Estimation of Iron concentration by using a support vector machineand an artificial neural network - the case study of the Choghart deposit southeast of Yazd, Yazd, Iran

Shahoo Maleki[1*], Hamid Reza Ramazia[2], Sirvan moradi[2]
[1] *Department of Mining and Metallurgy Engineering, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran*
[2] *Department of mining, Imam Khomeini International Qazvin University, Qazvin, Iran*
**Corresponding author, e-mail: Sh.maleki.ch@gmail.com*

(*received: 12/04/2014 ; accepted: 08/12/2014*)

Abstract

Estimation of the metal value in the metallic deposits is one of the important factors to evaluate the deposits in exploration studies and mineral processing. Therefore, one accurate estimator is essential to obtain a fine insight into the accumulation of the ore body. There are geostatistical methods for the estimation of the concentration of iron which performance of these models is complexity of analysis. The support vector machine (SVM) is by far one of the most robust artificial intelligence techniques used successfully for predictions and estimations of deposits because of its ability to generalize. Keeping this is view, the aim of this article is to use the SVM and back propagation neural networks (BPNN) to estimate the concentration of the iron element in the Choghartdeposit, in Iran. Comparing the obtained results with those of the validation process demonstrates that the SVM method is faster than the BPNN method and is more precise for the estimation of the iron concentration in the Choghartmine. The results of this study show that artificial intelligence–based models can evaluate the iron concentration with an acceptable accuracy.

Keywords: *Iron concentration, Ore deposit, Support Vector Machine (SVM), Back Propagation Neural Networks (BPNN).*

Introduction

Artificial neural network (ANN) techniques have been applied in the past to grade and resource estimation, with some success, for example Wu and Zhou (1993), Clarici *et al*. (1993) and Burnett (1995). ANNs require example inputs and outputs to discover this relationship. The input data normally comes in the form of samples at known locations in three-dimensional (3D) space and the output data is their grade at these locations. Most of the neural network systems treat the unknown parameter estimation as a problem of function approximation in the data coordinate space (Kapageridis & Denby, 1998; Kapageridis, 2005; Badel *et al*., 2010). The error-back propagation algorithm is the most popular learning algorithm (Rumelhart *et al*., 1986; Fahlman, 1988). Surprisingly, the most popular error-back propagation algorithmcannot handle more complex problems, while other more advanced algorithms (Hagan & M. Menhaj, 1994; Wilamowski *et al*., 2008) can. In recent times, the support vector machine (SVM) techniques (Cortes & Vapnik, 1995) have often been used to replace neural networks, because training is relatively easy, there is good generalization in theory and estimation, it is globally the best model, and there are no local optima, unlike in neural networks.

These are some of the advantages of the SVM. In recent years, artificial intelligence techniques have attracted the attention of many researchers. These networks rely on understanding the complexity of the input and output of a system, which can provide acceptable results. The support vector machine is one of the most powerful methods of artificial intelligence. In 1990, Vladimir Vepnayk has presented and proved their ability to anticipate problems of nonlinear systems (Behzad *et al*., 2009). The results show that this method has a high power for extending and dealing with noise and lack of data (Cristianini & Shawe-Taylor, 2000; Wang, 2005; Martinez-Ramon, Cristodoulou, 2006; Steinwart, 2008). In this study, the iron concentration values in the Choghartmine were studied by both the BPNN- and SVM-based models. It was observed that the SVM method was more precise and could evaluate the iron concentration with an acceptable accuracy.

Case study description

The Choghart deposit, and accumulation with a northwest–southwest orientation, is approximately 600m long and 200m wide. It can be seen on the ground surface and its thickness varies between 400 and 700m. A considerable diversity of various rocks like intrusive and metamorphous rocks pieces are seen around the deposit. The composition, which is formed by deposit host rocks has two completely different and distinct appearances: 1. Rocks with a high percentage of Quartz and Feldspar, named by geologists as Quartzite, Porphyry, Quartz, and Granophyres and 2. Rocks with a high percentage of Amphibolite, which are of Actinolite, Tremolite, and Feldspartypes (Albite), as well as, altered and amphibolitized alien rock pieces called Amphibolite, Amphiblol, Pyroxenite, Hornnblendite, and Metasmotite, with different compositions. About 30 m of dark lime stone with many Calcite-filled joints were found in the holes bored for exploration in the northeast of Choghart (Moor and Modabberi, 2003).

Back-propagation neural network

The goal of the Artificial Neural Network is to develop a mathematical model of biological events, to imitate the capability of biological neural structures, for the purpose of designing an intelligent information processing system. Recurrent neural networks (RNNs) have been an active research topic in recent years and they have been proposed as efficient techniques for implementing nonlinear adaptive filtering, due to their promising ability to model nonlinear dynamic systems (Narendra & Parthasarathy, 1990; Kolen, 2001). Numerous applications can be found in various disciplines (see Haykin, 1999 and Plett, 2003 for some examples). The back-propagation neural network (BPNN) is usually recognizedfor its prediction capability and ability to generalizewell on a wide variety of problems. Liang and Gupta studied the stability of the dynamic back propagation training algorithm, using the Lyapunov method (Liang and Gupta, 1999; Maleki *et al*., 2013).The back propagation model is a supervised type of network, in other words, trained with both inputs and target outputs. During training, the network tries to match the outputs with the desired target values. Learning starts with the assignment of random weights. The output is then calculated and the error is estimated. This error is used to update the weights until the stopping criterion is reached. It should be noted that the stopping criteria is usually the average error of the epoch.

Support vector machine

The SVM has been employed for regression estimation, the so-called support vector regression (SVR), in which the real value functions are estimated.

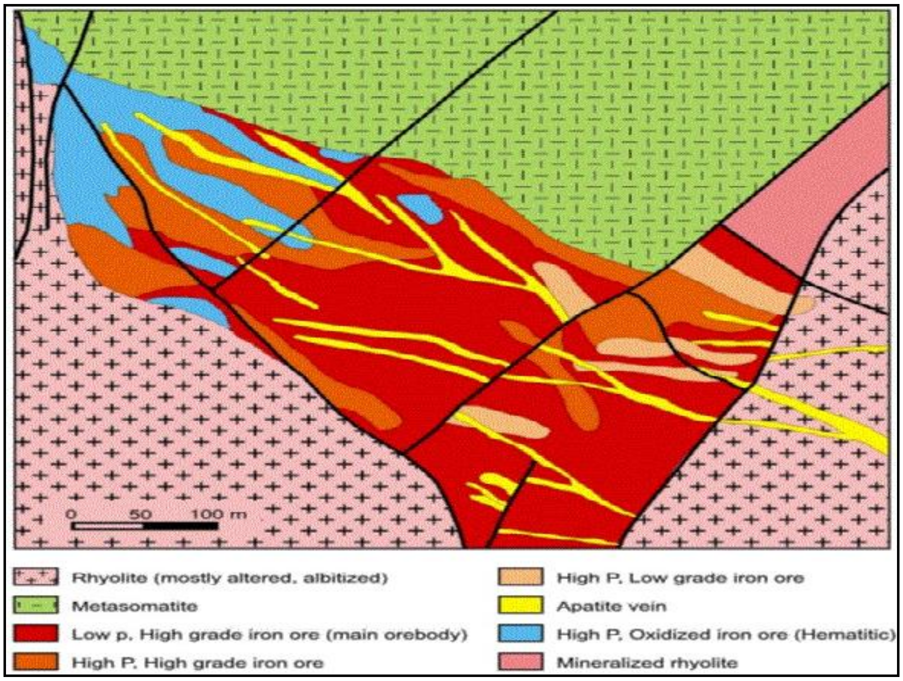
0 50 100 m
Rhyolite (mostly altered, albitized)
Metasomatite
Low p. High grade iron ore (main orebody)
High P. High grade iron ore
High P. Low grade iron ore
Apatite vein
High P. Oxidized iron ore (Hematitic)
Mineralized rhyolite

Figure 1. Geological map of the Choghartdeposit (Moor and Modabberi, 2003)

In this case, the aim of learning process is to find a function f(x), as an approximation of the value y(x), with minimum risk, and based only on the available independent and identically distributed data, that is, (Scholkopf *et al*., 1998; Maleki *et al*., 2013; Maleki *et al*., 2014a; Maleki *et al*., 2014b).

$$(x_1, y_1), \ldots, (x_m, y_m) \subseteq (X \subseteq R^n \times Y \subseteq R) \qquad (1)$$

In the SVR algorithm, the estimation function is determined by a small subset of training samples called the support vectors (SVs). Also, in this algorithm, a specific loss function called ε-insensitive loss is developed to create a sparseness property for SVR. This function is described as follows (Scholkopf *et al*., 1998; Maleki *et al*., 2013; Maleki *et al*., 2014a; Maleki *et al*., 2014b).

$$\left| y - f(x) \right|_{\pounds} = \begin{cases} 0 & \text{if } \left| y - f(x) \leq \pounds \right| \\ \left| y - f(x) \right| - \pounds & \text{Otherwise} \end{cases} \qquad (2)$$

where f(x), which is computed by the SVR, is the estimated value of y, and the corresponding errors being less than the ε-boundary (ε-tube) are not penalized (Fig. 2).

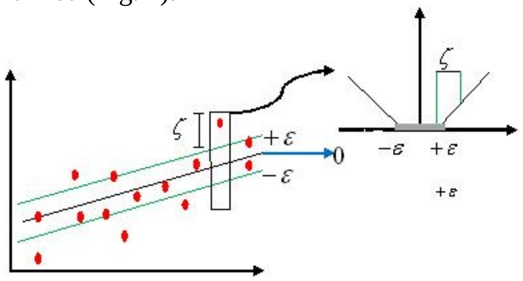

Figure 2. ε-Insensitive loss function (Liu *et al*., 2006)

For developing the regression algorithm, we begin with the linear function estimation. It is clear that every linear function of the input vector x has the following representation (Quang-Anh *et al*., 2005; Maleki *et al*., 2013; Maleki *et al*., 2014a; Maleki *et al*., 2014b).

$$f(x) = \langle w.x \rangle + b \,\text{Where}\, w, x \in X \subseteq R^n, b \in R \qquad (3)$$

Note that the angle bracket ($\langle . \rangle$) indicates the inner product of two vectors in the Hilbert space (i.e., a space in which the inner product of two vectors has a real value, also called inner (or dot) product space). With the ε-SVR, the aim is to find a function f(x) that estimates the values of the output variables, with deviations from the actual training data. The ε-values control the complexity of the approximating functions where small values tend to penalize a large portion of the training data, leading to tight approximating models, and large values tend to free data from penalization, leading to loose approximating models. Therefore, the proper choice of an ε-value is critical for the generalization of regression models.

For finding f(x), one should minimize the regulated risk functional (R_{reg}) (instead of just the empirical risk functional, which is used in traditional learning algorithms, such as, ANN), which is defined as follows (Al-Anazi and Gates, 2010; Maleki *et al*., 2014a; Maleki*et al*., 2014b).

$$R_{reg}[f] = \frac{1}{2}\|w\|^2 + C \,.\, R_{emp}^{\pounds}[f] \qquad \text{where } R_{emp}^{\pounds}[f] = \frac{1}{m}\sum_{i=1}^{m} \left| y_i - f(x_i) \right|_{\pounds} \qquad (4)$$

R_{emp} is the empirical error over training data, which is defined in the ε-insensitive loss function framework. The regularization coefficient *C* in Eq. (4) is an indicator of the complexity of function *f* and penalizes the error by setting the trade-off between training error minimization and model complexity. Briefly, the minimization of R_{reg} illustrates the principle idea of the structural risk minimization theory, which states that for achieving the minimum risk, the simultaneous control of the complexity of the model and the error owing to the training data is essential. This idea improves the generalization of the SVR.

It has been proven that minimizing Eq. (4) is equivalent to the following convex constrained quadratic optimization problem (Lia *et al*., 2007; Maleki *et al*., 2013; Maleki *et al*., 2014a; Maleki*et al*., 2014b).

$$L(w, \P, \P') = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\P_i + \P_i') \qquad (5)$$

$$Subject\ to \begin{cases} y_i - w^T.x - b \leq \P_i + \pounds \\ w^T.x + b - y_i \leq \P_i' + \pounds \\ \P_i, \P_i', x_i \geq 0 \end{cases}$$

where ξi and $\P_i'$ are slack variables, introduced to satisfy the constraints on the function. Therefore, SVR fits a function to the given data by not only minimizing the training error, but also by penalizing complex fitting functions. The first term of Eq. (5) is the Vapnik–Chervonenkis (VC) confidence interval, whereas, the second one is the empirical risk. Both terms limit the upper bound of the generalization error rather than limiting the training error. This means that SVR strikes a balance between the empirical error and VC-confidence interval, which leads to an improved generalization performance, better than the neural network models (Peng *et al*., 2004). In Eq. (5) C tries to ensure that the margin ε is maximized and the error of classification ξ is minimized. According to Eq. (5), any error smaller than ε does not require a nonzero ξi or $\P_i'$, and does not enter the objective function (Scholkopf *et al*., 1998; Maleki *et al*., 2013; Maleki *et al*., 2014a; Maleki *et al*., 2014b). By introducing Lagrange multipliers (α and α') and allowing $C > 0$, and $\varepsilon > 0$ chosen *apriori*, the equation of an optimum hyper plane is achieved by maximizing of the following relations.

$$L(\sim,\sim') = \frac{1}{2}\sum_{i=1}^{N}\sum_{i=1}^{N}(\sim_i - \sim_i')x_i'.x_i(\sim_i - \sim_i') + \sum_{i=1}^{N}((\sim_i - \sim_i')y_i - (\sim_i + \sim_i')\pounds) \qquad (6)$$

$$\mathit{Subject}\quad \mathit{to}\ \ 0 \le (\sim_i - \sim_i') \le C \qquad (7)$$

where x_i only appears inside an inner product. To get a potentially better representation of the data in a nonlinear case, the data points can be mapped into an alternative space, generally called feature space (a pre-Hilbert or inner product space) through a replacement.

$$x_i.x_j \rightarrow \ddot{}\,(x_i).\ddot{}\,(x_j) \qquad (8)$$

The functional form of the mapping $\varphi(x_i)$ does not need to be known, as it is implicitly defined by the choice of the kernel: $k(x_i,x_j) = \varphi(x_i).\varphi(x_j)$ or inner product in the Hilbert space. With a suitable choice of a kernel the data can become separable in the feature space, while the original input space is still nonlinear. Thus, while the problem of the data for n-parity or the two spirals is non-separable by a hyper plane, in the input space, it can be separated in the feature space by proper kernels (Walczack and Massart, 1996; Chih-Hung *et al*., 2009). Table 1 gives some of the common kernels.

Subsequently, the nonlinear regression estimate takes the following form (Maleki *et al*., 2013; Maleki *et al*., 2014a; Maleki *et al*., 2014b):

$$y_i = \sum_{i=1}^{N}\sum_{j=1}^{N}(\sim_i - \sim_i')\ddot{}\,(x_i)^T\,\ddot{}\,(x_j) + b = \sum_{i=1}^{N}\sum_{j=1}^{N}(\sim_i - \sim_i')K(x_i,x_j) + b \qquad (9)$$

where b is computed using the fact that the constraints of equation (5) become $\xi_i = 0$ if $0 < \alpha_i < C$, and $\xi_i' = 0$ if $0 < \alpha_i' < C$.

There are a quite a number of algorithms for SVM training, and Sequential Minimum Optimization (SMO) is an efficient one for this purpose (John, 1998). It is a simple algorithm that can quickly solve the SVM quadratic programming problem without any extra matrix storage and is exempt from using any numerical quadratic programming optimization steps. SMO decomposes the overall quadratic programming problem into sub-problems of quadratic programming by using the Osuna’s theorem to ensure convergence. There are two specific components in the structure of an SMO: An analytic method for solving the two Lagrange multipliers and a heuristic one for choosing multipliers in the optimization step (Khandelwal *et al*., 2010).

The advantage of an SMO lies in the fact that solving two Lagrange multipliers can be done analytically. Hence, numerical quadratic programming optimization can be avoided completely (Khandelwal *et al*., 2010). In addition, an SMO requires no extra matrix storage. Thus, very large SVM training problems can fit inside the memory of an ordinary personal computer or workstation. In this study, an SMO algorithm is used for both optimizing the structure of the SVM and helping to predict the permeability, in reasonable running time.

Dataset

For the present study, datasets that included iron concentrations were collected from four benches at the mine site (Figs. 3 to 6 show the location map of the data on the four benches). For each bench 10000 data points were gathered, and then the outlier and censored data were eliminated for all the benches (to show outlier and censored data we have used boxplots that are shown in Figs. 7a to 7d).The data used in this study is from 1000data samples for each bench, the other data set has outlier values.

Table 1. Polynomial, normalized polynomial, Radial Basis Function (Gaussian), and Pearson Universal (PUK) Kernels (Wang, 2005)

Kernel Function	**Type of Classifier**
$K(x_i,x_j)\,Z\,(x_i^{T}x_j\,H1)^{°}$	**Complete polynomial of degree °**
$K(x_i,x_j)\,Z\,\frac{(x_i^{T}x_j\;H\,1)^{°}}{\sqrt{(x_i^{T}x_j)\,J\,(y_i^{T}y_j)}}$	**Normalized polynomial kernel of degree °**
$K(x_i,x_j)=\exp\left[\left\Vert x_i-x_j\right\Vert^{2}/2±^{2}\right]$	**Gaussian (RBF) with parameters ± (sigma) control the half-width of the curve fitting peak**
$K(x_i,x_j)=\frac{1}{\left[1+\left(\frac{2\sqrt{\left\Vert x_i-x_j\right\Vert^{2}}\sqrt{2^{1/µ}-1}}{±}\right)^{2}\right]^{µ}}$	**Pearson VII Universal Kernel (PUK) with two parameters of ± (sigma) and ω (omega), which control the Pearson width and the tailing factor of the curve-fitting peak**

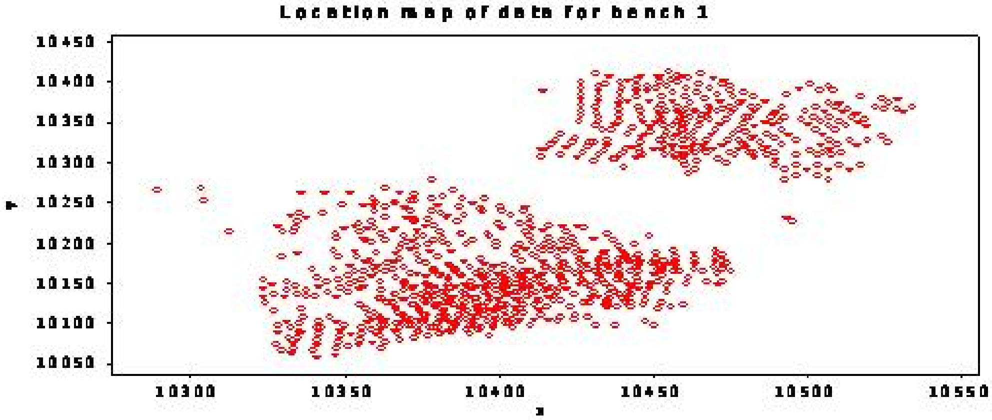

Figure 3. Shows location map of data for bench 1 in this study

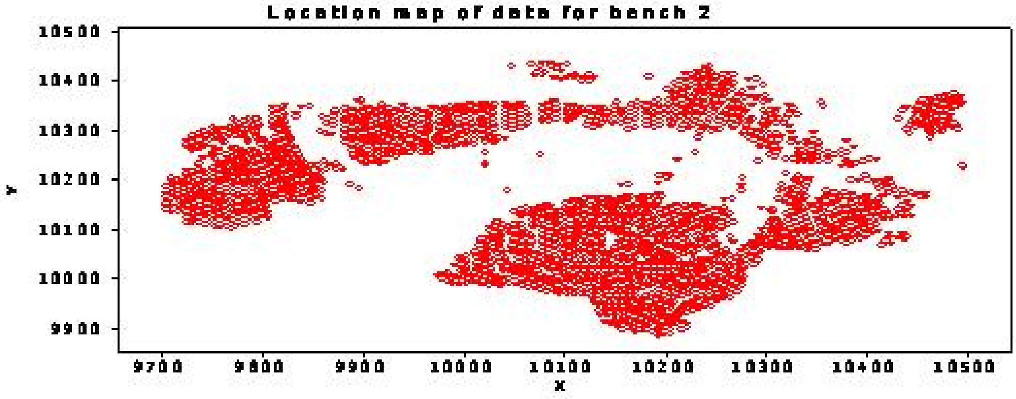

Figure 4. Shows location map of data for bench 2 in this study

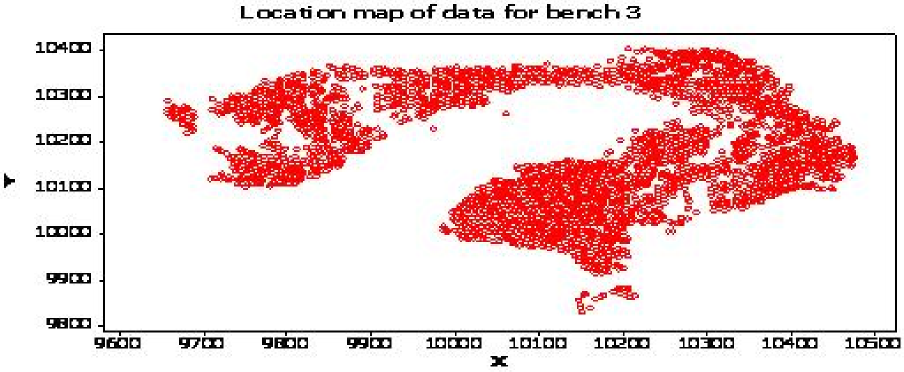

Figure 5. Shows location map of data for bench 3 in this study

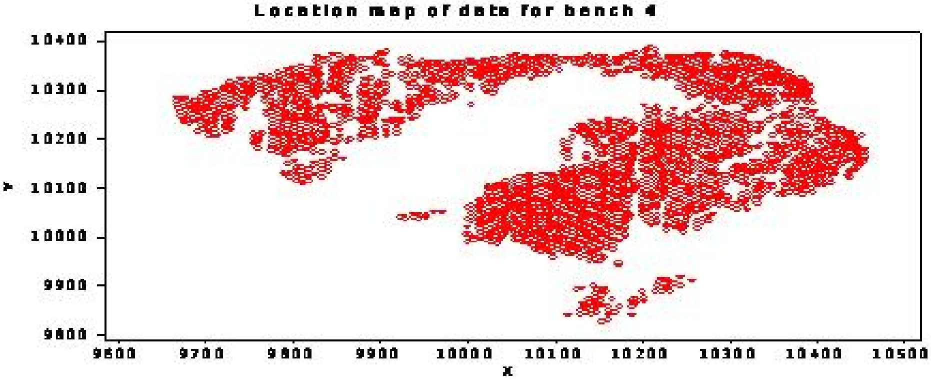

Figure 6. Shows location map of data for bench 4 in this study

Generally, datasets of the three benches were used to predict the iron concentration at the fourth bench. Therefore, the database of the first three benches was used for training in the SVM and the database of the fourth bench was used for testing, by using the MATLAB multipurpose software, in order to implement the automated Bayesian regularization. This type of regularization could significantly reduce a large amount of error called over-fitting. The very popular MATLAB Neural Network Toolbox (Matrix Laboratory (MATLAB) Neural Network Toolbox [Online]) was not able to handle the arbitrarily connected back-propagation neural network (BPNN) and SVM methods. Therefore, codes were prepared and implemented in the MATLAB, without using the Toolbox. To begin with, using these codes, three data points in the vicinity were selected, and finally they had been used in the estimation phase. In view of the requirements of the networks computation algorithms, the data of the input and output variables were normalized. In addition, for increasing the model strength for recognizing the relationship between the inputs and the corresponding outputs, it is necessary that the data be normalized (Boser, 1992; Amini *et al*., 2011). Therefore, the initial data were normalized using the Cox and Box method, as follows (Howarth and Earle, 1979):

$$z = \frac{x^{a} - 1}{a} (a = 0), x > 0 \qquad (10)$$

where X is the normalized value, λ is the actual value, and Z is the transformation value. Subsequently, the previous datasets were normalized between range (-1, 1) in the standard data, as (Govett, 1983; Amini *et al*., 2011):

$$X = 2 * \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} - 1 \qquad (11)$$

where X is the normalized value and X_i is the actual value in the columns, X_{max} and X_{min} are the maximum and minimum values of each related column (Figs. 8a to 8d shows the Histogram of raw data and the normalized data of the four benches).

In addition, the cross-validation of the whole training set was used for adjusting the associated parameters of the networks (Liu *et al*., 2009).

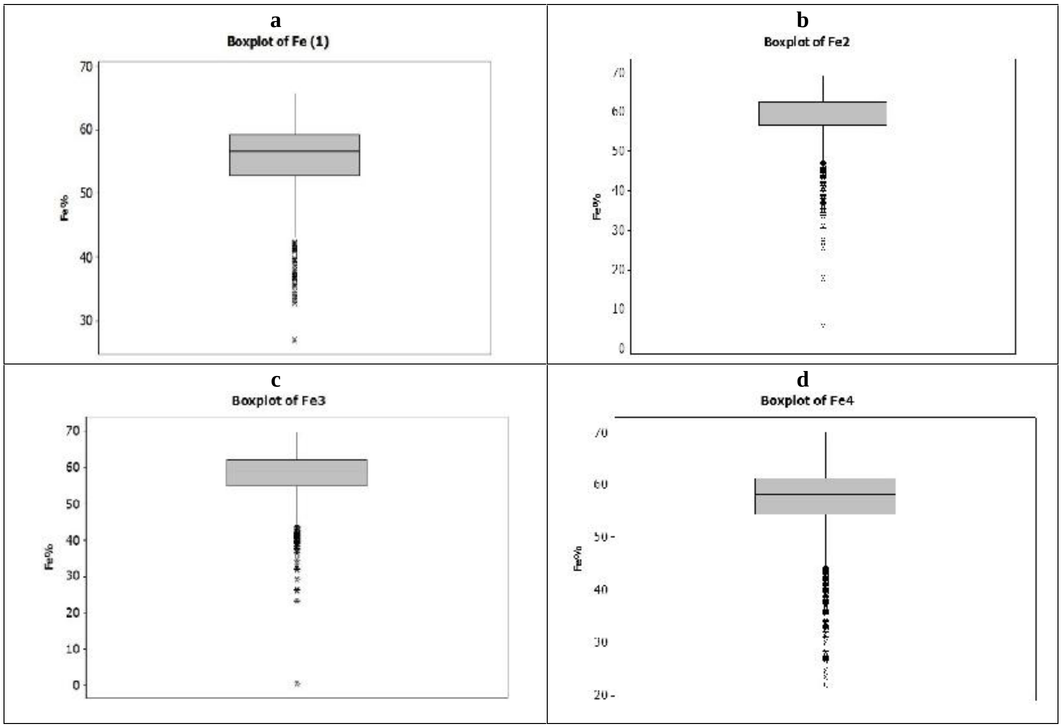

Figure 7. Shows the Boxplot of data the four benches in this study for identifying the outlier data. Figure 7a to 7d shows Boxplot bench 1 to 4, respectively

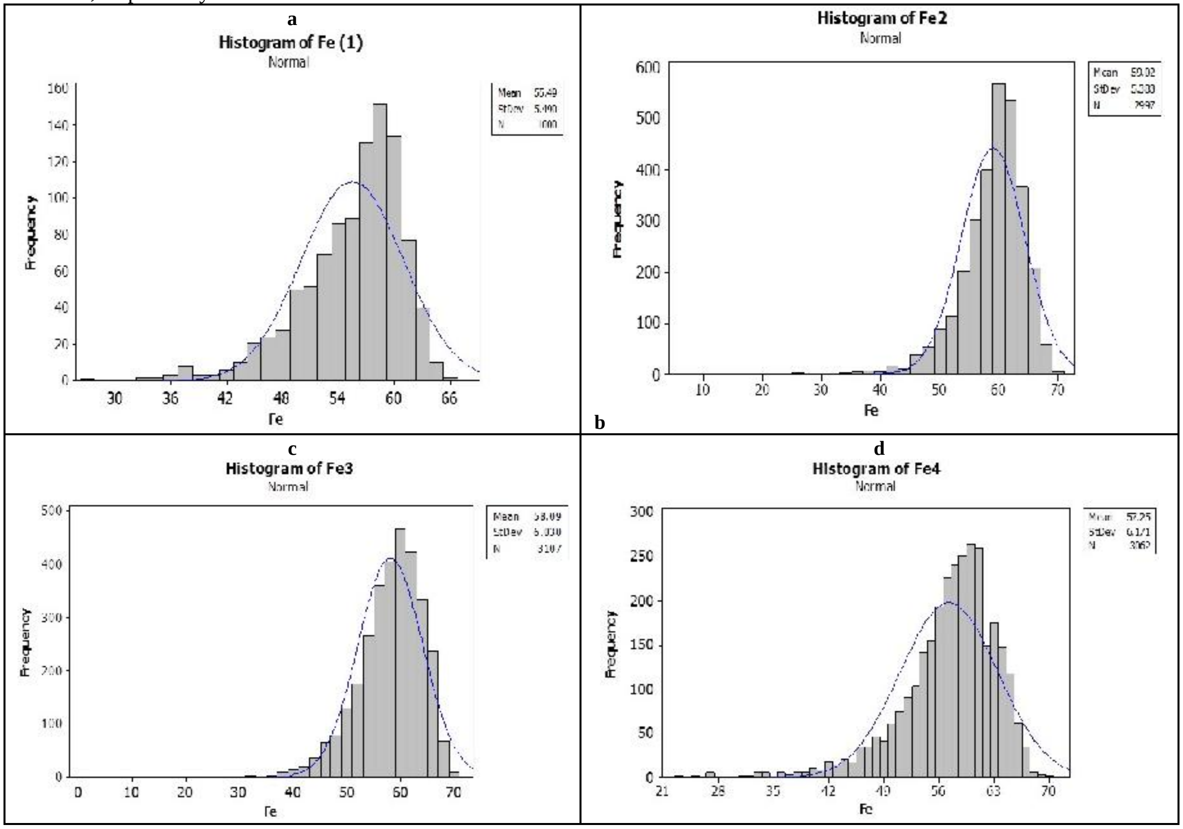

Figure 8. Frequency distribution chart and statistical parameters of Fe and histogram of the transferred data into a standard normal distribution among the benches of this study, Figs.8a to 8d shows a histogram of raw data and normalized data for benches 1 to 4, respectively.

Results

Estimation of iron concentration by the Support Vector Machine

As mentioned, the performance of the SVM depended mostly on the choice of the kernel function, which was in a sense equivalent to the choice of the BPNN structure. In this regard, despite the obtained results of the previous research studies (Dibike *et al*., 2001; Han and Cluckie, 2004) indicating the Gaussian radial basis function as a superior kernel, the form of the Gaussian kernel was as follows:

$$k(x_i, x_j) = e^{-\|x_i - x_j\|^2/2\pm^2} \qquad (12)$$

where ± is a constant parameter of the kernel, which can either control the amplitude of the Gaussian function or the generalization ability of SVM. We have to optimize ± and find the optimal one. Moreover, for implementation of the SVM, the appropriate values of the optimal parameter C (trade-off parameter), ± ¸ and ω (parameters of Pearson Universal kernel) need to be determined prior to building the model. For managing this issue, among all model selection tools, the cross-validation techniques can be rigorous for adjusting the associated parameters of SVM, because they make no biased assumptions about the data and noise distribution. The Leave One Out (LOO) is a cross-validation procedure consisting of removing one example from the training set, constructing the decision function only on the basis of the remaining training data, and then testing on the removed example (Liu *et al*., 2009).In this fashion one tests all examples of the training data and measures the fraction of errors over the total number of training examples. The root mean square error (RMSE) was used as an error function to evaluate the quality of the model. To obtain the optimal value of ± , the SVM with different ±*s* was trained, the ± varying from 0.05 to 0.22, every 0.01. At last, the optimal ± was found to be 0.13. To find an optimal£ and ω, the RMSEs on different£*s* and ωs were calculated. The optimal£ and ω were found to be 0.09 and 0.215, respectively. Figure 9 shows the LOO cross-validation step used for selecting the best values of ± ,£ ¸ and ω. Similarly, with other multivariate statistical models, the performances of SVM for regression depended on a combination of several parameters.

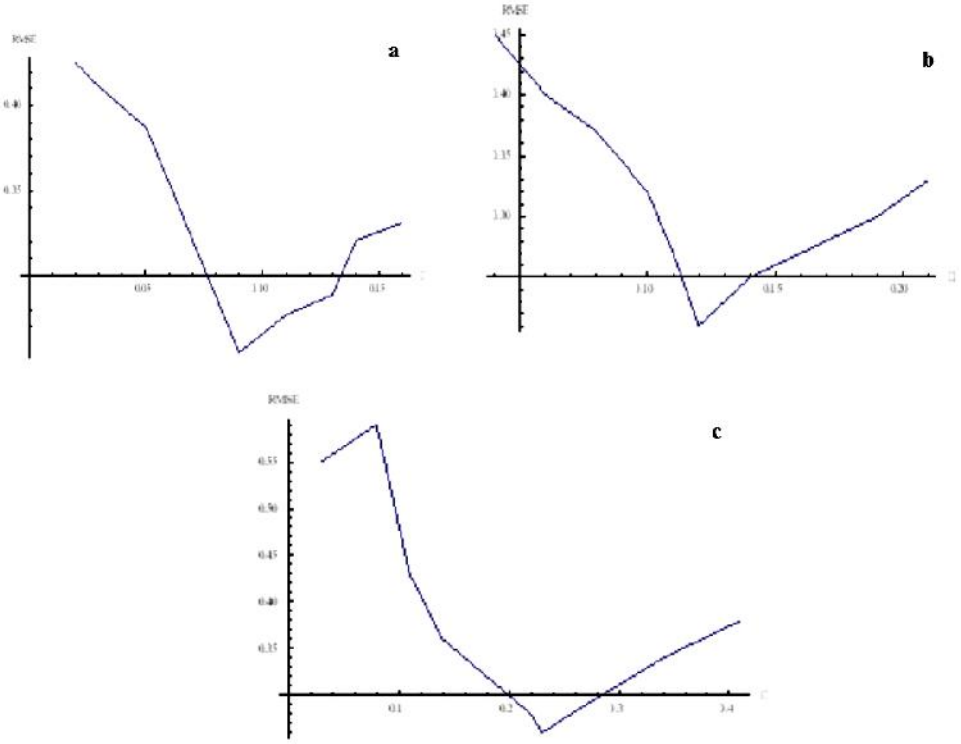

Figure 9. RMSE error versus ± (a), versus £ (b), and versus ω (c) in the LOO cross-validation step

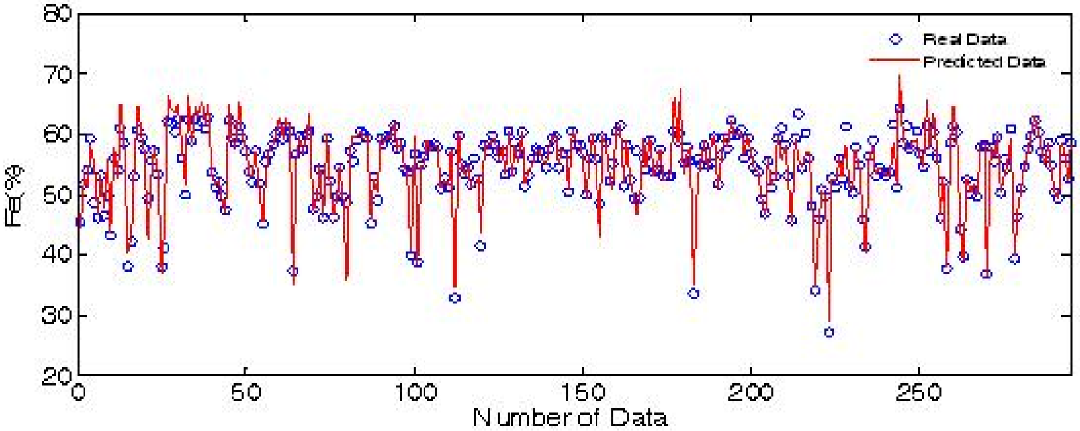

Figure 10. Performance of SVM in the estimation of the concentration of iron for the test data set

They are capacity parameter C, £ of the £ -insensitive loss function, and the kernel type K and its corresponding parameters. C is a regularization parameter that controls the trade-off between maximizing the margin and minimizing the training error. If C is too small then insufficient stress will be placed on fitting the training data. If C is too large then the algorithm will over-fit the training data. However, it was indicated that the prediction error was scarcely influenced by C (Wang *et al*., 2003; Maleki*et al*., 2014a; Maleki*et al*., 2014b). To make the learning process stable, a large value should be set up for C (e.g., C = 2100).

Figure 10 presents the estimation performance of SVM for concentration iron based on the test step.

Estimation of iron concentration by BPNN

To check the accuracy of SVM in the estimation of the concentration of iron, the results obtained by the SVM were compared with those of the back propagation neural network (BPNN). In this manner, to optimize the best network topology (e.g., number of hidden layers and corresponding neurons) a trial and error process was usually utilized. The structure of the BPNN model includes one input layer consisting of four neurons, two hidden layers of the8-4 neurons, and an output layer containing only one neuron (Fig. 11). Multiple layers of neurons with nonlinear transfer functions allowed the network to study nonlinear and linear relationships between the input and output vectors.

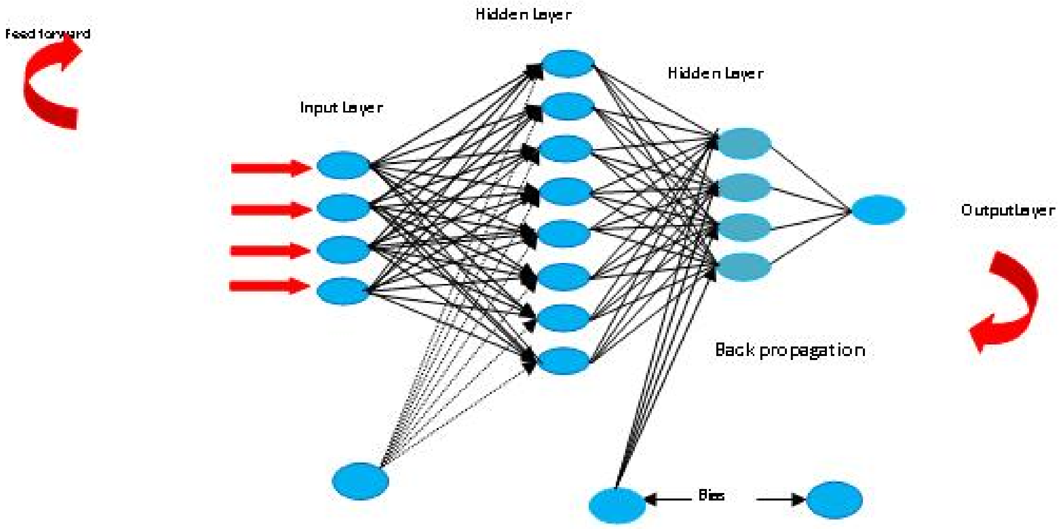

Figure 11. Block diagram of the optimum external BPNN Structure

The model performance for the estimation of the concentration of iron is done by calculating the mean square error (MSE) and coefficient correlation (R). The results, based on the normalized values obtained by the neural network code (m-file) in the MATLAB workspace, are demonstrated in Figure 12.

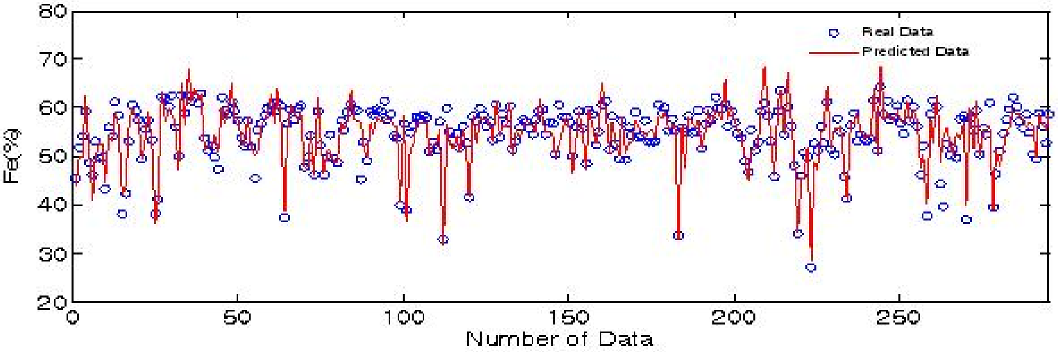

Figure 12. Performance of BPNN in the estimation of the concentration of iron for the test data set

Discussion

In this study, the performance of the SVM algorithm was demonstrated in the estimation of iron concentration. In this regard, two MATLAB software codes were developed and utilized for evaluating and analogy the performance of SVM with the best performed work of the BPNN model. When we compared the obtained results of SVM with those of the BPNN model, the SVM presented an overall better performance over the BPNN approach, in both the training and testing processes. Figure 13 shows the scatter plot estimation of the iron concentration of each method in the test data set.

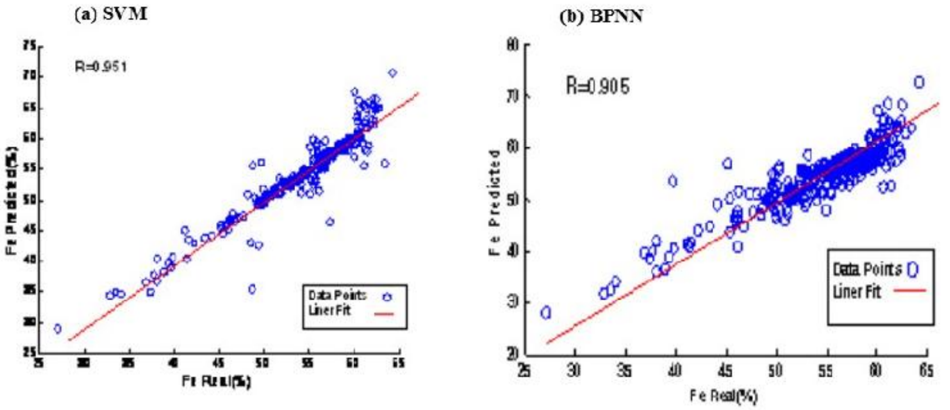

Figure 13. Relationship between the measured and estimated concentration of iron by the two artificial methodsconsidered for a test dataset

The plots indicate that an acceptable estimation (i.e., R=0.951) was obtained through SVM modeling. In addition, the SVM calculation had taken considerably less time for estimation compared to that of the BPNN. The performance parameters of both models are shown in Table 2. All of these expressions introduced the SVM as a suitable algorithm for the estimation of iron concentration.

Table 2. Performance parameters for evaluating the models

Model	**RMSE**	**R_{test}**
BPNN algorithm	0.455	0.905
SVM algorithm	0.235	0.951

Conclusion

In this study, we have shown the application of the SVM model compared to the BPNN model for the estimation of iron concentration of four benches at the Choghartmine of Iran, based on the borehole data. According to the results obtained at the end of the prediction, it seems that the SVM method (R=0.95 and RMSE=0.23) is a better and more accurate method for the estimation of the concentration of iron. However, the prediction performance of the BPNN method (R=0.9 and RMSE=0.45) is some what less than that of the SVM model and can not be considered as an alternative method for deposit mining characterization. Applying them ethodologies presented in this article, it appears that it will be possible to characterize deposit mining, using the available data point from the exploration borehole or extraction wells.

References

Al-Anazi, A.F., Gates, I.D., 2010. Support vector regression for porosity prediction in a heterogeneous reservoir: A comparative study, Computers & Geosciences, 36: 1494-1503.

Amini, H., Gholami, R., Monjezi, M., Torabi, R., Zadhesh, J., 2011. Evaluation of flyrock phenomenon due to blasting operation by support vector machine. Neural Comput & Applic, 11: 631-635.

Badel, M., Angorani, S., Shariat Panahi, M., 2010.The application of median indicator kriging and neural network in modeling mixed population in an iron ore deposit. Computers & Geosciences, 37: 530-540.

Behzad, M., Asghari, K., Morteza, E., Palhang, M., 2009.Generalization performance of support vector machines and neural networks in run off modeling. Elsevier, Expert Systems with Applications, 36: 7624-7629.

Boser, B.E., 1992. A training algorithm for optimal margin classifiers. In: Proceedings of the 5th annual workshop on computational learning theory, Pittsburgh, 5: 144-152.

Burnett, C., 1995. Application of neural networks to mineral reserve estimation. Ph.D. Dissertation, Department of Mineral Resources Engineering, University of Nottingham, Nottingham, 254 pp.

Chih-Hung, W., Gwo-Hshiung, T., Rong-Ho, L., 2009.A Novel hybrid genetic algorithm for kernel function and parameter optimization in support vector regression. Expert Systems with Applications, 36: 4725-4735.

Clarici, E., Owen, D., Durucan, S., Ravencroft, P., 1993.Recoverable reserve estimation using a neural network. In: Proceedings of the 24th International Symposium on the Application of Computers and Operations Research in the Minerals Industries (APCOM), Montreal, Quebec. 145-152

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn 20(3): 273-297.

Cristianini, N., Shawe-Taylor, J., 2000. An Introduction to Support Vector Machines (and other kernel-based learning methods). Cambridge University Press, UK.

Dibike, Y.B., Velickov, S., Solomatine, D., Abbott, M.B., 2001. Model Induction with support vector machines: Introduction and Application. Journal of Computing in Civil Engineering 15(3): 208-216.

Fahlman, S.E., 1988. Faster-learning variations on back-propagation: An empirical study. Connectionist Models Summer School, Sejnowski, T.J., Hinton, G.E., Touretzky, D.S. (Eds.), San Mateo, CA: Morgan Kaufmann.

Govett, G.J.S., 1983. Statistical Data Analysis in Geochemical Prospecting.Handbook of Exploration Geochemistry, Elsevior.

Hagan M.T., Menhaj, M., 1994. Training feedforward networks with the Marquardt algorithm. IEEE Trans. Neural Networks 5(6), 989-993.

Han, D., Cluckie, I., 2004. Support vector machines identification for runoff modeling. In Liong, S.Y., Phoon, K.K., Babovic, V. (Eds.), Proceedings of the sixth international conference on hydroinformatics 21-24.

Haykin, S., 1999.Neural Networks. Upper Saddle River, NJ: Prentice-Hall.

Howarth, R.J., Earle, S.A.M., 1979. Application of a Generalized Power Transformation to Geochemical Data. Mathematical Geology 11(1): 45-58.

John, C.P., 1998. Sequential Minimal Optimization: a Fast Algorithm for Training Support Vector Machines. MSRTR: Microsoft Research.

Kapageridis, I., 2005. Input space configuration effects in neural network-based grade estimation. Computers & Geosciences, 31: 704-717.

Kapageridis, I., Denby, B., 1998. Ore grade estimation with modular neural networkSystems-a casestudy.In:Panagiotou,G.(Ed.),ProceedingsoftheFirstInternational Conference on Information Technology in the Minerals Industry(MineIT ‘97). AA Balkema, Rotterdam CDROM Paper Code: KI16.

Khandelwal, M., Kankar, P.K., Harsha, S.P., 2010. Evaluation and prediction of blast induced ground vibration using support vector machine. Mining Science and Technology, 20: 64-70.

Kolen, J.F., 2001. A Field Guide to Dynamical Recurrent Networks. New York: IEEE Press.

Lia, Q., Licheng, J., Yingjuan, H., 2007. Adaptive simplification of solution for support vector machine, Pattern Recognition, 40: 972 -980.

Liang J., Gupta, M.M., 1999. Stable dynamic backpropagation learning in recurrent neural networks. IEEE Trans. Neural Netw 10 (6): 1321-1334.

Liu, H., Wen, S., Li, W., Xu, C., Hu, C., 2009. Study on Identification of Oil/Gas and Water Zones in Geological Logging Base on Support-Vector Machine. Fuzzy Information and Engineering 2 (62) 849-857.

Liu, H., Yao, X., Zhang, R., Liu, M., Hu, Z., Fan, B., 2006. The accurate QSPR models to predict the bioconcentration factors of nonionic organic compounds based on the heuristic method and support vector machine. Chemosphere 63: 722-733.

Maleki, Sh., Moradzadeh, A., Ghavami, R., Sadeghzadeh, F., 2013. A Robust Methodology for Prediction of DT Wireline Log. Iranian Journal of Earth Sciences, 5: 33-40.

Maleki, Sh., Moradzadeh, A., GhavamiRiabi, R., Sadaghzadeh, F., 2014a.Comparison of Several Different Methods of in situ stress determination, International JournalofRockMechanics&MiningSciences, 71: 395-404.

Maleki, Sh., Moradzadeh, A., GhavamiRiabi, R., Gholami, R., Sadaghzadeh, F., 2014b. Prediction of shear wave velocity using empirical correlations and artificial intelligence methods, NRIAG Journal of Astronomy and Geophysics, In Press, Corrected Proof, Available online 29 July 2014.

Martinez-Ramon, M., Cristodoulou, Ch., 2006. Support Vector Machines for Antenna Array Processing and Electromagnetic. Universidad Carlos III de Madrid, Spain, Morgan & Claypool, USA.

MATLAB Neural Network Toolbox [Online]. Available: www.mathworks.com; products; neural net.

Moor, F., Modabberi,S., 2003. Origin of Choghart iron oxide deposit, Bafq mining district, CentralIran: new isotopic and geochemical evidence, Journal of science, Islamic Republic of Iran, 14(3): 259-269.

Narendra K., Parthasarathy, K., 1990. Identification and control of dynamical systems using neural networks. IEEE Trans. Neural Netw, 1: 4-27.

Peng, K.L., Wu, C.H., Goo, Y.J., 2004. The development of a new statistical technique for relating financial information to stock market returns. International Journal of Management 21(4): 492-505.

Plett, G.L., 2003. Adaptive inverse control of linear and nonlinear systems using dynamic neural networks. IEEE Trans. Neural Netw 14 (3): 360-376.

Quang-Anh, T., Xing, L., Haixin, D., 2005.Efficient performance estimate for one-class support vector machine. Pattern Recognition Letters, 26: 1174-1182.

Rumelhart, D.E., Hinton, G.E., Wiliams, R.J., 1986. Learning representations by back-propagating errors. Nature 323, 533-536.

Scholkopf, B., Smola, A.J., Muller, K.R., 1998. Nonlinear component analysis as a kernel eigenvalues problem, Neural Computet, 10: 1299-1319.

Steinwart, I., 2008. Support Vector Machines. Los Alamos National Laboratory, information Sciences Group (CCS-3), Springer.

Walczack, B., Massart, D.L., 1996. The radial basis functions-partial least squares approach as a flexible non-linear regression technique. Anal. Chimical Acta, 331: 177-185.

Wang, L., 2005. Support Vector Machines: Theory and Applications, Nanyang Technological University, School of Electrical & Electronic Engineering, Springer Berlin Heidelberg New York.

Wang, W.J., Xu, Z.B., Lu, W.Z., Zhang, X.Y., 2003. Determination of the spread parameter in the Gaussian kernel for classification and regression. Neurocomputing, 55: 643-663.

Wilamowski, B.M., 2007. Neural networks and fuzzy systems for nonlinear applications. Int. Conf. Intelligent Engineering Systems, Budapest, Hungary, 1: 13-19.

Wilamowski, B.M., Cotton, N.J., Kaynak, O., Dundar, G., 2008. Computing gradient vector and Jacobian matrix in arbitrarily connected neural networks. IEEE Trans. Ind. Electron 55(10): 3784-3790.

Wu, X., Zhou, Y., 1993.Reserve estimation using neural network techniques. Computers & Geosciences, 19: 567-575.