Computers and Concrete, *Vol. 24, No. 5 (2019) 399-412*
DOI: https://doi.org/10.12989/cac.2019.24.5.399

Elman ANNs along with two different sets of inputs for predicting the properties of SCCs

Atefeh Gholamzadeh-Chitgar*[1] and Javad Berenjian[2a]

[1]*Department of Civil Engineering, Construction Management and Engineering, Tabari University of Babol, Babol, Iran*
[2]*Department of Civil Engineering, Tabari University of Babol, Babol, Iran*

(*Received April 10, 2019, Revised August 31, 2019, Accepted September 26, 2019*)

Abstract. In this investigation, Elman neural networks were utilized for predicting the mechanical properties of Self-Compacting Concretes (SCCs). Elman models were designed by using experimental data of many different concrete mix-designs of various types of SCC that were collected from the literature. In order to investigate the effectiveness of the selected input variables on the network performance in predicting intended properties, utilized data in artificial neural networks were considered in two sets of 8 and 140 input variables. The obtained outcomes showed that not only can the developed Elman ANNs predict the mechanical properties of SCCs with high accuracy, but also for all of the desired outputs, networks with 140 inputs, compared to ones with 8, have a remarkable percent improvement in the obtained prediction results. The prediction accuracy can significantly be improved by using a more complete and accurate set of key factors affecting the desired outputs, as input variables, in the networks, which is leading to more similarity of the predicted results gained from networks to experimental results.

Keywords: self-compacting concrete; Elman artificial neural networks; mechanical properties; input variables

1. Introduction

Self-compacting concrete (SCC) can be introduced as an advanced kind of concrete that is able to compact under its own weight without requiring any mechanical vibration (Krishna *et al.* 2010). The development and use of this concrete can lead to a reduction in noise pollution in the plants and construction sites, faster construction, improvement in the working conditions and finally, making the concrete productions with very high surface quality (Ouchi *et al.* 2003). Similar to other concretes, the mechanical properties of this concrete are obtained by experimental works that are time-consuming and costly. Therefore, using a new technique in order to decrease these experimental works can be very useful.

In the past two decades, many researchers have made use of different computational methods, especially artificial neural network technique, which are able to solve extremely complicated problems, to predict the various properties of concrete (Topcu and Sarıdemir 2007, Yeh 1998).

Altun *et al.* (2008) predicted the strength of steel fiber added lightweight concrete using artificial neural network (ANN) and multi linear regression (MLR) technique. To construct the models, of the 126 produced experimental examples, the 105 examples were used for training the network and 21 examples were used for testing. Different combinations of various variables including; steel fiber dosage (SFD), water (W), water-cement ratio (W/C), cement dosage (C), pumice sand (PS), pumice gravel (PG), and super plasticizer content (SC) were used as input variables in order to estimate the desired output. The study concluded that the model with a combination of SFD, W and W/C as input variables has the smallest mean square error (MSE) 1.49, and the highest correlation coefficient (R) 0.859 in the prediction of the compressive strength. It was also found that the ANN can predict the compressive strength of steel fiber added lightweight concrete better than the multi linear regression technique.

Barbuta *et al.* (2012) estimated the compressive and flexural strength of polymer concrete with Fly Ash successfully using multilayer perceptron (MLP) neural network. Boga *et al.* (2013) constructed a four-layered feed-forward neural network for predicting the mechanical and chloride permeability properties of concrete containing ground granulated blast furnace slag (GGBFS) and calcium nitrite-based corrosion inhibitor (CNI). Bilim *et al.* (2009) developed a feed-forward, single hidden layer neural network model with six inputs including the cement, ground granulated blast furnace slag, water, hyperplasticizer, aggregate and age of samples for predicting the compressive strength of ground granulated blast furnace slag concrete. From 225 experimental examples, 113 examples were used for training the network, and the remaining 112 examples were randomly selected and used for testing the network. To train the network, the number of neurons in the input, hidden and output layer were 6, 15 and 1 respectively. The scaled conjugate gradient (SCG) algorithm, Levenberg-Marquardt (LM) algorithm, one step

*Corresponding author, MSc.
E-mail: b_gh_ch@yahoo.com,
gholamzadehchitgar@tabari.ac.ir
[a]Associate Professor
E-mail: berenjian@nit.ac.ir

http://www.techno-press.org/?journal=cac&subpage=8 ISSN: 1598-8198 (Print), 1598-818X (Online)

secant backpropagation algorithm (OSS) and BFGS quasi-Newton backpropagation algorithm were used for learning the network. The results showed that Ann, as a feasible tool, are able to predict the compressive strength of ground granulated blast furnace slag concrete using the components of concrete as inputs. In addition, among the various algorithms used in this research, the Levenberg-Marquardt algorithm was found as the best learning algorithm.

Demir (2008) applied ANNs with different architectures to predict the elastic modulus of normal and high strength concrete. Pala *et al.* (2007) predicted long-term effects of fly ash and silica fume on compressive strength of concrete using ANN. For this aim, a computer program was performed in MATLAB. A feed forward neural network with the back-propagation learning algorithm was developed. The number of neurons in the input, hidden and output layer were 8, 9 and 1 respectively. According to the results, ANNs can evaluated the effect of cementitious material on the compressive strength of concrete with high accuracy. It was also found that fly ash content has a little effect on the strength of concrete at early ages in comparison with later ages. Furthermore, addition of silica fume to mixes can cause a reduction in the early compressive strength of concrete, however it increases the long-term compressive strength.

Li *et al.* (2011) used ANN with six input variables including the cement, fly ash, blast furnace slag, super plasticizer, sand ratio and water/binder for predicting the workability of SCC. Ghafoori *et al.* (2013) used linear and nonlinear regressions, and also neural network models to approximate rapid chloride permeability of self-consolidating concretes based on their mixture ratios. For this purpose, various models were designed by varying number of independent variables and mixtures allocated to training and testing. The obtained outcomes of this research demonstrated the better performance of neural networks than the prediction models obtained by linear and nonlinear regressions.

Sonebi *et al.* (2016) designed a multilayer feed forward neural network with seven inputs namely, the cement content, the dosages of limestone powder, water, fine aggregate, coarse aggregate, superplasticizer and time to predict the fresh properties of self-compacting concrete. In addition, the slump flow, T50, T60, V-funnel flow time, Orimet flow time, and blocking ratio (L-box) were considered as outputs. The results of this paper showed that the ANN model is able to predict the fresh properties of SCC instantly and accurately.

The main objective of the current study is to examine the potential of Elman ANNs to predict the mechanical properties of SCC under conditions that the designed networks contain different types of SCC. It means, unlike some researches, which were focused on only one type of SCC, such as (Prasad Meesaraganda *et al.* 2019, Azizifar and Babajanzadeh 2018, Vakhshouri and Nejadi 2015), this study covers the different types of SCC containing various materials (i.e., SCC with fiber, lightweight aggregates, recycled materials and pozzolans). For this purpose, experimental data of many different concrete mix-designs of various types of SCC were gathered from different sources (section 3.1) to make comprehensive models, which can predict each of the desired properties of SCCs. Furthermore, a review of the past studies shows that in spite of the different works reported on using artificial neural networks in the concrete field (mentioned above), little attention has been paid to study the effectiveness of the different input variables on the mechanical properties of SCCs. Therefore, checking the effectiveness of the different factors, as input variables, on the network performance in predicting the intended properties is another aim of this study. A summary of the intended purposes is presented as follows:

• Performance evaluation of Elman artificial neural network in predicting compressive strength, tensile strength, flexural strength and elastic modulus of SCC, in the case of that the expansive and dispersion of sources and mixed designs, which were used in this research, are large. This study clearly shows the high potential and reliability of this network in prediction.

• Making almost comprehensive models to predict the mechanical properties of SCC, in which the designed models contain different mix designs of various types of SCC. (Not just one type of SCC).

• Performance comparison of the constructed networks under conditions that the considered effective variables are different in them as inputs. Therefore, once, 8 variables and again, in order to simulate a real experimental conditions, 140 variables were entered as inputs in Elman ANNs. Then the results obtained from the models with 8 and 140 inputs were compared with each other and finally, the results obtained from the best models were compared with the experimental results.

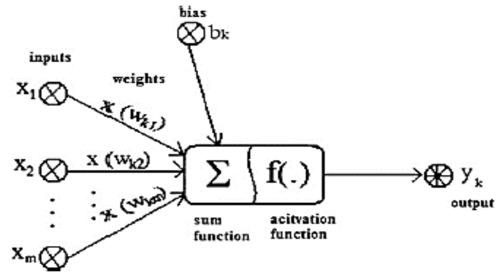

Fig. 1 A schematic of an artificial neuron (Haykin 1999)

2. Artificial neural networks

The human brain consists of multimillions of neurons in which each neuron works as a numerical processor and they are connected together extremely in a complex way with parallel operation (Ayazi *et al.* 2009). Similar to the human brain, artificial neural networks consist of many interconnected artificial neurons that are linked to each other with connection weights. Each neuron has a transfer function in order to determine the output (Uysal and Tanyildizi 2012). A schematic of an artificial neuron is given in Fig. 1 (Haykin 1999). According to Haykin (1999), the output of the neuron k can be calculated by the following Equations

$$u_k = \sum w_{km} \times x_m \tag{1}$$

$$y_k = f(u_k + b_k) \quad (2)$$

Where: $x_1, x_2, x_3, \ldots x_m$ are inputs; $w_{k1}, w_{k2}, w_{k3}, \ldots w_{km}$ are synaptic weights of neuron k; u_k is the linear combiner output due to inputs; b_k is the bias; $f(u_k + b_k)$ is the transfer function and finally, y_k is the output.

Typically, the network architecture has three main layers including an input layer, an output layer, and one (or more) hidden layers. The input variables are fed to the ANN in the input layer. The output layer represents the response of the network to the input of the system. The hidden layer helps network to prepare nonlinear mapping of the data to forecast the desired output (Ashtiani *et al.* 2018).

ANNs have shown to provide good approximates to nonlinear data. It has been demonstrated that neural networks have the ability to fit any data properly when utilizing the right amount of neurons and the appropriate topology (Cybenko 1989, Hornik *et al.* 1989, Hornik 1991, Funahashi 1989). Other advantages of these networks contain their fitness for parallel computing, their relative insensitivity to computational errors and their speed and precision in predicting generalized data (Haykin 1999, Kalogirou 2000). Once trained, artificial networks are also easy to retrain as new or more precise data becomes available (Haykin 1999). However, a disadvantage of the neural network, is its black box system approach, which is incapable to explain the weights and the interrelationship between the inputs and output (Barbuta *et al.* 2012). Another problem is that the optimal form or value of most network design parameters (such as the number of hidden layer neurons) can vary for each application and cannot be theoretically defined. For this reason, the appropriate structure of the network and also the number of neurons in the hidden layers are usually determined via trial and error technique (Devos and Rientjes 2005). Furthermore, large amounts of data are needed for training and calibration of these networks (Piryonesi and El-Diraby 2018). In some investigations, other algorithms such as decision trees are utilized due to their easiness of performance, but the accuracy of these algorithms might be lower than the neural networks (Provost and Fawcett 2013, Hastie *et al.* 2009, Shahin *et al.* 2009).

There are different types of neural networks, which prepare flexibility and strong means to disclose underlying relationships between input variables and the output of the system (Ashtiani *et al.* 2018). One of these is Elman ANN, which is used in this study. Elman ANN is a recurrent neural network that was suggested by Elman in 1990 (Elman 1990). The recurrent networks not only are able to transmit the information forwards, but also can transmit it backwards. The structure of Elman ANN has a feedback loop from hidden layer to input layer (Krenker *et al.* 2011) as shown in Fig. 2. This feedback loop permits the ANN to both detect and generate time varying patterns (Desai *et al.* 2011). This network is usually introduced as a special type of feed-forward ANN, containing additional memory neurons and feedback loop (Koker 2006).

3. Project data and details of the networks preparation

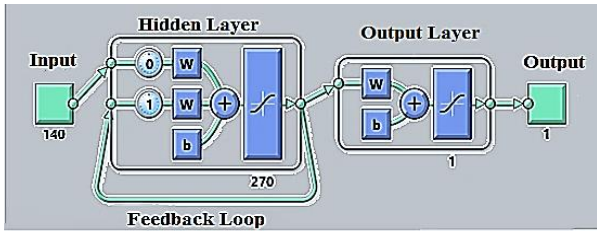

Fig. 2 A schematic of Elman recurrent neural network

3.1 Description of the data

To establish almost comprehensive models, which include different kinds of SCC (containing fiber, lightweight aggregates, recycled materials and pozzolans), experimental samples from different sources were collected (Douglas 2004, Yazicioglu *et al*. 2006, Turk and Karatas 2011, Mahajan and Singh 2013, Abdul Hameed 2005, Kumar *et al.* 2013, Corinaldesi and Moriconi 2011, Krishna *et al.* 2010, Venkateswara Rao *et al.* 2012, Dubey and Kumar 2012, Dumne 2014, Ranjbar *et al.* 2013, Sahmaran *et al.* 2005, Khaleel *et al.* 2011, Jalal *et al.* 2012, Beigi *et al.* 2013, Hossain *et al.* 2013, Aydin 2007, Guneyisi *et al.* 2010, Uysal and Yilmaz 2011, Patel *et al.* 2011). In fact, for predicting 3, 7, 28, 56 and 90-day compressive strength 75, 275, 549, 52 and 203 samples, for elastic modulus 38 samples and for 28-day tensile and flexural strengths 274 and 102 samples were used respectively.

Separating data into training and testing sets has a remarkable effect on the performance of the final model (Wu *et al.* 2012). The size of the training samples has a significant effect on the generalization abilities of the neural networks, because it is responsible for adjusting weights during the learning process of these networks (Richards 1991). Furthermore, small training sample sizes can cause a poor performance in the ANNs (Moyo and Sibanda 2015). Different sampling methods can be used to divide the data (May *et al*. 2010, Lohr 2009). In most articles, samples are selected randomly. Fathi *et al.* (2019) divided the data into the training and testing sets randomly. They used 80% of the data for training the network and 20% of the data for testing. Wang *et al.* (2018) randomly selected 85% of the data for training the network and 15% of the data for testing.

In this study, in order to attain a good generalization ability in the neural networks, the samples were divided into two sets randomly:

Training set: 85 percent of all samples were selected randomly for training the networks.

Testing set: 15 percent of the remaining samples were applied for testing the trained ANNs.

It should be noted an early stopping technique, (the default method for improving generalization), was used during the training process to avoid over-fitting. In this technique, a validation set is separated out of the training set automatically. The validation set is a fraction of the training set such that monitors and controls the training process (MATLAB Software 2013).

First, in order to take prediction conditions closer to experimental conditions, data utilized in ANNs were

Table 1 Introducing the different input variables for networks with 140 inputs

The amounts of gravel, sand, lightweight aggregates, recycled materials, cement, pozzolan, limestone powder, fiber, water, Nano-silica and polymer. ($^{Kg}/_{m^3}$)
The shapes of gravel including: fully rounded corner, rounded corner, relatively rounded corner, relatively sharp corner and sharp corner.
Length (mm), diameter (mm), tensile strength (MPa) and the shapes of the fibers including: fiber with straight end and fiber with hooked end.
Specific gravity of gravel, sand, lightweight aggregates, recycled materials, cement, pozzolan, limestone powder, fiber, super plasticizer, Nano-silica, viscosity-modifying agent (VMA) and high water reduction agent (HWRA). ($^{gr}/_{cm^3}$)
Maximum size of gravel and lightweight aggregates. (mm)
Water absorption of gravel, sand and lightweight aggregates. (%)
Grading of gravel, sand and lightweight aggregates.
Chemical properties of recycled materials, cement, pozzolan and limestone powder.
Curing conditions (Dry, Wet, Sealed).
Solid contents of super plasticizer and Nano-silica. (%)
PH of super plasticizer.
Dosage of super plasticizer, viscosity-modifying agent (VMA) and high water reduction agent (HWRA). ($^{Kg}/_{m^3}$)
Concrete's delivery time (min).
Temperature operation (C°).

Table 2 Introducing the different shapes of sand, fiber and also various curing conditions to the ANNs

Sand	Fiber	Curing Conditions
Fully rounded corner = 0	Fiber with straight end = 0.5	Dry conditions = 0
Rounded corner = 0.25	Fiber with hooked end = 1	Wet conditions = 1
Relatively rounded corner = 0.5	-	Sealed conditions = 2
Relatively sharp corner = 0.75	-	-
Sharp corner = 1	-	-

categorized in a large set of 140 inputs, according to Table 1.

In other words, the reason for choosing this number of variables is that the authors have tried to simulate prediction conditions to experimental conditions, by using a large part of key factors affecting the desired outputs, in order to measure the accuracy of the constructed networks in predicting the intended properties.

It should be noted that introducing the different shapes of sand, fiber and also various curing conditions to the neural networks were conducted by allocating constant digits for each shape and curing condition according to Table 2.

Second, some of the factors affecting the mechanical properties of concrete were ignored and, only 8 more common variables which have been seen frequently in related studies in this field, were used; namely: water-to-cement materials ratio, the amounts of cement, sand and gravel, super plasticizer dosages and specific gravity of cement, sand and gravel (Khademi and jamal 2016, Oztas *et al.* 2006, Altun *et al.* 2008, Shah *et al.* 2018, Das *et al.* 2015, Soleymani and Karimi Livary 2012, Rajaram *et al.* 2018, Dias and Pooliyadda 2001, Yuan *et al.* 2014, Nazari and Riahi 2011). It means, at the first time, the authors collected input variables with more sensitivity (in order to simulate a real experimental conditions). But, the second time, this sensitivity was ignored and the authors just tried to gather the factors which had the most selection as input variables among similar articles. Finally, these collected variables were arranged in a format of 8 inputs.

It is worth mentioning that, some methods, such as data analytics and multi-stage methodology can be helpful for the data-collection and prediction (Piryonesi and El-Diraby 2018, Panagoulia *et al.* 2017). For each network, one of these mentioned mechanical properties i.e. elastic modulus, tensile strength, flexural strength or compressive strength of SCCs was considered as output variable.

3.2 Networks architecture

The MATLAB Neural Network Toolbox was utilized to make and train Elman ANNs. For structure of these networks, one input layer, one hidden layer and also one output layer were considered. The number of neurons in each of the input and output layers is fixed. The input layer, by the size of the input vector, i.e., the number of neurons in the input layer is equal to the length of the input vector (input variables). Also, the number of neurons in the output layer is equal to the number of response variables (size of the output vector) (Gopi 2007, Champa and AnandaKumar 2010). Therefore, for networks with 8 and 140 inputs, the number of neurons in the input layer were 8 and 140 with one neuron in the output layer (equal to the number of input and output variables). In order to determine the number of hidden layer neurons trial and error technique usually is applied, because there are no fixed rules to determine it (Masters 1993). According to the Kanellopoulas and Wilkinson (1997), in this study, the number of neurons in the hidden layer were chosen at least twice as many as the number of input variables, it means for networks with 8 and 140 inputs, 16 and 300 neurons were considered respectively. In order to achieve the optimum ANNs with appropriate structures, which can predict the desired properties with the minimum values of the test errors, it is necessary to find the optimal number of hidden layer neurons. Therefore, for networks with 8 and 140 inputs, the number of hidden layer neurons varied from 4 to 16 and 30 to 300 neurons respectively.

3.3 Data transformation

The training and test sets were scaled between 0 and 1 by utilizing below codes in MATLAB software (MATLAB Software 2013)

$$[pn, ps] = mapminmax\,(p, 0, 1) \tag{3}$$

$$[tn, ts] = mapminmax\,(t, 0, 1) \tag{4}$$

Where p and t represent the original inputs and targets, *pn* and *tn* are the normalized inputs and targets, *ps* and *ts* contain the settings, in this case the minimum and maximum values of the original inputs and targets (MATLAB Software 2013). It should be noted that in this study, in order to show the effect of using scaled and raw data on the values of the test MSE, a set of raw data was also utilized in Elman networks.

3.4 Neural networks training

For training Elman models, a type of back propagation algorithm, namely Scaled Conjugate Gradient (SCG) algorithm, which has an excellent and fast performance for large networks with large number of weights, was used (Indra Kiran *et al.* 2010). In order to reach the optimum iteration numbers of Maximum fail, the number of maximum fail iterations of the networks were set between 6 and 500. This domain was selected randomly. Furthermore, different combinations of the transfer functions that are usually utilized in ANN modeling, including tan-sigmoid, purelin and the log-sigmoid were applied in the networks in order to determine the optimum ANNs. All networks were trained for a maximum of 1000 epochs.

3.5 ANN performance functions

Mean Square Error (MSE) and the Correlation Coefficient (*R*) were used to evaluate the results of the networks. These functions are defined as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(X_{ex\,(i)} - X_{cal\,(i)})^2 \quad (5)$$

$$R = [1 - \frac{\sum_{i=1}^{N}(X_{ex\,(i)} - X_{cal\,(i)})^2}{\sum_{i=1}^{N}(X_{ex\,(i)} - \overline{X_{ex}})^2}]^{1/2} \quad (6)$$

Where $X_{ex\,(i)}$, $X_{cal\,(i)}$, $\overline{X_{ex}}$ and N are the i th experimental value, i th calculated value, mean of experimental values and the number of data points respectively (Badrnezhad and Mirza 2014).

4. Results and discussions

4.1 Optimization of Elman ANNs

In this investigation, Elman ANNs are proposed to predict the mechanical properties of SCCs. Three main characteristics of the networks, including the number of hidden layer neurons, the number of maximum fail iterations and the transfer functions in each layer are considered as variable characteristics and are optimized in each network separately, in order to achieve the optimum networks with the best structures (based on the minimum values of the test errors). The required settings to optimize models is shown in Table 3. It should be noted that the effect of using scaled and raw data on the values of the test MSE, which was mentioned previously, is also evaluated in this section of the paper. The details and results of this process are presented as follows:

• The results obtained from using raw and scaled data in predicting the mechanical properties of SCCs are compared with each other in Table 4. As shown in this table, for all networks with 8 and 140 input variables, the results of the test errors for scaled data are smaller than the raw data. For example, for 3-day compressive strength (in the networks with 8 inputs), the minimum test MSE is obtained 32.99 using raw data, but by using scaled data, the minimum test error is obtained 27.62 and it shows the test MSE is decreased. Accordingly, for all networks, scaling data leads to improve the test errors. Therefore, the scaled data are used for doing all sections of this study. Furthermore, the optimum number of neurons in the hidden layer is determined based on the minimum test errors that were obtained using scaled data in Table 4. Therefore, for networks with 8 and 140 inputs, the obtained optimum number of hidden layer neurons are equal to (4, 120; 4, 150; 8, 240; 16, 30; and 12, 120) for 3, 7, 28, 56 and 90-day

Table 3 Required settings to optimize Elman ANNs

Purposes	Network type	The number of hidden layer neurons	Training algorithm	The number of max-fail iterations	Transfer functions between layers
Determination of the optimum number of hidden layer neurons	8 inputs	Variable between 4-16	SCG	6 (Network Default)	T*1-T (Network Default)
	140 inputs	Variable between 30-300	SCG	6 (Network Default)	T-T (Network Default)
Determination of the optimum iteration numbers of maximum fail	8 inputs	The optimum number	SCG	Variable between 6-500	T-T (Network Default)
	140 inputs	The optimum number	SCG	Variable between 6-500	T-T (Network Default)
Determination of the best combination of the transfer functions between layers	8 inputs	The optimum number	SCG	The optimum iteration	T-T, T-P*2, T-L*3, P-T, P-L, P-P, L-T, L-P, L-L
	140 inputs	The optimum number	SCG	The optimum iteration	T-T, T-P, T-L, P-T, P-L, P-P, L-T, L-P, L-L

*1: Tansig; *2: Purelin; *3: Logsig

Table 4 The obtained minimum values of the test errors by using raw and scaled data in the ANNs with 8 and 140 inputs

Data type	Number of inputs	Outputs							
		3-day FC*1	7-day FC	28-day FC	56-day FC	90-day FC	28-day Ft*2	28-day FL*3	ES*4
		Test MSE							
Raw	8	N*12-32.99	N16-78.40	N8-172.24	N16-82.17	N12-107.51	N4-0.87	N8-1.17	N16-7.97
	140	N300-12.72	N150-35.77	N240-47.85	N120-24.66	N120-48.60	N120-0.33	N270-0.91	N300-1.07
Scaled	8	N4-27.62	N4-75.52	N8-148.05	N16-63.95	N12-107.34	N16-0.69	N4-1.05	N8-7.82
	140	N120-4.12	N150-21.81	N240-40.07	N30-17.32	N120-48.36	N60-0.31	N210-0.71	N300-0.44

*: Number of neurons; *1: Compressive strength; *2: Tensile strength; *3: Flexural strength; *4: Elastic modulus

Table 5 The obtained values of the test MSE in the different iteration numbers of maximum fail for networks with 8 and 140 inputs

Number of Max_ fail iterations	Number of inputs	Outputs							
		3-day FC	7-day FC	28-day FC	56-day FC	90-day FC	28-day Ft	28-day FL	ES
		Test MSE							
6	8	27.62	75.52	148.05	63.95	107.34	0.69	1.05	**7.82**
	140	**4.12**	21.81	40.07	17.32	48.36	0.31	0.71	0.44
10	8	18.98	70.38	135.76	432.19	95.12	0.60	**0.98**	40.45
	140	5.26	39.83	75.77	19.21	81.31	0.45	1.23	1.64
50	8	**12.66**	55.39	142.89	34.67	**61.90**	0.46	2.02	11.31
	140	34.40	24.23	31.19	22.46	19.55	**0.18**	0.71	**0.39**
100	8	13.81	**39.34**	**63.92**	69.23	119.82	0.42	1.62	8.69
	140	28.81	**15.05**	43.42	9.61	**19.15**	0.33	2.89	1.23
500	8	13.25	40.58	82.67	**13.61**	100.65	**0.37**	1.95	8.33
	140	35.55	27.28	**28.18**	**4.92**	27.59	0.33	**0.24**	0.92

compressive strength, and (16, 60; 4, 210; and 8, 300) for tensile strength, flexural strength and elastic modulus, respectively.

• Maximum fail parameter is related to an early stopping technique, (the default method for improving generalization), during the training process to avoid overfitting. In this method, a validation set is separated out of the training set automatically. The number of "validation checks" indicates the number of continuous iterations that the validation performance fails to decrease (MATLAB Software 2103, Muluneh 2014). It means the training process continues until the validation error fails to decrease. When the validation error increases for a specified number of iterations (max_fail iterations), the training process stops. In the other words, the validation check indicates whether the current completed iteration has minimized error compared to the previous iterations. This criterion can be changed by setting the parameter *net. trainParam. max_fail* in MATLAB software (MATLAB Software 2013). The network is trained while the validation checks is periodically utilized to evaluate the model performance during the training in order to avoid over-training (Reitermanova 2010). In this study, as mentioned earlier, in order to reach the optimum iteration numbers of maximum fail (maximum validation checks), the number of maximum fail iterations of the networks was set between 6 and 500. For each iteration in the training process, the performance of the networks was checked. For models with two different sets of inputs, the values of the test MSE obtained from each output in the different iteration numbers of maximum fail is given in Table 5. As it can be seen, for networks with 8 and 140 inputs, the minimum test errors, which are specified in the table, are obtained in (50, 6; 100, 100; 100, 500; 500, 500; and 50, 100) iterations for 3, 7, 28, 56 and 90-day compressive strength, and in (500, 50; 10, 500; and 6, 50) iterations for 28-day tensile strength, 28-day flexural strength and elastic modulus, respectively. These iterations show that the training process should be stopped and no further iteration should be done. In fact, the designed networks in these iterations, have the best performance in the training process and no further training is necessary and if done, may forecast the results incorrectly. Therefore, these iterations are chosen as the optimum iterations of maximum fail in the networks due to having the most influence over the improved value of the test errors.

• The results are shown in Table 6. According to this table, for networks with 8 and 140 inputs, respectively, the

Table 6 The obtained results by using different combinations of three transfer functions (Tangsig, Logsig, Purelin) in the networks with 8 and 140 inputs

Outputs	Number of inputs	Transfer functions between layers								
		Tansig-Tansig	Tansig-Logsig	Tansig-Purelin	Purelin-Logsig	Purelin-Purelin	Purelin-Tansig	Logsig-Logsig	Logsig-Purelin	Logsig-Tansig
		Test MSE								
Fc3*[1]	8	**12.66**	310.57	14.29	310.57	40.58	29.80	310.57	32.79	32.80
	140	**4.12**	336.45	62.40	338.31	648.96	265.78	242.88	43.76	9.14
Fc7*[2]	8	39.34	162.33	62.32	174.60	82.32	81.10	176.96	62.28	**36.57**
	140	15.05	163.17	32.96	185.44	659.25	44.30	167.47	**13.54**	15.34
Fc28*[3]	8	**63.92**	308.06	104.47	308.06	212.09	224.40	308.06	113.87	129.32
	140	28.18	260.08	60.97	306.36	144.76	45.96	260.65	29.54	**18.91**
Fc56*[4]	8	13.61	224.18	**11.95**	221.42	95.43	88.21	222.64	12.91	16.75
	140	**4.92**	313.56	146.98	396.27	56.81	831.97	317.86	23.00	16.53
Fc90*[5]	8	61.90	181.80	91.52	192.77	227.99	270.42	470.82	**61.02**	79.13
	140	19.15	154.80	101.11	206.12	4.1817e+04	3.2214e+03	192.40	46.43	**17.79**
Ft28*[6]	8	**0.37**	1.81	0.49	1.81	0.99	1.01	1.59	0.40	0.52
	140	**0.18**	1.25	0.39	1.63	0.51	4.11	1.29	0.40	0.38
Fl28*[7]	8	0.98	7.21	1.59	6.88	1.29	1.14	6.76	1.34	**0.89**
	140	**0.24**	9.33	0.33	72.60	116.35	19.97	8.90	0.47	1.57
ES*[8]	8	**7.82**	27.90	9.13	27.66	10.85	47.36	26.02	23.40	37.79
	140	**0.39**	2.77	37.98	16.85	7.8401e+03	14.30	0.47	3.67	2.22

*[1]: 3-day compressive strength; *[2]: 7-day compressive strength; *[3]: 28-day compressive strength; *[4]: 56-day compressive strength:*[5]: 90-day compressive strength; *[6]: 28-day tensile strength; *[7]: 28-day flexural strength; *[8]: elastic modulus

Table 7 Comparison between statistical values of the optimum networks with 8 and 140 inputs

Outputs	Number of input variables	Test MSE	Train MSE	Test R	Train R	140-input-network test error improvement in comparison with 8-input- network test error
3-day Compressive Strength	8	12.66	13.78	0.95	0.95	67.45
	140	**4.12**	**8.56**	**0.99**	**0.97**	
7-day Compressive Strength	8	36.57	42.67	0.86	0.83	62.97
	140	**13.54**	**10.87**	**0.95**	**0.96**	
28-day Compressive Strength	8	63.92	58.16	0.88	0.90	70.41
	140	**18.91**	**17.15**	**0.97**	**0.97**	
56-day Compressive Strength	8	11.95	9.56	0.98	0.98	58.82
	140	**4.92**	**12.77**	**0.99**	**0.98**	
90-day Compressive Strength	8	61.02	109.90	0.92	0.86	70.84
	140	**17.79**	**19.03**	**0.97**	**0.97**	
28-day Tensile Strength	8	0.37	0.30	0.87	0.90	51.35
	140	**0.18**	**0.15**	**0.95**	**0.95**	
28-day Flexural Strength	8	0.89	1.64	0.84	0.83	73.03
	140	**0.24**	**0.38**	**0.97**	**0.95**	
Elastic Modulus	8	7.82	2.78	0.96	0.96	95.01
	140	**0.39**	**5.00**	**0.98**	**0.95**	

*: The specified rows show the best networks

obtained optimum results for each output are delineated as follows:

◦ 3-day compressive strength: the minimum test MSE (12.66 and 4.12), with the combination of (Tansig-Tansig).

◦ 7-day compressive strength: the minimum test MSE (36.57 and 13.54), with the combinations of (Logsig-Tansig and Logsig-Purelin).

◦ 28-day compressive strength: the minimum test MSE (63.92 and 18.91), with the combinations of (Tansig-Tansig and Logsig-Tansig).

◦ 56-day compressive strength: the minimum test MSE (11.95 and 4.92), with the combinations of (Tansig-Purelin and Tansig-Tansig).

◦ 90-day compressive strength: the minimum test MSE (61.02 and 17.79), with the combinations of (Logsig-Purelin and Logsig-Tansig).

◦ 28-day Tensile strength: the minimum test MSE (0.37 and 0.18), with the combination of (Tansig-Tansig).

◦ 28-day flexural strength: the minimum test MSE (0.89 and 0.24), with the combinations of (Logsig-Tansig and Tansig-Tansig).

◦ Elastic modulus: the minimum test MSE (7.82 and 0.39), with the combination of (Tansig-Tansig).

After building the optimum models with 8 and 140 inputs, in the next section of this research, a comparison is carried out between these models in order to determine the best networks, which are more successful in the prediction of the desired properties of SCCs.

4.2 Performance comparison of the optimum ANNs with 8 and 140 input variables, determination and performance evaluation of the best networks

The statistical values obtained from the optimum neural networks with 8 and 140 inputs are compared with each other in Table 7. As shown in this table, for all outputs, Elman ANNs with 140 inputs have a smaller test MSE and consequently a better predictive capability than the

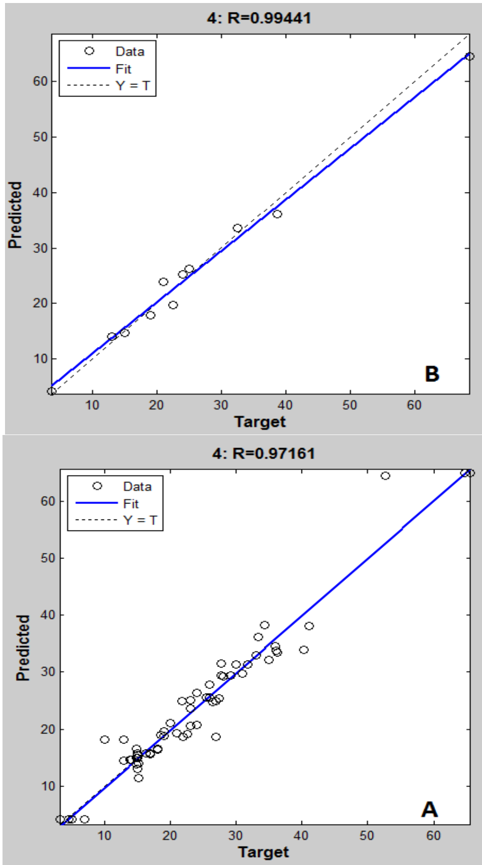

Fig. 3 Elman ANN predicted results versus experimental results (Target) for 3-day compressive strength: (A) training set; (B) testing set

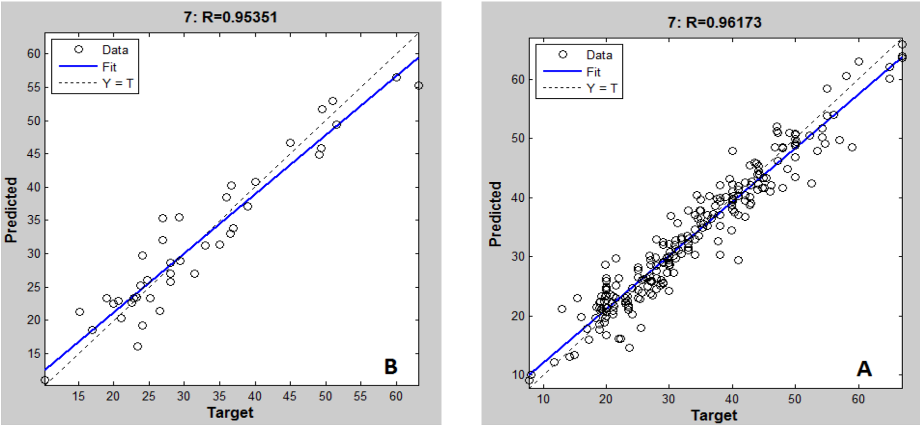

Fig. 4 Elman ANN predicted results versus experimental results (Target) for 7-day compressive strength: (A) training set; (B) testing set

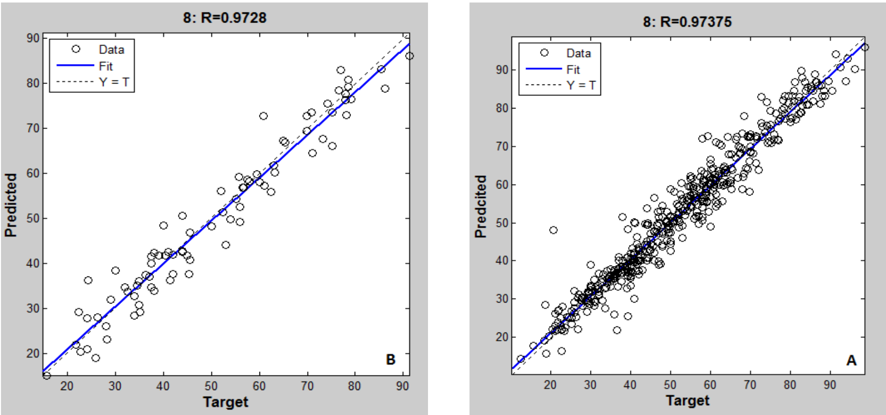

Fig. 5 Elman ANN predicted results versus experimental results (Target) for 28-day compressive strength: (A) training set; (B) testing set

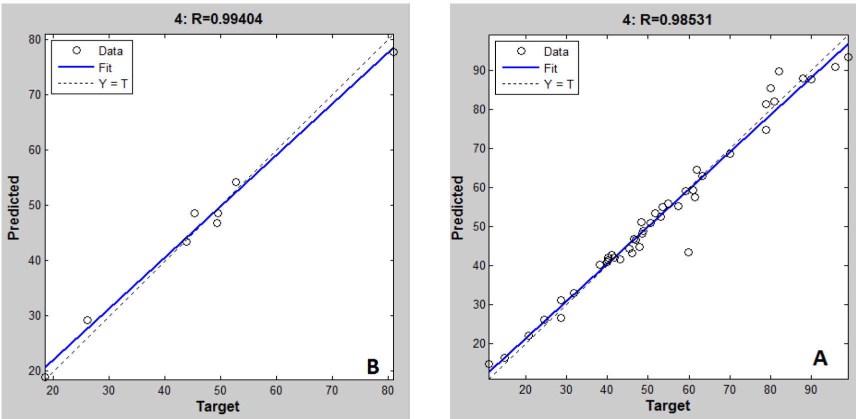

Fig. 6 Elman ANN predicted results versus experimental results (Target) for 56-day compressive strength: (A) training set; (B) testing set

networks with 8 input variables. In other words, in the prediction of the desired outputs, networks with 140 inputs compared to ones with 8, have 67.45, 62.97, 70.41, 58.82 and 70.84 (for 3, 7, 28, 56 and 90-day compressive strength respectively), 51.35 (for 28-day tensile strength), 73.03 (for 28-day flexural strength) and 95.01 (for elastic modulus) percent improvement regarding their test errors. This indicates the prediction accuracy can greatly be improved by the more simulation of the predicted conditions to the experimental conditions, which is achievable via optimizing neural networks by considering a more complete set of key factors affecting the desired outputs, as input variables. Therefore, in this paper, the optimum networks with 140 inputs are selected as the best networks in predicting the mechanical properties of SCC due to having the minimum test errors and the highest correlation coefficients.

The performance of training and testing sets of the best networks, networks with 140 inputs, can be seen in Figs. 3-

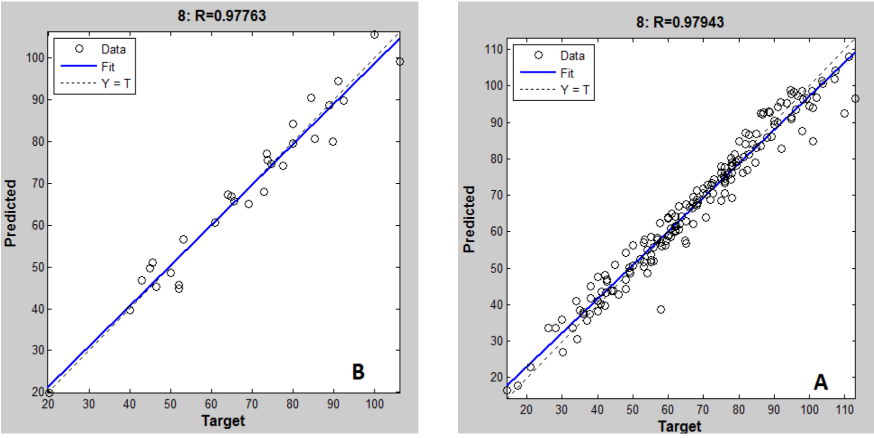

Fig. 7 Elman ANN predicted results versus experimental results (Target) for 90-day compressive strength: (A) training set; (B) testing set

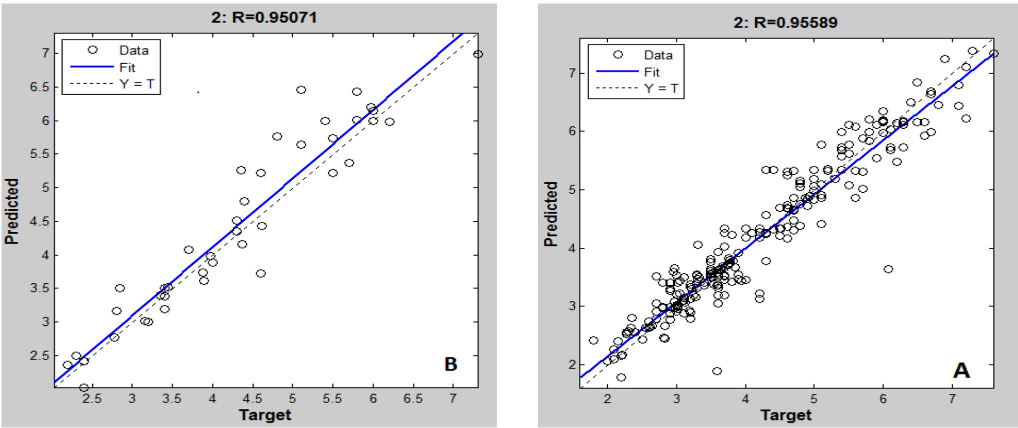

Fig. 8 Elman ANN predicted results versus experimental results (Target) for 28-day tensile strength: (A) training set; (B) testing set

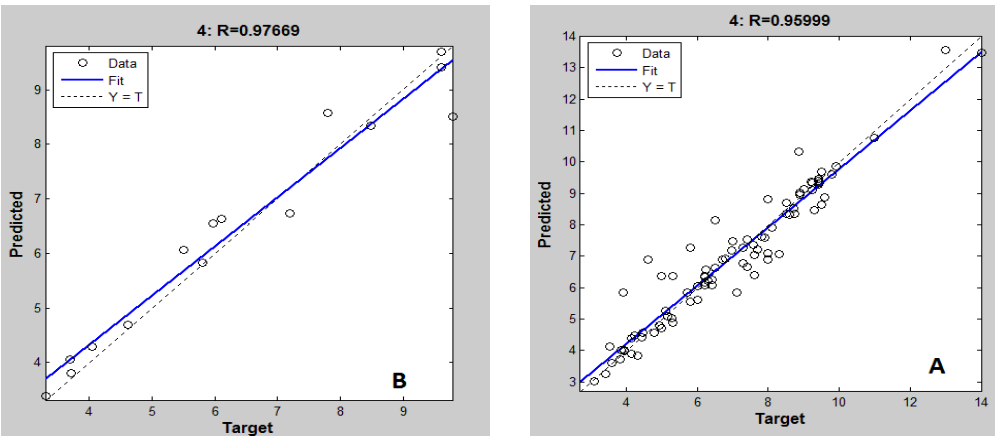

Fig. 9 Elman ANN predicted results versus experimental results (Target) for 28-day flexural strength: (A) training set; (B) testing set

10. As shown in these Figures, the values obtained during the training and testing Elman ANNs are in an excellent correlation with experimental values and it demonstrates that the networks have an acceptable performance in predicting all outputs. Also, this clearly shows that the optimal neural networks learned well the relationship between the considered input and output variables. It should be noted that in some cases (Figs. 3-6-9 and 10), the performance of the model in predicting, for the test set is slightly higher than that for the training set. It means that although due to this fact that ANN models are trained and calibrated based on the training set, the performance of the ANN models for the training data is expected to be higher as compared to that of the test data, but this is not always the case (Akbari and Afshar 2013, Oztas *et al.* 2006). Because the all of the data are randomly divided into the

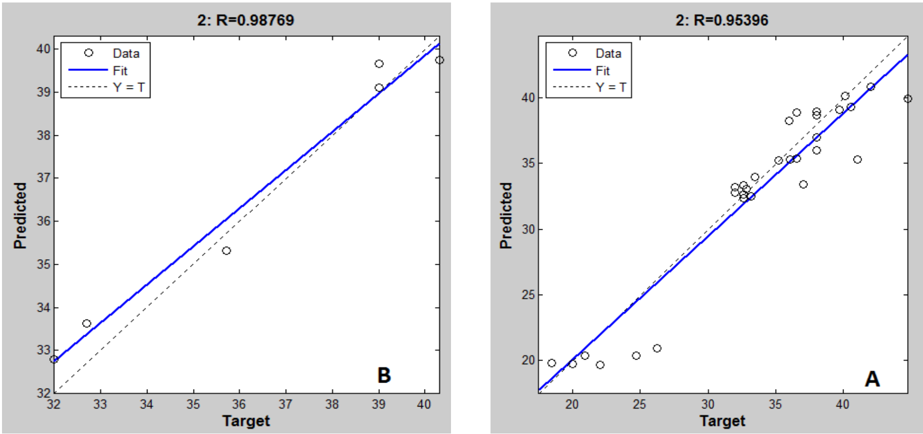

Fig. 10 Elman ANN predicted results versus experimental results (Target) for elastic modulus: (A) training set; (B) testing set

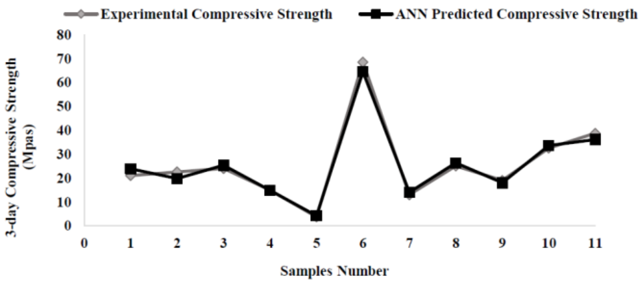

Fig. 11 Comparison between 3-day experimental compressive strength and 3-day ANN predicted compressive strength

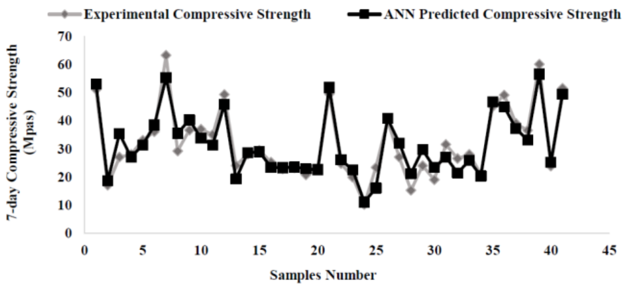

Fig. 12 Comparison between 7-day experimental compressive strength and 7-day ANN predicted compressive strength

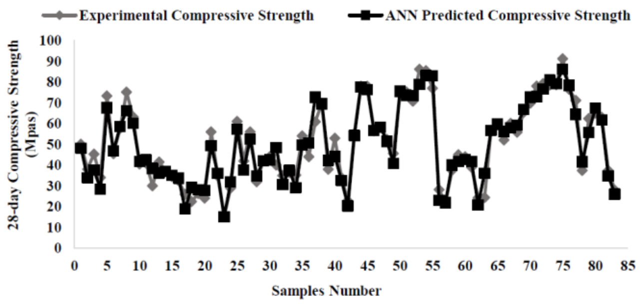

Fig. 13 Comparison between 28-day experimental compressive strength and 28-day ANN predicted compressive strength

training and test sets and any unexpected result may occur. In addition, the differences between the performances in the mentioned cases are not considerable and this issue shows

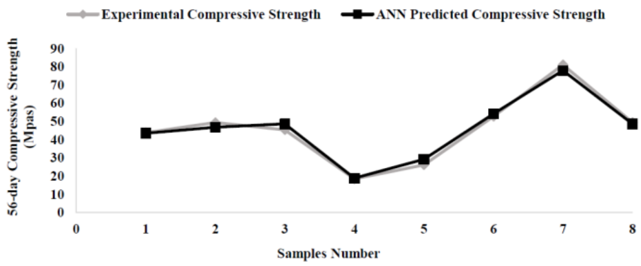

Fig. 14 Comparison between 56-day experimental compressive strength and 56-day ANN predicted compressive strength

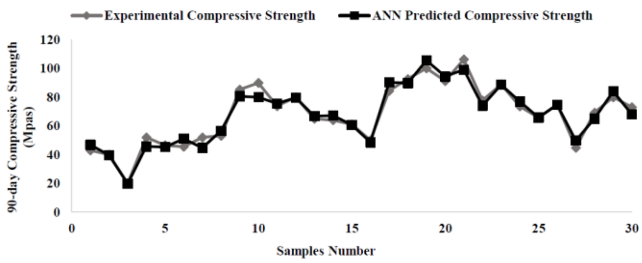

Fig. 15 Comparison between 90-day experimental compressive strength and 90-day ANN predicted compressive strength

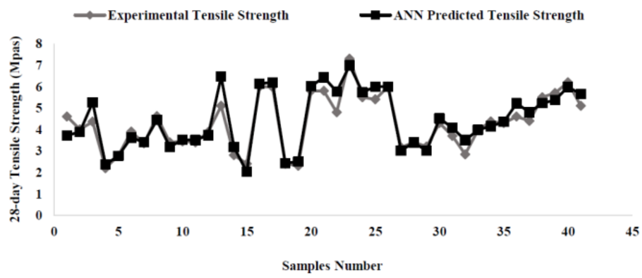

Fig. 16 Comparison between 28-day experimental tensile strength and 28-day ANN predicted tensile strength

that both training and test sets employed have similar characteristics.

Moreover, the experimental and predicted results of the best networks with 140 inputs are compared with each other in Figs. 11-18. As can be seen in these figures, the results obtained from the optimal ANNs have a great similarity to

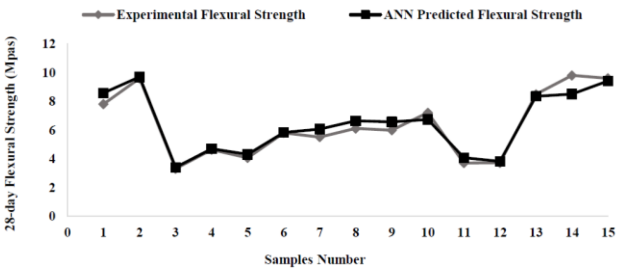

Fig. 17 Comparison between 28-day experimental flexural strength and 28-day ANN predicted flexural strength

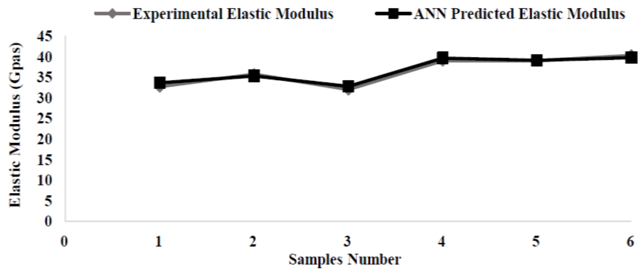

Fig. 18 Comparison between experimental elastic modulus and ANN predicted elastic modulus

the experimental values. This case confirms that the developed Elman networks with a more complete set of effective input variables, are quite successful in predicting all of the desired outputs.

5. Conclusions

Elman artificial neural networks with 8 and 140 input variables were designed to predict the compressive strength, tensile strength, flexural strength and elastic modulus of SCCs. Experimental samples of different concrete mix-designs of SCC were taken from various publications for developing the networks. The number of neurons in the hidden layer, the number of maximum fail iterations and the transfer functions in each layer were optimized based on the minimum values of the test MSE. Furthermore, for training the networks, both raw and scaled data sets were used, which was deduced that for all constructed networks scaling data can lead to improve the test errors. According to the obtained outcomes, despite the dispersion of the used sources in the research, which is expected to reduce the prediction accuracy of the network, Elman ANNs can effectively predict the desired mechanical properties of SCCs without demanding to do any experimental works along with high accuracy. The excellent agreement between the designed models and experimental data illustrated that the ANNs learned well the behavior of the considered variables in the prediction of the desired outputs. It was also found that for all of the desired properties, the optimized networks with 140 inputs have a more satisfactory performance than networks with 8 inputs. This clearly shows that the variables which are selected as the input, directly influence the network errors. Therefore, if inputs are identified and selected more sensitively, the network is modeled by more efficient factors that are available in the experimental conditions, and ultimately the constructed model presents a better performance in predicting the desired output. In other words, whatever the conditions of predicting ANNs through utilizing a more complete set of key factors affecting the desired outputs, as input variables, get closer to experimental conditions, these networks predict the desired outputs more precisely.

Acknowledgements

The authors are thankful to Dr. M. Akbari (Assistant Professor of Civil Engineering, University of Kashan) and Mr. N. Forouhar for their guidance during this investigation.

References

Abdul Hameed, M. (2005), "A study of mix design and durability of self compacting concrete", MSC. Dissertation, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia.

Akbari, M. and Afshar, A. (2013), "Similarity-based error prediction approach for real-time inflow forecasting", *Hydrol. Res*., **45**(4-5), 589-602. https://doi.org/10.2166/nh.2013.098.

Altun, F., Kisi, O. and Aydin, K. (2008), "Predicting the compressive strength of steel fiber added lightweight concrete using neural network", *Comput. Mater. Sci.,* **42**(2), 259-265. https://doi.org/10.1016/j.commatsci.2007.07.011.

Ashtiani, R.S., Little, D.N. and Rashidi, M. (2018), "Neural network based model for estimation of the level of anisotropy of unbound aggregate systems", *Transp. Geotech*., **15**, 4-12. https://doi.org/10.1016/j.trgeo.2018.02.002.

Ayazi, M.H., Tosee, V.R. and Jumaat, M.Z. (2009), "Application of artificial neural networks in compressive strength prediction of lightweight concrete with various percentage of scoria instead of sand", *Eng. e-Tran*., **4**(2), 64-68.

Aydin, A.C. (2007), "Self compactability of high volume hybrid fiber reinforced concrete", *Constr. Build. Mater*., **21**(6), 1149-1154. https://doi.org/10.1016/j.conbuildmat.2006.11.017.

Azizifar, V. and Babajanzadeh, M. (2018), "Compressive strength prediction of self-compacting concrete incorporating silica fume using artificial intelligence methods", *Civil Eng. J*., **4**(7), 1542-1552. https://doi.org/10.28991/cej-0309193.

Badrnezhad, R. and Mirza, B. (2014), "Modeling and optimization of cross-flow ultrafiltration using hybrid neural network-genetic algorithm approach", *J. Ind. Eng. Chem*., **20**(2), 528-543. https://doi.org/10.1016/j.jiec.2013.05.012.

Barbuta, M., Diaconescu, R.M. and Harja, M. (2012), "Using neural networks for prediction of properties of polymer concrete with fly ash", *J. Mater. Civil Eng*., **24**(5), 523-528. https://doi.org/10.1061/(ASCE)MT.1943-5533.0000413.

Beigi, M.H., Berenjian, J., Lotfi Omran, O., Sadeghi Nik, A. and Nikbin, I. (2013), "An experimental survey on combined effects of fibers and nanosilica on the mechanical, rheological, and durability properties of self-compacting concrete", *Mater. Des*., **50**, 1019-1029. https://doi.org/10.1016/j.matdes.2013.03.046.

Bilim, C., Atis, C.D., Tanyildizi, H. and Karahan, O. (2009), "Predicting the compressive strength of ground granulated blast furnace slag concrete using artificial neural network", *Adv. Eng. Softw*., **40**(5), 334-340. https://doi.org/10.1016/j.advengsoft.2008.05.005.

Boga, A.R., Ozturk, M. and Topcu, I.B. (2013), "Using ANN and ANFIS to predict the mechanical and chloride permeability properties of concrete containing GGBFS and CNI", *Compos.*

Part B, **45**(1), 688-696. https://doi.org/10.1016/j.compositesb.2012.05.054.

Champa, H.N. and AnandaKumar, K.R. (2010), "Artificial neural network for human behavior prediction through handwriting analysis", *Int. J. Comput. App.*, **2**(2), 36-41. https://doi.org/10.5120/629-878.

Corinaldesi, V. and Moriconi, G. (2011), "Characterization of self-compacting concretes prepared with different fibers and mineral additions", *Cement Concrete Compos.*, **33**(5), 596-601. https://doi.org/10.1016/j.cemconcomp.2011.03.007.

Cybenko, G. (1989), "Approximation by superpositions of a sigmoidal function. Mathematics of control", *Math. Control. Signal.. Syst.*, **2**(4), 303-314. https://doi.org/10.1007/BF02551274.

Das, S., Pal, P. and Singh, R.M. (2015), "Prediction of concrete mix proportion using ANN technique", *Int. Res. J. Eng. Technol.*, **2**(5), 820-825.

Demir, F. (2008), "Prediction of elastic modulus of normal and high strength concrete by artificial neural networks", *Constr. Build. Mater.,* **22**(7), 1428-1435. https://doi.org/10.1016/j.conbuildmat.2007.04.004.

Desai, V.V., Deshmukh, V.B. and Rao, D.H. (2011), "Pseudo random number generator using elman neural network", *IEEE Recent Advances in Intelligent Computational Systems (RAICS),* Trivandrum, India, September. https://doi.org/10.1109/RAICS.2011.6069312.

Devos, N.J. and Rientjes, T.H.M. (2005), "Constraints of artificial neural networks for rainfall-runoff modelling: trade-offs in hydrological state representation and model evaluation", *Hydrol. Earth. Syst. Sci.*, **9**(1-2), 111-126. https://doi.org/10.5194/hess-9-111-2005.

Dias, W.P.S. and Pooliyadda, S.P. (2001), "Neural networks for predicting properties of concretes with admixtures", *Constr. Build. Mater.*, **15**(7), 371-379. https://doi.org/10.1016/S0950-0618(01)00006-X.

Douglas, R.P. (2004), "Properties of self-consolidating concrete containing type F fly ash", MSC. Dissertation, Northwestern University, Evanston, Illinois.

Dubey, R. and Kumar, P. (2012), "Effect of superplasticizer dosages on compressive strength of self-compacting concrete", *Int. J. Civil Struct. Eng.,* **3**(2), 360-366. https://doi.org/10.6088/ijcser.201203013034.

Dumne, S.M. (2014), "Effect of superplasticizer on fresh and hardened properties of self-compacting concrete containing fly ash", *Am. J. Eng. Res.*, **3**(3), 205-211.

Elman, J.L. (1990), "Finding structure in time", *Cog. Sci.*, **14**(2), 179-211. https://doi.org/10.1016/0364-0213(90)90002-E.

Fathi, A., Mazari, M. and Saghafi, M. (2019), "Multivariate global sensitivity analysis of rocking responses of shallow foundations under controlled rocking", *Proceedings of the 8th International Conference on Case Histories in Geotechnical Engineering*, Philadelphia, Pennsylvania, March.

Funahashi, K.I. (1989), "On the approximate realization of continuous map- pings by neural networks", *Neur. Netw.*, **2**(3), 183-192. https://doi.org/10.1016/0893-6080(89)90003-8.

Ghafoori, N., Najimi, M., Sobhani, J. and Agel, M.A. (2013), "Predicting rapid chloride permeability of self-consolidating concrete: A comparative study on statistical and neural network models", *Constr. Build. Mater.*, **44**, 381-390. https://doi.org/10.1016/j.conbuildmat.2013.03.039.

Gopi, E.S. (2007), *Algorithm Collections for Digital Signal Processing Applications Using Matlab*, Springer, Netherlands, Heidelberg, Baden-Württemberg, Germany.

Guneyisi, E., Gesoglu, M. and Ozbay, E. (2010), "Strength and drying shrinkage properties of self-compacting concretes incorporating multi-system blended mineral admixtures", *Constr. Build. Mater.*, **24**(10), 1878-1887. https://doi.org/10.1016/j.conbuildmat.2010.04.015.

Hastie, T., Tibshirani, R. and Friedman, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction,* Springer-Verlag New York, NY. USA.

Haykin, S. (1994), *Neural Networks: A Comprehensive Foundation*, MacMillan Publishing Company, New York, NY, USA.

Hornik, K. (1991), "Approximation capabilities of multilayer feedforward networks", *Neur. Netw.*, **4**(2), 251-257. https://doi.org/10.1016/0893-6080(91)90009-T.

Hornik, K., Stinchcombe, M. and White, H. (1989), "Multilayer feedforward networks are universal approximators", *Neur. Netw.*, **2**(5), 359-366. https://doi.org/10.1016/0893-6080(89)90020-8.

Hossain, K.M.A., Lachemi, M., Sammour, M. and Sonebi, M. (2013), "Strength and fracture energy characteristics of self-consolidating concrete incorporating polyvinyl alcohol, steel and hybrid fibers", *Constr. Build. Mater.*, **45**, 20-29. https://doi.org/10.1016/j.conbuildmat.2013.03.054.

Indra Kiran, N.V.N., Pramila devi, M. and Lakshmi, G.V. (2010), "Effective control chart pattern recognition using artificial neural networks", *Int. J. Comput. Sci. Netw. Secur.*, **10**(3), 194-199.

Jalal, M., Mansouri, E., Sharifipour, M. and Pouladkhan, A.R. (2012), "Mechanical, rheological, durability and microstructural properties of high performance self-compacting concrete containing SiO_2 micro and nanoparticles", *Mater. Des.*, **34**, 389-400. https://doi.org/10.1016/j.matdes.2011.08.037.

Kalogirou, S.A. (2000), "Applications of artificial neural-networks for energy systems", *Appl. Energy*, **67**(1-2), 17-35. https://doi.org/10.1016/S0306-2619(00)00005-2.

Kanellopoulas, I. and Wilkinson, G.G. (1997), "Strategies and best practice for neural network image classification", *Int. J. Remote Sens.*, **18**(4), 711-725. https://doi.org/10.1080/014311697218719.

khademi, F.S. and Jamal, S.M.M. (2016), "Predicting the 28 days compressive strength of concrete using artificial neural network", *J. Civil Eng.*, **6**(2), 1-7. https://doi.org/10.26634/jce.6.2.5936.

Khaleel, O.R., Al-Mishhadani, S.A. and Abdul Razak, H. (2011), "The effect of coarse aggregate on fresh and hardened properties of Self-Compacting Concrete (SCC)", *Procedia Eng.*, **14**, 805-813. https://doi.org/10.1016/j.proeng.2011.07.102.

Koker, R. (2006), "Design and performance of an intelligent predictive controller for a six-degree-of-freedom robot using the Elman network", *Inf. Sci.*, **176**(12), 1781-1799. https://doi.org/10.1016/j.ins.2005.05.002.

Krenker, A., Bešter, J. and Kos, A. (2011), *Introduction to the Artificial Neural Networks*, Ed. Suzuki, K., *Artificial Neural Networks: Methodological Advances and Biomedical Applications*, Intech, Rijeka, Carotia.

Krishna, A.V., Krishna Rao, B. and Rajagopal, A. (2010), "Effect of different sizes of coarse aggregate on the properties of NCC and SCC", ***Int. J. Eng. Sci. Technol*., 2**(10), 5959-5965.

Kumar, R., Madan, S.K., Devgan, N.P. and Roshan, L. (2013), "An experimental study on performance of self compacting concrete contaning lime stone quarry fines and fly ash", *Asian J. Civil Eng.*, **15**(3), 421-433.

Li, F.X., Yu, Q.J., Wei, J.X. and Li, J.X. (2011), "Predicting the workability of self-compacting concrete using artificial neural network", *Adv. Mater. Res.*, **168-170**, 1730-1734. https://doi.org/10.4028/www.scientific.net/AMR.168-170.1730.

Lohr, S.L. (2009), *Sampling: Design and Analysis*, 2th Edition, Cengage Learning, Boston, Massachusetts, USA.

Mahajan, S. and Singh, D. (2013), "Fresh & hardened preperties of self compacting concrete incorporating different binder materials", *Int. J. Emerg. Technol. Adv. Eng.*, **3**(12), 689-693.

Masters, T. (1993), *Practical Neural Network Recipes in C++*, Academic Press Professional, San Diego, CA, USA.
MATLAB Software (2013), Neural Network Toolbox: mapminmax, R2013b, Version: (8.2.0.701).
May, R.J., Maier, H.R. and Dandy, G.C. (2010), "Data splitting for artificial neural networks using som-based stratified sampling", *Neur. Netw.*, **23**(2), 283-294. https://doi.org/10.1016/j.neunet.2009.11.009.
Moyo, V. and Sibanda, K.H. (2015), "Training set size for generalization ability of artificial neural networks in forecasting TCP/IP traffic trends", *Int. J. Comput. Appl.*, **113**(13), 14-19.
Muluneh, A. (2014), "Control chart pattern recognition for multivariate autocorrelated processes using artificial neural network", MSC. Dissertation, Addis Ababa University of Mechanical and Industrial Engineering, Addis Ababa, Ethiopia.
Nazari, A. and Riahi, S.H. (2011), "Prediction split tensile strength and water permeability of high strength concrete containing TiO_2 nanoparticles by artificial neural network and genetic programming", *Compos. Part B.*, **42**(3), 473-488. https://doi.org/10.1016/j.compositesb.2010.12.004.
Ouchi, M., Nakamura, S., Osterson, T., Hellberg, S. and Lwin, M. (2003), "Applications of self- compacting concrete in Japan, Europe and the United States", *Proceedings of the 5th International Symposium on High Performance Computing, ISHPC*, Tokyo-Odaiba, Japan, October.
Oztas, A., Pala, M., Ozbay, E., Kanca, E., Caglar, N. and Bhatti, A. (2006), "Predicting the compressive strength and slump of high strength concrete using neural network", *Constr. Build. Mater.*, **20**(9), 769-775. https://doi.org/10.1016/j.conbuildmat.2005.01.054.
Pala, M., Özbay, E., Öztas, A. and Yuce, M.I. (2007), "Appraisal of long-term effects of fly ash and silika fume on compressive strength of concrete by neural networks", *Constr. Build. Mater.*, **21**(2), 384-94. https://doi.org/10.1016/j.conbuildmat.2005.08.009.
Panagoulia, D., Tsekouras, G.J. and Kousiouris, G.A. (2017), "Multi-stage methodology for selecting input variables in ANN forecasting of river flows", *Global. Nest. J.*, **19**(1), 49-57. https://doi.org/10.30955/gnj.002067.
Patel, A., Bhuva, P., George, E. and Bhatt, D. (2011), "Compressive strength and nodulus of elasticity of self-compacting concrete", *National Conference on Recent Trends in Engineering & Technology*, Gujarat, India, May.
Piryonesi, S.M. and El-Diraby, T.E. (2018), "Using data analytics for cost-effective prediction of road conditions: Case of the pavement condition index", Research Report No. FHWA-HRT-18-065, U.S. Department of Transportation, Federal Highway Administration, Washington, DC.
Prasad Meesaraganda, L.V., Saha, P. and Tarafder, N. (2019), *Artificial Neural Network for Strength Prediction of Fibers' Self-compacting Concrete*, Eds. Bansal, J., Das, K., Nagar, A., Deep, K., and Ojha, A., *Soft Computing for Problem Solving, Advances in Intelligent Systems and Computing*, Springer, Singapore, Malaysia.
Provost, F. and Fawcett, T. (2013), *Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking*, O'Reilly Media, Sebastopol, CA.
Rajaram, M., Ravichandran, A. and Muthadh, A. (2018), "Prediction on mechanical properties of hybrid fiber concrete using MatLab", *Int. Res. J. Eng. Technol.*, **5**(7), 2426-2432.
Ranjbar, M.M., Madandoust, R., Mousavi, S.Y. and Yosefi, S. (2013), "Effects of natural zeolite on the fresh and hardened properties of self-compacted concrete", *Constr. Build. Mater.*, **47**, 806-813. https://doi.org/10.1016/j.conbuildmat.2013.05.097.
Reitermanova, Z. (2010), "Data splitting", *Proceedings of the 19th Annual Conference of Doctoral Students, Part I, Mathematics and Computer Sciences*, Prague, Jun.
Richards, E.L. (1991), "Generalization in neural networks: experiments in speech recognition", Ph.D. Dissertation, University of Colorado, Boulder, Colorado.
Sahmaran, M., Yurtseven, A. and Ozgur Yaman, I. (2005), "Workability of hybrid fiber reinforced self-compacting concrete", *Build. Environ.*, **40**(12), 1672-1677. https://doi.org/10.1016/j.buildenv.2004.12.014.
Shah, S.V., Pawar, D.A., Patil, A.S., Bhosale, P.S., Subhedar, A.S. and Bhosale, G.D. (2018), "Concrete mix design using artificial neural network", *Int. J. Adv. Res. Sci. Eng.*, **7**(3), 251-259.
Shahin, M.A., Jaksa, M.B. and Maier, HR. (2009), "Recent advances and future challenges for artificial neural systems in geotechnical engineering applications", *Adv. Artif. Neur. Syst.*, **2009**, 1-9. http://dx.doi.org/10.1155/2009/308239.
Soleymani, F. and Karimi Livary, A. (2012), "Artificial neural network for predicting flexural strength of concrete containing Cr_2O_3 nanoparticles", *J. Am. Sci.*, **8**(8), 155-162.
Sonebi, M., Grünewald, S., Cevik, A. and Walraven, J. (2016), "Modelling fresh properties of self-compacting concrete using neural network technique", *Comput. Concrete*, **18**(4), 903-921. https://doi.org/10.12989/cac.2016.18.4.903.
Topcu, I.B. and Sarıdemir, M. (2007), "Prediction of properties of waste AAC aggregate concrete using artificial neural network", *Comp. Mater. Sci.*, **41**(1), 117-125. https://doi.org/10.1016/j.commatsci.2007.03.010.
Turk, K. and Karatas, M. (2011), "Abrasion resistance and nechanical preperties of self-compacting concrete with different dosage of fly Ash/Silica fume", *Ind. J. Eng. Mater. S.*, **18**(1), 49-60.
Uysal, M. and Tanyildizi, H. (2012), "Estimation of compressive strength of self- compacting concrete containing polypropylene fiber and mineral additives exposed to high temperature using artificial neural network", *Constr. Build. Mater.*, **27**(1), 404-414. https://doi.org/10.1016/j.conbuildmat.2011.07.028.
Uysal, M. and Yilmaz, K. (2011), "Effect of mineral admixtures on properties of self-compacting concrete", *Cement Concrete Compos.*, **33**(7), 771-776. https://doi.org/10.1016/j.cemconcomp.2011.04.005.
Vakhshouri, B. and Nejadi, SH. (2015), "Predication of compressive strength in light-weight self-compacting concrete by ANFIS analytical mModel", *Arch. Civil Eng.*, **2**, 53-72. https://doi.org/10.1515/ace-2015-0014.
Venkateswara Rao, S., Seshagiri Rao, M.V. and Rathish Kumar, P. (2010), "Effect of size of aggregate and fines on standard and high strength self compacting concrete", *J. Appl. Sci. Res.*, **6**(5), 433-442.
Wang, H., Guo, W., Zhao, R., Zhou, B. and Hu, C. (2018), *A Real-Time Online Security Situation Prediction Algorithm for Power Network Based on Adaboost and SVM*, Eds. Yang, C.N., Peng S.L. and Jain, L., *Security with Intelligent Computing and Big-data Services*, *Advances in Intelligent Systems and Computing,* Springer, Cham, Basel, Switzerland.
Wu, W., May, R., Dandy, G.C. and Maier, H.R. (2012), "A method for comparing data splitting approaches for developing hydrological ANN models", *Proceedings of the 6th International Congress on Environmental Modelling and Software Managing Resources of a Limited Planet*, Leipzig, Germany, July.
Yazicioglu, S., Caliskan, S. and Turk, K. (2006), "Effect of curing condition on the engineering properties of self-compacting concrete", *Ind. J. Eng. Mater.*, **13**(1), 25-29.
Yeh, I. (1998), "Modeling of strength of high-performance concrete using artificial neural networks", *Cement Concrete Res.*, **28**(12), 1797-1808. https://doi.org/10.1016/S0008-8846(98)00165-3.
Yuan, Z., Wang, L.N. and Ji, X. (2014), "Prediction of concrete compressive strength: Research on hybrid models genetic based

algorithms and ANFIS", *Adv. Eng. Softw.,* **67**, 156-163. https://doi.org/10.1016/j.advengsoft.2013.09.004.

CC