



# A hybridized intelligence model to improve the predictability level of strength index parameters of rocks

Abbas Abbaszadeh Shahri<sup>1</sup> · Reza Asheghi<sup>1</sup> · Mohammad Khorsand Zak<sup>2</sup>

Received: 9 October 2019 / Accepted: 17 July 2020 / Published online: 19 September 2020  
© Springer-Verlag London Ltd., part of Springer Nature 2020

## Abstract

In the current paper, the uniaxial compressive strength (*UCS*) and Young modulus (*E*) of rocks were predicted using a hybridized intelligence method. The model was developed using an optimum multi-objective generalized feedforward neural network (*GFFN*) incorporated with an imperialist competitive metaheuristic algorithm (*ICA*) and managed using 208 datasets of different physical and mechanical quarries from almost all over of Iran. Rock class, density, porosity, *P*-wave velocity, point load index and water absorption were datacenter components. The predictability and accuracy performance of the hybrid *ICA-GFFN* model were discussed using different error criteria and confusion matrixes. The observed 5.4% and at least 32% improvement in hybrid *ICA-GFFN* than *GFFN* and multivariate regression (*MVR*) demonstrated feasible and accurate enough tools that can effectively be applied for multi-objective prediction purposes. The influence of inputs on predicted outputs was also identified using two different sensitivity analyses.

**Keywords** Hybrid model · Sensitivity analysis · Metaheuristic algorithm · Strength index parameters · Predictability level

## 1 Introduction

The real-world complicated problems due to nonlinear constraints, interdependencies among variables and large solution spaces need to be optimized using capable techniques. Optimization, as a core component in problem solving, refers to find the best value of a set of variables for an objective function subject to a given set of constraints. The performances, benefits and great successes of such processes have widely been notified in the literature [10, 22, 48].

A design problem in rock engineering using uniaxial compressive strength (*UCS*) and elasticity modulus (*E*) usually involves many parameters of which some are highly sensitive. These two strength index properties of rocks have significantly been quoted in design approaches of civil, mining and construction engineering-oriented

applications (e.g., tunneling, dam design, rock blasting, slope stability, rock mass classification, rock failure criteria, foundation engineering, underground excavation). Concerning approved and recognized difficulties in direct measurements of these parameters in both economical aspects and significant technical challenges in weak or highly weathered rocks [2, 4, 14, 16, 31, 37], producing optimized models that can provide more accurate results is demanded. From a practical perspective, a variety of predictive models for *UCS* and *E* have widely been highlighted using simpler indirect test methods through different statistical and multivariate techniques [4, 27, 37, 38, 40, 51, 55]. However, the accuracy of such empirical relations due to the large variability of rock properties cannot be generalized. Furthermore, the drawbacks of statistical techniques in the effectiveness of auxiliary factors (e.g., porosity, mineralogy and mineral composition, density and weathering degree), uncertainty of experimental tests as well as inaccurate prediction in a wide expanded range of data should be considered [1, 39, 41]. In recent years, such limitations in producing predictive models tremendously have been recuperated using different artificial intelligence techniques [2, 14, 20, 46]. Table 1 shows a brief summary of immense

✉ Abbas Abbaszadeh Shahri  
a\_abbaszadeh@iauh.ac.ir

<sup>1</sup> Department of Civil Engineering, Roudehen Branch, Islamic Azad University, Tehran, Iran

<sup>2</sup> Department of Applied Mathematics, Aligoudarz Branch, Islamic Azad University, Aligoudarz, Iran

**Table 1** Summary of some recently used techniques to predict *UCS* and *E*

References	Used method	Rock type	Input variables
[7]	ANN, MVR	Sedimentary rocks	$n, w, V_p, \gamma_d$
[9]	MVR, ANN, SVR	Carbonate rocks	$\rho, V_p, V_s$
[14]	Different hybrid ANN models, MVR	Quarried rocks	rock class, $V_p, \gamma, I_s, n, w$
[16]	Genetic programming	Carbonate rocks	$n, \rho, V_p$
[23]	ANN, regression	Travertine	$V_p, I_s, R_n, n$
[31]	ANN, regression analysis	Breccias rock samples	$V_{bp}, \rho, V_s, r, A_{bdf}$
[35]	MVR, ANN	Carbonate rocks	$n, \rho, V_p, v, I_s$
[44]	ANN	Coal rocks	$V_p, I_s, I_d, \rho$
[47]	ANFIS	Sedimentary rocks	$\rho, n, V_p$
[52]	ANN, MVR	Carbonate rocks	$R_n, n, \gamma_d, V_p, I_d$
[56]	ANN, hybrid neuro-fuzzy, MVR	Gypsum	$n, I_s, R_n, I_d$

ANN: artificial neural network, MVR: multivariate regression; SVR: support vector regression,  $V_p$ : p-wave velocity,  $I_d$ : slake durability;  $n$ : porosity;  $I_s$ : point load index;  $V_s$ : s-wave velocity;  $\gamma_d$ : dry unit weight;  $\rho$ : density;  $v$ : poison ratio;  $R_n$ : Schmidt hammer;  $A_{bdf}$ : average block diameter factor;  $r$ : roundness of blocks;  $V_{bp}$ : volumetric block proportion;  $w$ : water absorption

successes in producing more efficient predictive models for both *UCS* and *E* in rock engineering.

Further, it was approved that incorporated ANN-based models with metaheuristic algorithms can lead to remarkable progress in predictability level [2, 14, 29]. The metaheuristic algorithms as a subcategory of optimization processes have extensively been grown through the last two decades and outlined considerable popularity in a wide range of practical engineering applications, finance, planning, scheduling and designing [6, 18, 26, 28, 50]. These state-of-the-art algorithms have been drawn from various nature-inspired sources and aim to improve fitness function. Imperialistic competitive algorithm [12], firefly [53], gray wolf optimizer [36], ant colony [21], honey beam [42], particle swarm optimization [33], artificial bee colony [32] and simulated annealing [34] are some of these metaheuristics that have actively been incorporated with artificial intelligence models for a variety of engineering tasks. Efficiency, flexibility and model independent are some of the main substantial features of these algorithms [6, 14, 17, 54]. The performance of such algorithms not only takes relatively much less time than traditional optimization techniques but also provides appropriate accomplishments when the learning rule is not efficient or fails to deliver satisfactory results [17]. However, they cannot guarantee that the best solution found after termination criteria is satisfied or indeed its global optimal solution to the problem [2, 14, 26, 54].

Among these optimization methods, the imperialist competitive algorithm (*ICA*) is one of the recently developed metaheuristics inspired by socio-political behaviors [12]. This global search population-based algorithm is a component of swarm intelligence technique which can provide an evolutionary computation without the

requirement to the gradient of the function in its optimization process.

The *UCS* and *E* almost are predicted using localized data in terms of single objective models. This implies that developing a multi-objective model that can provide acceptable accuracy is greatly of interest. Furthermore, the approved performance of metaheuristics in different sizes of the search spaces motives for optimizing such multi-objective models in rock engineering problems. Comparing with the previous studies, this paper presents a robust automated hybrid multi-objective model, where the strength index properties were predicted through an optimum generalized feedforward network (*GFFN*) incorporated with *ICA*. Replacing the *GFFN* instead of the usual multilayer perceptron to increase the computability of the model and machine-driven tuned optimal internal parameters which yield the best performance are the main features of this study. The models were managed using 208 datasets corresponding to different physical and mechanical parameters (porosity,  $n$ ; density,  $\gamma$ ; water absorption,  $w$ ; rock class, point load index,  $I_s$ ; and *P*-wave velocity,  $V_p$ ) from almost all over quarry locations of Iran. It was observed that the classification error in *GFFN* for *UCS* and *E* from 19.1% and 23.8% was significantly decreased to 14.3% and 16.7% in *ICA-GFFN*. The assessed performance showed that the developed hybrid *ICA-GFFN* as a feasible tool can effectively be applied to provide more precise results than *GFFN*. Two different sensitivity analyses were applied to identify the most and least effective parameters on predicted *UCS* and *E*.

The remainder of this paper is organized as follows. A summary of *ICA* is presented in Sect. 2. The study is turned toward the modeling procedure, applied datasets and hybridizing layout in Sect. 3. Discussion, validation and

analysis of the obtained results then were placed in Sect. 4. The summary of remarkable findings then was outlined in conclusions.

## 2 The process of ICA

ICA is a recently developed evolutionary and robust optimization algorithm inspired by imperialist competitiveness based on the extending policy for the power and rule of a government beyond its own borders [12]. This algorithm mathematically is configured using a series of parameters including number of country ( $N_{cou}$ ), number of imperialist ( $N_{imp}$ ), number of decades ( $N_{dec}$ ), number of colonies ( $N_{col}$ ), direction of moved colony toward the imperialist ( $\beta$ ), deviation parameter ( $\theta$ ), arbitrary parameter describing the search condition ( $\varphi$ ) and effective factor on the total power of empire ( $\zeta$ ). As presented in Table 2, the initial guess of these parameters can be set using previous studies [5, 8, 12–14, 29].

More insights about the organized formula can be found in [12–14, 29]. Here, only a brief description about the theoretical concept is presented. This algorithm can be divided into eight essential steps including generating the initial empires, moving the colonies of an empire toward the imperialist, revolution, exchanging positions of the imperialist and colony, total power of an empire, imperialistic competition, elimination of empires and convergence. To start the ICA, the country and cost function should be defined. Among the initial generated population ( $N_{cou}$ ), those with minimum cost are selected to be imperialists and the rest play the role of colonies ( $N_{col}$ ). Then, the imperialistic empires begin to compete with each other to attract more colonies. Therefore, the colonies are moved toward an imperialist peak or new minimum area (assimilation process) to improve their situations and find better solutions. The movement process of colonies due to partially absorbed colonies can have a direct or deviated path toward the imperialist. The position of trapped colonies is then excited by sudden random changes using the

revolution process to escape from possible local optimum in the search space. The performance of revolution can be compared with a mutation in the genetic algorithm in preventing the early convergence to local optima [29]. Hence, if the new position of the colony possesses a lower cost function than the imperialist, the position of imperialist and colony will be exchanged. The more empire power, the more attracted colonies; thus, the weakest empire because of losing colonies is gradually collapsed and eliminated. This implies that all the countries then should be converged to only one robust empire in the domain of the problem as the desired solution.

Accordingly, the competition process among the empires represents the possession probability of each empire ( $p_n$ ) based on its total power and is calculated using normalized total cost of empire as follows:

$$p_n = \left| \frac{NTC_n}{\sum_{i=1}^{N_{imp}} NTC_i} \right|; \sum_{i=1}^{N_{imp}} p_i = 1 \tag{1}$$

where  $TC_n$  and  $NTC_n$  denote the total and normalized cost of the  $n$ th empire.

The distribution mechanism of ICA is the probability density function (PDF) which compared to a genetic algorithm requires less computation effort.

## 3 Overview of applied model

### 3.1 Summary on GFFN

The multilayer perceptrons (MLPs) as the main core of feedforward networks have widely been updated during the last decades. These structures are trained slowly but easy to use and can approximate any input/output map subjected to different internal characteristics. The GFFN is configured by replacing the perceptrons of the hidden layer in MLPs with the generalized shunting inhibitory neurons (GSN) (Fig. 1). This organization provides a subcategory of MLPs that is an extended form of the shunting inhibitory artificial neural networks (SIANNs) [19, 30]. The output of the  $j$ th neuron in the hidden layer ( $O_j$ ) using a set of adaptive weights ( $w_{i,j}$  and  $w_{jk}$ ) subjected to nonlinear activation functions ( $f$  and  $g$ ) are expressed as follows:

$$O_j = \frac{b_j + f(\sum_i w_{ij}x_i + w_{jk})}{a_j + g(\sum_i c_{ij}x_i + c_{jk})} \tag{2}$$

where  $x_i$  is the  $i$ th input;  $c_{ji}$  is the “shunting inhibitory” connection weight from input  $i$  to neuron  $j$  and  $w_{ij}$  denotes the connection weight from input  $i$  to neuron  $j$ .  $w_{j0}$  and  $c_{j0}$  are bias constants, and  $a_j$  is a positive constant that represents the passive decay rate of the neuron where  $b_j$  is the output bias.  $f$  and  $g$  are activation functions, respectively.

**Table 2** The range of used ICA parameters in previous studies

Parameter	Range in previous studies
$N_{cou}$	25–500
$N_{imp}$	5–65
$N_{dec}$	[0–1000]
$\beta$	[1–2]
$\theta$	$\pi/4$ ; [0, 1]
$\zeta$	[0.1–0.02]
$N_{cou}$ , $N_{imp}$ and $N_{dec}$ also can be managed using parametric investigations	

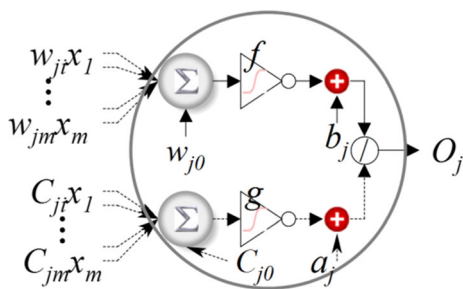


Fig. 1 The mathematical model of GSN in data processing

This reveals that the GSNs perform the nonlinear transformation on input datasets to speed up the training procedure and enhance the computability level to save the required memory. It also facilitates more freedom to select optimum topology and provides higher resolution in complex nonlinear decision classifiers [3, 11, 14, 24, 30]. Jumping over one or more layers is another characteristic of the GSN which allows neurons to operate as adaptive nonlinear filters [1, 11, 14, 24, 30]. Such ability in this classifier provides greater flexibility and more efficient performance than MLP in the same number of neurons [1, 11, 14]. The number of neurons in hidden layers and corresponding arrangements, activation function, learning rate and training algorithm are the main internal characteristics components which should be set to produce an optimum structure [1, 14, 15, 24]. The network error ( $E$ ) of the  $k$ th output neuron and corresponding root mean square error ( $RMSE$ ) is defined using the actual and predicted values ( $x_k$  and  $y_k$ ) as follows:

$$E = \frac{1}{2} \sum_{i=1}^n (x_k - y_k)^2 \rightarrow RMSE = \sqrt{\frac{\sum_{i=1}^n (x_k - y_k)^2}{n}} \quad (3)$$

To reduce the error between the desired and actual outputs, the weights are optimized using an updating procedure for the  $(n + 1)$ th pattern subjected to

$$\Delta w_{ik} = -\eta \frac{\partial E(W)}{\partial w_{ik}} \quad (4)$$

$$w_{ik}(n + 1) = w_{ik}(n) + \nabla w_{ik}(n) \quad (5)$$

where  $\eta$  is the learning rate.

### 3.2 Organized datacenter

In the current study, a number of 208 datasets comprising the rock class, point load index ( $I_s$ ),  $P$ -wave velocity ( $V_p$ ), porosity ( $n$ ), water absorption ( $w$ ) and density ( $\gamma$ ) from 47 quarry locations in Iran were acquired and compiled (Table 3). These datasets are the updated version of [14] with a greater number of instances containing both UCS and  $E$ . A sample of gathered datasets and executed simple

descriptive statistical analyses are presented in Tables 4 and 5. The rock classes including sedimentary, igneous, diagenetic and metamorphic were assigned and coded from 1 to 4, respectively.

Different strategies for randomization are used in optimization to alleviate the computational burden associated with robust control techniques. In this paper, the randomized data split into three sets for training, testing and validation was used. Comparing the K-fold cross-validation, elimination of the selection bias, balancing the groups with respect to uncertainties, and the statistical tests are the basic benefits of randomized data. The K-fold cross-validation always requires to be checked for stable performance over all the folds. Furthermore, insufficiently distributed data in the folds or significant differences from others provide instability in model performance. Such discrepancies then should be removed using some other different cross-validation methods. Trusting just to final aggregated score for the  $R^2$  will miss a lot of information about models' performance. The compiled components were then randomized by 55%, 25% and 20% to provide training, testing and validation sets. Further, the datasets due to different units were normalized within the range of [0, 1] using:

$$\text{Norm scaled value} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (6)$$

This procedure provides dimensionless input data which are necessary to improve the learning speed and model stability.

### 3.3 Assessment of optimum hybrid model

Combining a developed optimum GFFN with ICA is the overview of the hybridized layout in this study. An optimum network structure needs to be managed through internal characteristics, where there is no unified accepted method for such critical arrangements [1, 24]. Therefore, to decrease the complexities, it is advised to make the internal characteristics as few as possible [57]. However, there is no guarantee that can capture all possible alternatives for optimal solutions. In the current study, this drawback was covered using an automated iterative trial–error integrated with the constructive technique. Referring to Fig. 2a, the optimum GFFN model was captured by examining different training algorithms and activation functions. This strategy in addition to monitored error improvements was applied to escape from local minima, early convergence and prevent the overfitting problem. Quick propagation (QP), conjugate gradient descent (CGD), momentum (MO), quasi-Newton (QN) and Levenberg–Marquardt (LM) were implemented as training algorithms. Logistic (Log), hyperbolic tangent (HyT), linear (Lin) and squash

**Table 3** The specification of acquired datasets

Rock class	Rock type	Location (quarry name)	Number of samples
Igneous	Andesite, Granite, Dacite, Terakite, Vitric tuff, Ignemberite, Gabbro, Syenite	NW of Iran, Sefid Afshar, Nehbandan, Ghermez Zafreh, Borujerd, Iranshahr, Hamedan, Natanz, Piranshahr, Birjand, Yazd, Farahzad-Tehran, Mashhad, Naeen, Naybandan, Sabzevar	42
Sedimentary	Limestone, Travertine, Onyx travertine	Badjgah, Sarvestan, Kharama, Sadra, Firoozabad, Sarvie Neyriz, North and SW of Iran, Hamedan, Sanandaj, Anarak, Azarshahr, Dastjerd, Atashkuh, Mahallat, Abbaszabad, Abyar, Dare bokhari, Firuzkuh, Makou, Hajiabad	141
Metamorphic-diagenetic	Dolomitic limestone, Marble	Ghorveh, Kerman, Sirjan, Dehbid, Bavanat, Neyriz, Salsali, Haftoman, Anarak, Harsin	25

**Table 4** A brief presentation of compiled datasets

Rock type	Rock class	Location	$\gamma$ (gr/cm <sup>3</sup> )	$n$ (%)	$V_p$ (km/s)	$I_s$ (Mpa)	$w$ (%)	$UCS$ (Mpa)	$E$ (Gpa)
Travertine	1	Azarshahr	2.48	7.41	3.95	3.59	1.43	33.6	4.5
Travertine	1	Azarshahr	2.55	3.1	4.9	4.72	0.67	60.7	8.05
Onyx travertine	1	Dastjerd	2.66	1.77	5.26	5.3	1.7	65.7	8.72
Travertine	1	Atashkooh	2.47	4.2	4.6	4.2	1.47	49.3	6.61
Travertine	1	Abbasabad	2.43	4.86	4.15	3.64	2.69	41.3	5.56
Travertine	1	Abyar	2.41	3.58	4.69	4.66	1.27	51.4	6.86
Travertine	1	Mahallat	2.38	6.4	4.14	3.3	1.7	37.4	5.03
Travertine	1	Hajiabad	2.53	3.35	5.66	3	1.76	63.68	11.45
Andesite	2	Sanandaj-Sirjan	2.94	1.37	5.26	6.49	0.46	87.19	20.25
Andesite	2	Yazd	2.72	0.87	5.79	8.82	0.32	126	46.4
Granite	2	Nehbandan	2.8	0.32	6.1	10.67	0.23	145	36.5
Granite	2	Borujerd	2.71	0.81	5.82	10.22	0.29	128	65.9
Syenite	2	Iranshahr	2.54	1.88	5.3	6.27	0.59	91	20
Gabbro	2	Natanz	2.82	0.26	6.11	14.98	0.23	147	81.4
Vitric tuff	2	Farahzad-Tehran	2.18	1.15	3.72	7.2	5.61	101	32.4
Dolomitic limestone	3	Ghorveh	2.68	1.11	6.12	5.7	0.29	110	17.38
Marble	4	Sirjan	2.69	1.1	5.21	3.3	0.16	64	12.46
Dolomitic limestone	3	Dehbid	2.71	0.35	5.89	5.64	0.2	103.5	17.44
Dolomitic limestone	3	Bavanat	2.71	0.61	5.32	4.09	0.15	77	12.89
Marble	4	Neyriz	2.7	0.37	5.83	5.53	0.18	101.8	13.58
Limestone	1	Sarvestan	2.69	0.87	4.85	2.66	4.36	47	7.33
Limestone	1	Kharama	2.7	0.27	5.87	4.03	1.94	77	12.27
Limestone	1	Firoozabad	2.56	9.76	3.42	2.01	6.38	32.9	8.01
Limestone	1	Sarvie Neyriz	2.71	0.44	4.59	3.09	3.34	56.9	8.53
Limestone	1	Anarak	2.68	1.11	6.12	5.3	1.29	83	15.78

( $S_q$ ) were also used as activation transfer functions for hidden and output layers. The value of 0.7 for the learning rate was set for all implemented algorithms and the step sizes of hidden layers were changed in the domain of [1.0–0.001]. Furthermore, the sum of squares and cross-

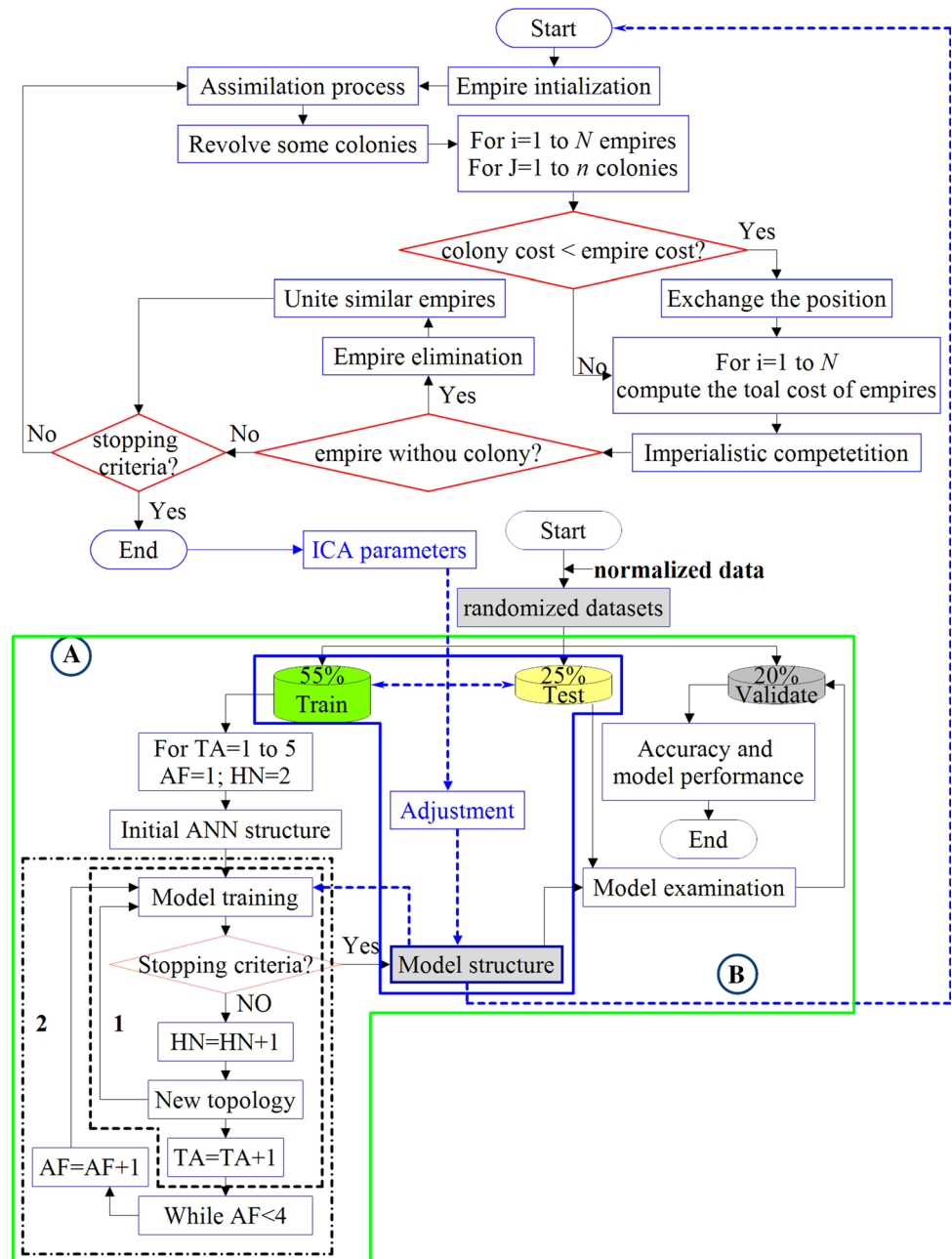
entropy were also employed as output error function, respectively. Following the embedded loops (1 and 2) in Fig. 2a, the number of neurons is a user-defined parameter that can frequently be increased to evaluate a wide range of topologies subjected to different training algorithms and

**Table 5** Descriptive statistics of acquired datasets

Variable	Mean	Mean SE	SD	Min	Median	Max	Skewness	MSSD
Rock class	1.51	0.058	0.846	1	1	4	1.65	0.019
$\gamma$ (gr/cm <sup>3</sup> )	2.59	0.012	0.175	2.18	2.58	3.06	0.42	0.0097
$n$ %	6.63	0.431	6.207	0.15	4.53	31.4	1.05	8.34
$V_p$ (m/s)	4.82	0.066	0.946	2.4	5.04	6.82	- 0.64	0.235
$W$ (%)	4.29	0.22	3.082	0.3	3.32	15.12	1.49	1.754
$I_s$ (MPa)	2.96	0.282	4.062	0.07	1.38	16.16	1.86	1.569
$UCS$ (Mpa)	58.78	2.61	37.98	4.75	45.9	193	0.79	274.08
$E$ (Gpa)	14.3	1.18	17.21	0.5	6.81	89.4	2.09	58.62

Mean SE mean standard error, SD standard deviation, MSSD mean of the squared successive differences

**Fig. 2** Simplified diagram of optimizing process for **a** GFFN and **b** hybrid ICA-GFFN in this study



activation functions. The root mean square error (*RMSE*) and iteration number were organized as termination criteria. If the *RMSE* is not achieved, then the number of iteration (in this study set for 1000) as the second stopping metric is implemented.

This implies on wide ranges of monitored topologies (> 860) even with similar architecture but various internal characteristics (e.g., number of neurons, layer arrangements, training algorithm, activation functions) that was run three times to control whether the stopping criteria are satisfied (Table 6). Accordingly, the *RMSE* and the network correlation ( $R^2$ ) of each tested structure were calculated. As an example of executed effort, the variation of calculated *RMSE* based on the number of neurons as well as some of the tested topologies with different layer arrangements subjected to *HyT* activation transfer function is presented in Fig. 3a, b. The minimum *RMSE* for optimum *GFFN* architecture was observed in number of neurons 12 correspond to 6-7-5-2 topology (Fig. 3b). The performance of the optimum model then is assessed using different accuracy criteria and statistical error indices.

To improve the predictability level of the developed optimum *GFFN* model, the procedure of Fig. 2a was incorporated into *ICA*. This metaheuristic algorithm by adjusting the weights and biases can minimize the error of optimum *GFFN*. However, for an appropriate optimizing process, the effective *ICA* parameters (Table 2) should properly be selected. These parameters can be determined using previous studies [5, 8, 12–14, 29]. Here, for  $\beta$ ,  $\theta$  and  $\zeta$ , the values 2,  $\pi/4$  and 0.02 were considered. The proper  $N_{cou}$  was specified through the analyzed  $R^2$  and minimum *RMSE* of 12 trained hybrid models using the introduced *GFFN* structure (Table 7). By applying a similar process, the monitored approximate boundary for low variation of *RMSE* against the  $N_{cou}$  was characterized as the optimum  $N_{dec}$  (Fig. 4). The  $N_{imp}$  similarly was determined through the calculated  $R^2$  and *RMSE* of *ICA-GFFN* models (Table 8). To provide the output, the hybrid *ICA-GFFN*

model then was trained using the optimized *GFFN* structure (6-7-5-2) but subjected to ascertained *ICA* parameters (Table 9). To assess the capacity of the network performance and evaluate the predictability level, the results of randomized training and testing datasets are reflected in Fig. 5a–d. The monitoring of error improvement for models was also carried out to control the overfitting and trapping in local minima (Fig. 5e). This criterion refers to network performance predictability during the last and/or each iteration and, consequently, can detect the situation when the network is not improving, and further training is unavailing.

### 4 Validation and discussion

Model evaluation as an important part of a data science project is utilized to quantify the improved facilities regarding previous versions. The confusion matrix as a compact representation of the model performance is the source of many scoring metrics for classification tasks [45]. Moreover, it is a benefit to identify the system confusing for different classes. The confusion matrixes of *GFFN*, *ICA-GFFN* and conducted multivariate regression (*MVR*) models (Eqs. 7–10) for randomized datasets were calculated. A sample of carried out efforts for hybrid *ICA-GFFN* using validation datasets is presented in Table 10.

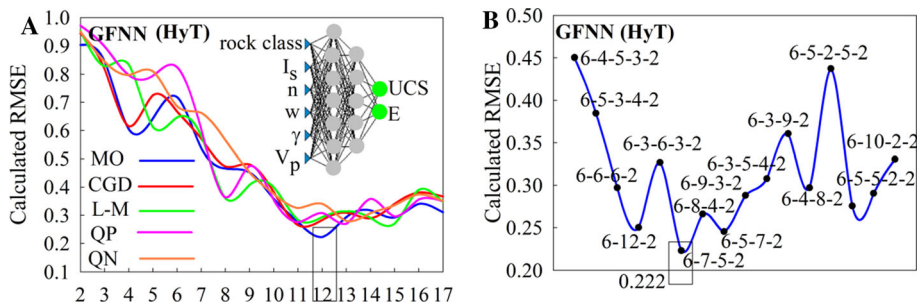
$$\begin{aligned}
 UCS &= -124.97 + 3.54(rockclass) + 44.4\gamma - 0.49n \\
 &\quad + 0.18w + 6.75V_P + 7.65I_{S(50)}R^2 \\
 &= 0.67
 \end{aligned}
 \tag{7}$$

$$\begin{aligned}
 E &= -47.5 + 0.79(rockclass) + 9.93\gamma + 0.02n + 0.91w \\
 &\quad + 1.99V_P + 5.06I_{S(50)} \\
 R^2 &= 0.60
 \end{aligned}
 \tag{8}$$

**Table 6** Implemented training algorithms and characteristics of corresponding optimum *GFFN* model

<i>GFFN</i> training algorithm	Min <i>RMSE</i>	Number of neuron	Corresponding structure	Layer activation transfer function		$R^2$					
				Hidden 1	Hidden 2	Train		Test		Validate	
						<i>UCS</i>	<i>E</i>	<i>UCS</i>	<i>E</i>	<i>UCS</i>	<i>E</i>
<i>QP</i>	0.276	15	6-9-6-2	<i>Log</i>	<i>HyT</i>	0.89	0.86	0.88	0.84	0.89	0.87
<i>CGD</i>	0.265	14	6-5-9-2	<i>HyT</i>	<i>Log</i>	0.90	0.88	0.91	0.89	0.90	0.89
<i>QN</i>	0.280	12	6-6-6-2	<i>Log</i>	<i>Log</i>	0.88	0.83	0.90	0.85	0.90	0.86
<i>L-M</i>	0.270	15	6-8-7-2	<i>HyT</i>	<i>Log</i>	0.90	0.87	0.92	0.86	0.90	0.87
<i>MO</i>	0.222	12	6-7-5-2	<i>HyT</i>	<i>HyT</i>	0.93	0.91	0.92	0.90	0.94	0.92

**Fig. 3** **a** Variation of network RMSE using different training algorithms against the number of neurons subjected to HyT activation functions and **b** a series of examined structures with 12 neurons

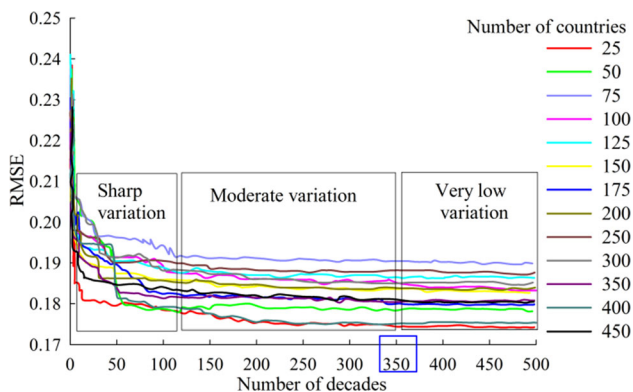


**Table 7** Characterizing the  $N_{cou}$  among the trained models

Tested ICA-GFFN	$N_{cou}$	Train		Test	
		$R^2$	RMSE	$R^2$	RMSE
1	25	0.71	0.181	0.74	0.167
2	50	0.72	0.172	0.66	0.227
3	75	0.67	0.185	0.50	0.196
4	100	0.70	0.169	0.58	0.202
5	150	0.69	0.175	0.60	0.263
6	200	0.72	0.164	0.75	0.150
7	250	0.68	0.179	0.77	0.148
8	300	0.75	0.173	0.55	0.256
9	350	0.68	0.207	0.70	0.191
10	400	0.72	0.166	0.61	0.171
11	450	0.67	0.201	0.88	0.119
12	500	0.64	0.190	0.68	0.177

**Table 8** Determination of  $N_{imp}$  using both training and testing datasets

Tested ICA-GFFN	$N_{imp}$	Train		Test	
		$R^2$	RMSE	$R^2$	RMSE
1	5	0.73	0.206	0.81	0.167
2	10	0.69	0.178	0.84	0.140
3	15	0.84	0.125	0.86	0.127
4	20	0.51	0.528	0.63	0.479
5	25	0.94	0.088	0.96	0.070
6	30	0.92	0.093	0.95	0.097
7	35	0.91	0.103	0.94	0.135
8	40	0.93	0.098	0.96	0.072
9	45	0.90	0.128	0.95	0.075
10	50	0.89	0.135	0.96	0.134
11	55	0.87	0.115	0.79	0.176
12	60	0.73	0.206	0.81	0.167



**Fig. 4** Performance of ICA-GFFN models using different  $N_{cou}$  to find the optimum  $N_{dec}$

$$UCS = 2.004 \text{ rockclass}^{0.305} \gamma^{0.268} n^{-0.093} w^{0.062} V_P^{1.406} I_{S(50)}^{0.524} \quad (9)$$

$$R^2 = 0.54$$

$$E = 0.194 \text{ rockclass}^{0.574} \gamma^{1.615} n^{-0.096} w^{0.925} V_P^{0.53} I_{S(50)}^{1.003} \quad (10)$$

$$R^2 = 0.49$$

Accordingly, the calculated correct classification rate (CCR) and classification error (CE) and then the

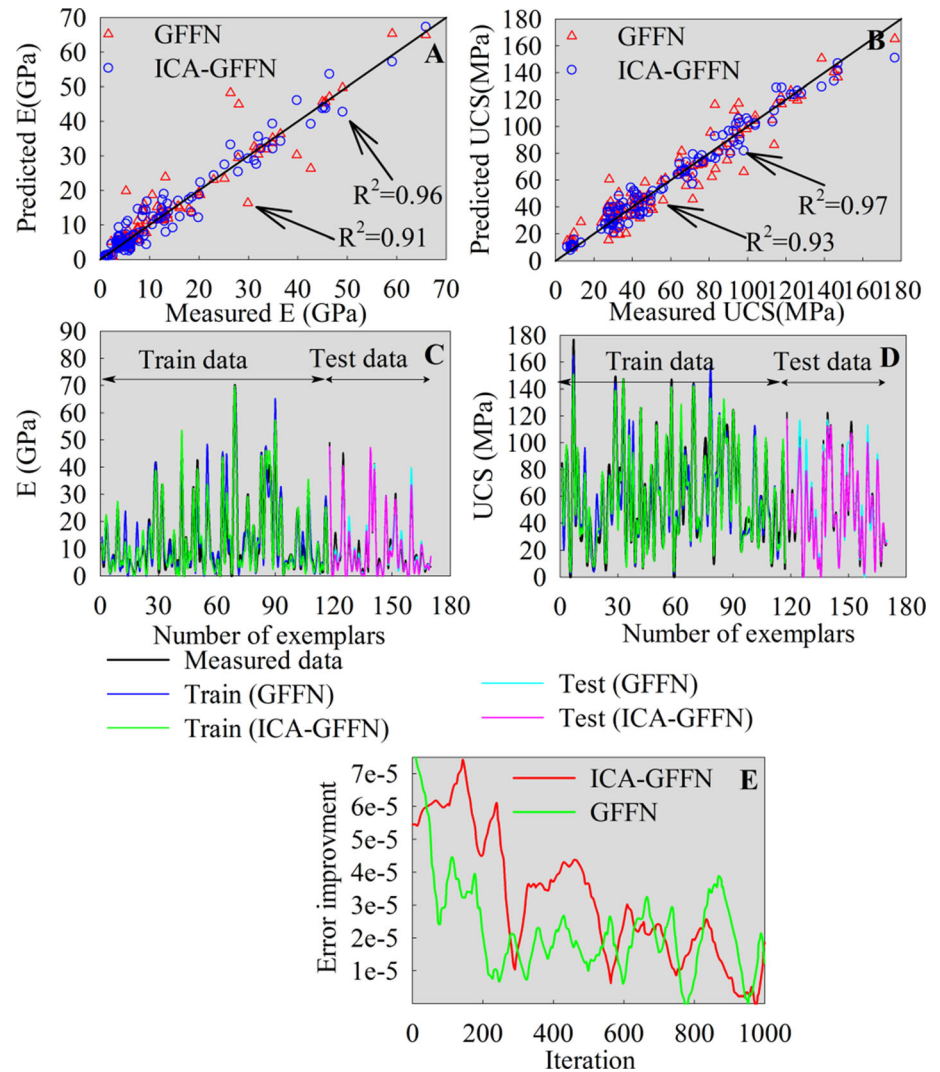
**Table 9** Handled parameters for optimization process in this study

Parameter	Considered ICA parameters in this paper
$N_{cou}$	Carried out parametric analyses $\rightarrow$ 200
$N_{imp}$	Carried out parametric analyses $\rightarrow$ 25
$N_{dec}$	Carried out parametric analyses $\rightarrow$ 350
$\beta$	2
$\theta$	$\pi/4$
$\zeta$	0.02

improvement progress of all models were compared, and the results are presented in Tables 11 and 12, respectively. Further, the accuracy performance of all models was pursued and cross-examined using known statistical error indices, respectively (Table 12). Mean absolute percentage error (MAPE) is one of the most popular indexes for a description of the accuracy and size of the forecasting error. The performance of the model can be evaluated using variance account for (VAF) as an intrinsically



**Fig. 5** Predictability of optimum *GFFN* and hybrid *ICA-GFFN* models for **a** and **c** *E* and **b**, **d** *UCS* using randomized training and testing datasets as well as **e** error improvement of applied models



connected index between predicted and actual values. The generic index of agreement (*IA*) [49] indicates the compatibility of modeled and observations. The formulation of these indices can widely be found in statistical textbooks. Higher values of *IA* and  $R^2$  as well as smaller values of *MAPE* and *RMSE* exhibit better model performance (Table 12).

The precision-recall curve as a useful tool to measure the success of prediction can reflect the relevant and corresponding number of truly turned results (Precision) as well as show the tradeoff between them for different thresholds (recall). Therefore, both high precision and recall express an ideal performance which returns many results and all are labeled correctly. The high precision shows a low false-positive rate while high recall refers to a low false-negative rate. Thereby, a large area under the curve is connected to both high recall and precision returning accurate as well as the majority of all positive results. High recall but low precision returns many data but

most of predictions comparing to training are labeled incorrect and vice versa in high precision but low recall. In Fig. 6, the precision-recall curves for *GFFN*, *ICA-GFFN* and generated multivariate regressions (*MVR*) using the same randomized datasets are presented.

The robustness and performance capacity of sensitivity analyses in the presence of uncertainty for the purpose of model calibration, determining the importance of inputs and enhancing the predictability of a system has been approved [2, 15, 43]. This also implies that removing the least effective inputs may lead to the development of better results [25]. Here, the effectiveness of input parameters on predicted outputs using two sensitivity analyses methods known as the Cosine amplitude and partial derivative (PaD) [15, 25] according to Eqs. 11 and 12 is identified and reflected in Fig. 7.

**Table 10** Confusion matrix of *ICA-GFFN* model for *UCS* and *E*

Target output	Network output <i>ICA-GFFN</i> (UCS)																		Results		
	<7.88	7.88-19.45	19.45-31.02	31.02-42.59	42.59-54.16	54.16-65.73	65.73-77.3	77.3-88.87	88.87-100.44	100.44-112.01	112.01-123.58	123.58-135.15	135.15-146.72	146.72-158.29	158.29-169.86	169.86-181.43	181.43-193	>193	Total	True	False
7.88-19.45	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	0
19.45-31.02	0	0	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	3	2
31.02-42.59	0	0	1	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6	4	2
42.59-54.16	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	6	6	0
54.16-65.73	0	0	0	0	1	6	0	0	0	0	0	0	0	0	0	0	0	0	7	6	1
65.73-77.3	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2	2	0
77.3-88.87	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0
88.87-100.44	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
100.44-112.01	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	2	2	0
112.01-123.58	0	0	0	0	0	0	0	0	1	0	2	0	0	0	0	0	0	0	3	2	1
123.58-135.15	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	3	3	0
135.15-146.72	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
146.72-158.29	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
158.29-169.86	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
169.86-181.43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
181.43-193	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
<b>Note</b>	0	4	4	6	8	6	2	1	2	2	2	3	0	1	0	0	1	0	42	36	6

Target output	Network output <i>ICA-GFFN</i> (E)																		Results		
	<0.51	0.51-6.07	6.07-11.62	11.62-17.18	17.18-22.73	22.73-28.29	28.29-33.84	33.84-39.4	39.4-44.95	44.95-50.51	50.51-56.07	56.07-61.62	61.62-67.18	67.18-72.73	72.73-78.29	78.29-83.84	83.84-89.4	>89.4	Total	True	False
<0.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.51-6.07	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6.07-11.62	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11.62-17.18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17.18-22.73	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22.73-28.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28.29-33.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33.84-39.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
39.4-44.95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44.95-50.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50.51-56.07	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
56.07-61.62	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
61.62-67.18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
67.18-72.73	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
72.73-78.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
78.29-83.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
83.84-89.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
>89.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
<b>Total</b>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
<b>True</b>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
<b>False</b>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 10** continued

0.51-6.07	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	19	0
6.07-11.62	0	1	4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	7	4	3
11.62-17.18	0	0	1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	3	1
17.18-22.73	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	2	1	1
22.73-28.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28.29-33.84	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	2	2	0
33.84-39.4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0
39.4-44.95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44.95-50.51	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	3	2	1
50.51-56.07	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	2	2	0
56.07-61.62	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
61.62-67.18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
67.18-72.73	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0
72.73-78.29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
78.29-83.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
83.84-89.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0
<b>Note</b>	0	20	5	4	1	1	2	2	1	2	2	0	0	1	0	0	1	0	0	42	35	7

**Table 11** Compared CCR and CE of the models for validation and test datasets

Model	CCR (%)				CE (%)			
	Test		Validate		Test		Validate	
	UCS	E	UCS	E	UCS	E	UCS	E
ICA-GFFN	80.3	79.5	85.7	83.3	19.7	20.5	14.3	16.7
GFFN	76.8	75.1	78.6	76.2	21.2	24.9	19.1	23.8
Equations 7, 8	57.7	51.9	59.5	54.7	42.3	48.1	40.5	45.3
Equations 9, 10	48.1	44.2	52.3	45.2	51.93	55.8	47.7	54.8

$$R_{ij} = \frac{\sum_{k=1}^m (x_{ik} \times x_{jk})}{\sqrt{\sum_{k=1}^m x_{ik}^2 \sum_{k=1}^m x_{jk}^2}}, x_i \text{ and } x_j$$

: elements of data pairs (11)

$$\text{contribution of } i\text{th variable} = \frac{SSD_i}{\sum_i SSD_i}; SSD_i = \sum_p \left( \frac{\partial O_k^p}{\partial x_i^p} \right)^2$$

(12)

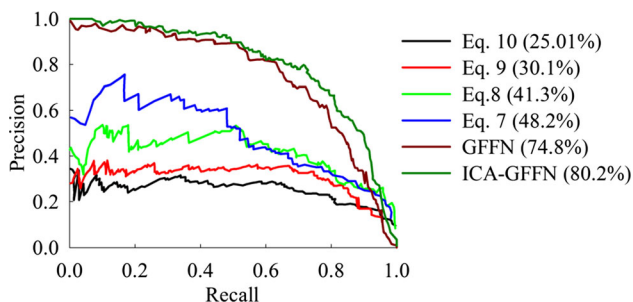
where  $O_k^p$  and  $x_i^p$  are the output and input values for pattern  $P$ , and  $SSD_i$  is the sum of the squares of the partial derivatives.

### 5 Conclusion

A new efficient multi-objective GFFN model using an iterative trial and error procedure integrated with the constructive technique was proposed and applied on a

**Table 12** Improvement of progresses for the applied models

	<i>ICA-GFFN</i>	<i>GFFN</i>	Equation 7	Equation 9	Compared statistical error indices subjected to validation datasets				
					<i>MAPE</i>	<i>VAF</i>	<i>RMSE</i>	<i>IA</i>	$R^2$
<i>Predicted UCS</i>									
<i>ICA-GFFN</i>	–	8.28%	30.57%	38.97%	6.14	95.80	0.178	0.90	0.96
<i>GFFN</i>	– 8.28%	–	24.3%	33.46%	8.65	90.36	0.226	0.85	0.94
Equation 7	– 30.57%	– 24.3%	–	12.14%	11.83	76.12	0.541	0.69	0.73
Equation 9	– 38.97%	– 33.46%	– 12.14%	–	15.62	66.54	0.738	0.57	0.60
	<i>ICA-GFFN</i>	<i>GFFN</i>	Equation 8	Equation 10	Compared statistical error indices subjected to validation datasets				
					<i>MAPE</i>	<i>VAF</i>	<i>RMSE</i>	<i>IA</i>	$R^2$
<i>Predicted E</i>									
<i>ICA-GFFN</i>	–	8.52%	34.33%	45.73%	7.22	91.55	0.241	0.88	0.95
<i>GFFN</i>	– 8.52%	–	28.21%	40.68%	8.05	89.77	0.297	0.81	0.92
Equation 8	– 34.33%	– 28.21%	–	17.36%	12.58	74.62	0.668	0.64	0.69
Equation 10	– 45.73%	– 40.68%	– 17.36%	–	18.94	64.83	0.804	0.49	0.56

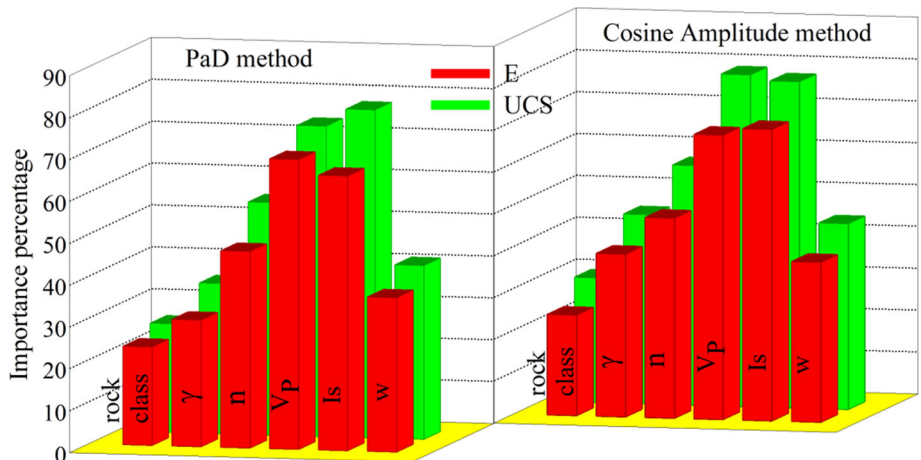


**Fig. 6** Comparing the area under computed precision-recall curves for *MVR*, *GFFN* and *ICA-GFFN*

comprehensive dataset of building stones of Iran to predict the *UCS* and *E*. Comparison of numerous structures subjected to different internal characteristics that showed a four-layer *GFFN* model with 6-7-5-2 topology (12 neurons

in two hidden layers) can be selected as the optimum. The predictability of the introduced *GFFN* model then successfully with at least 8% improvement progress was optimized using *ICA*. According to established confusion matrixes, the correct classification rate of *UCS* and *E* from 52.3% and 45.2% in *MVR* models increased to 85.7% and 83.3% in hybrid *ICA-GFFN*. Furthermore, the *GFFN* and hybrid *ICA-GFFN* with at least a 24% improvement in progress showed the superior capability to *MVR* models. The compared areas under precision-recall curves show that incorporating with *ICA* provides 80.2% accuracy in the prediction process. This implies that proposed *ICA-GFFN* as a viable tool for optimizing multi-objective problems decreases the classification errors which can be interpreted as more precious results. Accordingly, the models were evaluated using statistical criteria where the *ICA-GFFN*

**Fig. 7** Influence of input parameters on predicted *UCS* and *E* using different sensitivity analyses



with 6.14, 0.178, 95.80 and 0.96 for *UCS* and 7.22, 0.241, 91.55 and 0.95 for *E* corresponding to *MAPE*, *RMSE*, *VAF* and  $R^2$  reflected better performance capacities than *GFFN*. The calculated *IA* index (0.90 for *UCS* and 0.88 for *E*) was another supplementary indicator that showed that the *ICA-GFFN* model produces closer predicted values to observations. The implemented sensitivity analyses showed that  $I_s$ ,  $V_p$  and  $n$  are the most effective factors on predicted *UCS* and *E* values. The result of sensitivity analyses can also be interpreted as the appropriate trend with previous empirical correlations which mostly have been established by these three factors and thus the obtained *MVR* correlations in this paper also can be calibrated with these effective factors.

## Compliance with ethical standards

**Conflicts of interest** The authors declare no conflict of interest.

## References

1. Abbaszadeh Shahri A, Larsson S, Renkel C (2020) Artificial intelligence models to generate visualized bedrock level: a case study in Sweden. *Model Earth Syst Environ*. <https://doi.org/10.1007/s40808-020-00767-0>
2. Abbaszadeh Shahri A, Maghsoudi Moud F, Mirfallah Lialestani SP (2020) A hybrid computing model to predict rock strength index properties using support vector regression. *Eng Comput*. <https://doi.org/10.1007/s00366-020-01078-9>
3. Abbaszadeh Shahri A, Asheghi R (2018) Optimized developed artificial neural network-based models to predict the blast-induced ground vibration. *Innov Infrastruct Solut*. <https://doi.org/10.1007/s41062-018-0137-4>
4. Abbaszadeh Shahri A, Larsson S, Johansson F (2016) Updated relations for the uniaxial compressive strength of marlstones based on P-wave velocity and point load index test. *Innov Infrastruct Solut* 1:17. <https://doi.org/10.1007/s41062-016-0016-9>
5. Abdechiri M, Faez K, Bahrami H (2010) Neural network learning based on chaotic imperialist competitive algorithm. In: IEEE proceedings of the 2nd international workshop on intelligent systems and applications (ISA), pp 1–5. <https://doi.org/10.1109/iwisa.2010.5473247>
6. Abdel-Basset M, Abdel-Fatah L, Sangaiah Ak (2018) Metaheuristic algorithms: a comprehensive review. In book: computational intelligence for multimedia big data on the cloud with engineering applications. *Intelligent data-centric systems*, pp 185–231. <https://doi.org/10.1016/b978-0-12-813314-9.00010-4>
7. Abdi Y, Taheri Gravand A, Zarei Sahamieh R (2018) Prediction of strength parameters of sedimentary rocks using artificial neural networks and regression analysis. *Arab J Geosci*. <https://doi.org/10.1007/s12517-018-3929-0>
8. Abdollahi M, Isazadeh A, Abdollahi D (2013) Imperialist competitive algorithm for solving systems of nonlinear equations. *Comput Math Appl* 65:1894–1908. <https://doi.org/10.1016/j.camwa.2013.04.018>
9. Aboutaleb Sh, Behnia M, Bagherpour R, Bluekian B (2017) Using non-destructive tests for estimating uniaxial compressive strength and static Young's modulus of carbonate rocks via some modeling techniques. *Bull Eng Geol Environ*. <https://doi.org/10.1007/s10064-017-1043-2>
10. Arora JS, Elwakeil OA, Chahande AI, Hsieh CC (1995) Global optimization methods for engineering applications: a review. *Struct Optim* 9:137–159. <https://doi.org/10.1007/BF01743964>
11. Arulampalam G, Bouzerdoum A (2002) Expanding the structure of shunting inhibitory artificial neural network classifiers. *IJCNN, IEEE*. <https://doi.org/10.1109/IJCNN.2002.1007601>
12. Atashpaz Gargari E, Lucas C (2007) Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: *Proceedings of the IEEE congress evolutionary of computer*, pp 4661–4667. <https://doi.org/10.1109/cec.2007.4425083>
13. Atashpaz-Gargari E, Hashemzadeh F, Rajabioun R, Lucas C (2008) Colonial competitive algorithm, a novel approach for PID controller design in MIMO distillation column process. *Int J Intell Comput Cybern*. 1:337–355. <https://doi.org/10.1108/17563780810893446>
14. Asheghi R, Abbaszadeh Shahri A, Khorsand Zak M (2019) Prediction of uniaxial compressive strength of different quarried rocks using metaheuristic algorithm. *Arab J Sci Eng*. <https://doi.org/10.1007/s13369-019-04046-8>
15. Asheghi R, Hosseini SA, Saneie M, Abbaszadeh Shahri A (2020) Updating the neural network sediment load models using different sensitivity analysis methods: a regional application. *J Hydroinf* 22(3):562–577. <https://doi.org/10.2166/hydro.2020.098>
16. Beiki M, Majidi A, Givshad AD (2013) Application of genetic programming to predict the uniaxial compressive strength and elastic modulus of carbonate rocks. *Int J Rock Mech Min Sci* 63:159–169. <https://doi.org/10.1016/j.ijmms.2013.08.004>
17. Biswas K, Vasant PM, Vintaned JAG, Watada J (2020) A review of metaheuristic algorithms for optimizing 3D well-path designs. *Arch Comput Methods Eng*. <https://doi.org/10.1007/s11831-020-09441-1>
18. Blum C, Roli A (2008) Hybrid metaheuristics: an introduction. In: Blum C, Aguilera MJB, Roli A, Sampels M (eds) *Hybrid metaheuristics. Studies in computational intelligence*, vol 114. Springer, Berlin, pp 1–30. [https://doi.org/10.1007/978-3-540-78295-7\\_1](https://doi.org/10.1007/978-3-540-78295-7_1)
19. Bouzerdoum A, Mueller R (2003) A generalized feedforward neural network architecture and its training using two stochastic search methods. In: Cantú-Paz E, et al. (eds) *Genetic and evolutionary computation—GECCO 2003*. GECCO 2003. Lecture notes in computer science, vol 2723. Springer, Berlin. [https://doi.org/10.1007/3-540-45105-6\\_89](https://doi.org/10.1007/3-540-45105-6_89)
20. Briševac Z, Hrženjak P, Buljan R (2016) Models for estimating uniaxial compressive strength and elastic modulus. *Grđevinar* 68(1):19–28. <https://doi.org/10.14256/JCE.1431.2015>
21. Colorni A, Dorigo M, Maniezzo V (1991) Distributed optimization by ant colonies, actes de la première conference européenne sur la vie artificielle. Elsevier Publishing, Paris, pp 134–142
22. Dede T, Kripka M, Togan V, Yepes V, Ravipudi V (2018) Advanced optimization techniques and their applications in civil engineering. *Adv Civ Eng*. <https://doi.org/10.1155/2018/5913083>
23. Dehghan S, Sattari GH, Chehreh Chelgani S, Aliabadi MA (2010) Prediction of uniaxial compressive strength and modulus of elasticity for Travertine samples using regression and artificial neural networks. *Min Sci Technol* 20:0041–0046. [https://doi.org/10.1016/S1674-5264\(09\)60158-7](https://doi.org/10.1016/S1674-5264(09)60158-7)
24. Ghaderi A, Abbaszadeh Shahri A, Larsson S (2019) An artificial neural network based model to predict spatial soil type distribution using piezocone penetration test data (CPTu). *Bull Eng Geol Environ* 78:4579–4588. <https://doi.org/10.1007/s10064-018-1400-9>
25. Gevrey M, Dimopoulos I, Lek S (2003) Review and comparison of methods to study the contribution of variables in artificial

- neural network models. *Ecol Model* 160(3):249–264. [https://doi.org/10.1016/S0304-3800\(02\)00257-0](https://doi.org/10.1016/S0304-3800(02)00257-0)
26. Gogna A, Tayal A (2013) Metaheuristics: review and application. *J Exp Theor Artif Intell* 25(4):503–526. <https://doi.org/10.1080/0952813X.2013.782347>
  27. Haghnejad A, Ahangari K, Noorzad A (2014) Investigation on various relations between uniaxial compressive strength, elasticity and deformation modulus of Asmari Formation in Iran. *Arabian J Sci Eng* 39:2677–2682. <https://doi.org/10.1007/s13369-014-0960-7>
  28. Hopper E, Turton BCH (2001) A review of the application of meta-heuristic algorithms to 2D strip packing problems. *Artif Intell Rev* 16:257–300. <https://doi.org/10.1023/A:1012590107280>
  29. Hosseini S, Al Khaled A (2014) A survey on the imperialist competitive algorithm metaheuristic: implementation in engineering domain and directions for future research. *Soft Comput J, Appl*. <https://doi.org/10.1016/j.asoc.2014.08.024>
  30. Jadhav S, Nalbalwar S, Ghatol A (2012) Performance evaluation of generalized feedforward neural network based ECG arrhythmia classifier. *Int J Comput Sci Issues* 9(4):379–384
  31. Kahraman S, Gunaydin O, Alber M, Fener M (2009) Evaluating the strength and deformability properties of misis fault breccia using artificial neural networks. *Expert Syst Appl* 36:6874–6878. <https://doi.org/10.1016/j.eswa.2008.08.002>
  32. Karaboğa D (2005) An idea based on honeybee swarm for numerical optimization. Technical report TR06, Department of computer engineering, Erciyes University, Türkiye
  33. Kennedy J, Eberhart R (1995) Particle swarm optimization. Proceedings of IEEE, international conference on neural networks. IV, 1942–1948. <https://doi.org/10.1109/icnn.1995.488968>
  34. Kirkpatrick S, Gelatt JCD, Vecchi MP (1983) Optimization by simulated annealing. *Science* 220(4598):671–680. <https://doi.org/10.1126/science.220.4598.671>
  35. Madhubabu N, Singh PK, Kainthola A, Mahanta B, Tripathy A, Singh TN (2017) Prediction of compressive strength and elastic modulus of carbonate rocks. *Measurement* 88:202–213. <https://doi.org/10.1016/j.measurement.2016.03.050>
  36. Mirjalili SA, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. *Adv Eng Softw* 69:46–61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
  37. Moradian ZA, Behnia M (2009) Predicting the uniaxial compressive strength and static Young's modulus of intact sedimentary rocks using the ultrasonic test. *Int J Geomech* 9(114):1532–3641. [https://doi.org/10.1061/\(asce\)1532-3641\(2009\)9:1\(14\)](https://doi.org/10.1061/(asce)1532-3641(2009)9:1(14))
  38. Najibi AR, Ghafoori M, Lashkaripour GR, Asef MR (2015) Empirical relations between strength and static and dynamic elastic properties of Asmari and Sarvak limestones, two main oil reservoirs of Iran. *J Petrol Sci Eng* 126:78–82. <https://doi.org/10.1016/j.petrol.2014.12.010>
  39. Onodera TF, Yoshinaka R, Oda M (1974) Weathering and its relation to mechanical properties of granite. In: Proceedings of the 3rd congress of ISRM, vol II(A), pp 71–78, Denver, USA
  40. Palchik V (2011) On the ratios between elastic modulus and uniaxial compressive strength of heterogeneous carbonate rocks. *Rock Mech Rock Eng* 44(1):121–128. <https://doi.org/10.1007/s00603-010-0112-7>
  41. Palchik V (1999) Influence of porosity and elastic modulus on uniaxial compressive strength in soft brittle porous sandstones. *Rock Mech Rock Eng* 32:303–309. <https://doi.org/10.1007/s006030050050>
  42. Pham DT, Ghanbarzadeh A, Koc E, Otri S, Rahim S, Zaidi M (2005) The bees algorithm. Technical note, Manufacturing engineering centre, Cardiff University, UK
  43. Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D, Saisana M, Tarantola S (2008) *Global sensitivity analysis: the primer*. Wiley, Chichester
  44. Sarkar K, Tiwary A, Singh TN (2010) Estimation of strength parameters of rock using artificial neural networks. *Bull Eng Geol Environ* 69(4):599–606. <https://doi.org/10.1007/s10064-010-0301-3>
  45. Stehman S (1997) Selecting and interpreting measures of thematic classification accuracy. *Remote Sens Environ* 62(1):77–89. [https://doi.org/10.1016/S0034-4257\(97\)00083-7](https://doi.org/10.1016/S0034-4257(97)00083-7)
  46. Torabi-Kaveh M, Naseri F, Saneie S, Sarshari S (2014) Application of artificial neural networks and multivariate statistics to predict UCS and E using physical properties of Asmari limestones. *Arab J Geosci*. <https://doi.org/10.1007/s12517-014-1331-0>
  47. Umrao RK, Lk Sharma, Singh R, Singh TN (2018) Determination of strength and modulus of elasticity of heterogenous sedimentary rocks: an ANFIS predictive technique. *Measurement*. <https://doi.org/10.1016/j.measurement.2018.05.064>
  48. Venter G (2010) Review of optimization techniques. In book: encyclopedia of aerospace engineering, <https://doi.org/10.1002/9780470686652.eae495>
  49. Willmott CJ (1984) On the evaluation of model performance in physical geography. *Spat Stat Models*. [https://doi.org/10.1007/978-94-017-3048-8\\_23](https://doi.org/10.1007/978-94-017-3048-8_23)
  50. Wong WK, Ming CI (2019) A review on metaheuristic algorithms: recent trends, benchmarking and applications. In: IEEE, 7th international conference on smart computing and communications (ICSCC), Malasiya, <https://doi.org/10.1109/icssc.2019.8843624>
  51. Yagiz S (2009) Predicting uniaxial compressive strength, modulus of elasticity and index properties of rocks using the Schmidt hammer. *Bull Eng Geol Environ* 68(1):55–63. <https://doi.org/10.1007/s10064-008-0172-z>
  52. Yagiz S, Sezer EA, Gokceoglu C (2012) Artificial neural networks and nonlinear regression techniques to assess the influence of slake durability cycles on the prediction of uniaxial compressive strength and modulus of elasticity for carbonate rocks. *Int J Numer Anal Met Geomech* 36(14):1636–1650. <https://doi.org/10.1002/nag.1066>
  53. Yang XS (2008) *Firefly algorithm*. Luniver Press, Bristol
  54. Yang XS (2011) Metaheuristic optimization: algorithm analysis and open problems. In: Pardalos PM, Rebennack S (eds) *Experimental algorithms*, SEA 2011, Lecture notes in computer science, vol 6630, Springer, Berlin, [https://doi.org/10.1007/978-3-642-20662-7\\_2](https://doi.org/10.1007/978-3-642-20662-7_2)
  55. Yasar E, Erdogan Y (2004) Correlating sound velocity with the density, compressive strength and Young's modulus of carbonate rocks. *Int J Rock Mech Min Sci*. 41(5):871–875. <https://doi.org/10.1016/j.ijrmms.2004.01.012>
  56. Yilmaz I, Yuksek AG (2009) Prediction of the strength and elasticity modulus of gypsum using multiple regression, ANN and ANFIS models. *Int J Rock Mech Min Sci* 46(4):803–810. <https://doi.org/10.1016/j.ijrmms.2008.09.002>
  57. Yun Y, Moon C, Kim D (2009) Hybrid genetic algorithm with adaptive local search scheme for solving multistage-based supply chain problems. *Comput Ind Eng* 56(3):821–838. <https://doi.org/10.1016/j.cie.2008.09.016>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.