Bulletin of Environment, Pharmacology and Life Sciences
Bull. Env. Pharmacol. Life Sci., Vol 2 (9) August 2013: 30-38

Online ISSN 2277-1808
Journal's URL:http://www.bepls.com

ORIGINAL ARTICLE

The Study of Artificial Neural Network (ANN) Efficiency with Neuro-Fuzzy Inference System (ANFIS) in Dissolved oxygen Simulation of River Water

Mojtaba Ahour[1], AlirezaNouri[1], Mohammad Sadegh Sadeghian[1]
[1]Islamic Azad University Central Tehran Branch, Tehran,Iran

ABSTRACT
Several simulation methods were used for simulating water quality status of river. An adaptive neuro fuzzy inference system (ANFIS), artificial neural networks (ANN) were developed to simulate concentration dissolved oxygen (DO) for twelve months at Khoram-Abad River station in Khoram-Abad Province-Iran. This paper will present a comprehensive case study based on key input parameters such as river flow, reach distance, velocity of stream, dissolved oxygen of soluble water at the last n station (DO^{-n}), water temperature, width of water surface, depth of flow which is collected at above mentioned station. The purpose of this paper is to study and illustrate the several errors correlation coefficient (R^2), root mean square error (RMSE),normalized mean square errors (NMSE) and mean absolute percentage error (MAPE) based upon ANFIS and ANN results, ultimately the results of two models analyzed and were compared.
Keywords: *adaptive neuro fuzzy inference system (ANFIS); artificial neural networks (ANN); dissolved oxygen; river water quality*

Received 12.05.2013 Accepted 29.06.2013

A REVIEW OF POST STUDY

In the middle of 1970s , river quality modelling was expanded by rapid expansion of measuring instruments and computers and development of numerical methods in solving differential equations [1].during two lost decades, varied modellings of considered types of process, the number of simulation features, being one or multi-dimension, dominated partial numerical solve in methods, transmission mechanism and analysis of uncertainty have been developed and they are used in different areas of the world. There are some limitation on calibration of coefficient reactions and their accuracy. It decreases in the anticipation of quality behavior of rivers, especially, when the numbers of features and their mutual effects on one another increases.

In order to calibration and accuracy of river quality modelling, many researches have been carried out from which we can indicate the works of Yih and Davison [2] in determining longitudinal dispersion coefficient, Mulligan and /brown in using Genetic Algorithm in calibration of river quality modeling [3] Van Griensven [4] in calibration sensitivityanalysis of river quality modellings in a dynamic state. Maier and Dandy carried out a study to estimate the degree of salinity in mary River using the artificial neural network, concluded that neural network modelling and back propagation network were exact instruments in estimating of lowering quality degree in this river. The difference between the observed and estimated value lies between 46 and 53 Umhos/cm . they suggested the comparison between this model and other mathematical and physical models[5]. Sandhu and Finch Also emphasized on the ability of artificial neural network for daily estimating and the real degree of salinity of the cation, anion concentration degree,ES and TDs value in this basin[6]. Ha and StenstormAchieved the determining index 92.3-94.5 percent and very low root mean square errors RMSE about 0.154-0.157 when they studied 173 cases of land use by studying their water quality using a neural network including 10 input variables water quality, four hidden layers and 5 output variables of land use[7]. In a research in ebgland .Walley and Fontana Investigated water quality of Britain River by artificial neural networks using back-propagation algorithm and achieved the accuracy of 95.62 percent.[8]. Detailed information on concentrations of N and P in river water in response to non-point source pollution (NPS) is fundamental for researchers, planners and government agencies if effective strategies to reduce the negative impacts are to be found and adopted. A modelling approach is an efficient way to investigate river water quality status and patterns at different spatial and temporal scales. To date, mechanism-based models, such as

soil and water assessment tool (SWAT) [9], agricultural non-point source pollution modelling system (AGNPS)[10], and hydrological simulation program Fortran model (HSPF)[11] , have been developed for NPS simulation at a watershed scale. These models have been widely applied to simulate NPS pollution in agricultural watersheds with distributed or lumped water quality simulation[12,13,14] . However, these mechanism-based models require so many data and parameters for calibration that their application is rarely successful to sites and times other than those they were developed within [15,16] . ANNs are a very powerful computational technique for modelling complex non-linear relationships, particularly in situations where the explicit form of the relation between the variables involved is unknown [17]. ANNs have been successfully used for water quality modelling, for example: for predicting physical properties [18,19,20] , nutrients [21,16,22] and aquatic biological indices in rivers [23,24].

INSTRUMENTS AND METHODS

Artificial neural network

Back-propagation algorithm neural network. In order to map relationships between biological, hydrological and environmental variables, a BPNN was used as a non-linear predictor [25] (Maier and Dandy, 2000). The BPNN is the most popular type of neural network and is more frequently used than other types for water resources applications [26], although other types of network have been developed more recently that are clearly superior with respect to convergence, robustness and efficiency. The BPNN used for this study is a supervised iterative learning algorithm. It is designed to minimize the mean square error between the computed output of the network and the desired output. It requires input vectors in the input layer, as well as target values in the output layer corresponding to each input vector. The learning algorithm of the BPNN is very popular and common, and detailed descriptions of the learning rules can be found in, for instance, [27]. After the learning process, a dataset that has not been used in the training process is applied to validate the reliability of the trained BPNN.

Preliminary experiments demonstrated that input data pre-processing was crucial for training success. The use of raw input data leads to slow training and frequent convergence at very bad local optima. However, when each of the input fields was standardized to have a mean of 0 and a standard deviation of 1, training was very fast and bad local optima were never encountered. Output data was scaled to lie between 0 and 1 so as to be compatible with the sigmoid transfer functions used in BPNN designs. In order for the training to proceed, network weights must be initialized to small, random values. In this application, network weights were initialized to random values lying between −0.1and 0.1. The random number generator was reseeded with the current time prior to each training run. Weight initialization at the start of the modelling was a random process and was found to have a large influence on the results. To allow for this variation, 50 replicates were used for every network topology in each training run. The batch gradient decent with a momentum algorithm [28] (Demuth and Beale, 2000) was used as the training function of a BPNN can significantly influence its capacity for generalization and performance. To select the network of the best size, two different approaches are pursued. The first starts with a small initial network and gradually added new inputs or hidden units; thus, growing the model to optimum size. Other examples of this method are as follows: cascade-correlation leaning architecture [29]; the upstart algorithm [30] and the tiling algorithm [31]. The second, referred to as pruning, starts with a large network and removes unnecessary weights and/or units to achieve a satisfactory network size. Well-known examples of this method are as follows: optimal brain damage [32]; optimal brain surgeon [33]; interactive pruning [34] and skeletonization [35]. These two approaches require prior judgement of what size of initial network would be appropriate to grow or prune for the problem at hand. This is especially so for pruning approaches where training time required increases with larger networks. In this study, a method derived from these two approaches was used to determine the BPNN structure, which started with a suitable network for both pruning and growing[36]. First, initial input variables for the BPNN were selected using correlation analysis and quadratic polynomial stepwise regression. Correlation analyses were carried out between target output variables of the BPNN and other available data to identify those variables that were potentially directly related to the target outputs. Then quadratic polynomial stepwise regression analyses were carried out for the target output variables to reveal the effects of combined independent variables on the target outputs. Initial input variables for the BPNN were selected as the intersection of the results from both correlation and quadratic polynomial stepwise regression [36]. Second, the initial numbers of hidden layers and hidden nodes were both selected according to prior research. It has been shown that ANNs with one hidden layer can approximate any function, given that sufficient degrees of freedom (i.e. connection weights) are provided [37]. Robert [38] has also proved that a three-layer BPNN can be satisfied with any map from n-dimension to m-dimension. In this study, one hiddenlayer was selected. The function of the hidden layer nodes is to detect relationships between

network inputs and outputs. The initial number of hidden nodes (Nh) was selected using a method suggested by Zhang [39]:

$N_h = \sqrt{n \times m}$ (1)

where n is the number of inputs and m is the number of outputs.

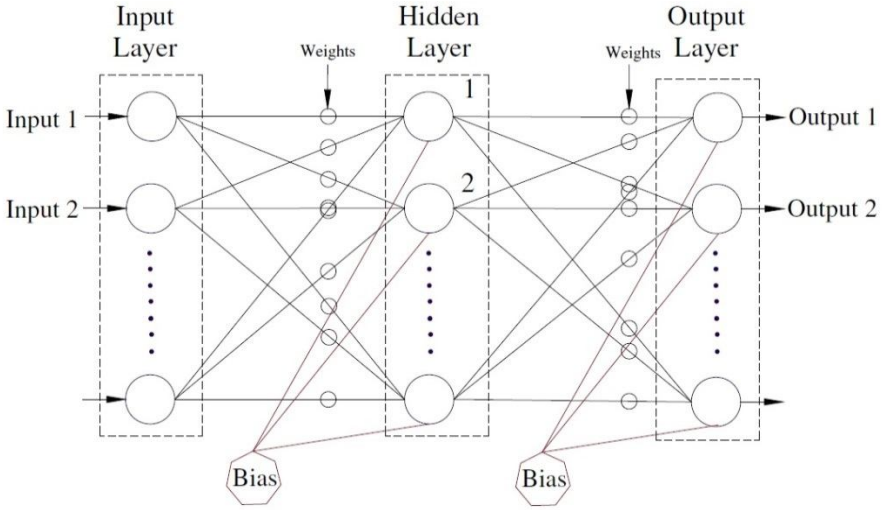

Figure 1 Structure of ANN

2-2 Adaptive neuro-fuzzy inference system (ANFIS)

Adaptive neuro-fuzzy inference system (ANFIS), first introduced by Jang [40], is a universal approximator, which can represent highly nonlinear functions more powerfully than conventional statistical methods. ANFIS is functionally equivalent to a fuzzy interference system consisting of a set of rules. It can construct an input–output mapping based on both the fuzzy if–then rules and the stipulated input–output data pairs. ANFIS employs the neural network training procedure to adjust the membership function and the associated parameters. To present the ANFIS architecture, two fuzzy if–then rules based on a first-order Sugeno model are considered:

Rule 1 : If x is A1 and y is B1; then f 1 = p1x + q1y r1; (2)

Rule 2 : If x is A2 and y is B2; then f 2 = p2x + q2y r2; (3)

where x and y are the inputs, Ai and Bi are the fuzzy sets, fi are the outputs within the fuzzy region specified by the fuzzy rule, pi, qi and ri are the design parameters which are determined during the training process. Generally, the ANFIS model consists of five layers configured analogously to any multi-layer feed-forward neural network (Figure. 2). The functionality of nodes in ANFIS, as a five-layered feed forward neural structure can be summarized as follows [40]:

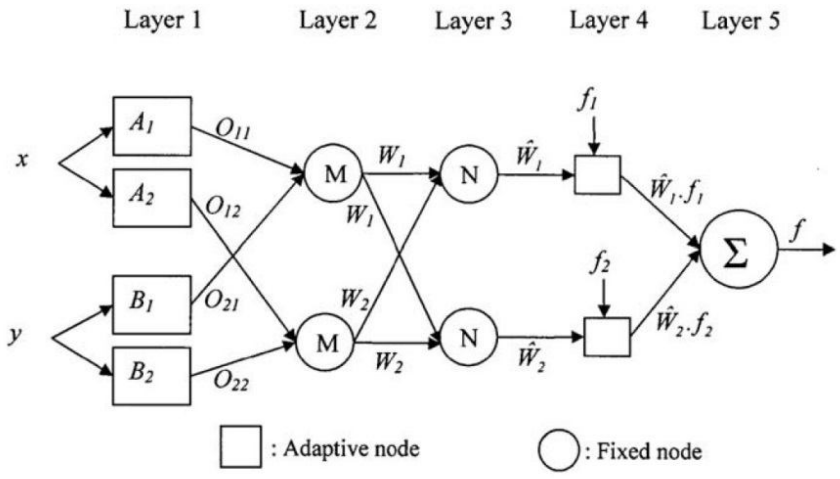

Figure 2 Structure of ANFIS network

Layer 1: Nodes are adaptive; membership functions (MFs) of input variables are used as node functions:

$Q_{1i} = \mu_{A_i}(x)$, i = 1,2, (4)

$Q_{2i} = \mu_{B_i}(y)\,, i = 1,2,$ (5)

where$\mu_{A_i}(x)$ and $\mu_{B_i}(y)$ can adopt any fuzzy membership function. For example, if the bell shaped membership function is employed $\mu_{A_i}(x)$is given by

$$\mu_{A_i}(x) = \frac{1}{1 + \{(\frac{x - c_i}{a_i})^2\}^{b_i}}$$

where a_i, b_i and c_i are the parameters of the membership function, governing the bell shaped functions accordingly.

Layer 2: Nodes are fixed and labeled with M, indicating that they perform as a simple multiplier. The outputs of this layer can be represented as

$W_i = \mu_{A_i}(x).\mu_{B_i}(y)\,, \quad i = 1,2,$

Layer 3: Nodes are fixed are labeled with N, indicating that they play a normalization role:

$\widehat{W}_i = \frac{W_i}{W_1 + W_2}\,, \quad i = 1,2.$

Layer 4: Nodes are adaptive with node function given by Layer 1 for a first-order model, and with parameters referred to as defuzzifier of consequent parameters.

Layer 5: The single node is fixed with output equal to the sum of all the rules' outputs.

The aim of learning algorithm for this architecture is to tune all the modifiable parameters, namely {ai, bi, ci} and {pi, qi, ri}, to make the ANFIS output match the training data. A hybrid algorithm combining the least squares method and the gradient descent method is usually adopted to solve this training problem. The least squares method is used to optimize the consequent parameters {pi, qi, ri} with the premise parameters {ai, bi, ci} fixed. The output of the ANFIS is calculated by employing the consequent parameters found in the previous step. The output error is used to adapt the premise parameters by means of a standard error backpropagationalgorithm[41].

2-3 Normalized of the parameters

The existence of the different units will result in the erroneous contribution of the parameters, especially those with large value. This article conducts the normalization with the following equation[42]

$PN = \frac{2\times(P - \min P)}{(\max P - \min P)} - 1$ (6)

In this equation, P and PN represent the input data before and after the change, separately, max p and min p the maximum value and the minimum value that the premnmx function gets.

Training networks

Considering the obtained results of multi-variable regression, four models were designed based on table 1 in which DO^{-n}dissolved oxygen of soluble water at the last n.station, $\bar{x}$ the distance between two stations, Q river flow, V velocity of river stream , W the width of water surface , d depth and The temperature of river water at the under investigation station.

Four mentioned models were trained by means of artificial neural network and neuro-fuzzy network. The utilized network for training neural network models is perceptron type and training type after error dispersion beyond BP. It was considered two rules for a series of input data in neural fuzzy networks. It is also used various transmission functions available in MATLAB toolbox. For training of neuro-fuzzy networks and among which the best transmission function with lower error was applied as the transmission function of mentioned network. Hybrid method aided us in ANFIS network training. For each model from total available data, 70% was used for training networks, 15% for testing and the last 15% for validation. For the purpose of studying and appropriate evaluating of network efficiency, proportions were chosen randomly and training process carried out 100 times. For either model and the average of 100 times reported as that model error.

Number Of Models	Type Of Network	Structure
1	ANFIS	$DO^{-1}, DO^{-2}, \bar{x}, Q, V$
2		$DO^{-1}, DO^{-2}, \bar{x}, Q, V, W, d$
3		$DO^{-1}, DO^{-2}, \bar{x}, Q, V, T$
4		$DO^{-1}, DO^{-2}, \bar{x}, Q, V, W, d, T$
5	ANN	$DO^{-1}, DO^{-2}, \bar{x}, Q, V$
6		$DO^{-1}, DO^{-2}, \bar{x}, Q, V, W, d$
7		$DO^{-1}, DO^{-2}, \bar{x}, Q, V, T$
8		$DO^{-1}, DO^{-2}, \bar{x}, Q, V, W, d, T$

Table 1: Structure of models

MODEL VERFICATION

It was used different standards to investigate the performance of proposed models and comparsion of models. To be more precise, four standards were applied. Between the observed and estimated value: R^2 correlation coefficient , RMSE root mean square errors,NMSE normalized mean square errors and MAPE mean absolute percentage Error, as you see below

$$R^2 = \frac{\sum_{i=1}^{n}(x_i-\bar{x}).(\hat{x}_i-\bar{\hat{x}})}{(\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(\hat{x}_i-\bar{\hat{x}})^2)^{\frac{1}{2}}} \quad (7)$$

$$RMSE = [\frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2]^{\frac{1}{2}} \quad (8)$$

$$NMSE = \frac{\sum_{i=1}^{n}(x_i-\hat{x}_i)^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2} \quad (9)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|x_i-\hat{x}_i|}{x_i} \quad (10)$$

In this equationx_i is the observative value , $\hat{x}_i$ the estimated value and n is the number of data.

Since presented models did not show the same procedure in confrontation with standard error, to reach a total conclusion in proportion to the analoy of each model performance , two techniques were applied. The first technique is the idea of colouring for each model with different standard errors. In this tech nique ,put different colours on models for mentioned standards. Eventually the sum of all colours in a certain model results the major rank for every one of four standards in comparison to other models.

The second technique is normalization of all determined standards .By this technique ,different standards , wich were desirable in different ranges, change into a value between and so that lower value represents lower error and desirability in simulation.

$$Np = \alpha - \frac{(maxp-p)}{(maxp-minp)} \quad (11)$$

In this equation p and NP represent standard error and normalized standard error ,respectively and α is a multiplier. Its value for R^2 is 0 and for other standards is 1. Notice that maximum and minimum standard error are determined separately.

1- Case study

Represented models from different structures of ANN and ANFIS network with quantitative and qualitative data were trained and investigated in Khoramabad River.

Khoramabad river is formed from joining two rivers. Robat and karganeh ,in the center of the twon. North basin of khorramabadwichkio park lies in this area ,is situated in the range of 48 12 to 48 and 45 eastern longitude and 33 28 to 33 44 latitude (figure 3) .Applied quantitative and qualitative data in training and trial models are inomontbly monitoring that totally it equals 36 months. In table 2, it has been shown a report if the applied maximum,minimum ,average and standard derivation data.

Table 2: Maximum,minimum ,average and standard derivation data

	Min	**Max**	**Average**	**Std. Div**
Q(m^3/s)	6.34	10.24	8.14	1.28
V(m/s)	1.92	3.96	3.21	0.54
W(m)	1.34	3.36	2.13	0.44
D(m)	0.98	1.3	1.15	0.08
T(c°)				
DO(mg/lit)	4.1	11	7.35	1.60

Figure 3:Investigated reach of lorestan province, khorramabad and khorramabad river

RESULTS AND DISCUSSION

Represented models in table 1 has been trained with quantitative and qualitative data. The errors result from training and trail represented for each model in table 3and figure 4.

Table3: Summary of R^2,RMSE,NMSE and MAPE

		R2		RMSE		NSME		MAPE	
		Train	Test	Train	Test	Train	test	train	Test
ANFIS	1	0.794	0.781	1.422	1.755	0.254	0.331	0.055	0.062
	2	0.926	0.914	1.309	1.652	0.236	0.342	0.060	0.060
	3	0.843	0.830	2.535	2.728	0.436	0.516	0.077	0.085
	4	0.881	0.869	1.241	1.596	0.290	0.394	0.047	0.054
ANN	5	0.860	0.848	1.480	1.783	0.272	0.369	0.074	0.082
	6	0.821	0.809	1.260	1.595	0.481	0.602	0.050	0.058
	7	0.781	0.769	2.597	2.895	0.360	0.468	0.082	0.090
	8	0.904	0.892	1.376	1.708	0.247	0.344	0.058	0.058

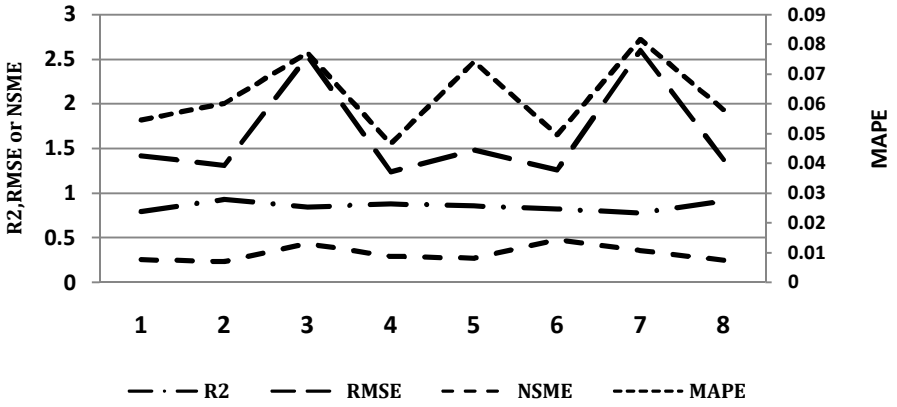

Figure 4:Graph corresponding to table 3

As it is shown in table 2 Different standard errors relevant to a model did not confirm one another absolutely. Number 2 , for instance , showed the best error for$R^2$and NSME standard error , but it was not the best rather than MAPE and RMSE standard error. Two presented techniques in the previous part have been used to remove this problem and to compare the performance of models together.In table 4 , every eight models of ANN and ANFIS networks has been compared and devoted lower rank with the better performance.so , different ranks were placed on each standard error for all of eight models. Eventually , the sum of ranks gained from each standard error of each model , the total rank of the model determined.Using this ranking technique brought an obvious distinction among the performance of models so that comparison of the eight models became easier. but there was a problems in this procedure that some models appeared with the same rank. For example, the sum of ranks in two models (1 and 6) was the same (=18). As a result , both of these models placed on the fourth total rank.

		ANFIS				ANN			
		1	2	3	4	5	6	7	8
R2	**train**	0.794	0.926	0.843	0.881	0.860	0.821	0.781	0.904
	test	0.781	0.914	0.830	0.869	0.848	0.809	0.769	0.892
	Rank	7	1	5	3	4	6	8	2
RMSE	**train**	1.422	1.309	2.535	1.241	1.480	1.260	2.597	1.376
	test	1.755	1.652	2.728	1.596	1.783	1.595	2.895	1.708
	Rank	5	3	7	1	6	2	8	4
NSME	**train**	0.254	0.236	0.436	0.290	0.272	0.481	0.360	0.247
	test	0.331	0.342	0.516	0.394	0.369	0.602	0.468	0.344
	Rank	3	1	7	5	4	8	6	2
MAPE	**train**	0.055	0.060	0.077	0.047	0.074	0.050	0.082	0.058
	test	0.062	0.060	0.085	0.054	0.082	0.058	0.090	0.058
	Rank	3	5	7	1	6	2	8	4
Sum Of Rank		18	10	26	10	20	18	30	12
RSR		4	1	7	1	6	4	8	3

Table 4: Ranking ofR^2,RMSE ,NMSE and MAPE for train and test

To eliminate this problem each of standard errors was replaced with relative error using equation 6.in this way different standards in different ranges changed between 0 to 1 so that the performance of a superior model approximates to 0 and for an inappropriate model approximates to 1 for every standard error. It is also possible sum up relative standards of each model by using this method and considers their mean value as the mean relative error of individual model. It has been shown different standards of relative error for an individual model in table 5. it has also drawn the corresponding diagram to the mentioned table for each training and experiment errors in figures5 and 6 , respectively.

		R2		RMSE		NSME		MAPE		train	test	Total
		train	test	Train	test	train	test	train	Test			
ANFIS	1	0.912	0.914	0.133	0.123	0.074	0.000	0.229	0.229	0.337	0.316	0.327
	2	0.000	0.000	0.050	0.044	0.000	0.038	0.389	0.173	0.110	0.064	0.087
	3	0.575	0.575	0.955	0.871	0.819	0.682	0.870	0.852	0.805	0.745	0.775
	4	0.311	0.314	0.000	0.000	0.222	0.230	0.000	0.000	0.133	0.136	0.135
ANN	5	0.456	0.456	0.177	0.145	0.145	0.138	0.788	0.783	0.391	0.381	0.386
	6	0.725	0.725	0.015	0.000	1.000	1.000	0.086	0.103	0.457	0.457	0.457
	7	1.000	1.000	1.000	1.000	0.506	0.506	1.000	1.000	0.876	0.877	0.876
	8	0.154	0.155	0.100	0.087	0.046	0.048	0.328	0.113	0.157	0.101	0.129

Table 5:Normalize error in table 3

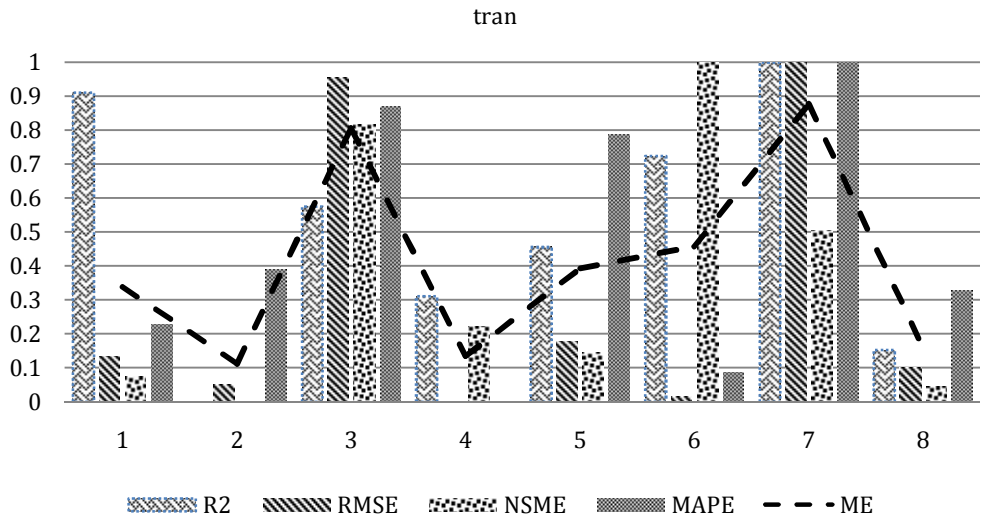

Figure 5:Graph corresponding to table5 for the training

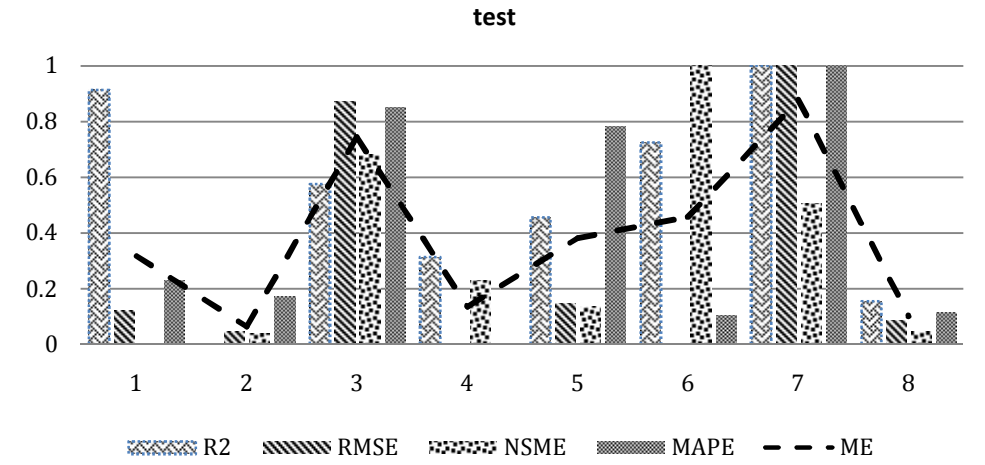

Figure 6:Graph corresponding to table5 for the test

Analogy of introduced models by relative mean error of each model, these results were inferred :

On the analogy of each model of a network with the same model of another network (Analogy between model 1-4 with model 5-8), neuro-fuzzy network showed an absolute superiority to the similar neural models. In models related to neuro-fuzzy network, model no.2 showed the best efficiency, then model no.4 showed the best one relative mean error for the process of training and trial of model 2 and 4 is 0.087 and 0.135, respectively. The structural difference between network 4 with network 4 is the temperature. Apparently, by inserting temperature to model 2, the efficiency of the model is decreased and after that model 1 and 3 were taken lower mean error the difference between model 1 and 3 is also in the temperature. As a result, temperature in neuro-fuzzy network made disturbance in training process.In neural networks, model 8 showed the best performance. Then model 5, 6, 7 were the superior models, respectively. The performance of neural models showed that having lower efficiency than neuro-fuzzy model, however, in comparison of network no.6 to 8, in the presence of temperature, the efficiency of the network in relation to correspondent neural network in the absence of temperature not only it does not have lower efficiency, but also its performance has been improved. But, in comparison between model no.5 to 7, the model in which there is no temperature, showed better performance.

REFERENCES

1. Little,K.W., and R.E.,Williams. 1975. Least-squares calibration of QUAL2E. Water Environment Research, 64(2), 179-185.
2. Yih,S.M., B.,Davidson. 1975. Identification in nonlinear, distributed water quality models. Water Resources Research, 11 (5), 693-704.
3. Mulligan,A.E., and L.C.,Brown. 1998. Genetic algorithms for calibrating water quality models. Journal of Environmental Engineering, 124 (3), 202-211.
4. Van Griensven,A., A.,Francos, and W.,Bauwens. 2002. Sensitivity analysis and auto calibration of an integral dynamic model for river water quality. Water Science and Technology, 43(7), 321-328.
5. Maier, H. R., and Dandy, G. C. (1996).“The use of artificial neural networks for the prediction of water quality parameters.” Water Resources Research, 32(4), 1013-1022.
6. Sandhu, N., and Finch, R. (1995). Methodology for flow and salinity estimation in the Sacramento-San Joaquin Delta and Suisun Marsh, Chapter 7: Artificial neural networks and their applications, 16th Annual Progress Report, New York, 85.
7. Ha, H., and Stenstorm, M. K. (2003). “Identification of land use with water quality data in storm water using a neural network.”Water Research, 37(3), 4222-4230.
8. Walley, W. J., and Fontana, V. N. (1998). “Neural network prediction of average scope per taxon and number of families at unpolluted river sites in Great Britain.” Water Rersources Research, 32(3), 613-622.
9. Arnold JG, Srinivasan R, Muttiah RS, Williams JR. 1998. Large area hydrologic modeling and assessment part I: model development. Journal of American Water Resources Association 34: 73–89.
10. Young RA, Onstad CA, Bosch DD, Anderson WP. 1989. AGNPS: A nonpoint source pollution model for evaluating agricultural watersheds. Journal of Soil and Water Conservation 44: 168–173.
11. Donigian AS, Imhoff JC, Bicknell BR, Kittle JL. 1984. Application Guide for the Hydrological Simulation Program FORTRAN . GA7 Environmental Research Laboratory, US Environmental Protection Agency: Athens.
12. Fitzhugh TW, Mackay DS. 2001. Impact of subwatershed partitioning on modeled source- and transport-limited sediment yields in an agricultural nonpoint source pollution model. Journal of Soil and Water Conservation 56: 137–143.
13. Luo B, Li JB, Huang GH, Li HL. 2006. A simulation-based interval two-stage stochastic model for agricultural nonpoint source pollution control through land retirement. Science of the Total Environment 361: 38–56.
14. Easton ZM, Fuka DR, Walter MT, Cowan DM, Schneiderman EM, Steenhuis TS. 2008. Re-conceptualizing the soil and water assessment tool (SWAT) model to predict runoff from variable source areas. Journal of Hydrology 348: 279–291.
15. Borah DK, Bera M. 2004. Watershed-scale hydrologic and nonpointsource pollution models: reviews of application. Transactions of the American Society of Agricultural Engineers 47: 789–803.
16. Maria H, Martin F, Michael S, Markus H. 2006. An application of artificial neural networks to carbon, nitrogen and phosphorus concentrations in three boreal streams and impacts of climate change. Ecological Modelling 95: 51–60.
17. Smith M. 1993. Neural Networks for Statistical Modeling. Van Nostrand Reinhold: New York.
18. Huang WR, Simon F. 2002. Neural network modeling of salinity variation in Apalachicola River. Water Research 6: 356–362.
19. Alp M, Cigizoglu HK. 2007. Suspended sediment load simulation by two artificial neural network methods using hydrometeorological data. Environmental Modelling and Software 22: 2–13.
20. Chenard JF, Daniel C. 2008. Stream temperature modelling using artificial neural networks: application on Catamaran Brook, New Brunswick, Canada. Hydrological Processes 22: 3361–3372. DOI: 10.1002/hyp.1468.
21. Lek S, Guiresse M, Giraduel JL. 1999. Predicting stream nitrogen concentration from watershed features using neural networks. Water Research 33: 3469–3478.
22. Mohamed HN, Daniel WS, Mohamed GED, Ellie EP. 2006. The application of artificial neural networks to flow and phosphorus dynamics in small streams on the Boreal Plain, with emphasis on the role of wetlands. Ecological Modelling 191: 19–32.

23. Park YS, R´egis C, Arthur C, Lek S. 2003. Applications of artificial neural networks for patterning and predicting aquatic insect species richness in running waters. Ecological Modelling 160: 265–280.
24. Jeong KS, Kim DK, Joo GJ. 2006. River phytoplankton prediction model by Artificial Neural Network: Model performance and selection of input variables to predict time-series phytoplankton proliferations in a regulated river system. Ecological Informatics 1: 235–245.
25. Maier HR, Dandy GC. 2000. Neural networks for the prediction and forecasting of water resources variables: a review of modeling issues and applications. Environmental Modelling and Software 15: 101–124.
26. Daniel BM, Muttucumaru S. 2009. Prediction of urban stormwater quality using artificial neural networks.Environmental Modelling and Software. 24: 296–302.
27. Rumelhart DE, Hinton GE, Williams RJ. 1986. Learning internal representations by error propagation. In Parallel Distributed Processing: Explorations in the Microstructure of Cognition 1 ,Rumelhart DE, McCelland JL (eds). Foundations, MIT Press: Cambridge.
28. Demuth H, Beale M. 2000. Neural Network Toolbox User's Guide for Use with MATLAB Version 4.0 . The MathWorks: Natick.
29. Fahlman SE, Lebiere C. 1990. The cascade-correlation architecture. In Advances in Neural Information Processing Systems (Denver, 1989), vol. 2, Touretsky DS (ed.) Morgan Kaufmann: San Mateo; 524–532.
30. Frean M. 1990. The upstart algorithm: a method for constructing and training feedforward neural networks. Neural Computation 2: 198–209.
31. M´ezard M, Nadal JP. 1989. Learning in feedforward layered networks: The tiling algorithm. Journal of Physics A 22: 2191–2203.
32. Cun LY, Denke JS, Solla SA. 1990. Optimal brain damage. In Advances in Neural Information Processing Systems (Denver, 1989), vol. 2, Touretzky DS (ed.) Morgan Kaufmann: San Mateo; 598–605.
33. Hassibi B, Stork DG. 1993. Optimal brain surgeon. In Advances in Neural Information Processing Systems (Denver, 1992), vol. 5, Hanson SJ, Cowan JD, Giles CL (eds). Morgan Kaufmann: San Mateo; 164–171.
34. Sietsma J, Dow RJF. 1991. Creating artificial neural networks that generalize. Neural Networks 4: 67–79.
35. Mozer MC, Smolensky P. 1989. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In Advances in Neural Information Processing Systems (Denver, 1988),vol. 1, Touretzky DS (ed.) Morgan Kaufmann: San Mateo; 107–115.
36. Dingjiang Chen,, Jun Lu and YenaShen " Artificial neural network modelling of concentrations of nitrogen, phosphorus and dissolved oxygen in a non-point source polluted river in Zhejiang Province, southeast China" HYDROLOGICAL PROCESSES Hydrol. Process. 24, 290–299 (2010)
37. Hornik K, Stinchcombe M, White H. 1989. Multilayer feed forward networks are universal approximators. Neural Networks 2: 359–366.
38. Robert HN. 1989. Theory of the back-propagation neural network. Proceedings of International Joint Conference on Neural Networks. IEEE Press: Washington, USA; vol. 1, 593–605.
39. Zhang LM. 1993. The Model of Artificial Neural Network and its Application (in Chinese). Fudan University Press: Shanghai, China.
40. Jang, J. S. R. (1993). ANFIS: Adaptive-network-based fuzzy inference system. IEEE Transactions on Systems Man and Cybernetics, 23, 665–685.
41. FoadSoltani , Reza Kerachian, EhsanShirangi " Developing operating rules for reservoirs considering the water quality issues: Application of ANFIS-based surrogate models" Expert Systems with Applications 37 (2010) 6639–6645
42. (Zhou Kaili, Kang Yaohong. Neural Network Model and MATLAB Emulation Programming[M]. Beijing: Tsinghua University Press, 2005.)

How to cite this article

Mojtaba Ahour, AlirezaNouri, Mohammad Sadegh Sadeghian.The Study of Artificial Neural Network (ANN) Efficiency with Neuro-Fuzzy Inference System (ANFIS) in Dissolved oxygen Simulation of River Water. Bull. Env. Pharmacol. Life Sci., Vol 2 (9) August 2013: 30-38