processed for the elements.

Network analysis and results

Network analysis

The network architecture is the first important stage to construct network modelling. Normally, a trial and error method is assumed to select the optimum net architecture. This is because there are not any created rules to define the net architecture in back-propagation neural network (Lin et al., 2003). A net architecture is a frame work, as shown in Figure 9, consisting of an input layer, hidden layers, and an output layer. There is no common statute to select the number of neurons in a hidden layer (Ahmet et al., 2006). The number of neurons in a hidden layer is variable ≤15. The minimum number of nodes in the hidden layer is defined by below formula (Carpenter and Barthelemy, 1994):

$$HN = IN + 1$$

Where, HN is the number of nodes in the hidden layers and IN is the number of nodes in the input layer. Therefore, the number of neurons in the hidden layers varies from 3 to 15.

The number of training data is the second important part to define network modelling. Although increasing the number of training data increases the time required to train the network (Ayman, 2005), the number of prototypes in the training step considerably influence the ability of a network generation. The minimum number of

Table 4. ANNs information.

The number of data	105 data = 80 (Training) + 15 (Validation) + 10 (Testing)
Input layer	(Scoria - %) and (Curing - day)
The number of Neurons in hidden layers(HNs)	3≤HNs≤15
Output layer	Compressive and tensile strength
Net architecture	(2-HN_1-2) and (2-HN_1-HN_2-2)
Network type	Multilayer feed-forward
Net algorithm	Back-propagation
Training function	Trainbr and Trainlm
Learning function	LEARNGD and LEARNGDM
Output transfer function	Log-Sigmoid and Purelin
Hidden transfer function	Log-Sigmoid and Tangent Sigmoid
Performance function	RMSE

training data sets, should be greater than the number of undetermined parameters (Carpenter and Hoffman, 1995) and (Oreta and Kawashima, 2003a), is defined by the formula below (Oreta and Kawashima, 2003b):

NT = NH*(NI+1)+NO*(NH+1)

Where: NT, is the minimum number of training data; NH, is the number of hidden layer nodes; NI, is the number of input layer nodes; NO, is the number of output layer nodes.

In considering 15 neurons for hidden layers, 2 input layer nodes (scoria instead of sand percent and curing day), and 2 output layer nodes (compressive and tensile strength), the minimum number of training data is:

NT = 15*(2+1)+2*(15+1) = 77

In addition, for more improved performance, all the data normalized within a range of (0.1, 0.9) rather than from 0 to 1. This is because the sigmoid used as a transfer function shows a slow rate of learning in the end points on the function (Ayman, 2005). The following equation is used:

XT = 0.1 + 0.8[(X - X[submin])/(X[submax] - X[submin])]

Where: XT is the normalized value, X is the original value, X[submax] is the maximum value of the output, X[submin] is the minimum value of the output.

As a summarized result, before training, the following three important steps have to be determined:

(i) Network parameters, that is the number and size of different layers of the network, nodal function, normalization factors, etc.
(ii) The validation of the experimental data available for training and testing
(iii) The learning algorithm rule

Also, in back-propagation neural networks, some parameters, such as learning rate, mutation rate, population size, and cross over rate have to be selected for network learning and more developmental optimization. The neural network information is shown in Table 4.

The NN tool toolbox in MATLAB software was used to generate 52 different networks, shown in Tables 5, 6, 7 and 8, based on the net information mentioned in Table 4 and network topology. The network topology included hidden layer neurons between 3 and 5. The net calculation report presented in Table 5 investigated for the net topology, such as 3, 7, 11 and 15 neurons in the hidden layer, for the training function such as Trainbr and Trainlm, for the learning function such as Learngd and Learngdm, and for the sigmoid output transfer function. The network generation shown in Table 6 is for the 5, 10 and 15 neurons in one hidden layer network, trainlm and learngdm as training and learning function and Purelin as output transfer function. Tables 7 and 8 are related to generate networks with two hidden layers and different neurons. In the two calculation reports, the training and learning function are similar but the output transfer function is different. In each set of the defined network, the validation of the network has been performed after training on the test pattern. The assessment and validation of the network performance has been done using root mean square error (RMSE).

Network results

The generation of the artificial neural network consisted of five main parts: (a) data gathering and problem

Table 5. Networks calculation report for one hidden layer and LOGSIS as output transfer function.

Train function	Learn function	Transfer function	RMS error and R^2	The number of neurons in hidden layer							
				3		7		11		15	
				Train	Test	Train	Test	Train	Test	Train	Test
Trainbr	LGD	Logsig	RMS	0.153	0.161	0.152	0.159	0.157	0.161	0.170	0.162
			R^2	0.841	0.886	0.882	0.843	0.822	0.875	0.894	0.767
		Tansig	RMS	0.145	0.161	0.167	0.160	0.139	0.161	0.142	0.159
			R^2	0.860	0.902	0.841	0.866	0.843	0.827	0.706	0.806
	LGDM	Logsig	RMS	0.157	0.159	0.147	0.161	0.158	0.159	0.146	0.161
			R^2	0.869	0.849	0.867	0.826	0.869	0.875	0.830	0.844
		Tansig	RMS	0.655	0.625	0.171	0,161	0.167	0.161	0.161	0.166
			R^2	0.841	0.909	0.866	0.870	0.717	0.791	0.74	0.79
Trainlm	LGD	Logsig	RMS	0.154	0.213	0.160	0.155	0.141	0.158	0.135	0.154
			R^2	0.87	0.85	0.884	0.9	0.886	0.865	0.87	0.89
		Tansig	RMS	0.16	0.158	0.186	0.213	0.17	0.154	0.138	0.163
			R^2	0.881	0.887	0.856	0.878	0.9	0.87	0.914	0.865
	LGDM	Logsig	RMS	0.149	0.161	0.154	0.155	0.152	0.154	0.142	0.171
			R^2	0.869	0.893	0.881	0.908	0.909	0.889	0.906	0.855
		Tansig	RMS	0.165	0.154	0.144	0.154	0.138	0.155	0.159	0.154
			R^2	0.903	0.892	0.905	0.855	0.879	0.978	0.879	0.902

Table 6. Networks calculation report for one hidden layer and PURELIN as output transfer function.

Neurons in hidden layer	Training and learning function: TRAINLM and LEARNGDM				Transfer function Logsig=L , Tansig=T
	R^2			Training RMSE	
	Training	Validation	Testing		Hidden layer
5	0.994	0.991	0.989	0.0294	L
	0.997	0.993	0.996	0.0214	T
10	0.996	0.984	0.995	0.0233	L
	0.997	0.994	0.997	0.020	T
15	0.998	0.992	0.993	0.0368	L
	0.998	0.993	0.996	0.0227	T

Table 7. Networks calculation report for two hidden layers and LOGSIS as output transfer function.

Neurons in hidden layer		Training and learning function: TRAINLM and LEARNGDM				Transfer Function Logsig=L , Tansig=T	
		R^2			Training RMSE		
HL1	HL2	Training	Validation	Testing		1st hidden	2nd hidden
5	10	0.882	0.913	0.896	0.153	L	T
		0.881	0.891	0.93	0.145	T	T
5	5	0.886	0.897	0.907	0.152	L	T
		0.887	0.919	0.92	0.139	T	T
10	5	0.907	0.862	0.855	0.147	L	L
		0.887	0.863	0.922	0.153	T	L
		0.897	0.883	0.875	0.143	T	T

Table 8. Networks calculation report for two hidden layers and PURLIN as output transfer function.

Neurons in hidden layer		Training and learning function: TRAINLM and LEARNGDM				Transfer function Logsig=L , Tansig=T	
		R^2			Training RMSE		
HL1	HL2	Training	Validation	Testing		1st hidden	2nd hidden
5	10	0.997	0.975	0.991	0.029	L	T
		0.997	0.996	0.992	0.022	T	T
5	5	0.99	0.98	0.994	0.032	L	T
		0.996	0.990	0.989	0.024	T	T
10	5	0.998	0.996	0.982	0.017	L	L
		0.997	0.992	0.989	0.0243	T	L
		0.997	0.992	0.996	0.0286	T	T

Table 9. The parameters and RMSE of selected network.

No.	Net architecture	Neurons and transfer function in hidden layer		Training function	Learning function	Output transfer function	R^2 (All)	RMSE	
		HL1	HL2					Train	Test
N1	2-10-2	10 Tansig	-	Trainlm	Learngdm	Purelin	0.996	0.020	0.027
N2	2-10-5-2	10 Tansig	5 Logsig	Trainlm	Learngdm	Purelin	0.992	0.017	0.018

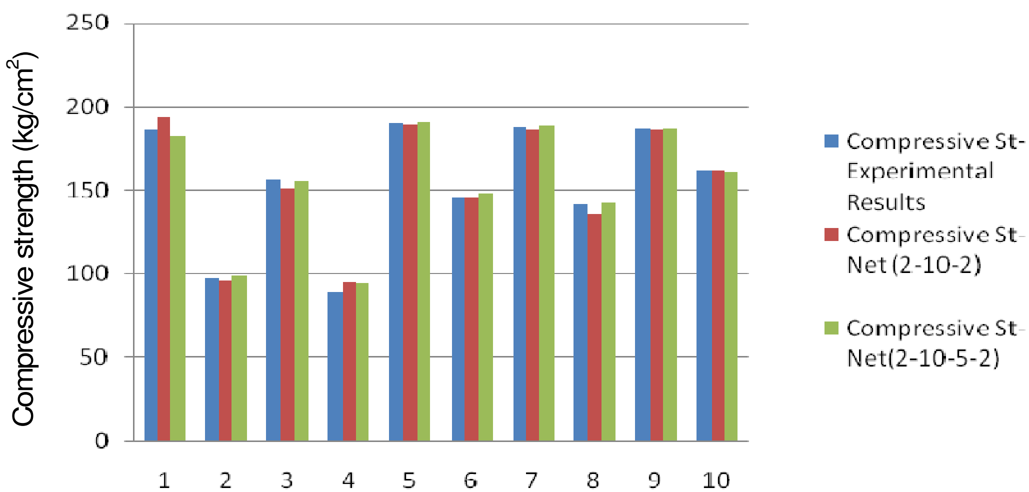

Figure 10. Evaluation of target and predicted compressive strength.

evaluation; (b) architecture determination; (c) learning process determination; (d) training of the networks; and (e) testing of the trained network for generalization evaluation (Wu and Lim, 1993). The performance of the generated network for prediction depends on the data training and testing and the area this data covers (Maru and Nagpal, 2004) and evaluates based on the minimum error between data after training. The RMSE method was used to select the optimum network. Concerning the investigation of different networks defined in Tables 5, 6, 7 and 8, two networks based on RMSE and data correlation coefficient were selected. The parameters and net training and testing error of the selected networks are shown in Table 9. Also, a comparison of the experimental and predicted compressive strength and tensile strength by the aforementioned network N1(2-10-2) and N2(2-10-5-2) are shown in Figures 10 and 11. These figures presented that the experimental and predicted results are

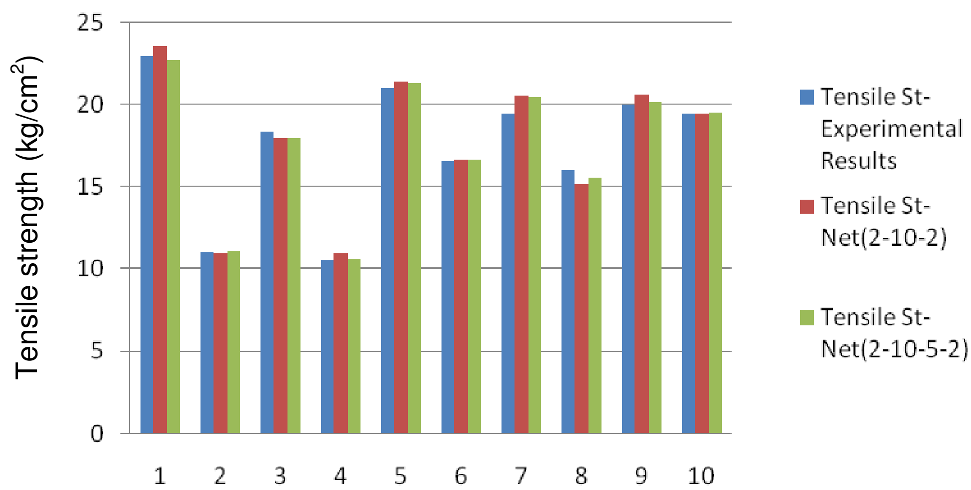

Figure 11. Evaluation of target and predicted tensile strength.

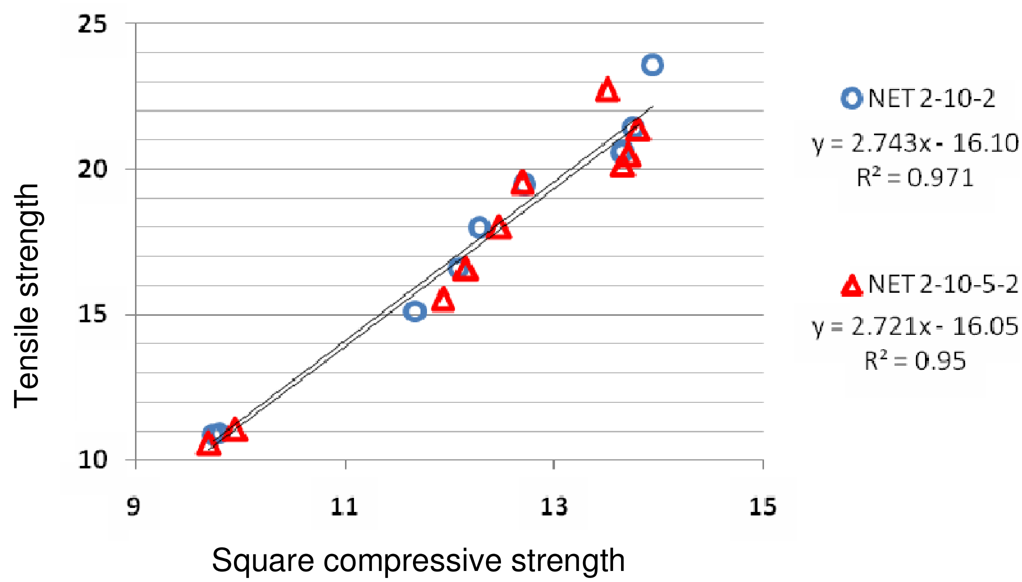

Figure 12. Relationship between squared compressive strength and tensile strength.

close together. The relationship between squared compressive strength and tensile strength of network output is given in Figure 12. The accurate correlation coefficient closed to 1 presented the effective relation between them in both networks 2-10-2 and 2-10-5-2.

Conclusion

(1) The mixtures made with equal to and more than 60% scoria instead of sand are lightweight mortar.
(2) By increasing the percentage of scoria instead of sand in mortar from 0 to 100%, the mortar 28 days density, compressive strength, and tensile strength reduced 49, 7.52 and 16.67%, respectively.
(3) The performances of the generated networks were evaluated by RMSE and correlation coefficient (R^2) between data.
(4) The networksN1(2-10-2) and N2(2-10-5-2) with TRAINIM as training function, LEARNGDM as adaption learning function, and PURELIN as output transfer function can predict the compressive and tensile strength of the mortar with minimum training and testing error.
(5) The training RMSE for networks N1 and N2 was 0.02 and 0.017.
(6) The (testing RMSE, R^2) for the networks N1 and N2 were (0.027, 0.997) and (0.018, 0.982), respectively.
(7) The relationship between squared compressive

strength and tensile strength gathered from both networks 2-10-2 and 2-10-5-2 presented an effective correlation coefficient equal to 0.975 and 0.95, respectively.

REFERENCES

Abedi J, Mollahi A (2005). An Investigation in Mechanical Property of Lightweight Concrete with Rice Stalk. 2nd International Conferences on Concrete & Development. Tehran. Iran. pp. 112

Ahmet Ö, Murat P, Erdog Ö, Erdog K, Naci Ç, M Asghar B (2006). Predicting the compressive strength and slump of high strength concrete using neural network . Constr. Building Mater., 9: 769-775.

Aydin AC, Karakoç MB, Düzgün OA, Bayraktutan MS (2010). Effect of low quality aggregates on the mechanical properties of lightweight concrete. Scientific Research and Essays, 5: 1133-1140.

Ayman AS (2005). A neural network model for predicting maximum shear capacity of concrete beams without transverse reinforcement. Canadian J. Civil Eng., 32: 44-57.

BRITISH STANDARDS INSTITUTION. BS EN (1996). Eurocode 6 – Design of masonry structures.

Carpenter WC, Barthelemy JF (1994). Common Misconceptions about Neural Networks as Approximators. ASCE J. Comput. Civil Eng., 8:345-358.

Carpenter W, Hoffman M (1995). Training backprop neural networks. J AI Expert, pp. 30-33.

Dias WPS, Pooliyadda SP (2001). Neural networks for predicting properties of concretes with admixtures. Constr Build Mater., 15: 371-379.

Famili H (1997). A Project about Lightweight Concrete. University of Elmo-sanat publishers. Iran. Tehran.

Gadea J, Rodríguez A, Campos PL, Garabito J, Calderón V (2010). Lightweight mortar made with recycled polyurethane foam. Cement and Concrete Composites, 32: 672-677.

Guang NH, Zong WJ (2000). Prediction of compressive strength of concrete by neural networks. Cem Concr Res., 30: 1245-1250.

Haykin S (1999). Neural networks: a comprehensive foundation. Prentice Hall Inc. Englewood Cliffs.N.J.

Hornik K, Maxwell S and White H (1990). Universal Approximation of an Unknown Mapping and its Derivative using Multilayer Feedforward Networks. Neural Networks, 3:551-560.

İlker BT, Mustafa S (2007). Prediction of properties of waste AAC aggregate concrete using artificial neural network. Eskişehir Osmangazi University. Department of Civil Engineering. Turkey, Computational Materials Sci., 41(1): 117-125.

Joachim S (1999). Influence of water-cement ratio and cement content on the properties of polymer-modified mortars. Cement Concr. Res., 29: 909-915.

Kasperkiewics J, Racz J, Dubrawski A (1995). HPC strength prediction using ANN. ASCE J. Comput. Civil Eng., 9: 279-284

Lai S, Sera M (1997). Concrete strength prediction by means of neural network. Constr Build Mater., 11: 93-98.

Lee SC (2003). Prediction of concrete strength using artificial neural Networks. Eng Struct., 25: 849-857

Lin Y, Lai CP, Yen T (2003). Prediction of ultrasonic pulse velocity (UPV) in concrete. ACI Materials J. 100:21-28.

Lo TY, Cui HZ (2004). Effect of porous lightweight aggregate on strength of concrete. MaterialsLetters.58:916-919.

Marcos L, García-Ruiz PA (2008). Lightweight cement mortars: Advantages and inconveniences of expanded perlite and its influence on fresh and hardened state and durability. Construction and Building Materials. 22: 1798-1806.

Marcos LT, García-Ruiz PA (2009). Lightweight pozzolanic materials used in mortars: Evaluation of their influence on density. mechanical strength and water absorption. Cement and Concrete Composites. 31: 114-119.

Maru S, Nagpal AK (2004). Neural network for creep and shrinkage deflections in reinforced concrete frames. J. Computing in Civil Engineering. ASCE, 18(4): 350-359.

McCullagh P, Nelder JA (1989). Generalized Linear Models. 2nd ed. London: Chapman & Hall.

Merikallio T Mannonen R (1996). Drying of Lightweight Concrete Produced From Crushed Expended Clay Aggregates. Com Concer Res., 26: 1423-1433.

Mukherjee A, Deshpande JM (1995). Modelling Initial Design Process using Artificial Neural Networks. ASCE Computing in Civil Engineering, 7(1): 556-559.

Oreta AWC, Kawashima K (2003). Neural network modeling of confined compressive strength. J. Struct Eng., 129: 554-561.

Oreta AWC, Kawashima K (2003). Neural network modeling of confined compressive strength and strain of circular concrete columns. ASCE J. Struct. Eng., 129(4): 554-561.

Rajagopalan PR, Prakash J, Naramimhan V (1973). Correlation between ultrasonic pulse velocity and strength of concrete. Indian Concrete J. 47: 416–418.

Reda MMT, Noureldin A, El-Sheimy N, Shrive NG (2003). Artificial neural networks for predicting creep with an example application to structural masonry. Canadian J. Civil Eng., 30: 523-532.

Ripley BD (1996). Pattern recognition and neural networks. Cambridge University Press. New York.

Sanahi G (1998). Application of Perlit in Construction Process. Master Dissertation. University of Tabriz. Tabriz. Iran.

Sari D, Pasamehmetoglu A (2004). The Effect of Grading and Admixture on the Pumice Lightweight Aggregate Concrete. 2nd Department of Civil Engineering. Atilim University. Ankara. Turkey, Cement and Concrete Res., 35(5): 936-942.

Shideler J (1975). Lightweight Aggregate Concrete for Structural Use. ACI J., 54: 299-328.

Shorabi, M., Rigi, A. (2005). Application of Lightweight Concrete Properties with Lightweight Grain of Taftan in Construction Methoh. 2nd International Conferences on Concrete & Development. Tehran. Iran. pp. 109.

Short M, Kinniburgh W (1978). Lightweight Concrete. Applied Science Publishers. London. pp. 443-455.

Topcu IB, Odler I (1996). Semi lightweight concretes produced by volcanic slags. Cement and Concrete Research. pp. 15-21.

Unal O, Uygunog T Yildiz A (2007). Investigation of properties of low strength lightweight concrete for thermal insulation. Building and Environment, 42: 584-590.

Yeh I C (1998). Modeling of strength of HPC using ANN. Cem Concr Res 28(12):1797-1808.

Yu-Ling W , Jing-Chiang Y, Yong-Yang L, Shih-Yu C, Paul HW (2008). Recycling of harbor sediment as lightweight aggregate. Marine Pollution Bulletin. 5th International Conference on Marine Pollution and Ecotoxicology, 57: 867-872.

Wasserman R, Bentur A (1996). Interfacial interactions in lightweight aggregate concretes and their influence on the concrete strength. Cement and Concrete Composites, 18: 67-76.

Wu X, Lim SY (1993). Prediction maximum scour depth at the spur dikes with adaptive neural networks. Neural networks and combinatorial optimization in civil and structural engineering. Civil-Comp Press, Pp. 61-66.