Caspian J. Env. Sci. 2008, Vol. 6 No.1 pp. 53~58

CJES
Caspian Journal of Environmental Sciences

[Research]

Rainfall-runoff modelling using artificial neural networks (ANNs): modelling and understanding

A. M. Kalteh[1, 2]

1- Dept. of Forestry, Faculty of Natural Resources, University of Guilan. O.Box 1144, Somehsara, Iran.
2- Dept. of Water resource Engineering, Lund University, Sweden.
E-mail: kalteh@guilan.ac.ir

ABSTRACT
In recent years, artificial neural networks (ANNs) have become one of the most promising tools in order to model complex hydrological processes such as the rainfall-runoff process. In many studies, ANNs have demonstrated superior results compared to alternative methods. ANNs are able to map underlying relationship between input and output data without prior understanding of the process under investigation. However, they have been known as black-box models due to their problem in providing insight into the relationship learned. In this study, firstly we develop a rainfall-runoff model using an ANN approach, and secondly we describe different approaches including Neural Interpretation Diagram, Garson's algorithm, and randomization approach to understand the relationship learned by the ANN model. The results indicate that ANNs are promising tools not only in accurate modelling of complex processes but also in providing insight from the learned relationship, which would assist the modeller in understanding of the process under investigation as well as in evaluation of the model.

Keywords: Rainfall-runoff modeling, Artificial neural networks, Neural interpretation diagram, Garson's algorithm, Randomization approach.

INTRODUCTION

In recent years, artificial neural networks (ANNs), a system theoretic/black-box method, have been used for modelling complex hydrological processes, such as rainfall-runoff (e.g. Hsu *et al*., 1995; Lorrai & Sechi, 1995; Minns & Hall, 1996; Dawson & Wilby, 1998; Tokar & Johnson, 1999; Rajurkar *et al*., 2002; Wilby *et al*., 2003; Giustolisi & Laucelli, 2005; Jain & Srinivasulu, 2006) and shown to be one of the most promising tools in hydrology (ASCE Task Committee, 2000a, b; Maier & Dandy, 2000; Dawson & Wilby, 2001). ANN models are built upon the input and output observations without the detailed understanding of the complex physical laws governing the process under investigation and are able to provide reasonably accurate model for the process under investigation, as a great number of the applications in hydrology along with the comparison of their predictive performance with other methods in many studies have demonstrated. However, they have been mostly criticized for their black-box nature due to the fact that the primary application of an ANN is the nonlinear modelling of input output observations in order to obtain accurate modelling of system's response without gaining any understanding of the mechanisms learned by the network. In other words, the trained network is not able to provide any explanation regarding how the model was built; hence, there is no way to evaluate the model obtained. According to the ASCE Task Committee (2000b), for ANNs to expand their acceptability it is very impo-rtant that they provide some explanation after training has been completed. In order to fulfill this, by using an ANN model developed to describe the rainfall-runoff relationship in a watershed in northern Iran, we use three approaches including the Neural Interpretation Diagram, Garson's algorithm, and randomization approach to obtain some explanation on trained ANN about the modelled relationships of the process under investigation. This paper starts with a brief description of the study area and data and then gives the details of the method used for modelling the rainfall-runoff process and methods for interpretation of the ANN model before discussing the results and drawing the conclusions.

STUDY AREA AND DATA

In this study, the monthly precipitation totals, average runoff and temperature database derived from a watershed located in northern Iran were used in order to develop an ANN rainfall-runoff model. The stations of 13001 (54 4 E, 36 38 N), 13004 (53 40 E, 36 37 N), 13005 (53 54 E, 36 35 N), 13007 (54 44 E, 36 36 N), 13009 (53 36 E, 36 35 N) and 13013 (53 19 E, 36 38 N) are located in the watershed so that the station 13013 with drainage area 1962 km^2 is located in the downstream end of the watershed under study. Five rainfall variables measured at 13001, 13004, 13005, 13007, and 13009 along with average runoff and temperature variables, measured at 13005 were used as input variables in order to model runoff at downstream station 13013. The time series span the period from 1982-83 to 1996-97. Out of this available data, the first three years were reserved as validation period. Moreover, in order to reflect the seasonality in the watershed, time variables represented by a sine, and a cosine curve was also used as the inputs to develop the model. All time series were standardized prior to use by subtracting the mean and dividing by the standard deviation.

METHODS

Feed-forward multilayer perceptron (MLP)

The feed-forward multilayer perceptron (MLP) is the most commonly used ANN in hydrological applications. The structure of a three-layer MLP is shown in Fig. 1. It consists of three layers; an input layer, a hidden layer, and an output layer. The number of neurons in the input and output layers are defined based on the number of input and output variables of the system under investigation respectively. However, the number of neurons in the hidden layer(s), in this study a single hidden layer with six neurons, is usually defined via a trial-and-error procedure. As seen from the figure, the neurons of each layer are connected to the neurons of the next layer by weights. In order to obtain optimal values of these connection weights, ANNs must be trained. In this study, we used a back-propagation algorithm for training the network, in which the inputs are presented to the network and the outputs obtained from the network are compared with the real output values (target values) of the system under investigation in order to compute error and then the computed error is back-propagated through the network and the connection weights are updated. This procedure, called training procedure, continues until an acceptable level of convergence is reached. In this study, in order to avoid instability, the neural network was trained 20 times, and by averaging the output from all a final output was obtained. Details about ANN structures, training algorithms and applications in hydrology are thoroughly covered by ASCE Task Committee (2000a, b), Maier & Dandy (2000) and Dawson & Wilby (2001).

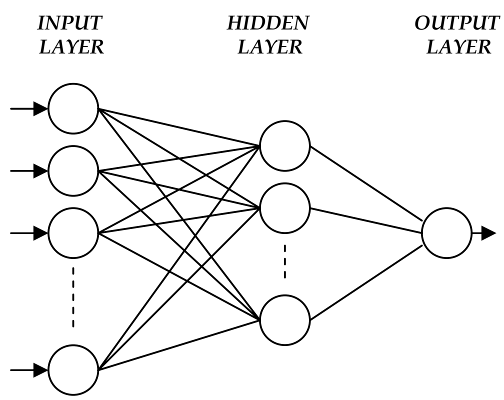

Fig 1. A structure of a three-layer feed-forward multilayer perceptron (MLP).

The results of the ANN model applied in this study were evaluated by means of:

1. Root mean square error (*RMSE*)

$$RMSE = \left[\frac{\sum_{i=1}^{n} (Q(i) - \hat{Q}(i))^2}{n} \right]^{0.5}$$

2. Correlation coefficient (r)

$$r = \frac{\sum_{i=1}^{n} (Q(i) - \overline{Q})(\hat{Q}(i) - \tilde{Q})}{\sqrt{\sum_{i=1}^{n} (Q(i) - \overline{Q})^2 \sum_{i=1}^{n} (\hat{Q}(i) - \tilde{Q})^2}}$$

where $\hat{Q}(i)$ is the n estimated runoff value, $Q(i)$ is the n observed runoff value, $\overline{Q}$ is the mean of the observed runoff values, and $\widetilde{Q}$ is the mean of the estimated runoff values.

Neural interpretation methods

Neural interpretation diagram (NID)

The neural interpretation diagram (NID) was introduced by Özesmi and Özesmi (1999) in order to interprete the connection weights of a trained ANN model based on the visualisation of their magnitude and direc-
tion, which are represented by line thickness and shading respectively. In this study, the NID was used for investigation of the rainfall-runoff relationship in which the magnitude and direction of the relationship are represented based on the line thickness and state (dotted or solid). As the thickness is scaled to the value of the weight of the connection (magnitude), the solid connection weights are used to represent the connections that are positive (excitor), and the dotted connection weights are used to represent the connections that are negative (inhibitor). Consequently, the visual inspection of the magnitude and direction of the connection weights of a trained ANN model will help hydrologists to identify the individual and interacting effects of the input variables on the output variables of the modelled system.

According to Fig. 1, there are two types of connection weights, one representing weights for the input-hidden layer connection and the other representing the hidden-output layer connection weights; hence, the rainfall-runoff relationship or the relationship between input and output variables is determined by the input-hidden and hidden-output layer connection weights as follows.The positive effects of input variables are achieved by the same signs of the input-hidden and hidden-output connection weights (both positive or both negative) and the negative (inhibitory) effects of input variables are achieved by the opposite signs of the input-hidden and hidden-output connection weights (i.e. positive input-hidden and negative hidden-output connection weights or negative input-hidden and positive hidden-output layer connection weights). Interactions among input variables can be identified as input variables with the connection weights entering the same hidden neuron with opposing signs.

Garson's algorithm

Garson (1991) introduced a method for using the connection weights obtained from ANNs to determine the relative contribution of each input variable in modelling the output of the system. Referring to Fig. 2, an example of calculation procedures needed for the Garson's algorithm with three input neurons (1, 2, and 3); two hidden neurons (A and B) and one output neuron (O) can be summarized as follows:

Table 1. Input-hidden-output neuron connection weights.

	Hidden A	Hidden B
Input 1	$W_{A1} = 0.087$	$W_{B1} = -0.526$
Input 2	$W_{A2} = 0.119$	$W_{B2} = -0.284$
Input 3	$W_{A3} = 0.039$	$W_{B3} = 0.793$
Output	$W_{OA} = -0.738$	$W_{OB} = -0.987$

Table 2. Contribution of each input neuron to the output via each hidden neuron (e.g., $C_{A1}=W_{A1}\times W_{OA}$).

	Hidden A	Hidden B
Input 1	$C_{A1} = -0.064$	$C_{B1} = 0.519$
Input 2	$C_{A2} = -0.088$	$C_{B2} = 0.280$
Input 3	$C_{A3} = -0.029$	$C_{B3} = -0.783$

Table 3. Relative contribution of each input neuron (e.g., $r_{A1}=\frac{|C_{A1}|}{|C_{A1}|+|C_{A2}|+|C_{A3}|}$) and sum of input neuron contributions (e.g., $S_1= r_{A1}+r_{B1}$).

	Hidden A	Hidden B	Sum
Input 1	$r_{A1} = 0.355$	$r_{B1} = 0.328$	$S_1 = 0.683$
Input 2	$r_{A2} = 0.486$	$r_{B2} = 0.177$	$S_2 = 0.663$
Input 3	$r_{A3} = 0.159$	$r_{B3} = 0.495$	$S_3 = 0.654$

Table 4. Relative importance of each input variable (e.g. $RI_1=\frac{S_1}{(S_1+S_2+S_3)}\times 100=\frac{0.683}{(0.683+0.663+0.654)}\times 100=34.16\%$

	Relative importance (%)
Input 1	34.16
Input 2	33.14
Input 3	32.69

It should be mentioned that the Garson's algorithm uses the absolute values of the connection weights (i.e., without considering the direction of the relationship) which would lead to the misinterpretation of the importance of the input variables in modelling the output of the system.

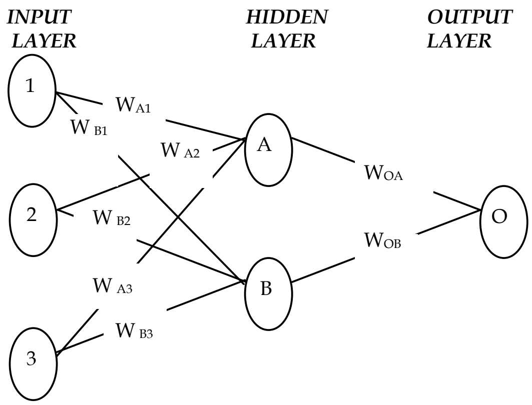

Fig 2. An ANN structure for the illustration of Garson's algorithm.

Randomization approach

Olden and Jakson (2002) proposed the randomization approach, which uses an additional measure to investigate the importance of input variables, i.e. the overall connection weight as defined below in 2b. According to Olden and Jakson (2002) this approach consists of the following procedures:

1. Construct a number of ANN models using the original input output data with randomly generated initial weights.
2. Select the ANN model with the best performance and calculation and record:

a. Input-hidden-output connection weights: the product of input-hidden and hidden-output connection weights, for instance C_{A1}: see Garson's algorithm, step 2.

b. Overall connection weight: the sum of the products of the input-hidden and hidden-output connection weight for each input variable, for instance $C_1 = C_{A1} + C_{B1}$.

3. Randomly permute the original output data.
4. Construct an ANN using randomized outputs.
5. Repeat steps 2, 3 and 4 a large number of times.

In this study, we used the overall connection weights in order to assess the importance of the input variables on the output in a way that if the overall connection weight of an input variable is greater than 95% of the randomized overall connection weights for the same input variable then the input variable can be considered to be significant with a 95% confidence level. This approach was also tested by Kingston *et al*., (2003) and found that it can correctly identify the significant input variables on the output. It might be mentioned that if input variables are found to be insignificant, they will subsequently be removed from the ANN structure. This procedure can be repeated (iterative process) until only significant input variables remain.

RESULTS

Runoff estimation

In this study, a three-layer feed-forward MLP model was developed in order to estimate the monthly runoff in a watershed in northern Iran. As mentioned in the study area and data section, we have developed the feed-forward MLP model using 9 input variables, and 1 output variable, which are defined based on the problem at hand. However, we have chosen six neurons in the hidden layer based on a trial-and-error procedure.

Figs. 3 and 4 show the scatter diagrams of the observed versus simulated monthly runoff obtained from the feed-forward MLP model for training and validation period respectively. It is seen that runoff can be reasonably well simulated by using the developed feed-forward MLP model. However, we are not sure about how this model works. In order to get some explanation from the model, three approaches were used and their results will be explained in next subsection.

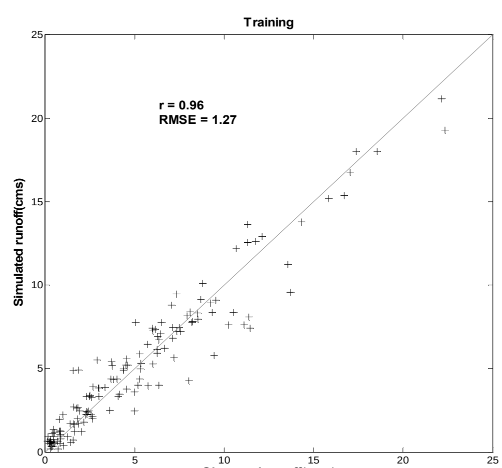

Fig 3. Observed versus simulated runoff for training period.

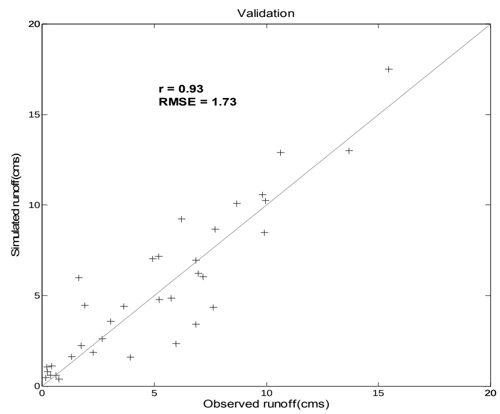

Fig 4. Observed versus simulated runoff for the validation period.

Sensitivity

Fig. 5 represents the NID for the employed ANN rainfall-runoff model in order to depict the influence of each input variable on the output. As can be noticed from Fig. 5, this approach qualitatively depicts the contribution of each input variable via hidden neurons on the output variable. For instance, the temperature variable has positive effects to all neurons in the hidden layer except the second neuron. The outgoing signal from the fifth and sixth hiddenneurons have the most positive and negative effects on the output variable respectively. Moreover, it is possible to depict the interactions among the input variables (described before). For example, the effect of temperature and runoff on the sixth neuron in the hidden layer. As mentioned earlier, this approach qualitatively depicts the connection weights of the ANN rainfall-runoff model.

However, in order to obtaine quantitatively the contribution of each input variable on the output, we used Garson's algorithm. Fig. 6 shows the results. As can be noticed, the input variable contributions ranged from 4.12% to 19.73 %. The highest contribution belongs to the temperature variable measured on station 13005, and the lowest contribution belongs to the rainfall variable measured at station 13009. It does not seem to be appropriate to compare the NID and Garson's algorithm because the NID visually represents the connection weights flowing from input variables via hidden neurons to the output neuron while the latter provides the relative importance of each input variable on the output without considering the direction of the connection weights.

But, it is also interesting to get information about significant input variable (s) on the output. To do so, the randomization approach (by following the procedures mentioned before) was used, and the runoff input variable was found to be the only significant input on the output.

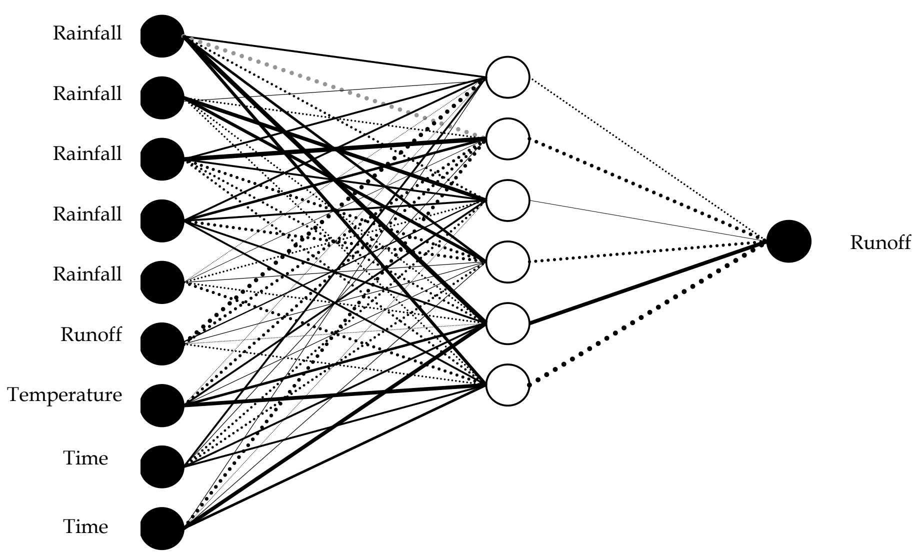

Fig 5. Neural Interpretation Diagram (NID) for artificial neural network (ANN) rainfall-runoff modelling. The line thickness represents the magnitude of the connection weights and the format of the lines as dotted (negative effect), and solid (positive effect) represents the direction of the connection weights.

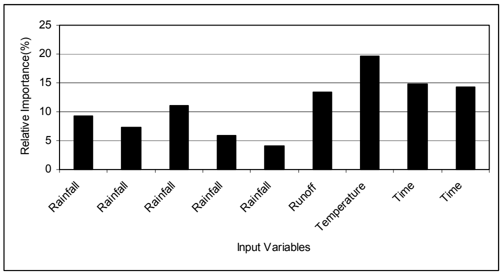

Fig 6. Relative importance of each input variable (%) to simulate runoff based on Garson's algorithm.

CONCLUSIONS

We presented an ANN model to estimate monthly runoff in a watershed in northern Iran. The ANN model found out to be reasonably accurate. However, in order to provide some explanations for those who criticize ANNs as a black-box model, we considered three approaches, including neural interpretation diagram (NID), Garson's algorithm, and randomization approach in order to understanding the mechanisms of being modelled and the results showed the utility. Consequently, ANNs would get greater acceptability among hydrologists by combining their interpretation and predictive abilities.

REFERENCES

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000a) Artificial neural networks in hydrology. I: preliminary concepts. *J. Hydrol. Eng. ASCE* **5(2),** 115-123.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000b) Artificial neural networks in hydrology. II: hydrologic applications. *J. Hydrol. Eng. ASCE* **5(2),** 124-137.

Dawson, C.W. & Wilby, R.L. (1998) An artificial neural network approach to rainfall-runoff modeling. *Hydrol. Sci. J.* **43(1),** 47-65.

Dawson, C.W. & Wilby, R.L. (2001) Hydrological modelling using artificial neural networks. *Prog. Phys. Geog.* **25(1),** 80-108.

Garson, G.D. (1991) Interpreting neural-network connection weights. *Artificial Intell. Expert.* **6**, 47-51.

Giustolisi, O. & Laucelli, D. (2005) Improving generalization of artificial neural networks in rainfall-runoff modelling. *Hydrol. Sci. J.* **50(3),** 439-457.

Hsu, K.L., Gupta, H.V. & Sorooshian, S. (1995) Artificial neural network modeling of the rainfall-runoff process. *Wat. Resour. Res.* **31(10),** 2517-2530.

Jain, A. & Srinivasulu, S. (2006) Integrated approach to model decomposed flow hydrograph using artificial neural network and conceptual techniques. *J. Hydrol.* **317**, 291-306.

Kingston, G.B. Maier, H.R. & Lambert, M.F. (2003) *Understanding the mechanisms modelled by artificial neural networks for hydrological prediction.* MODSIM 2003 congress, Townsville, Australia, July 14-17, Vol. **2,** 825-830.

Lorrai, M. & Sechi, G.M. (1995) Neural nets for modelling rainfall-runoff transformations. *Wat. Res. Manage.* **9**, 299-313.

Maier, H.R. & Dandy, G.C. (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environ. Modelling Software* **15**, 101-124.

Minns, A.W. & Hall, M.J. (1996) Artificial neural networks as rainfall runoff models. *Hydrol. Sci. J.* **41**(3), 399-417.

Olden, J.D. & Jackson, D.A. (2002) Illuminating the "black box": a randomization approach for understanding variable contributions in artificial neural networks. *Ecol. Modelling.* **154**, 135-150.

Özesmi, S.L. & Özesmi, U. (1999) An artificial neural network approach to spatial habitat modelling with interspecific interaction. *Ecol. Modelling*. **116**, 15-31.

Rajurkar, M.P., Kothyari, U.C. & Chaube, U.C. (2002) Artificial neural networks for daily rainfall-runoff modelling. *Hydrol. Sci. J.* **47(6),** 865-877.

Tokar, A.S. & Johnson, P.A. (1999) Rainfall-runoff modeling using artificial neural networks. *J. Hydrol. Eng. ASCE.* **4(3),** 232-239.

Wilby, R.L., Abrahart, R.J. & Dawson, C.W. (2003) Detection of conceptual model rainfall-runoff processes inside an artificial neural network. *Hydrol. Sci. J.* **48(2),** 163-181.