-RESEARCH PAPER-

International Journal of Mining and Geo-Engineering IJMGE

IJMGE 56-2 (2022) 97-105 DOI: 10.22059/IJMGE.2021.314154.594880

Comparison between the performances of four metaheuristic algorithms in training a multilayer perceptron machine for gold grade estimation

Andisheh Alimoradi [a, *], Hossein Hajkarimian [a], Hamidreza Hemati Ahooi [a], Mohammad Salsabili [b]

[a] *Department of Mining Engineering, Imam Khomeini International University, Qazvin, Iran*

[b] *Departement des sciences appliquees, Universite du Quebec a Chicoutimi, Quebec, Canada*

Article History:

Received: 23 November 2020.

Revised: 29 December 2020.

Accepted: 08 January 2021.

ABSTRACT

Reserve evaluation is a very difficult and complex process. The most important and yet most challenging part of this process is grade estimation. Its difficulty derived from challenges in obtaining required data from the deposit by drilling boreholes, which is a very time-consuming and costly act itself. Classic methods which are used to model the deposit are based on some preliminary assumptions about reserve continuity and grade spatial distribution which are not true about all kind of reserves. In this paper, a multilayer perceptron (MLP) artificial neural network (ANN) is applied to solve the problem of ore grade estimation of highly sparse data from Zarshouran gold deposits in Iran. The network is trained using four metaheuristic algorithms in separate stages for each algorithm. These algorithms are artificial bee colony (ABC), genetic algorithm (GA), imperialist competitive algorithm (ICA), and particle swarm optimization (PSO). The accuracy of predictions obtained from each algorithm in each stage of experiments was compared with real gold grade values. We used unskillful value to check the accuracy and stability of each network. Results showed that the network trained with the ABC algorithm outperforms other networks that trained with other algorithms in all stages having the least unskillful value of 13.91 for validation data. Therefore, it can be more suitable for solving the problem of predicting ore grade values using highly sparse data.

Keywords: *Multilayer perceptron, Metaheuristic machine learning, Grade estimation, Inverse modeling, Optimization*

1. Introduction

Ore grade estimation is a very complicated, time-consuming, and costly process. The complexity of this process derives from scientific uncertainty and the necessity for human intervention. Also, in almost all grade estimation cases, drilling is the major method to obtain samples from ore bodies and constructing a database for use in the estimation process. Drilling is one of the most expensive operations in the mining industry since more than 80 percent of the exploration costs are related to the drilling activities (Edwards and Atkinson, 1986) and it makes the grade estimation a very costly act (Kapageridis, 1999). However, it is very important to do this phase with the maximum possible accuracy, since it can have a significant role in the mining future planning and furthermore, it is applied as the major tool to distinguish the borders between economic and non-economic deposits (Journel and Huijbregts, 1978).

For the past forty years, geostatistical-based methods have been the main approach for solving the problem of grade estimation. These methods have been based on certain assumptions about the spatial distribution of ore grades within the deposit (Bárdossy and Fodor, 2004). Negative effects of these assumptions made researchers establish more complicated methods for reducing the role of assumptions. However, these newer established methods require a large amount of knowledge and expertise in order to be effectively applied (Bárdossy and Fodor, 2004). Geostatistical methods such as kriging are indeed robust and powerful tools, but in some cases in which spatial patterns relationships and the grade distribution among ore body are complicated, Geostatistics are not always able to give the most optimum answer (Strebelle, 2002). Therefore, using nonlinear estimators like artificial neural networks (ANN) may be a proper resolution to overcome the problem of finding a complex spatial relationship among ore body.

Many researchers have employed ANN and its various types for grade estimation in recent years. For example, a wavelet neural network (WNN) has been used successfully for grade estimation of a porphyry copper deposit (Li, 2010). In another case, a radial basis function network (RBF) has been employed for grade estimation in an iron deposit and its results have been compared to geostatistics (Kapageridis and Denby, 1999). Comparison between a multilayer perceptron (MLP) ANN results and geostatistics has been reported for a limestone deposit in another paper (Chatterjee, 2006). In another case, one of the newer kriging methods which are called median indicator kriging has been compared with an ANN for grade estimation in an iron ore deposit (Badel et al, 2011). Grade estimation results of a placer gold deposit obtained from a feedforward ANN network have been compared to estimation results from a support vector machine (SVM) in another paper (Dutta, 2010). A similar comparison has been made for an iron mine in another study (Maleki et al, 2014). In one research, a method called ANNMG is presented to integrate ANNs and geostatistics for optimum mineral reserve evaluation. The results are very promising (Jalloh, 2016). Several studies which are more or less similar to discussed ones, published in recent years (Koike, 2002; Samanta, 2004; Samanta et al 2005; Samanta et al, 2006; Samanta and Bandopadhyay, 2009; Tahmasebi and Hezarkhani, 2011). All of these researches are showing that the ANN can be used as a reliable approach to obtain mostly accurate grade estimations.

* Corresponding author. *E-mail address:* alimoradi@eng.ikiu.ac.ir (A. Alimoradi).

Metaheuristic algorithms are often nature-inspired, and they are now among the most widely used algorithms for optimization. Metaheuristic algorithms are very diverse, including but not limited to, simulated annealing, ant and bee algorithms, genetic algorithms, harmony search, differential evolution, particle swarm optimization, imperialist competitive algorithm, firefly algorithm, cuckoo search, etc. (Yang, 2013).

In this paper, in order to solve the problem of grade estimation for a Carlin-type gold deposit (Alimoradi et al, 2020), four metaheuristic algorithms which are artificial bee colony (ABC), genetic algorithm (GA), imperialist competitive algorithm (ICA) and particle swarm optimization (PSO), have been used as the training algorithm for an MLP neural network to find the unknown nonlinear relations between known and unknown gold grade values in the boreholes. A more detailed explanation of these metaheuristics and also the method of training an MLP with these algorithms have been discussed in the third chapter.

2. Zarshouran Gold deposit

Zarshouran is a Gold-Arsenic deposit located 42 km north of the town of Takab in the Province of West Azerbaijan, northwest Iran (Alimoradi et al, 2020). Zarshouran belongs to a group of sedimentary-rock-hosted gold deposits which are very similar to Carlin-type sediment-hosted gold deposits (Paar, 2009). Both deposits are located within the active geothermal field of the Northern Takab region where thermal springs locally precipitate high amounts of gold and silver (Daliran, 2002). Figure 1 shows the location of the Zarshouran gold deposit.

3. Methodology

For a better understanding of the method that has been used in this paper for grade estimation, first, it is necessary to explain some theorems that are relevant to the design of the integrated algorithm. Figure 2. Shows the workflow of the machine learning procedure.

3.1. Multilayer Perceptron neural network

ANNs are some sort of computing system that contain processing elements in an interrelated network structural form. ANNs are the mathematical simulation of human neural systems and use the process of learning from available examples for gaining the ability to recognize the patterns among them. In an ANN, the neurons are placed in layers, where neurons of each layer are linked to neurons of the adjacent layer. The transfer function (TF) is the processing tool for neurons to work on signals (data) they receive. Usually, the integration of linear and non-linear TFs is used to assist the network in solving non-linear and complex problems. The whole process of learning can be defined as the operation of setting the synaptic weights of the links between neurons of different layers. Details of basic and advanced mechanisms of ANNs have been described at length in the literature (Daliran, 2003). There are various types of ANNs such as multilayer perceptron (MLP), general regression, RBF and time delay networks, etc. in this paper an MLP network has been used for grade estimation in the Zarshouran gold deposit.

The multilayer perceptron neural network is considered one of the best approximation methods for the prediction of the nonlinear relationship between inputs and outputs of a given dataset. In MLP, in order to gain the network output, elements in the input and hidden layer are manipulated by a weighting function. In addition, constant values, which are called biases of each layer, allows the network to shift the activation function to the left or right, which is so critical for better learning. The process of determining the weights and biases is called training which is done by using learning or training algorithms. In most cases, gradient-based learning algorithms have been used to do this task. But in this paper, in order to alleviate some disadvantages of gradient-based training algorithms, such as becoming trapped in local minima, the network has been trained with four metaheuristic algorithms. The theory of these algorithms will be discussed in continue.

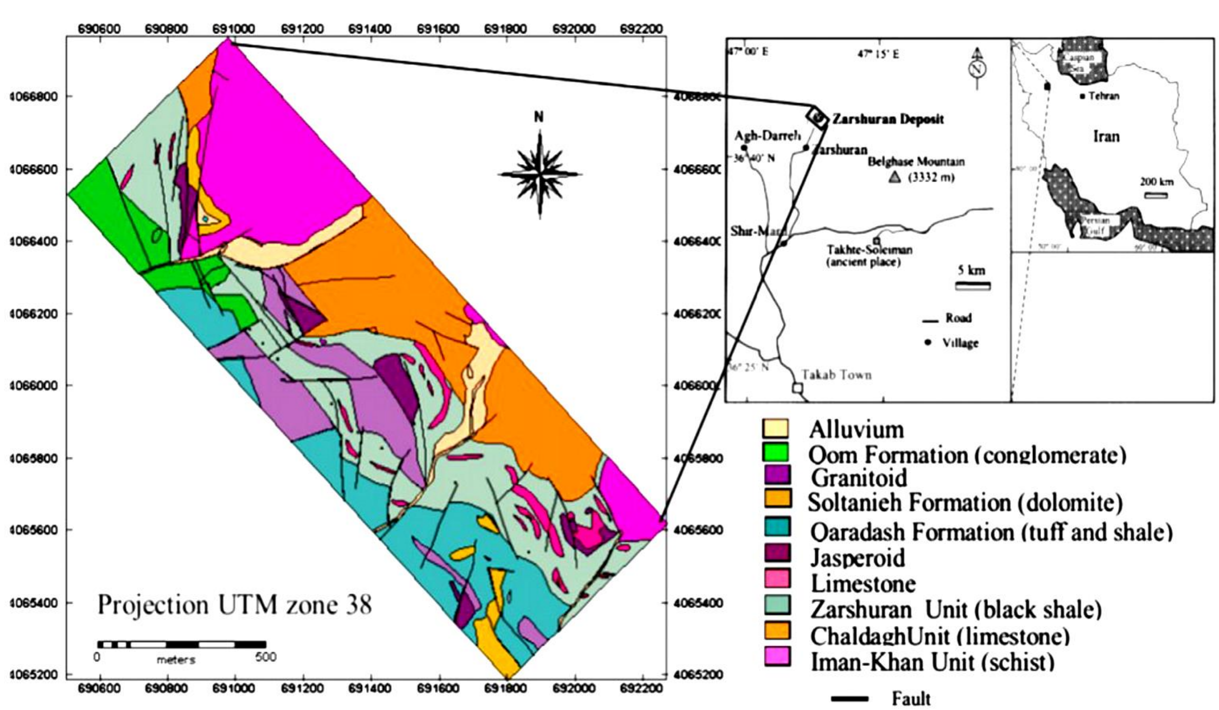

Figure 1. Zarshouran location and simplified lithological map [31]

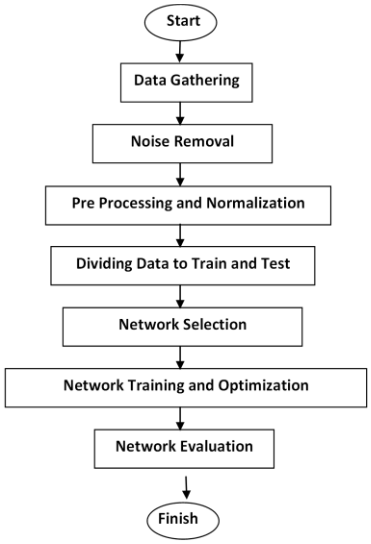

Figure 2. Workflow chart

3.2. ABC algorithm

Karaboga developed an Artificial Bee Colony (ABC) algorithm based on the behavior of bees in searching for flowers, which is called waggle dancing, and proposed it for solving optimization problems (Paravarzar, 2014). In recent years, the ABC algorithm has been efficiently used for solving a vast spectrum of engineering optimization problems. It can be used as the training algorithm for better and more optimum learning in neural networks (Haykin, 1999).

Like in nature, there are three groups of bees in the ABC algorithm: employed bees, onlookers, and scouts. In most cases, half of the colony population are considered as the employed bees and obviously, the other half will be onlookers. Each food source is occupied by one and only one employed bee. Therefore, the numbers of employed bees and food sources are equal. Each food source position corresponds to a potential solution to the optimization problem. The position of each food source is generated randomly by using the following equation:

$$x_{i,j} = x_j^{min} + \Phi(x_j^{max} - x_j^{min}) \tag{1}$$

where i = 1, 2..., NP, D is the optimization problem dimension and ϕ is a random number uniformly distributed in the interval [0,1]. x_j^{min} and x_j^{max} denote the lower bound and upper bound of the optimization problem, respectively. After the generating of initial positions, each position is updated according to:

$$v_{i,j} = x_{i,j} + \delta(x_{i,j} - x_{k,j}) \tag{2}$$

where k ∈ 1, 2..., NP and δ is a random number uniformly distributed in the interval [−1,1]. If the current position is beyond the lower or upper bounds of the optimization problem, boundary value would be considered for it. After that, the quality of each food source is evaluated using a fitness function or a cost function. Then, the onlooker bees start their working process and choose a food source based on its probability of selection which is calculated as follows:

$$p_i = \frac{fit_i}{\sum_{i=1}^{NP} fit_i} \tag{3}$$

In this equation, fit_i is the fitness (or cost) value of the selected food source, which is corresponding to the quality of the i_{th} food source. Each food source is used to generate a candidate position by Eq. (2). If the candidate has better quality, the old food source is replaced by it. The maximum number of updates for each food source is controlled by a parameter, which is called "*limit*. If the number of updates for a source position reaches the limit value, that source is rejected, and scout bees generate a new food source randomly by Eq. (1). ABC algorithm is very flexible to solve both continuous and discrete optimization problems. Therefore, it can be a great and suitable substitute for gradient-based training algorithms.

3.3. GA

Ga is a nature-based computation method for solving a wide range of real-world optimization problems. GA can be used for various types of optimization problems that cannot be solved easily by other standard optimization algorithms; such as problems with discontinuous objective function, stochastic, highly nonlinear, or non-differentiable optimization problems (Karaboga, 2005). GA major tools to find a proper and optimum solution are crossover and mutation processes. GA algorithms use a crossover operator to mix two initial solutions, which are called parents, to produce offsprings. Then mutation operator is applied randomly on offsprings in order to make them more unique in comparison to their parents. After that, if the offsprings have better fitness values based on the type of optimization problem, the parents are replaced by their offsprings in a process which is called survivor selection (Karaboga, 2007). Advantageous features of GA and the reason behind choosing it for training an MLP for grade estimation are that it can be used with both continuous or discrete parameters, does not need any assumptions about the problem, and, unlike gradient methods, it does not require computation of derivative information to reach to an optimum result.

3.4. ICA

Atashpaz-Gargari and Lucas proposed the ICA as the proper method for solving various optimization problems in 2007 (Goldberg and Holland, 1988). Like other population-based algorithms, ICA starts with an initial population. Each individual of the population is called a country in which some having the least cost are considered imperialist and the rest are the colonies of these imperialists. The division of the colonies of initial countries is based upon the power of the imperialist. so, at first, an imperialist's normalized cost is defined by Eq. (4):

$$C_n = max_i\{c_i\} - c_n \tag{4}$$

In this equation, C_n is the cost of n_{th} imperialist and c_n is the normalized cost of that imperialist. After the normalized cost of all imperialists is calculated, the normalized power of each imperialist would be:

$$p_n = \left| \frac{c_n}{\sum_{i=1}^{n_{imp}} c_i} \right| \tag{5}$$

The normalized power of an imperialist is the number of colonies that are dominated and controlled by that imperialist. the initial number of colonies of an empire would be determined by the following equation:

$$N.C_n = round\{p_n.(N_{col})\} \tag{6}$$

$N.C_n$ is the initial number of lesser countries or colonies of nth empire and N_{col} is the number of all colonies. $N.C_n$ is chosen randomly to divide the colonies for each imperialist (Goldberg and Holland, 1988).

The colonies in each of the empires start moving towards their imperialist, based on the assimilation policy. Figure 3 shows this movement. In this movement, θ and x are arbitrary numbers that are generated based on $x \sim U(0, \beta \times d)$, $\theta \sim U(-\gamma, \gamma)$. d is the notion of distance between imperialist and its colony and β must be greater than 1. This constraint causes the colonies to get closer to the imperialist state from both sides. Moreover, γ is a parameter that adopts the deviation from the main direction. Although β and γ are random numbers, most of the times the best-fitted value of β and γ are approximately 2 and π/4 (Rad).

The total power of an empire is defined by the imperialist's power and the percentage of the colony's power. Therefore, the total cost defines by:

$$T.C_n = Cost(imperialist_n) + \xi\ mean\ \{Cost(colonies\ of\ empire_n)\} \tag{7}$$

$T.C_n$ is the total cost of nth empire and ξ is a positive number, which is considered to be less than 1.

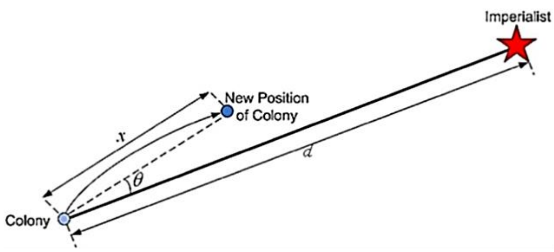

Figure 3. Movement of colonies toward their relevant imperialist [38]

The process of ICA begins after defining the above equations. Like real-world superpower competition for controlling the smaller countries, in ICA empires try to get the colonies of other empires into their domain. When the ICA gets to its next iterations, the power of more powerful empires will increase and therefore will end up in the reduction of the power of weaker empires. At the beginning of the competition, first one must find the probability of possessions for each empire based on its total power. The normalized total cost is simply obtained by:

$$N.T.C_n = max_i(T.C_i) - T.C_n \tag{8}$$

$N.T.C_n$ is the normalized cost of the n_{th} empire. Having the normalized total cost, the possession probability of each empire would be:

$$P_{p_n} = \left|\frac{N.T.C_n}{\sum_{i=1}^{N_{imp}} N.T.C_i}\right| \tag{9}$$

Finally, these processes will successfully cause all the countries to converge to a situation in which only one empire exists in the world and all the other countries are colonies of that empire and they have the same position and power as the imperialist (Goldberg and Holland, 1988). ICA is not tested before for training a neural network on the problem of grade estimation in any available published papers. So, its ability in this field can be evaluated and compared with the other three algorithms in this paper for the first time.

3.5. PSO

Eberhart and Kennedy (Talbi, 2009) introduced the PSO as an optimization algorithm that derives its inspiration from the social behavior of birds and fishes. PSO can be used to solve a wide range of optimization problems, from nonlinear continuous functions to the most complex engineering problems. (Atashpaz-Gargari and Lucas, 2007).

In this paper, the global PSO algorithm has been used, which is described as follows. If the search space of optimization is considered as a D-dimensional space, then a D-dimensional vector (X_i) can be the representative of the i_{th} particle of the swarm, $X_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^T$. Another D-dimensional vector can be considered as the velocity or position change vector for each particle, $V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})^T$. Parameter g is defined as the best particle of the swarm index and the superscripts will show the iteration number. After each iteration, the position of every particle is updated based on that particle best exploration, best exploration among all the swarm, and also the previous velocity vector of the particle by using the following two equations (Talbi, 2009):

$$v_{id}^{n+1} = v_{id}^n + cr_1^n(P_{id}^n - x_{id}^n) + cr_2^n\left(P_{gd}^n - x_{id}^n\right) \tag{10}$$

$$x_{id}^{n+1} = x_i^d + v_{id}^{n+1} \tag{11}$$

In these equations, d = 1, 2, . . .,D; i = 1, 2,. . .,N and N is the swarm size; a constant value which is called acceleration constant is noted by c; r_1 and r_2 are random numbers. A fitness or objective function which is suitable for the defined problem is used to evaluate the performance of each particle.

Shi proposed using a parameter which is called maximum velocity (V_{max}), which would improve the precision of the algorithm. It can make the particle continue the search in the region. based on this proposition, Eqs. (11) and (12) were modified as following equations in the later versions of the PSO (Eberhart and Kennedy, 2015):

$$v_{id}^{n+1} = \chi\left(wv_{id}^n + c_1r_1^n(p_{id}^n - x_{id}^n) + c_2r_2^n(p_{gd}^n - x_{id}^n)\right) \tag{12}$$

$$x_{id}^{n+1} = x_i^d + v_{id}^{n+1} \tag{13}$$

w is called inertia weight; c_1 and c_2 are constant values called cognitive and social parameters, respectively; and χ is a constriction factor.

PSO is extremely computationally inexpensive in terms of both memory requirements and speed. It also has the flexibility to be integrated with other optimization and soft-computing techniques to form hybrid tools (Gopalakrishnan, 2013). In addition to these advantages, it was never coded as the training algorithm of a neural network to be used for grade estimation before. So, the performance of this optimizer in these kinds of operations can be tested for the first time in this research.

3.6. Training MLP network by Metaheuristic algorithms for Grade Estimation

Applying the metaheuristic algorithms to train neural networks is relatively straightforward. The goal of metaheuristic optimizers is minimizing a cost function or maximizing a fitness function. So, if the process of defining the weights and biases be determined as an optimizing problem, then metaheuristic algorithms can be used to solve this problem and in other words, train the neural network.

To do this, first, we need to extract the weights and biases from an initial network. Then the process of defining these parameters should be coded as the cost function of metaheuristics. After that, the whole process of the metaheuristics should be considered as a function which would be the training function for the network.

In this paper, the ending criteria is to stop training and optimizing the network after some fixed number of iterations [32]. This method helps better understanding the results change with the different number of iterations. Also, the hyperbolic tangent sigmoid transfer function was set as the network's TF.

To compare the performance of each metaheuristic algorithm in training the MLP network for grade estimation, it is needed to use some criteria such as mean absolute error (MAE) and route mean square error (RMSE). Unskillful value (Dutta, 2010) is a quantity that shows the inability of the methods in their task. it is obvious that lower values are better and showing that the respective algorithm is more suitable for doing the estimation. It is defined as follows:

$$unskillful\ value = MAE + RMSE + ((1 - R^2) \times 100) \tag{14}$$

where MAE is mean absolute error, RMSE is Root Mean Square Error and R^2 is the coefficient of determination.

3.7. Data Statistics

Usually, the grade estimation computations are done with borehole log data which are taken directly from the deposit. The borehole data represent assay and surveying values of 49 boreholes of varying depth intervals from 60 to 230 m. Assay and surveying files typically contain information on the sample coordinates (easting, northing, and elevation), length and ore grades. Unfortunately, the boreholes were drilled at irregular intervals which can make many problems in the way of reaching an optimal grade estimation. The assay value for each borehole was collected at intervals of 1 m depth. Figure 4 shows the histogram of the distribution of gold grades from the assay data of drill holes, composited into 5 m samples, together with brief descriptive statistics.

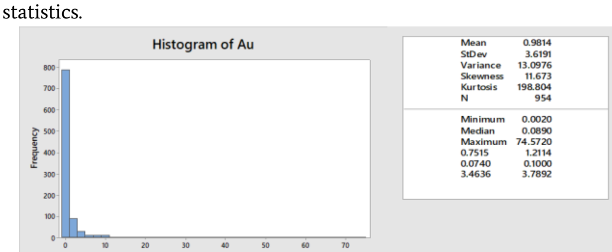

Figure 4. Histogram and the summary of descriptive statistics (using MINITAB)

The highly positive skewed shape is seen in the histogram graph. This feature of gold grade distribution is a norm as most of the gold deposits in nature typically occur sporadically in a few small patches with a high gold concentration in the region of the low-grade zone. Deeper studies of the drill holes demonstrate that high-grade values don't show any regular trend and occasionally appear amid low values. Therefore, modeling such data is difficult and the accuracy of predictive models is reduced due to the nature of data. The kurtosis of this dataset is 198.8 which is far away from the kurtosis of a normally distributed dataset (= 3). In addition, the high variance value of the dataset showing the sparseness of grade distribution around the mean value.

However, it is discernible that the available gold data are statistically sparse and of course because of the random nature of gold, the spatial modeling of this dataset is a very complex operation.

3.8. Data Preparation

In order to evaluate the performance of the proposed method on ore grade estimation, at first, data are divided into training and test subsets randomly. unfortunately, there is no geological information available in the dataset. Therefore, northing, easting, and elevation are considered as inputs for the networks and Au value is the outputs (Pyrcz et al, 2006). Data are normalized to the range between −1 and 1 by using the following equation:

$$X_{norm} = \frac{(X_{old} - \frac{1}{2}(X_{max} - X_{min}))}{\frac{1}{2}(X_{max} - X_{min})} \qquad (15)$$

where X_{norm} is the normalized value, X_{old} denotes the original value, and X_{min}, X_{max} are the minimum and maximum of the original values, respectively.

As it said before, reaching a completely accurate estimation for a random nature metal like gold is so hard and it is almost impossible in a vast area. So, if the aim of estimation reduced to finding the spatial correlation between grades of smaller sections of the area of study, it is possible to gain better and more accurate estimations in these parts and after that, with the integration of obtained results, reaching to reliable estimation of the whole area will be attainable. Therefore, in addition to using the whole area dataset for training the neural network, a clustering method named Self-organizing map (SOM) was applied to data in order to divide the boreholes into three sections of more compacted ones in each group to train network with them separately. The explanation of this method has been discussed in the literature (Abraham et al, 2006). The first cluster is in the west part of the area and consists of 16 boreholes and 311 composites. The second one is in the center of the area and consists of 24 boreholes and 404 composites. The last one is on the east side of the area and consists of 9 boreholes and 239 composites. Figure 5 shows these clusters.

4. Results

Table 1 shows the parameter settings for each algorithm. These values have been set according to the best results obtained from trial and error in 40 executions of each algorithm.

Table 1. Control parameters of algorithms, using whole data

Algorithm	Parameters
MLP/ABC	11 neurons in 1 hidden layer, iterations = 130, employed = 350, onlookers = colony size
MLP/GA	16 and 9 neurons in 2 hidden layers, generations(iterations) = 80, population = 300
MLP/ICA	12 neurons in 1 hidden layer, total countries = 400, initial imperialists: 40, decades = 40, revolution rate = 0.3
MLP/PSO	20 neurons in 1 hidden layer, swarm size = 350, iterations = 100

Figures 5, 6, 7 and 8 show outputs vs. targets comparison and regression graphs of test subset which related to ABC, GA, ICA, and PSO algorithms, respectively.

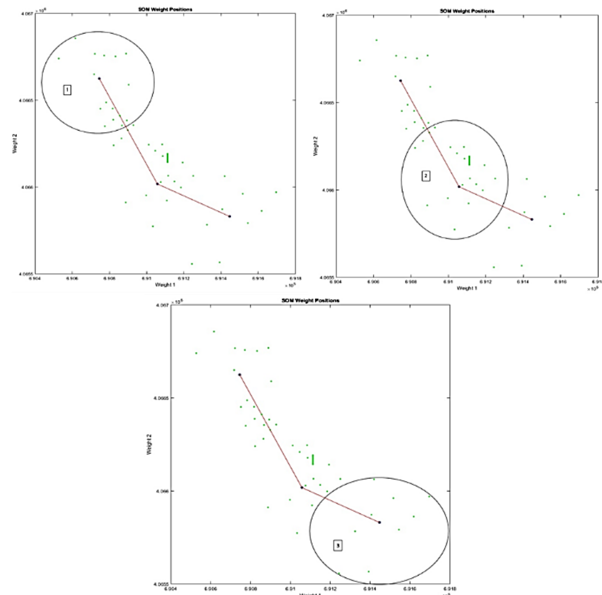

Figure 5. Clustered boreholes: 1. west part, 2. center part, 3. east part

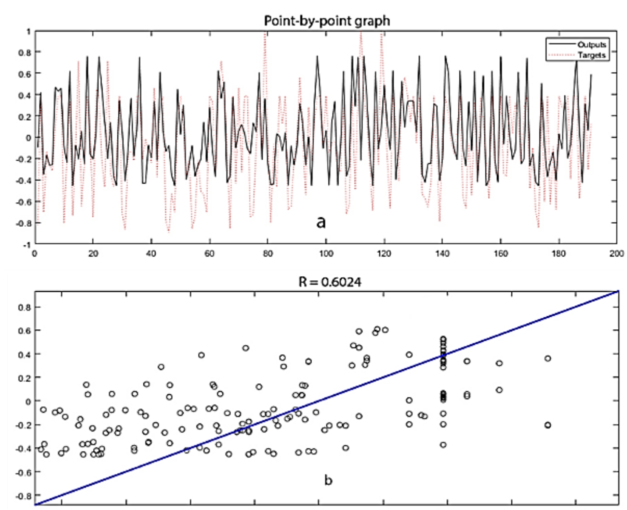

Figure 6. MLP/ABC results, using whole data: a) outputs vs. targets b) regression

Table 2 shows unskillful values of training algorithms. Based on results obtained from all methods, it can be said that none of the estimations are completely reliable.

The random nature of gold and lack of proper data, especially the fact that geological and lithological analyses were not available, must be the reasons behind not desirable results. However, the results are considerably more accurate in comparison to studies that implemented the A.I. approach for gold grade estimation) Samanta and Bandopadhyay, 2009; Dutta, 2010). In this stage, the network which was trained with the ABC algorithm has been shown a better performance. Although the PSO results are very close to being the best. So, it can be said these two algorithms outperform others in this phase of estimation. For further investigation, clustered data were used. In this stage, only outputs vs. targets graphs for the first cluster are reported and for other clusters, just control parameters and unskillful values are appended, because for comparison it would be sufficient.

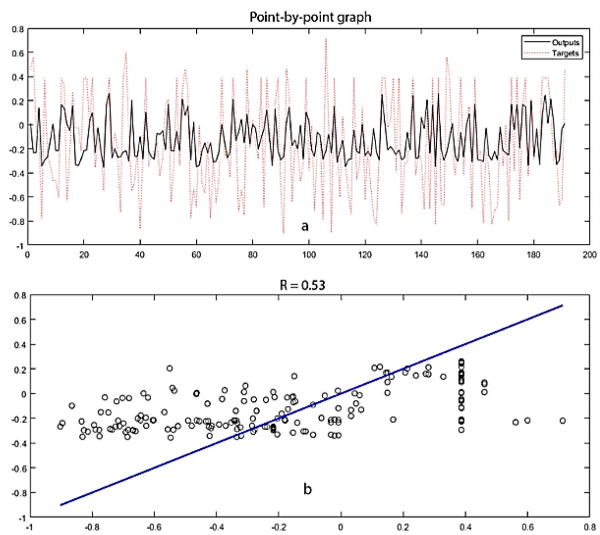

Figure 7. MLP/GA results, using whole data: a) outputs vs. targets b) regression

Table 2. Unskillful values of algorithms, using whole data
(* sign shows the best result)

Training algorithm	ABC	GA	ICA	PSO
Unskillful value	64.69*	72.54	76.62	66.56

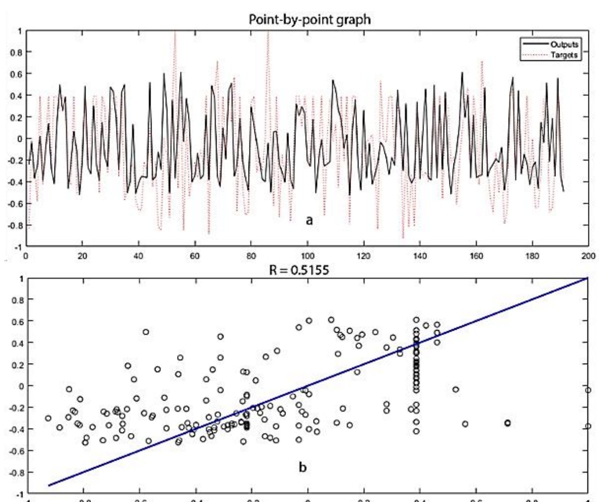

Figure 8. MLP/ICA results, using whole data: a) outputs vs. targets b) regression

Table 3 and Table 4 show control parameters and unskillful values of algorithms for doing the estimations with first cluster data, respectively.

Table 3. Control parameters of algorithms, using 1st cluster data

Algorithm	Parameters
MLP/ABC	9 neurons in 1 hidden layer, iterations = 100, employed = 300, onlookers = colony size
MLP/GA	20 neurons in 1 hidden layer, generations(iterations) = 80, population = 260
MLP/ICA	22 neurons in 1 hidden layer, total countries = 300, initial imperialists: 50, decades = 40, revolution rate = 0.3
MLP/PSO	18 neurons in 1 hidden layer, swarm size = 350, iterations = 100

Table 4. Unskillful values of algorithms, using 1st cluster data
(* sign shows the best result)

Training algorithm	ABC	GA	ICA	PSO
Unskillful value	17.7*	39.02	39.04	28.03

Figures 9, 10, 11, 12 and 13 show outputs vs. targets comparison graphs obtained from each algorithm estimation with first cluster data. In these figures, sample numbers are the number of output data that has been selected randomly as Train, Test, and Validation samples.

Table 5 and Table 6 are control parameters and unskillful values of algorithms using second cluster data, respectively.

In the same way, Table 7 and Table 8 are reporting the same information about the third cluster.

Table 5. Control parameters of algorithms, using 2nd cluster data

Algorithm	Parameters
MLP/ABC	14 neurons in 1 hidden layer, iterations = 180, employed = 350, onlookers = colony size
MLP/GA	16 neurons in each one of 2 hidden layers, generations(iterations) = 100, population = 500
MLP/ICA	16 neurons in 1 hidden layer, total countries = 500, initial imperialists: 55, decades = 38, revolution rate = 0.3
MLP/PSO	15 neurons in 1 hidden layer, swarm size = 380, iterations = 100

Table 6. Unskillful values of algorithms, using 2nd cluster data
(* sign shows the best result)

Training algorithm	ABC	GA	ICA	PSO
Unskillful value	48.82*	65.78	57.08	54.27

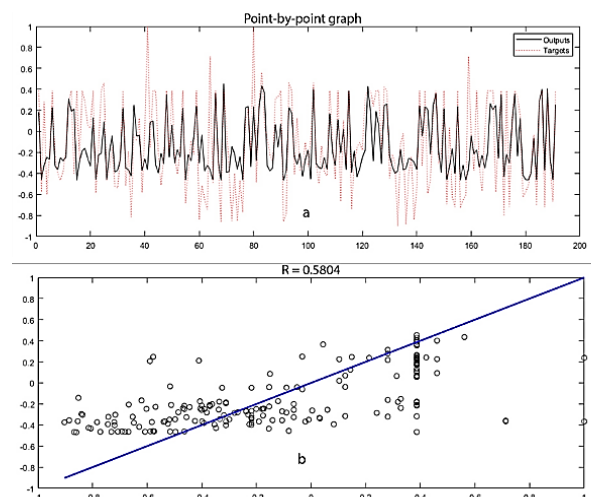

Figure 9. MLP/PSO results, using whole data: a) outputs vs. targets b) regression

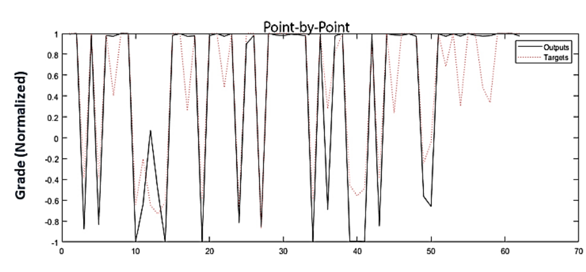

Figure 10. MLP/ABC results, using 1st cluster data

The analysis of results demonstrates that by dividing a gold-rich area into smaller parts, it would be possible to reach a more precise estimation for each section despite all complexities. Improvements in the accuracy of estimations can be seen clearly in all three clusters, but the results obtained from the first cluster in the west part of the area of study are the most accurate ones and therefore are so promising. It seems the boreholes' intervals play a major role in this issue because in the first cluster the boreholes are more compact and were drilled in a more regular pattern. It is just a hypothesis that should be tested.

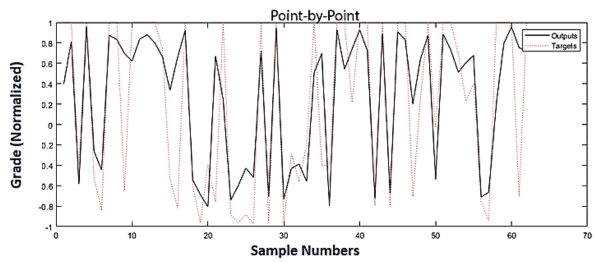

Figure 11. MLP/GA results, using 1st cluster data

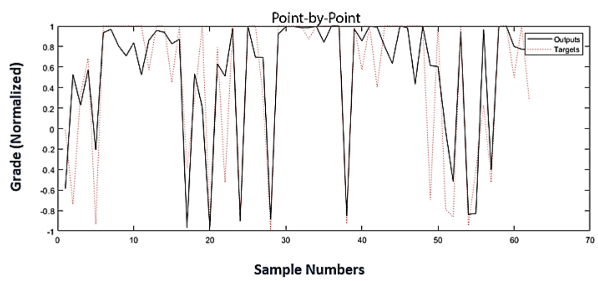

Figure 12. MLP/ICA results, using 1st cluster data

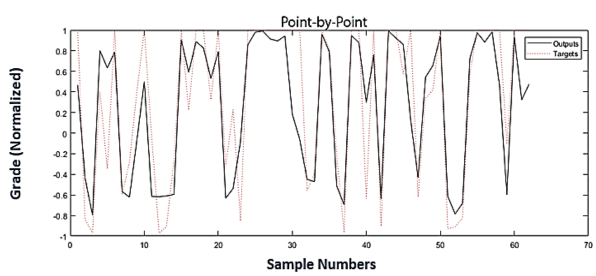

Figure 13. MLP/PSO results, using 1st cluster data

Table 7. Control parameters of algorithms, using 3rd cluster data

Algorithm	Parameters
MLP/ABC	11 and 13 neurons in 2 hidden layers, iterations = 100, employed = 150, onlookers = colony size
MLP/GA	25 neurons in 1 hidden layer, generations(iterations) = 100, population = 250
MLP/ICA	20 neurons in 1 hidden layer, total countries = 300, initial imperialists: 50, decades = 33, revolution rate = 0.3
MLP/PSO	9 neurons in 1 hidden layer, swarm size = 200, iterations = 100

Table 8. Unskillful values of algorithms, using 3rd cluster data

(* sign shows the best result)

Training algorithm	ABC	GA	ICA	PSO
Unskillful value	57.19*	67.06	65.77	60.94

In order to evaluate this probability, the data that belong to the most compact part of the first cluster boreholes, as can be seen in Fig 14, were considered as the input dataset for grade estimator networks.

This section contains 8 boreholes and 95 composites. Fig 15 shows outputs vs. targets comparison of test subset which obtained by implementation of MLP/ABC algorithm. The graphs of other algorithms were excluded from the report to prevent the article from being too lengthy.

However, control parameters and unskillful values of all algorithms are reported in Tables 9 and 10, respectively.

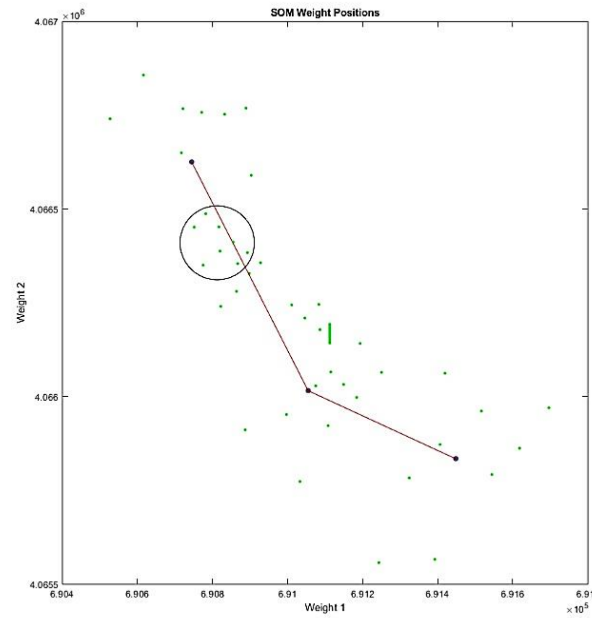

Figure 14. Compact boreholes of 1st cluster

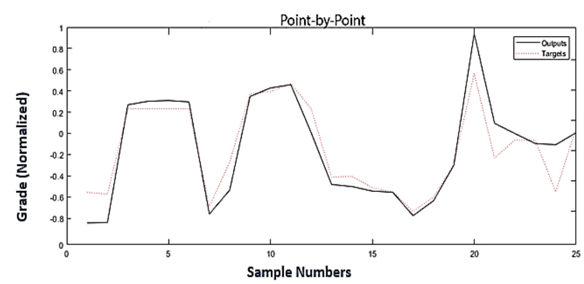

Figure 15. MLP/ABC results using compact boreholes data

Table 9. control parameters of algorithms, using compact boreholes data

Algorithm	Parameters
MLP/ABC	8 neurons in 1 hidden layer, iterations = 100, employed = 75, onlookers = colony size
MLP/GA	14 neurons in 1 hidden layer, generations(iterations) = 100, population = 80
MLP/ICA	12 neurons in 1 hidden layer, total countries = 35, initial imperialists: 7, decades = 40, revolution rate = 0.3
MLP/PSO	10 neurons in 1 hidden layer, swarm size = 75, iterations = 100

Table 10. unskillful values of algorithms, using compact boreholes data

(* sign shows the best result)

Training algorithm	ABC	GA	ICA	PSO
Unskillful value	13.91*	23.08	21.23	19.45

Based on these results, it can be said that all algorithms performed very well in this stage and the estimations are very accurate. Like before, the network which was trained with the ABC algorithm outperformed others. It should be noted the dataset which was used in this stage is very small and the results obtained from using such small data for estimation, cannot be generalized for the whole area. But if reaching an optimal estimation is not possible in a large area, especially when a random nature metal like gold is under study, implementing the methods on smaller parts and then integrating their high accuracy estimations, may allow us to achieve desirable results.

5. Conclusion

Grade estimation is one of the most crucial steps in mine development. It is very hard and almost impossible to achieve a reliable grade estimation, despite the method employed or data that has been used; but it is getting even more complex when the lack of proper data is the issue.

The estimations which were reported in this paper were done with data that doesn't have any geological and lithological information. However, the proposed method could reach more accurate results in comparison to other gold grade estimation researches. In addition to doing the estimations with whole data, the clustering area of study to three sections based on boreholes intervals proved that regardless of all complexities, it is possible to obtain more accurate results for each section. For even deeper investigation, the most compacted part in one of the clusters was used as the training data to assess the importance of the regular drilling. The results of this stage are very precise. It should be noted that the network which was trained with the ABC algorithm, outperformed other networks in all stages. So, the ABC algorithm is the most suitable in comparison to ICA, GA, and PSO, to train a neural network for grade estimation by using statistically sparse data.

One of the most important things that can be realized from this research is that regular drilling, even if it is done in low numbers of boreholes and even when the element of study has a very random nature, can play a major role in the preciseness of grade estimation; because it gives the operator the ability to do an accurate estimation in smaller parts; then and by integrating all estimations, it would be possible to gain to a good and relatively reliable estimation for the whole area.

Acknowledgements

The authors would like to express their gratitude to the Zarshouran gold mine industries and Kavoshgaran Consulting Engineers Company for their collaboration and support. Permitting us to use the data made this research possible.

REFERENCES

[1] Abraham, A., H. Guo, and H. Liu, Swarm Intelligence: Foundations, Perspectives, and Applications, in Swarm Intelligent Systems, N. Nedjah and L.d.M. Mourelle, Editors. 2006, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 3-25.

[2] Alimoradi, A., Maleki, B., Karimi, A., Sahafzadeh, M., Abbasi, S. Integrating Geophysical Attributes with New Cuckoo Search Machine-Learning Algorithm to Estimate Silver Grade Values– Case Study: Zarshouran Gold Mine, Journal of Mining and Environment, 2020, Vol. 11, No. 3, 865-879.

[3] Atashpaz-Gargari, E. and C. Lucas, Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition. Vol. 7. 2007. 4661-4667.

[4] Badel, M., S. Angorani, and M. Shariat Panahi, The application of median indicator kriging and neural network in modeling mixed population in an iron ore deposit. Computers & Geosciences, 2011. 37(4): p. 530-540.

[5] Bárdossy, G. and J. Fodor, Evaluation of Uncertainties and Risks in Geology. 2004, Springer Berlin Heidelberg.

[6] Chatterjee, S., Ore grade estimation of a limestone deposit in India using an Artificial Neural Network. 2006.

[7] Daliran, F., Agdarreh & Zarshuran SRHDG deposits, Takab region, NW-Iran. in Proceedings: GSA - Annual Meeting, Fall 2002.

[8] Daliran, F., Discovery of 1.2 kg/t gold and 1.9 kg/t silver in mud precipitates of a cold spring from the Takab geothermal field, NW Iran, in Mineral Exploration and sustainable development, Vol.1. Ed.: D.G. Eliopoulos. 2003, Millpress, Rotterdam. p. 461-464.

[9] Dutta, S., Machine Learning Algorithms and Their Application to Ore Reserve Estimation of Sparse and Imprecise Data. Vol. 2. 2010. 86-96.

[10] Eberhart, R. and J. Kennedy, A new optimizer using particle swarm theory, in MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science (2015). IEEE.

[11] Goldberg, D.E. and J.H. Holland, Genetic Algorithms and Machine Learning. Machine Learning, 1988. 3(2): p. 95-99.

[12] Gopalakrishnan, K., Particle Swarm Optimization in Civil Infrastructure Systems: State-of-the-Art Review, in Metaheuristic Applications in Structures and Infrastructures, A.H. Gandomi, et al., Editors. 2013, Elsevier: Oxford. p. 49-76.

[13] Haykin, S.S., Neural networks: a comprehensive foundation. 1999, Upper Saddle River, N.J.: Prentice-Hall.

[14] Jalloh, A.B., Integrating artificial neural networks and geostatistics for optimum 3D geological block modeling in mineral reserve estimation: A case study. International Journal of Mining Science and Technology, 2016. 26(4): p. 581-585.

[15] Journel, A.G. and C.J. Huijbregts, Mining geostatistics. 1978, New York: The Blackburn Press.

[16] Kapageridis, I., Application of artificial neural network systems to ore grade estimation from exploration data. 1999, University of Nottingham. p. 14-35.

[17] Kapageridis, I.K. and B.H. Denby. Ore Grade Estimation with Modular Neural Network Systems – A Case Study. 1999.

[18] Karaboga, D., An idea based on honey bee swarm for numerical optimization, Technical Report TR06, Erciyes University, Kayseri, Turkey. 2005.

[19] Karaboga, D., B. Akay, and C. Ozturk, Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks, in Modeling Decisions for Artificial Intelligence. 2007, Springer Berlin Heidelberg. p. 318-329.

[20] Koike, K., Neural Network-Based Estimation of Principal Metal Contents in the Hokuroku District, Northern Japan, for Exploring Kuroko-Type Deposits. Vol. 11. 2002, Natural Resources Research. 135-156.

[21] Li, X.-l., Adaptive ore grade estimation method for the mineral deposit evaluation. Mathematical and Computer Modelling, 2010. 52(11-12): p. 1947-1956.

[22] Maleki, S., H. Ramazi, and S. Moradi, Estimation of Iron Concentration by Using a Support Vector Machine and an Artificial Neural Network - the Case Study of the Choghart Deposit southeast of Yazd, Yazd, Iran. Geopersia, 2014. 4(2): p. 75-86.

[23] Edwards, R. Atkinson, K., Ore Deposit Geology and its Influence on Mineral Exploration, 1986, Springer

[24] Paar, W., Daliranite, PbHgAs2S6, a new sulphosalt from the Zarshouran Au-As deposit, Takab region, Iran. Mineralogical Magazine, 2009. Vol. 73(5): p. 871–881.

[25] Paravarzar, S., Correlation between geological units and mineralized zones using fractal modeling in Zarshuran gold deposit (NW Iran). Arabian Journal of Geosciences, 2014. 8: p. 3845-3854.

[26] Pyrcz, M. J., Gringarten, E., Frykman, P., Deutsch, C. V., Representative Input Parameters for Geostatistical Simulation, 2006, University of Alberta

[27] Strebelle, S., Conditional Simulation of Complex Geological Structures Using Multiple-Point Statistics. Mathematical Geology, 2002. 34(1): p. 1-21.

[28] Tahmasebi, P. and A. Hezarkhani, Application of a Modular Feedforward Neural Network for Grade Estimation. Natural Resources Research. Vol. 20. 2011. 25-32.

[29] Talbi, E.-G., Metaheuristics: From Design to Implementation. 2009: Wiley Publishing. 593.

[30] Samanta, B., Sparse Data Division Using Data Segmentation and Kohonen Network for Neural Network and Geostatistical Ore Grade Modeling in Nome Offshore Placer Deposit. Natural Resources Research, 2004. 13(3): p. 189-200.

[31] Samanta, B., R. Ganguli, and S. Bandopadhyay, Comparing the predictive performance of neural networks with ordinary kriging in a bauxite deposit. Mining Technology, 2005. 114(3): p. 129-139.

[32] Samanta, B., S. Bandopadhyay, and R. Ganguli, Comparative Evaluation of Neural Network Learning Algorithms for Ore Grade Estimation. Mathematical Geology, 2006. 38(2): p. 175-197.

[33] Samanta, B. and S. Bandopadhyay, Construction of a radial basis function network using an evolutionary algorithm for grade estimation in a placer gold deposit. Computers & Geosciences, 2009. 35(8): p. 1592-1602.

[34] Yang, X.-S., Optimization and Metaheuristic Algorithms in Engineering, in Metaheuristics in Water, Geotechnical and Transport Engineering, X.-S. Yang, et al., Editors. 2013, Elsevier: Oxford. p. 1-23.