Neural Comput & Applic (2019) 31 (Suppl 1):S409–S424
DOI 10.1007/s00521-017-3007-7

ORIGINAL ARTICLE

Self-compacting concrete strength prediction using surrogate models

Panagiotis G. Asteris[1] · **Konstantinos G. Kolovos**[2]

Received: 7 October 2016 / Accepted: 8 April 2017 / Published online: 28 April 2017

Abstract Despite the extensive use of self-compacting concrete in constructions over the last decades, there is not yet a robust quantitative method, available in the literature, which can reliably predict its strength based on its mix components. This limitation is due to the highly nonlinear relation between the self-compacting concrete's compressive strength and the mixed components. In this paper, the application of artificial neural networks for predicting the mechanical characteristics of self-compacting concrete has been investigated. Specifically, surrogate models (such as artificial neural network models and a new proposed normalization method) have been used for predicting the 28-day compressive strength of admixture-based self-compacting concrete (based on experimental data available in the literature). The comparison of the derived results with the experimental findings demonstrates the ability of artificial neural networks to approximate the compressive strength of self-compacting concrete in a reliable and robust manner. Furthermore, the proposed formula for the normalization of data has been proven effective and robust compared to available ones.

Keywords Artificial neural networks · Back propagation neural networks · Compressive strength · Self-compacting concrete

✉ Panagiotis G. Asteris
asteris@aspete.gr; panagiotisasteris@gmail.com

Konstantinos G. Kolovos
kolovosk@gmail.com

[1] Computational Mechanics Laboratory, School of Pedagogical and Technological Education, 141 21 Heraklion, Athens, Greece

[2] Department of Physical Sciences and Applications, Hellenic Army Academy, Vari, Greece

1 Introduction

Among numerous trends and developments in constructions area over the last decades, the introduction of self-compacting concrete (SCC) is of high interest for the exploitation of alternative raw materials, by-products, wastes and secondary materials as mineral additives. It is commonly characterized as a special concrete with enhanced fluid properties such as increased flowability and good segregation resistance and can settle by its own weight even at the presence of congested reinforcement at deep and narrow element sections of nonconventional geometry. Therefore, SCC can consolidate itself without requiring the use of internal or external vibration during the placing process, thus avoiding segregation and bleeding and at the same time maintaining its stability [4, 23]. Moreover, the potential use of SCC in lightweight applications has drawn significant attention [32].

Due to its complex composition, a proper mix design process is necessary for SCC in order to accomplish its desirable properties. For this design process, the available materials must be taken into account, proportioned with one or more mineral as well as chemical admixtures. The challenge of enhancing grain size distribution and particle packing, thus ensuring greater cohesiveness for SCC, is addressed by seeking optimum balance between coarse and fine materials and the chemical admixtures. According to [18], variations in cement and/or mineral additives due to changes in the production process as well as in the aggregate type may cause large variations on the properties of fresh SCC. Therefore, it is of great importance to have a robust mixture, which is minimally affected by the external sources of variability. Towards this direction, the utilization of powder industrial by-products and wastes as environmental friendly mineral additives for the production of lightweight SCC has attracted the attention of researchers [11, 32, 35, 37]. Furthermore, a wide variety of

secondary materials exists for the mix design process, such as limestone powder (LP), fly ash (FA), ground granulated blast furnace slag (GGBFS), silica fume (SF), rice husk ash (RHA) and, as chemical admixtures, new generation superplasticizers (SP) and viscosity-modifying admixtures (VMA) [4, 11, 17, 18, 23, 32, 35, 37, 56, 67].

Artificial neural networks (ANNs) have emerged the last decades as an attractive meta-modelling technique applicable to a vast number of scientific fields including material science, among others. The main characteristic of this method is that a surrogate model can be constructed after a training process with only a few available data, which can be used in order to predict pre-selected model parameters, reducing the need for time- and money-consuming experiments. So far, literature includes publications in which ANNs were used for predicting the compressive strength and modulus of elasticity [22, 39, 63, 64] and for modelling the characteristics of concrete materials [1, 13, 44, 66]. Moreover, similar methods such as fuzzy logic and genetic algorithms have also been used for modelling the compressive strength of concrete materials [3, 12, 46]. Detailed and in-depth state of the art report can be found in [2, 42, 43, 57].

In this context, in the present work, the application of properly trained ANN models for the prediction of the 28-day compressive strength of admixture-based self-compacting concrete is presented. The database consists of 205 specimens (taken from the literature) having mixture composition with comparable physical and chemical properties. The developed ANN models take into consideration 11 SCC composition parameters in order to predict the compressive strength and have been proven to be very successful, exhibiting very reliable predictions.

2 Artificial neural networks

This section summarizes the mathematical and computational aspects of artificial neural networks. In general, ANNs are information-processing models configured for a specific application through a training process. A trained ANN maps a given input into a specific output and thereby can be considered to be similar to a response surface method. This main advantage of a trained ANN over conventional numerical analysis procedures (e.g. regression analysis) is that the results can be produced with much less computational effort [2, 5–8, 30, 33, 51–53].

2.1 General

The concept of an artificial neural network is based on the concept of the biological neural network of the human brain. The basic building block of the ANN is the artificial neuron, which is a mathematical model trying to mimic the behaviour of the biological neuron. Information is passed into the artificial neuron as input, and it is processed with a mathematical function leading to an output which determines the behaviour of the neuron (similar to fire or not situation for the biological neuron). Before the information enters the neuron, it is weighted in order to approximate the random nature of the biological neuron. A group of such neurons consists of an ANN in a manner similar to biological neural networks. In order to set up an ANN, one needs to define (i) the architecture of the ANN, (ii) the training algorithm, which will be used for the ANN learning phase, and (iii) the mathematical functions describing the mathematical model. The architecture or topology of the ANN describes the way the artificial neurons are organized in the group and how information flows within the network. For example, if the neurons are organized in more than one layer, then the network is called a multilayer ANN. Regarding the training phase of the ANN, it can be considered as a function minimization problem in which the optimum value of weights needs to be determined by minimizing an error function. Depending on the optimization algorithms used for this purpose, different types of ANN exist. Finally, the two mathematical functions that define the behaviour of each neuron are the summation function and the activation function. In the present study, we use a back-propagation neural network (BPNN) which is described in the next section.

2.2 Architecture of BPNN

A BPNN is a feed-forward, multilayer network i.e. information flows only from the input towards the output with no back loops and the neurons of the same layer are not connected to each other, but they are connected with all the neurons of the previous and subsequent layer. A BPNN has a standard structure that can be written as

$$\mathbf{N}\text{–}\mathbf{H}_1\text{–}\mathbf{H}_2\text{–}\cdots\text{–}\mathbf{H}_{\mathbf{NHL}}\text{–}\mathbf{M} \quad (1)$$

where $\mathbf{N}$ is the number of input neurons (input parameters), $\mathbf{H}_i$ is the number of neurons in the i^{th} hidden layer for $i = 1, \ldots,$ NHL, NHL is the number of hidden layers and $\mathbf{M}$ is the number of output neurons (output parameters). Figure 1 depicts an example of a BPNN composed of an input layer with five neurons, two hidden layers with four and three neurons respectively and an output layer with two neurons, i.e. a 5-4-3-2 BPNN.

A notation for a single node (with the corresponding R-element input vector) of a hidden layer is presented in Fig. 2.

For each neuron i, the individual element inputs $p_1, \ldots, p_R$ are multiplied by the corresponding weights $w_{i,1}, \ldots, w_{i,R}$ and the weighted values are fed to the junction of the summation function in which the dot product $(W \cdot p)$ of the weight vector $\mathbf{W} = [w_{i,1}, \ldots, w_{i,R}]$ and the input vector $\boldsymbol{p} = [p_1, \ldots, p_R]^T$ is generated. The threshold b (bias) is added

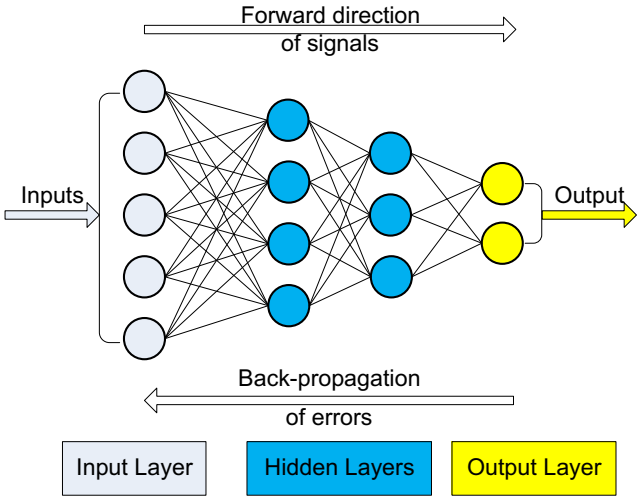

Fig. 1 A 5-4-3-2 BPNN

to the dot product forming the net input n which is the argument of the transfer function f:

$$n = \boldsymbol{W}\cdot\boldsymbol{p} = w_{i,1}p_1 + w_{i,2}p_2 + \ldots + w_{i,R}p_R + b \quad (2)$$

The choice of the transfer (or activation) function f may strongly influence the complexity and performance of the ANN. Although sigmoidal transfer functions are the most commonly used, one may use different types of functions. Previous studies [9, 36] have proposed a large number of alternative transfer functions. In the present study, the logistic sigmoid and the hyperbolic tangent transfer functions were found to be appropriate for the problem investigated. During the training phase, the training data are fed into the network which tries to create a mapping between the input and the output values. This mapping is achieved by adjusting the weights by minimizing the following error function

$$E = \sum (t_i - o_i)^2 \quad (3)$$

where t_i and o_i are the exact value and the prediction of the network, respectively, within an optimization framework. The

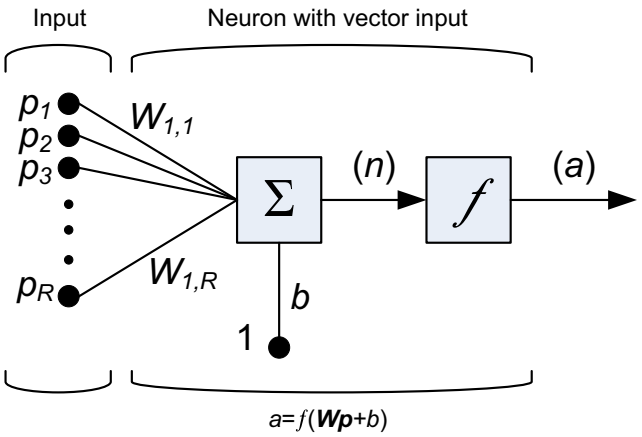

Fig. 2 A neuron with a single R-element input vector

training algorithm used for the optimization plays a crucial role in building a quality mapping, and an exhaustive investigation was performed in order to find the most suitable for this problem. The most common method used in literature is the back-propagation technique in which, as stated by its name, the information propagates to the network in a backward manner in order to adjust the weights and minimize the error function. To adjust the weights properly, a general method called gradient descent is applied in which the gradients of the error function with respect to the network weights are calculated. Further discussion on the training algorithms is made in the numerical example section.

2.3 Dealing with overfitting

One of the most common problems that occur during the training phase of an ANN is the overfitting. In this stage, the network has learned the available training data very well (very small value for the error function), but when new data are provided to the network, this error increases significantly and the network's prediction is poor. In order to prevent overfitting, several techniques/algorithms and criteria have been proposed for determining the number of hidden layers as well the number of neurons of each layer. Furthermore, the training of the ANN can be terminated before the network has the opportunity to learn the data very well and a regularization term can be added in the objective function in order to smooth the mapping [7, 8, 14–16, 20, 30, 38, 47].

2.4 Proposed algorithm

In the present work, a simple heuristic algorithm is proposed in order to obtain a reliable and robust ANN for predicting the 28-day compressive strength of admixture-based self-compacting concrete. The steps of the proposed algorithm are the following:

Step 1. Normalization of data: The normalization is a preprocessing phase which has been proved to be the most crucial step of any type of problem in the field of soft computing techniques such as the artificial neural network techniques.

Step 2. Development and training of several ANNs: The development and training of the ANNs occurs with a number of hidden layers ranging from 1 to 2 and with a number of neurons ranging from 4 to 20. Each one of the ANNs is developed and trained for a number (nf) of different activation functions as well as with and without the use of data preprocessing techniques (step 1).

Step 3. Determination of the mean square error: For each one of the above trained NNs, the mean square error (MSE) is computed for a set of data (validation

data), which have not been used during the training phase (training data) of the ANNs.

Step 4. Establishment of upper and lower limits: Upper and lower limits are introduced for each one of the output parameters based on experimental or numerical data as well as reasonable estimations by the users.

Step 5. Selection of optimum architecture: The optimum architecture is the one that gives the minimum mean square error while all the computed output parameters for all the validation data are between the upper and lower limits.

It should be emphasized that the importance of the limits established at step 4 is based on the user's expertise and experience to the specific field for making reasonable assumptions.

3 Results and discussion

In this section, the above proposed algorithm is presented step by step for tuning optimum ANNs used for the prediction of the 28-day compressive strength of admixture-based self-compacting concrete based on availability in the literature experimental data.

3.1 Experimental

The database used herein consists of 205 mixes obtained from literature ([10, 19, 23–29, 31, 32, 45, 50, 54, 55, 58–62, 65, 67]) (Table 1).

Each input training vector $\boldsymbol{p}$ is of dimension 1 × 11 and consists the values of the 11 fillers ($R = 11$) namely, the cement (C), the coarse aggregate (CA), the fine aggregate (FA), the water (W), the limestone powder (LP), the fly ash (FA), the ground granulated blast furnace slag (GGBFS), the silica fume (SF), the rice husk ash (RHA) and, as chemical admixtures, the new generation superplasticizers (SP) and the viscosity modifying admixtures (VMA). The corresponding output training vectors are of dimension 1 × 1 and consist the value of the 28-day compressive strength of the SCC specimens. Their mean values together with the minimum and maximum values are listed in Table 2.

3.2 Sensitivity analysis

In general, sensitivity analysis of a numerical model is a technique used to determine if the output of the model (response, stress, deformation, stresses, etc.) is affected by changes in the assumptions of the inputs (Young's modulus, fillers, etc.). During the development of an ANN, it is of high importance to know the effect of each one of the above 11 composition parameters (network inputs) on the compressive strength of SCC (network output). This will provide feedback as to which input parameters are the most significant, and thus, by removing the insignificant ones, the input space will be reduced and subsequently the complexity of the ANN as well as the training times required for its training will be also reduced. The results obtained from the sensitivity analysis are presented in Fig. 1. Based on these results, the viscosity-modifying admixtures (VMA) parameter has the strongest impact on the compressive strength while the fine aggregate (FA) parameter has the weakest impact (Fig. 3).

3.3 Normalization of data

As mentioned previously, the normalization of the input and output parameters has a significant impact on the ANN training. In the present study, during the pre-processing stage, the min–max [21] and the z-score normalization methods have been used. In particular, the 11 input parameters (Table 1) as well as the single output parameter have been normalized using the min–max normalization method. As stated in [34], in order to avoid problems associated with low learning rates of the ANN, the normalization of the data should be made within a range defined by appropriate upper and lower limit values of the corresponding parameter. In this work, the input and output parameters have been normalized in the range [0, 1] and [−1, 1], respectively. Moreover, in this work, a transformation technique called *Central* has been applied, in which the origin of the training data is shifted to the centre of the data with the following formula:

$$z_i = x_i - \frac{\max(x) + \min(x)}{2} \tag{4}$$

where $\boldsymbol{x}$ ($\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$) are the original data and $\boldsymbol{z_i}$ is the $\boldsymbol{i}^{\mathbf{th}}$ transformed data. The results obtained using this technique for pre-processing the data were better compared to the results obtained by other known normalization techniques found in literature, as will be demonstrated in the section of the numerical examples.

3.4 Training algorithms

In order to find the training algorithm that is more suitable to tackle the nonlinear behaviour of the SCC's compressive strength, the performance of various optimization techniques such as the quasi-Newton, resilient, one-step secant, gradient descent with momentum and adaptive learning rate and the Levenberg-Marquardt method have been investigated. It should be mentioned that all the ANNs under study (they will be presented in the next section) have been investigated by means of all the aforementioned training algorithms. Among these algorithms, the best—by far—ANN prediction of the output parameter was achieved by using the Levenberg-

Table 1 Experimental data/results and input and output parameters of BPNNs

Sample	Input											Output	Comments
	Cement	Limestone powder	Fly ash	GGBS	Silica fume	RHA	Coarse aggregate	Fine aggregate	Water	SP	VMA	Compressive strength	
1	225	0	225	0	0	0	865	898	179	4.5	0.315	33.94	T
2	231	0	231	0	0	0	862	864	175	4.62	0.37	38.9	T
3	258	0	258	0	0	0	835	836	176	5.16	0.413	41.21	V
4	330	0	220	0	0	0	826	827	176	5.5	0.44	45.95	T
5	360	0	240	0	0	0	797	796	180	6	0.48	53	T
6	360	0	240	0	0	0	812	813	168	6	0.48	56	Test
7	400	0	250	0	0	0	785	785	172	9.75	0.46	61.64	T
8	450	0	250	0	0	0	760	760	174	11.2	0.49	66.29	T
9	390	0	58	0	0	0	881	874	156	9.75	0	25.5	V
10	384	0	0	0	38	0	863	836	211.2	9.216	0	29	T
11	310	189	0	0	0	0	667	1018	170	6	0	51.2	T
12	315	164	0	0	0	0	673	1025	173	5.51	0	50.7	Test
13	320	153	0	0	0	0	687	1016	174	5.21	0	53.6	T
14	330.36	0	178.6	0	0	0	739.44	875.16	179.74	5.97	0.92	48.4	T
15	330.85	0	180.56	0	0	0	739.44	857.57	180.05	6.02	0.97	57.3	V
16	331.33	0	182.47	0	0	0	739.44	840	180.38	6.08	1.03	63.8	T
17	330.36	0	180.56	0	0	0	754.18	875.16	180.05	6.08	1.03	61.3	T
18	330.85	0	182.47	0	0	0	754.18	857.57	180.38	5.97	0.92	50.45	Test
19	331.33	0	178.6	0	0	0	754.18	840	179.74	6.02	0.97	53.1	T
20	330.85	0	178.6	0	0	0	768.88	875.16	180.38	6.02	1.03	61.1	T
21	331.33	0	180.56	0	0	0	768.88	857.57	179.74	6.08	0.92	55.9	V
22	330.36	0	182.47	0	0	0	768.88	840	180.05	5.97	0.97	55.3	T
23	331.33	0	182.47	0	0	0	739.44	875.16	180.05	6.02	0.92	57.3	T
24	330.36	0	187.6	0	0	0	739.44	857.57	180.38	6.08	0.97	65.6	Test
25	330.85	0	180.56	0	0	0	739.44	840	179.74	5.97	1.03	53.6	T
26	330.85	0	182.47	0	0	0	754.18	875.16	179.74	6.08	0.97	52.9	T
27	331.33	0	178.6	0	0	0	754.18	857.57	180.05	5.97	1.03	47.3	V
28	330.36	0	180.56	0	0	0	754.18	840	180.38	6.02	0.92	59.26	T
29	331.33	0	180.56	0	0	0	768.88	875.16	180.38	5.97	0.97	51.53	T
30	330.36	0	182.47	0	0	0	768.88	857.57	179.74	6.02	1.03	56.6	Test
31	330.85	0	178.6	0	0	0	768.88	840	180.05	6.08	0.92	58.45	T
32	540	0	60	0	0	0	750	900	200	12	0	78.05	T
33	420	0	180	0	0	0	750	900	192	12	0	79.19	V
34	570	0	0	0	30	0	750	900	200	12	0	80.42	T
35	540	0	0	0	60	0	750	900	192	12	0	79.18	T
36	540	0	0	0	0	60	750	900	209	12	0	77.82	Test
37	510	0	0	0	0	90	750	900	209	12	0	77.15	T
38	360	0	180	0	60	0	750	900	200	12	0	70.67	T
39	360	0	180	0	0	60	750	900	200	12	0	73.7	V
40	407	0	244	0	0	0	761	815	181	7.5	0	70.4	T
41	428	0	257	0	0	0	736	788	188	7.9	0	74.5	T
42	438	0	263	0	0	0	723	774	191	8.1	0	69.5	Test
43	458	0	275	0	0	0	698	748	190	8.4	0	68.2	T
44	350	0	0	0	150	0	600	900	175	7.35	0	48.88	T
45	300	0	0	0	200	0	600	900	175	6.21	0	42.23	V
46	250	0	0	0	250	0	600	900	175	5	0	35.14	T

Table 1 (continued)

Sample	Input											Output	Comments
	Cement	Limestone powder	Fly ash	GGBS	Silica fume	RHA	Coarse aggregate	Fine aggregate	Water	SP	VMA	Compressive strength	
47	500	71	0	0	0	0	639	967	173.5	6.75	0	62.2	T
48	350	69.2	150	0	0	0	621	939	173.5	6.75	0	57.3	Test
49	300	68.3	200	0	0	0	613	927	175.5	6.75	0	59.1	T
50	250	67.8	250	0	0	0	608	920	173.9	6.75	0	40.8	T
51	200	67.1	300	0	0	0	603	912	173.5	6.75	0	38.1	V
52	150	66.4	350	0	0	0	597	902	173.5	6.75	0	34.4	T
53	350	69	150	0	0	0	620	937	169	6.75	0	52.4	T
54	300	68.9	200	0	0	0	618	935	162	6.75	0	52.3	Test
55	250	69.3	250	0	0	0	622	941	149.5	6.75	0	47.5	T
56	200	68.4	300	0	0	0	614	929	149.5	6.75	0	39.9	T
57	150	68.2	350	0	0	0	613	927	142.8	6.75	0	32.8	V
58	467	0	83	0	0	0	762	865	182	8.53	0	74.2	T
59	412	0	138	0	0	0	752	887	182	8.25	0	73.4	T
60	357	0	193	0	0	0	742	878	182	7.98	0	67.5	Test
61	440	0	0	110	0	0	775	866	182	9.35	0	77.9	T
62	330	0	0	220	0	0	772	863	182	9.08	0	74.8	T
63	220	0	0	330	0	0	769	861	182	8.8	0	71.6	V
64	495	55	0	0	0	0	775	866	182	9.08	0	69.3	T
65	440	110	0	0	0	0	771	863	182	8.8	0	65.2	T
66	385	165	0	0	0	0	768	860	182	8.53	0	60.2	Test
67	377	239	0	0	0	0	562	861	227	3.7	0	56.1	T
68	376	246	0	0	0	0	577	886	203	6.5	0	49.7	T
69	377	247	0	0	0	0	593	898	181	7.9	0	46.8	V
70	376	263	0	0	0	0	609	932	158	9	0	42.8	T
71	377	272	0	0	0	0	630	963	140	13	0	36.3	T
72	290	0	100	0	0	0	837	913	130.5	2.32	0	42.7	Test
73	250	0	261	0	0	0	837	478	137.5	1.25	0	17	T
74	210	0	100	0	0	0	837	910	136.5	1.68	0	19.1	T
75	250	0	160	0	0	0	837	742	137.5	1.25	0	24.1	V
76	210	0	220	0	0	0	837	786	94.5	1.68	0	26.7	T
77	290	0	100	0	0	0	837	709	188.5	0.58	0	26.6	T
78	290	0	220	0	0	0	837	625	130.5	0.58	0	32.9	Test
79	250	0	160	0	0	0	837	742	137.5	1.25	0	26	T
80	250	0	160	0	0	0	837	742	137.5	1.25	0	28.5	T
81	250	0	160	0	0	0	837	742	137.5	1.25	0	26.4	V
82	250	0	160	0	0	0	837	739	137.5	0	0	27.3	T
83	210	0	100	0	0	0	837	1066	94.5	0.42	0	54.3	T
84	317	0	160	0	0	0	837	594	174.35	1.585	0	29.1	Test
85	210	0	220	0	0	0	837	562	136.5	0.42	0	10.2	T
86	250	0	160	0	0	0	837	742	137.5	1.25	0	25.3	T
87	250	0	160	0	0	0	837	919	95	1.25	0	36.3	V
88	250	0	160	0	0	0	837	746	137.5	2.5	0	26.7	T
89	250	0	160	0	0	0	837	566	180	1.25	0	11	T
90	183	0	160	0	0	0	837	891	100.65	0.915	0	22.1	Test
91	500	0	150	0	50	0	769.85	740.8	154	10.5	0.56	83.15	T
92	500	0	125	0	75	0	769.52	740.48	154	10.5	0.56	90.99	T

Table 1 (continued)

Sample	Input											Output	Comments
	Cement	Limestone powder	Fly ash	GGBS	Silica fume	RHA	Coarse aggregate	Fine aggregate	Water	SP	VMA	Compressive strength	
93	500	0	100	0	100	0	769.18	740.16	154	10.5	0.56	86.99	V
94	500	0	125	0	75	0	769.52	740.48	154	10.5	0.56	99.84	T
95	500	0	125	0	75	0	769.63	740.59	154	10.5	0.56	99.76	T
96	500	0	125	0	75	0	788.07	758.33	140	10.5	0.56	88.98	Test
97	500	0	100	0	100	0	787.73	758.01	140	10.5	0.56	83.94	T
98	500	0	125	0	75	0	788.07	758.33	140	10.5	0.56	85.37	T
99	500	0	125	0	75	0	790.4	781.44	140	13.3	0.35	91.13	V
100	500	0	125	0	75	0	774.98	766.19	154	11.2	0.35	117.03	T
101	408.5	0	21.5	0	0	0	630	1135	200	3.68	0	53.006	T
102	387	0	43	0	0	0	630	1135	200	3.29	0	51.039	Test
103	365.5	0	64.5	0	0	0	630	1135	200	3.11	0	50.982	T
104	344	0	86	0	0	0	630	1135	200	2.75	0	50.37	T
105	322.5	0	107.5	0	0	0	630	1135	200	2.58	0	43.984	V
106	301	0	129	0	0	0	630	1135	200	2.26	0	44.003	T
107	279.5	0	150.5	0	0	0	630	1135	200	2.10	0	44.344	T
108	258	0	172	0	0	0	630	1135	200	1.68	0	43.184	Test
109	350	175	0	0	0	0	500	1050	126	11.7	0	61.3	T
110	350	175	0	0	0	0	500	1050	143.5	8.9	0	60.1	T
111	350	175	0	0	0	0	500	1050	161	6.9	0	58.7	V
112	350	175	0	0	0	0	500	1050	178.5	3.8	0	49.5	T
113	350	175	0	0	0	0	500	1050	196	3.2	0	43	T
114	333	175	0	0	17	0	500	1050	158	7.3	0	52.9	Test
115	326	175	0	0	24	0	500	1050	158	5.1	0	54.3	T
116	319	175	0	0	31	0	500	1050	158	5.2	0	58.4	T
117	313	175	0	0	37	0	500	1050	158	6.8	0	63.3	V
118	307	175	0	0	43	0	500	1050	158	6.1	0	59.1	T
119	301	175	0	0	49	0	500	1050	158	8.8	0	65.5	T
120	500	0	0	0	0	25	750	875	200	17.5	0	38	Test
121	500	0	0	0	0	25	750	875	200	20	0	37.8	T
122	500	0	0	0	0	25	750	875	200	22.5	0	22.2	T
123	500	0	0	0	0	50	750	875	200	17.5	0	36.2	V
124	500	0	0	0	0	50	750	875	200	20	0	41.4	T
125	500	0	0	0	0	50	750	875	200	22.5	0	48.5	T
126	360	0	90	0	0	0	855	813	198	3.2	0	68	Test
127	270	0	180	0	0	0	842	801	198	2.9	0	60.3	T
128	180	0	270	0	0	0	829	788	198	3	0	42.5	T
129	360	0	0	90	0	0	866	824	198	3.7	0	72.6	V
130	270	0	0	180	0	0	863	821	198	3.4	0	74.9	T
131	180	0	0	270	0	0	861	819	198	2.8	0	65.7	T
132	428	0	0	0	22.5	0	865	823	198	4.9	0	71.2	Test
133	405	0	0	0	45	0	861	819	198	5.2	0	76.1	T
134	383	0	0	0	67.5	0	858	816	198	7.8	0	74.8	T
135	360	0	67.5	0	22.5	0	855	813	198	4.2	0	67.2	V
136	270	0	135	0	45	0	841	801	198	4.5	0	57.6	T
137	180	0	202.5	0	67.5	0	828	788	198	4.8	0	44.9	T
138	360	0	0	67.5	22.5	0	863	821	198	4	0	68	Test

Table 1 (continued)

Sample	Input											Output	Comments
	Cement	Limestone powder	Fly ash	GGBS	Silica fume	RHA	Coarse aggregate	Fine aggregate	Water	SP	VMA	Compressive strength	
139	270	0	0	135	45	0	857	816	198	4.6	0	68.2	T
140	180	0	0	202.5	67.5	0	852	810	198	5.8	0	70.7	T
141	360	0	45	45	0	0	861	819	198	3.2	0	78	V
142	270	0	90	90	0	0	853	811	198	3.2	0	69.2	T
143	180	0	135	135	0	0	845	803	198	2.8	0	60.6	T
144	360	0	33.8	33.8	22.5	0	859	817	198	4.2	0	76	Test
145	270	0	67.5	67.5	45	0	849	808	198	4.2	0	66.8	T
146	180	0	101.3	101.3	67.5	0	840	799	198	5	0	55.2	T
147	400	0	0	0	0	100	700	850	250	14	0	42.9	V
148	350	0	0	0	0	150	700	850	250	12.25	0	40.9	T
149	300	0	0	0	0	200	700	850	250	10.5	0	33.5	T
150	465	0	85	0	0	0	590	910	227.7	10.73	0	35.19	Test
151	440	0	110	0	0	0	590	910	228.6	11.01	0	33.15	T
152	415	0	135	0	0	0	590	910	233.3	9.91	0	31.47	T
153	385	0	165	0	0	0	590	910	234.4	9.91	0	30.66	V
154	355	0	195	0	0	0	590	910	241.6	9.91	0	29.62	T
155	300	125	0	0	0	0	733.6	1072.7	180	5.7	0	40.4	T
156	350	45	0	0	0	0	684.3	1080.1	210	5.25	1.23	36.9	Test
157	350	90	0	0	0	0	726.4	996.5	210	5.08	0	38.7	T
158	440	0	82.5	0	27.5	0	916	713	176	8.22	0	79.2	T
159	330	0	165	0	55	0	898	699	176	9.11	0	67.2	V
160	220	0	247.5	0	82.5	0	880	685	176	8.89	0	59.9	T
161	440	0	0	82.5	27.5	0	927	721	176	9.78	0	79.6	T
162	330	0	0	165	55	0	920	716	176	10.78	0	87.5	Test
163	220	0	0	247.5	82.5	0	912	710	176	10.22	0	84.5	T
164	440	0	55	55	0	0	924	720	176	8	0	76.9	T
165	330	0	110	110	0	0	913	711	176	7.5	0	62.2	V
166	220	0	165	165	0	0	903	703	176	4.44	0	69.3	T
167	440	0	41.3	41.3	27.5	0	922	717	176	6.4	0	78.6	T
168	330	0	82.5	82.5	55	0	909	707	176	6.48	0	72.7	Test
169	220	0	123.8	123.8	82.5	0	896	697	176	8	0	64.3	T
170	374	107	0	0	0	53	968	790	182	2.25	0	93	T
171	366	0	104	0	0	52	968	790	178	1.25	0	97	V
172	336	0	104	0	0	78	968	790	176	1.50	0	101	T
173	308	0	103	0	0	103	968	790	175	1.65	0	103	T
174	397	113	0	0	0	57	968	790	168	3.00	0	99	Test
175	388	0	111	0	0	55	968	790	170	1.50	0	109	T
176	413	0	108	0	0	59	968	790	155	2.00	0	122	T
177	413	0	108	0	59	0	968	790	155	2.00	0	118	V
178	413	0	108	0	59	0	968	790	155	1.75	0	119	T
179	460	0	0	0	0	0	1085.2	693.81	161	1.10	0	68	T
180	368	0	92	0	0	0	1085.2	693.81	161	0.88	0	66	Test
181	322	0	138	0	0	0	1085.2	693.81	161	0.77	0	58	T
182	276	0	184	0	0	0	1085.2	693.81	161	0.66	0	56	T
183	368	0	0	92	0	0	1085.2	693.81	161	0.88	0	66	V
184	322	0	0	138	0	0	1085.2	693.81	161	0.77	0	61.5	T

Table 1 (continued)

Sample	Input											Output	Comments
	Cement	Limestone powder	Fly ash	GGBS	Silica fume	RHA	Coarse aggregate	Fine aggregate	Water	SP	VMA	Compressive strength	
185	276	0	0	184	0	0	1085.2	693.81	161	0.66	0	58	T
186	319	0	0	0	0	0	1600	368	185	0	0	44.2	Test
187	110	0	440	0	0	0	821	698	246	1.1	0.25	48.3	T
188	500	0	0	0	0	0	1467	337	185	0	0	74.5	T
189	220	0	330	0	0	0	937	796	172	2.64	0.15	73.5	V
190	552	0	0	0	0	0	1486	342	160	5.52	0	91.3	T
191	330	0	220	0	0	0	981	835	144	4.95	0.2	92.6	T
192	600	0	0	0	0	0	1462	336	155	7.2	0	92.3	Test
193	440	0	110	0	0	0	989	841	142	6.6	0.2	94.6	T
194	412.5	0	137.5	0	0	0	520.1	612.7	203.5	0	0	44	T
195	401.5	0	137.5	0	11	0	518.6	610.9	203.5	0	0	50	V
196	385	0	137.5	0	27.5	0	516.8	608.8	203.5	0	0	52	T
197	412.5	0	137.5	0	0	0	523.1	612.7	203.5	0	0	54	T
198	401.5	0	137.5	0	11	0	521.6	610.9	203.5	0	0	62	Test
199	385	0	137.5	0	27.5	0	519.8	608.8	203.5	0	0	65	T
200	337.5	0	112.5	0	0	0	531.9	626.5	225	0	0	30	T
201	328.5	0	112.5	0	9	0	530.6	625.1	225	0	0	32	V
202	315	0	112.5	0	22.5	0	529.2	623.4	225	0	0	33	T
203	337.5	0	112.5	0	0	0	534.9	626.1	225	0	0	34	T
204	328.5	0	112.5	0	9	0	533.7	625.1	225	0	0	40	Test
205	315	0	112.5	0	22.5	0	532.2	623.4	225	0	0	46	T

T training data, *V* validation data, *Test* test data

Marquardt algorithm as implemented by levmar. This algorithm appears to be optimum for training moderate-sized (up to several hundred neurons per layer) feedforward neural networks dealing with nonlinear problems (Lourakis 2005).

Table 2 The input and output parameters used in the development of BPNNs

Code	Variable	Data used in NN models		
		Minimum	Average	Maximum
01	Cement	110.00	348.38	600.00
02	Limestone powder	0.00	25.67	272.00
03	Fly ash	0.00	105.20	440.00
04	GGBS	0.00	17.39	330.00
05	Silica fume	0.00	14.71	250.00
06	RHA	0.00	6.55	200.00
07	Coarse aggregate	500.00	787.67	1600.00
08	Fine aggregate	336.00	827.93	1135.00
09	Water	94.50	178.54	250.00
10	SP	0.00	5.96	22.50
11	VMA	0.00	0.14	1.23
12	Compressive strength (MPa (or N/mm^2))	10.20	58.08	122.00

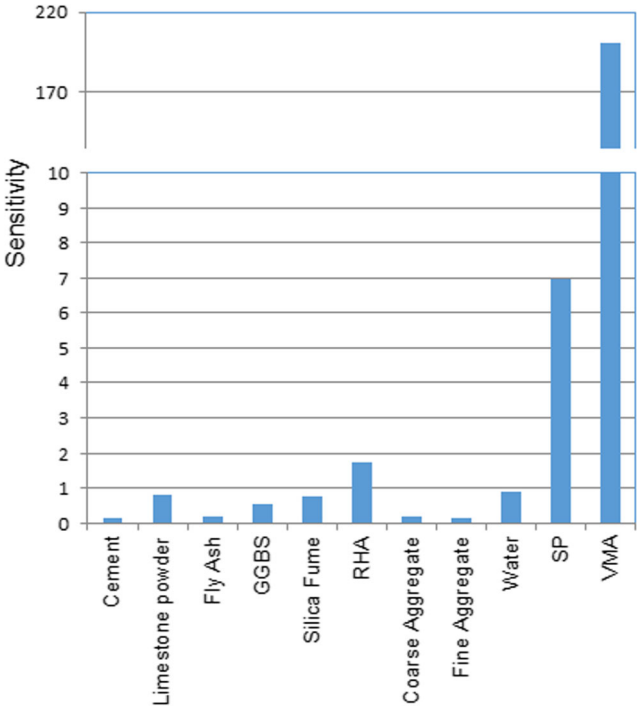

Fig. 3 Sensitivity analysis of the compressive strength to the composition parameters of SCC

3.5 BPNN model development

In this work, a total of 91,800 different BPNN models have been developed and investigated. More specifically, a number of 18,360 of these involve ANN architectures implemented in 5 different computers in order to investigate the sensitivity of the ANN results to the very nature of the floating-point arithmetic of each computer. Each one of these ANN models was trained over 113 datasets out of a total of 169 datasets (66.86% of the total number), and the validation and testing of the trained ANN were performed with the remaining 56 datasets. More specifically, 28 datasets (16.57%) were used for the validation of the trained ANN and 28 (16.57%) datasets were used for the testing (estimating the Pearson's correlation coefficient *R*). The parameters used for the ANN training are summarized in Table 3. In order to have a fair comparison of the various ANNs, the datasets used for their training are manually divided by the user into training, validation and testing sets using appropriate indices to state whether the data belongs to the training, validation or testing set. In the general case, the division of the datasets into the three groups is made randomly.

.

The 91,800 developed ANN models were sorted in a decreasing order based on Pearson's correlation coefficient value, and the architecture of the top 20 models are presented in Table 4 for the five computers used. Also, Tables 5, 6, 7, 8, and 9 present the top 20 models for each computer individually. Based on these results, the optimum BPNN model is that

Table 3 Training parameters of BBNN models

Parameter	Value
Training algorithm	Levenberg-Marquardt algorithm
Number of hidden layers	1; 2
Number of neurons per hidden layer	4 to 20 by step 1
Training goal	0
Epochs	1000
Cost function	MSE; SSE
Transfer functions	Tansig (T); Logsig (L)
Initial weights of hidden layers	0.10; 0.50; 0.90
−0.10; −0.50; −0.90	
Initial weights of bias	0.10; 1.00

MSE mean square error, *SSE* sum square error, *Tansig (T)* hyperbolic tangent sigmoid transfer function, *Logsig (L)* log-sigmoid transfer function

of 11-11-5-1 (Fig. 4) with Pearson's correlation coefficient *R* equal to 0.9828.

Figures 5 and 6 depict the comparison of the exact experimental values with the predicted values of the optimum BPNN model with topology 11-11-5-1. These results clearly show that the 28-day compressive strength of admixture-based self-compacting concrete predicted from the multilayer feed-forward neural network are very close to the experimental results.

From the presented results, we see that the following:

- Among the available literature training algorithms, the best, by far, ANN prediction of the SCC strength was achieved by using the Levenberg-Marquardt algorithm.
- The computational environment significantly affects the performance of the ANN training and subsequently its performance. This is due to the fact that the algorithms of the computational units ultimately rely on basic arithmetic operations that can yield different results when performed in different environments due to the very nature of floating-point arithmetic. Different optimum ANN architectures were found in different computers.
- Furthermore, the proposed new formula for the normalization of data proved effective and robust compared to available ones.
- For the top 20 models, the optimum number of hidden layers was found to be two.
- The initial values of weights significantly affect the results; different values of the initial weights result in different optimum ANN architectures.
- All 20 ANN models presented in Table 4 have been trained with a number of epochs between 47 and 73. This means that the developed multilayer feed-forward neural network models can predict the 28-day compressive strength of admixture-based self-compacting concrete

Table 4 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient *R* (all computers)

Ranking	Computer	Pre-process	Cost function	Training functions	Initial weights	Architecture (code)	Pearson's *R*	Number of epochs
1	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-11-5-1	0.9828	73
2	C03	Central	MSE	T-L-T	−0.9	11-19-15-1	0.9825	66
3	C05	*z*-score	MSE	T-L-T	−0.5	11-14-10-1	0.9824	63
4	C05	Central	MSE	T-L-T	0.5	11-15-14-1	0.9823	67
5	C01	Min–max [0, 1]	SSE	T-L-T	−0.9	11-16-9-1	0.9820	56
6	C01	*z*-score	SSE	T-L-T	−0.5	11-20-14-1	0.9819	52
7	C04	Min–max [0, 1]	SSE	T-L-T	−0.9	11-14-6-1	0.9818	51
8	C04	*z*-score	MSE	T-L-T	−0.5	11-19-4-1	0.9818	58
9	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-20-5-1	0.9817	66
10	C05	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-16-16-1	0.9816	69
11	C05	Central	SSE	T-L-T	−0.1	11-11-9-1	0.9815	63
12	C02	Central	MSE	T-L-T	−0.9	11-20-5-1	0.9814	49
13	C02	*z*-score	MSE	T-L-T	−0.5	11-13-8-1	0.9813	66
14	C01	Min–max [0, 1]	MSE	T-L-T	−0.1	11-20-20-1	0.9812	62
15	C05	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-18-6-1	0.9812	59
16	C01	Central	MSE	T-L-T	−0.9	11-20-5-1	0.9811	60
17	C01	No pre-process	SSE	T-L-T	0.1	11-9-8-1	0.9810	62
18	C05	Central	MSE	T-L-T	0.1	11-11-8-1	0.9810	47
19	C01	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-20-8-1	0.9809	53
20	C04	Min–max [0, 1]	MSE	T-L-T	−0.5	11-19-10-1	0.9808	51

with smaller error rates and less computational effort compared to the one presented in the literature. Furthermore, these ANN models predict the SCC compressive strength with values of the Pearson's correlation coefficient *R*

Table 5 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient *R* (computer C01)

Ranking	Computer	Pre-process	Cost function	Training functions	Initial weights	Architecture (code)	Pearson's *R*	Number of epochs
1	C01	Min–max [0, 1]	SSE	T-L-T	−0.9	11-16-9-1	0.9820	56
2	C01	*z*-score	SSE	T-L-T	−0.5	11-20-14-1	0.9819	52
3	C01	Min–max [0, 1]	MSE	T-L-T	−0.1	11-20-20-1	0.9812	62
4	C01	Central	MSE	T-L-T	−0.9	11-20-5-1	0.9811	60
5	C01	No pre-process	SSE	T-L-T	0.1	11-9-8-1	0.9810	62
6	C01	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-20-8-1	0.9809	53
7	C01	Min–max [0, 1]	MSE	T-T-T	−0.5	11-10-8-1	0.9807	53
8	C01	*z*-score	MSE	T-L-T	−0.5	11-20-5-1	0.9803	70
9	C01	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-17-14-1	0.9803	51
10	C01	Central	MSE	T-L-T	−0.9	11-20-8-1	0.9802	64
11	C01	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-20-6-1	0.9798	47
12	C01	No pre-process	MSE	T-L-T	−0.9	11-20-5-1	0.9797	67
13	C01	*z*-score	SSE	T-L-T	−0.5	11-20-13-1	0.9796	69
14	C01	Min–max [−1, 1]	SSE	T-L-T	−0.9	11-17-10-1	0.9793	49
15	C01	*z*-score	SSE	T-L-T	−0.5	11-19-18-1	0.9792	62
16	C01	Central	MSE	T-T-T	0.9	11-9-5-1	0.9790	234
17	C01	*z*-score	MSE	T-L-T	−0.5	11-18-9-1	0.9789	66
18	C01	*z*-score	MSE	T-L-T	−0.5	11-20-4-1	0.9789	62
19	C01	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-19-9-1	0.9789	62
20	C01	*z*-score	SSE	T-T-T	−0.1	11-14-13-1	0.9788	103

Table 6 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient *R* (computer C02)

Ranking	Computer	Pre-process	Cost function	Training functions	Initial weights	Architecture (code)	Pearson's *R*	Number of epochs
1	C02	Central	MSE	T-L-T	−0.9	11-20-5-1	0.9814	59
2	C02	*z*-score	MSE	T-L-T	−0.5	11-13-8-1	0.9813	66
3	C02	*z*-score	MSE	T-L-T	−0.9	11-10-5-1	0.9806	69
4	C02	Min–max [0, 1]	MSE	T-L-T	−0.9	11-6-5-1	0.9805	116
5	C02	Min–max [0, 1]	MSE	T-L-T	−0.5	11-9-9-1	0.9805	49
6	C02	*z*-score	MSE	T-L-T	−0.9	11-12-5-1	0.9804	58
7	C02	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-13-5-1	0.9804	48
8	C02	*z*-score	SSE	T-L-T	0.1	11-10-7-1	0.9801	78
9	C02	Central	MSE	T-L-T	−0.5	11-11-9-1	0.9800	55
10	C02	*z*-score	MSE	T-L-T	−0.9	11-15-6-1	0.9799	64
11	C02	No pre-process	MSE	T-T-T	−0.1	11-10-8-1	0.9798	100
12	C02	Min–max [0, 1]	MSE	T-L-T	−0.5	11-20-5-1	0.9797	63
13	C02	Min–max [−1, 1]	SSE	T-L-T	0.9	11-16-11-1	0.9796	151
14	C02	*z*-score	SSE	T-L-T	0.9	11-18-4-1	0.9796	107
15	C02	Central	SSE	T-T-T	0.9	11-13-7-1	0.9794	108
16	C02	Min–max [0, 1]	MSE	T-L-T	−0.5	11-20-17-1	0.9792	65
17	C02	Min–max [0, 1]	SSE	T-L-T	−0.9	11-16-8-1	0.9792	54
18	C02	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-19-4-1	0.9792	62
19	C02	Min–max [−1, 1]	SSE	T-L-T	−0.9	11-20-17-1	0.9791	70
20	C02	*z*-score	SSE	T-L-T	−0.5	11-20-14-1	0.9789	52

between 0.98083 and 0.98275 (for the optimum one, see Table 4) while the best values in the literature is 0.97 for ANN and 0.98 for the case of using fuzzy logic models [41].

Table 7 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient *R* (computer C03)

Ranking	Computer	Pre-process	Cost function	Training functions	Initial weights	Architecture (code)	Pearson's *R*	Number of epochs
1	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-11-5-1	0.9828	73
2	C03	Central	MSE	T-L-T	−0.9	11-19-15-1	0.9825	66
3	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-20-5-1	0.9817	66
4	C03	*z*-score	MSE	T-L-T	−0.5	11-19-6-1	0.9808	51
5	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-19-7-1	0.9807	57
6	C03	Min–max [0, 1]	MSE	T-L-T	−0.1	11-15-4-1	0.9807	61
7	C03	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-13-9-1	0.9803	43
8	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-8-5-1	0.9801	73
9	C03	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-18-5-1	0.9801	60
10	C03	Central	SSE	T-T-T	−0.1	11-19-7-1	0.9799	104
11	C03	*z*-score	SSE	T-T-T	−0.1	11-8-4-1	0.9797	65
12	C03	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-16-9-1	0.9797	47
13	C03	Min–max [−1, 1]	SSE	T-L-T	−0.9	11-20-16-1	0.9796	52
14	C03	Min–max [0, 1]	MSE	T-L-T	−0.5	11-18-10-1	0.9795	71
15	C03	*z*-score	MSE	T-L-T	0.5	11-18-9-1	0.9794	79
16	C03	*z*-score	MSE	T-L-T	−0.5	11-14-10-1	0.9793	45
17	C03	*z*-score	MSE	T-L-T	−0.9	11-13-5-1	0.9793	60
18	C03	*z*-score	SSE	T-T-T	−0.1	11-14-10-1	0.9792	81
19	C03	No pre-process	MSE	T-L-T	−0.9	11-18-4-1	0.9791	46
20	C03	Min–max [0, 2]	MSE	T-L-T	−0.5	11-19-5-1	0.9790	74

Table 8 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient *R* (computer C04)

Ranking	Computer	Pre-process	Cost function	Training functions	Initial weights	Architecture (code)	Pearson's *R*	Number of epochs
1	C04	Min–max [0, 1]	SSE	T-L-T	−0.9	11-14-6-1	0.9818	51
2	C04	*z*-score	MSE	T-L-T	−0.5	11-19-4-1	0.9818	58
3	C04	Min–max [0, 1]	MSE	T-L-T	−0.5	11-19-10-1	0.9808	51
4	C04	*z*-score	MSE	T-L-T	−0.5	11-14-8-1	0.9808	69
5	C04	Central	MSE	T-L-T	−0.9	11-20-5-1	0.9805	58
6	C04	*z*-score	SSE	T-L-T	−0.5	11-20-13-1	0.9803	72
7	C04	Min–max [0, 1]	SSE	T-L-T	−0.9	11-10-4-1	0.9803	181
8	C04	*z*-score	MSE	T-L-T	−0.5	11-20-10-1	0.9802	67
9	C04	Min–max [0, 1]	SSE	T-L-T	−0.5	11-17-9-1	0.9802	54
10	C04	*z*-score	MSE	T-L-T	−0.5	11-19-10-1	0.9801	75
11	C04	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-15-6-1	0.9799	50
12	C04	*z*-score	SSE	T-L-T	−0.9	11-14-6-1	0.9799	58
13	C04	*z*-score	MSE	T-L-T	−0.5	11-14-10-1	0.9799	54
14	C04	Min–max [−1, 1]	SSE	T-L-T	−0.9	11-19-8-1	0.9797	58
15	C04	Min–max [−1, 1]	MSE	T-T-T	−0.1	11-16-7-1	0.9797	83
16	C04	Min–max [0, 1]	MSE	T-L-T	−0.5	11-18-5-1	0.9796	60
17	C04	Min–max [0, 1]	MSE	T-L-T	−0.5	11-11-4-1	0.9795	57
18	C04	Central	MSE	T-T-T	−0.1	11-14-11-1	0.9795	72
19	C04	*z*-score	MSE	T-L-T	0.1	11-20-11-1	0.9792	61
20	C04	Min–max [0, 1]	MSE	T-L-T	−0.1	11-13-10-1	0.9792	72

Table 9 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient *R* (computer C05)

Ranking	Computer	Pre-process	Cost function	Training functions	Initial weights	Architecture (code)	Pearson's *R*	Number of epochs
1	C05	*z*-score	MSE	T-L-T	−0.5	11-14-10-1	0.9824	63
2	C05	Central	MSE	T-L-T	0.5	11-15-14-1	0.9823	67
3	C05	Min–max [−1, 1]	SSE	T-L-T	−0.5	11-16-16-1	0.9816	69
4	C05	Central	SSE	T-L-T	−0.1	11-11-9-1	0.9815	63
5	C05	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-18-6-1	0.9812	59
6	C05	Central	MSE	T-L-T	0.1	11-11-8-1	0.9810	47
7	C05	*z*-score	MSE	T-L-T	−0.5	11-20-9-1	0.9805	55
8	C05	Min–max [−1, 1]	MSE	T-L-T	−0.5	11-18-4-1	0.9803	70
9	C05	*z*-score	MSE	T-L-T	0.9	11-11-5-1	0.9803	123
10	C05	Min–max [0, 1]	SSE	T-L-T	−0.5	11-18-8-1	0.9801	68
11	C05	Min–max [0, 1]	SSE	T-L-T	0.1	11-16-5-1	0.9799	88
12	C05	Central	MSE	T-T-T	0.1	11-17-16-1	0.9798	104
13	C05	*z*-score	MSE	T-L-T	−0.9	11-14-10-1	0.9797	82
14	C05	Min–max [0, 1]	MSE	T-L-T	−0.5	11-15-4-1	0.9795	70
15	C05	*z*-score	SSE	T-L-T	−0.9	11-16-16-1	0.9793	79
16	C05	Central	MSE	T-T-T	0.1	11-12-6-1	0.9792	102
17	C05	No pre-process	MSE	T-L-T	−0.9	11-15-11-1	0.9791	54
18	C05	No pre-process	SSE	T-L-T	0.1	11-17-16-1	0.9791	54
19	C05	Min–max [0, 1]	MSE	T-L-T	−0.5	11-13-4-1	0.9789	61
20	C05	*z*-score	MSE	T-L-T	−0.5	11-18-8-1	0.9789	65

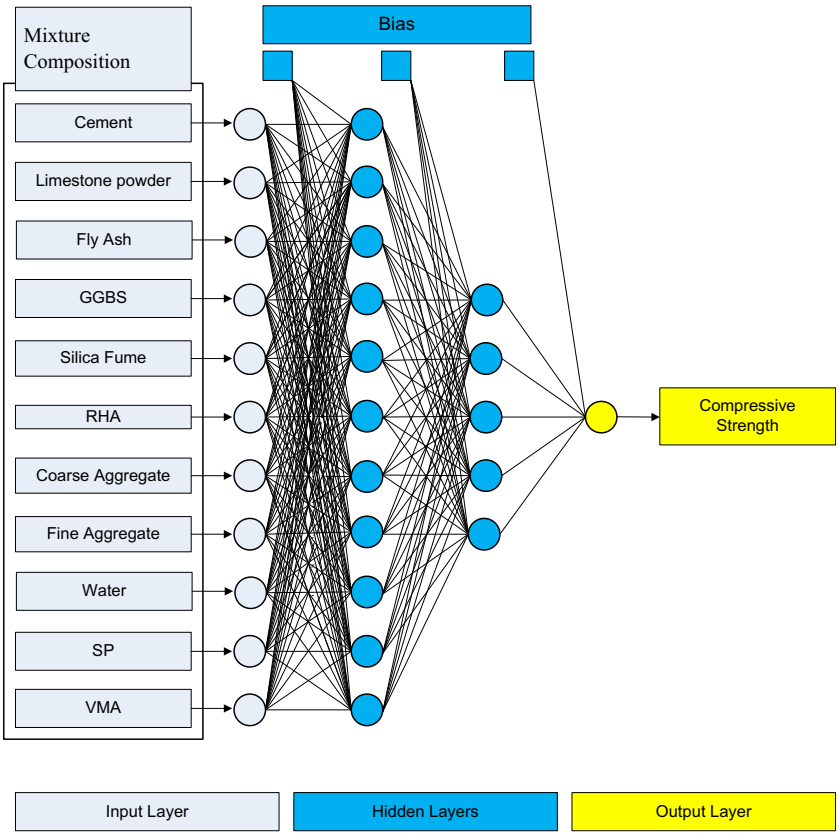

Fig. 4 The best with two hidden layers (11-11-5-1) of BPNN based on Pearson's correlation coefficient *R*

4 Conclusions

In this paper, artificial neural networks were trained in order to investigate their capability in predicting the 28-day compressive strength of admixture-based self-compacting concrete. In order to achieve that, a novel heuristic algorithm was proposed in order to find the optimum architectures for a set of multilayered feed-forward back-propagation neural networks based on the value of Pearson's correlation coefficient. The results showed that the prediction of the compressive strengths of admixture-based self-compacting concrete value obtained with the trained ANNs is very close to the experimental results making ANN a very promising metamodel for predicting the 28-day compressive strength of SCC mixtures. Furthermore, the proposed new formula for the normalization of data

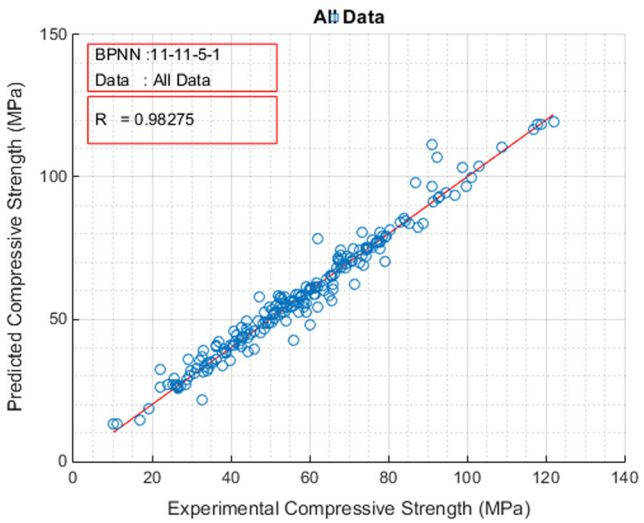

Fig. 5 Pearson's correlation coefficient *R* of the experimental and predicted compressive strength for the best with two hidden layers BPNN (11-11-5-1)

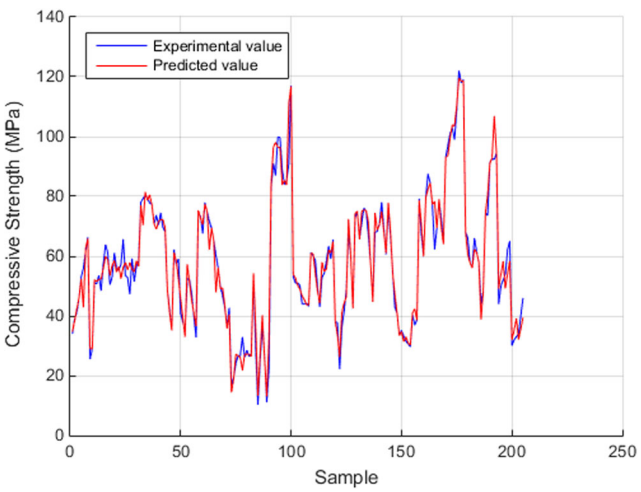

Fig. 6 Experimental vs predicted values of compressive strength for the best with two hidden layers BPNN (11-115-1)

proved effective and robust compared to available ones in the literature.

In a subsequent work, the reverse problem will be investigated in which the identification of the optimum ANN configuration will be the target when the compressive SCC strength will be the input and the 11 compositional parameters will be the output.

Acknowledgments The research was performed within the framework of the Master's Program in Applied Computational Structural Engineering (ACSE), which has been partially financed by the Research Committee of the School of Pedagogical and Technological Education, Athens, Greece. The authors would like to express their gratitude to MSc students Mrs. M.G. Douvika, Mr. K. Roinos and Mr. A.K. Tsaris and to the undergraduate students Mr. N. Margaris and Mr. D. Georgakopoulos for their assistance on the computational implementation of the ANN models.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Açikgenç M, Ulaş M, Alyamaç KE (2015) Using an artificial neural network to predict mix compositions of steel fiber-reinforced concrete. Arab J Sci Eng 40(2):407–419
2. Adeli H (2001) Neural networks in civil engineering: 1989-2000. Computer-aided civil and infrastructure engineering 16(2):126–142
3. Akkurt S, Tayfur G, Can S (2004) Fuzzy logic model for the prediction of cement compressive strength. Cem Concr Res 34(8): 1429–1433
4. Alyamac KE, Ince R (2009) A preliminary concrete mix design for SCC with marble powders. Constr Build Mater 23:1201–1210
5. Asteris, P.G., Plevris, V. (2013). Neural network approximation of the masonry failure under biaxial compressive stress, ECCOMAS Special Interest Conference—SEECCM 2013: 3rd South-East European Conference on Computational Mechanics, Proceedings—an IACM Special Interest Conference, pp. 584–598
6. Asteris PG, Plevris V (2016) Anisotropic masonry failure criterion using artificial neural networks. Neural Computing and Applications (NCAA). doi:10.1007/s00521-016-2181-3
7. Asteris PG, Tsaris AK, Cavaleri L, Repapis CC, Papalou A, Di Trapani F, Karypidis DF (2016a) Prediction of the fundamental period of infilled RC frame structures using artificial neural networks. Computational Intelligence and Neuroscience 2016: 5104907
8. Asteris PG, Kolovos KG, Douvika MG, Roinos K (2016b) Prediction of self-compacting concrete strength using artificial neural networks. European Journal of Environmental and Civil Engineering 20:s102–s122
9. Bartlett PL (1998) The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. IEEE Trans Inf Theory 44(2):525–536
10. Baskar I, Ramanathan P, Venkatasubramani R (2012) Influence of silica fume on properties of self-compacting concrete. Int J Emerg Trends Eng Dev 4:757–767
11. Baykal G, Döven AG (2000) Utilization of fly ash as pelletization process; theory, application, areas and research results. Resour Conserv Recycl 30:59–77
12. Baykasoğlu A, Dereli TU, Taniş S (2004) Prediction of cement strength using soft computing techniques. Cem Concr Res 34(11): 2083–2090
13. Belalia Douma O, Boukhatem B, Ghrici M, Tagnit-Hamou A (2016) Prediction of properties of self-compacting concrete containing fly ash using artificial neural network. Neural Comput & Applic. doi:10.1007/s00521-016-2368-7
14. Berry MJA, Linoff G (1997) Data mining techniques. Wiley, NY
15. Blum A (1992) Neural networks in C++. Wiley, NY
16. Boger, Z, Guterman, H (1997) Knowledge extraction from artificial neural network models, IEEE Systems, Man, and Cybernetics Conference, Orlando, FL, USA
17. Boukendakdji O, Kenai S, Kadri EH, Rouis F (2009) Effect of slag on the rheology of fresh self-compacted concrete. Constr Build Mater 23:2593–2598
18. Boukendakdji O, Kadri EH, Kenai S (2012) Effects of granulated blast furnace slag and superplasticizer type on the fresh properties and compressive strength of self-compacting concrete. Constr Build Mater 34:583–590
19. Brouwers HJH, Radix HJ (2005) Self-compacting concrete: theoretical and experimental study. Cem Concr Res 35:2116–2136
20. Chen Z (2013) An overview of bayesian methods for neural spike train analysis. Computational Intelligence and Neuroscience 2013: Article number 251905
21. Delen D, Sharda R, Bessonov M (2006) Identifying significant predictors of injury severity in traffic accidents using a series of artificial neural networks. Accid Anal Prev 38(3):434–444
22. Dias WPS, Pooliyadda SP (2001) Neural networks for predicting properties of concretes with admixtures. Constr Build Mater 15(7): 371–379
23. Dinakar P, Sethy KP, Sahoo UC (2013) Design of self-compacting concrete with ground granulated blast furnace slag. Mater Des 43: 161–169
24. Fathi A, Shafiq N, Nuruddin MF, Elheber A (2013) Study the effectiveness of the different pozzolanic material on self-compacting concrete. ARPN J Eng Applied Sci 8:229–305
25. Felekoglu B, Turkel S, Baradan B (2007) Effect of water/cement ratio on the fresh and hardened properties of self-compacting concrete. Build Environ 42:1795–1802
26. Gandage, AS, Ram, VV, Sivakumar, MVN, Vasan, A, Venu, M, Yaswanth, AB (2013) Optimization of class C flyash dosage in self-compacting concrete for pavement applications, Proceedings of the International Conference on Innovations in Concrete for Meeting Infrastructure Challenge, October 23–26, 2013, Hyderabad, Andhra Pradesh, India, pp: 213–226
27. Gesoglu M, Ozbay E (2007) Effects of mineral admixtures on fresh and hardened properties of self-compacting concretes: binary, ternary and quaternary systems. Mater Struct 40:923–937
28. Gesoglu M, Guneyisi E, Ozbay E (2009) Properties of self-compacting concretes made with binary, ternary and quarternary cementitious blends of fly ash, blast furnace slag and silica fume. Constr Build Mater 23:1847–1854
29. Gettu, R., Izquierdo, J., Gomes, P.C.C., Josa, A. (2002). Development of high-strength self-compacting concrete with fly ash: a four-step experimental methodology, Proceedings of the 27th Conference on Our World in Concrete and Structures, August 29–30, 2002, Singapore, pp: 217–224
30. Giovanis DG, Papadopoulos V (2015) Spectral representation-based neural network assisted stochastic structural mechanics. Engineering Structures, Volume 84:382–394
31. Grdic Z, Despotovic I, Curcic GT (2008) Properties of self-compacting concrete with different types of additives. Facta Universitatis-Ser: Archit Civil Eng 6:173–177
32. Güneyisi E, Gesoglu M, Ali Azez O, Öznur Öz H (2016) Effect of nano silica on the workability of self-compacting concretes having

untreated and surface treated lightweight aggregates. Constr Build Mater 115:371–380
33. Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. Neural Netw 2(5):359–366
34. Iruansi, O, Guadagnini, M, Pilakoutas, K, Neocleous, K (2010) Predicting the shear strength of RC beams without stirrups using Bayesian neural network, in 4th International Workshop on Reliable Engineering Computing (REC 2010)
35. Joseph G, Ramamurthy K (2009) Influence of fly ash on strength and sorption characteristics of cold-bonded fly ash aggregate concrete. Constr Build Mater 23:1862–1870
36. Karlik B, Olgac AV (2011) Performance analysis of various activation functions in generalized MLP architectures of neural networks. International Journal of Artificial Intelligence And Expert Systems (IJAE) 1(4):111–122
37. Kayali O (2008) Fly ash lightweight aggregates in high performance concrete. Constr Build Mater 22:2393–2399
38. Lamanna J, Malgaroli A, Cerutti S, Signorini MG (2012) Detection of fractal behavior in temporal series of synaptic quantal release events: a feasibility study, Computational Intelligence and Neuroscience, volume 2012, 2012. Article number 704673
39. Lee SC (2003) Prediction of concrete strength using artificial neural networks. Eng Struct 25(7):849–857
40. Lourakis MIA (2005). A brief description of the Levenberg-Marquardt algorithm implemened by levmar. Institute of Computer Science Foundation for Research and Technology - Hellas (FORTH), available at: http://www.ics.forth.gr/~lourakis/levmar/levmar.pdf.
41. Malagavelli V, Manalel PA (2014) Modeling of compressive strength of admixture-based self compacting concrete using fuzzy logic and artificial neural networks. Asian Journal of Applied Sciences 7(7):536–551
42. Mansouri I, Kisi O (2015) Prediction of debonding strength for masonry elements retrofitted with FRP composites using neuro fuzzy and neural network approaches. Compos Part B 70:247–255
43. Mansouri, I., Gholampour, A., Kisi, O., Ozbakkaloglu, T. (2016). Evaluation of peak and residual conditions of actively confined concrete using neuro-fuzzy and neural computing techniques, neural computing and applications, pp. 1-16
44. Mashhadban H, Kutanaei SS, Sayarinejad MA (2016) Prediction and modeling of mechanical properties in fiber reinforced self-compacting concrete using particle swarm optimization algorithm and artificial neural network. Constr Build Mater 119:277–287
45. Memon SA, Shaikh MA, Akbar H (2011) Utilization of rice husk ash as viscosity modifying agent in self compacting concrete. Constr Build Mater 25:1044–1048
46. Özcan F, Atiş CD, Karahan O, Uncuoğlu E, Tanyildizi H (2009) Comparison of artificial neural network and fuzzy logic models for prediction of long-term compressive strength of silica fume concrete. Adv Eng Softw 40(9):856–863
47. Papadopoulos V, Giovanis DG, Lagaros ND, Papadrakakis M (2012) Accelerated subset simulation with neural networks for reliability analysis. Comput Methods Appl Mech Eng 223-224:70–80
48. Pattnaik S, Karunakar DB, Jha PK (2014) A prediction model for the lost wax process through fuzzy-based artificial neural network, Proceedings of the Institution of Mechanical Engineers. Part C: Journal of Mechanical Engineering Science 228(7):1259–1271
49. Phadke MS (1989) Quality engineering using design of experiments. In Quality control, robust design, and the Taguchi method. Springer, US, pp 31–50
50. Phani SS, Sekhar ST, Rao S, Sravana P (2013) High strength self-compacting concrete using mineral admixtures. Indian Concr J 87: 42–47
51. Plevris, V, Asteris, PG (2014a) Modeling of masonry compressive failure using Neural Networks, OPT-i 2014—1st International Conference on Engineering and Applied Sciences Optimization, Proceedings, pp. 2843–2861
52. Plevris V, Asteris PG (2014b) Modeling of masonry failure surface under biaxial compressive stress using neural networks. Constr Build Mater 55:447–461
53. Plevris, V, Asteris, P (2015) Anisotropic failure criterion for brittle materials using Artificial Neural Networks, COMPDYN 2015—5th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering, pp. 2259–2272
54. Rahman ME, Muntohar AS, Pakrashi V, Nagaratnam BH, Sujan D (2014) Self-compacting concrete from uncontrolled burning of rice husk and blended fine aggregate. Mater Des 55:410–415
55. Rao, NVR, Rao, PS, Sravana, P, Sekhar, TS (2009). Studies on relationship of water-powder ratio and compressive strength of self-compacted concrete, Proceedings of the 34th Conference on Our World in Concrete and Structures, August 16–18, 2009, Singapore, pp: 1–8
56. Rouis F (2009) Effect of slag on the rheology of fresh self-compacted concrete. Constr Build Mater 23:2593–2598
57. Safiuddin M, Raman SN, Salam MA, Jumaat MZ (2016) Modeling of compressive strength for self-consolidating high-strength concrete incorporating palm oil fuel ash. Materials 9(5):396
58. Sahmaran M, Yaman IO, Tokyay M (2009) Transport and mechanical properties of self-consolidating concrete with high volume fly ash. Cem Concr Compos 31:99–106
59. Sfikas IP, Trezos KG (2013) Effect of composition variations on bond properties of self-compacting concrete specimens. Constr Build Mater 41:252–262
60. Siddique R (2011) Properties of self-compacting concrete containing class F fly ash. Mater Des 32:1501–1507
61. Sonebi M (2004) Medium strength self-compacting concrete containing fly ash: modelling using factorial experimental plans. Cem Concr Res 34:1199–1208
62. Sukumar B, Nagamani K, Raghavan RS (2008) Evaluation of strength at early ages of self-compacting concrete with high volume fly ash. Constr Build Mater 22:1394–1401
63. Topçu IB, Saridemir M (2008) Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic. Comput Mater Sci 41(3):305–311
64. Trtnik G, Kavčič F, Turk G (2009) Prediction of concrete strength using ultrasonic pulse velocity and artificial neural networks. Ultrasonics 49(1):53–60
65. Valcuende M, Marco E, Parra C, Serna P (2012) Influence of limestone filler and viscosity-modifying admixture on the shrinkage of self-compacting concrete. Cem Concr Res 42:583–592
66. Waszczyszyn Z, Ziemiański L (2001) Neural networks in mechanics of structures and materials—new results and prospects of applications. Comput Struct 79(22–25):2261–2276
67. Zhao H, Sun W, Wu X, Gao B (2015) The properties of the self-compacting concrete with fly ash and ground granulated blast furnace slag mineral admixtures. J Clean Prod 95:66–74