Journal of Hydrology 509 (2014) 379–386

Contents lists available at ScienceDirect

Journal of Hydrology

journal homepage: www.elsevier.com/locate/jhydrol

A comparative study of artificial neural network, adaptive neuro fuzzy inference system and support vector machine for forecasting river flow in the semiarid mountain region

Zhibin He [a,b,*], Xiaohu Wen [b], Hu Liu [a,b], Jun Du [a,b]

[a] Linze Inland River Basin Research Station, Chinese Ecosystem Research Network, China
[b] Key Laboratory of Ecohydrology of Inland River Basin, Cold and Arid Regions Environmental and Engineering Research Institute, Chinese Academy of Sciences, Lanzhou 730000, China

ARTICLE INFO

Article history:
Received 18 June 2012
Received in revised form 10 April 2013
Accepted 27 November 2013
Available online 2 December 2013
This manuscript was handled by Andras Bardossy, Editor-in-Chief, with the assistance of K.P. Sudheer, Associate Editor

Keywords:
River flow forecasting
Artificial neural network
Adaptive neuro fuzzy inference system
Support vector machine

SUMMARY

Data driven models are very useful for river flow forecasting when the underlying physical relationships are not fully understand, but it is not clear whether these data driven models still have a good performance in the small river basin of semiarid mountain regions where have complicated topography. In this study, the potential of three different data driven methods, artificial neural network (ANN), adaptive neuro fuzzy inference system (ANFIS) and support vector machine (SVM) were used for forecasting river flow in the semiarid mountain region, northwestern China. The models analyzed different combinations of antecedent river flow values and the appropriate input vector has been selected based on the analysis of residuals. The performance of the ANN, ANFIS and SVM models in training and validation sets are compared with the observed data. The model which consists of three antecedent values of flow has been selected as the best fit model for river flow forecasting. To get more accurate evaluation of the results of ANN, ANFIS and SVM models, the four quantitative standard statistical performance evaluation measures, the coefficient of correlation (R), root mean squared error (RMSE), Nash–Sutcliffe efficiency coefficient (NS) and mean absolute relative error (MARE), were employed to evaluate the performances of various models developed. The results indicate that the performance obtained by ANN, ANFIS and SVM in terms of different evaluation criteria during the training and validation period does not vary substantially; the performance of the ANN, ANFIS and SVM models in river flow forecasting was satisfactory. A detailed comparison of the overall performance indicated that the SVM model performed better than ANN and ANFIS in river flow forecasting for the validation data sets. The results also suggest that ANN, ANFIS and SVM method can be successfully applied to establish river flow with complicated topography forecasting models in the semiarid mountain regions.

1. Introduction

River flow forecasting is very important for water resources system planning and management, especially in arid area where water resources is scarce, river flow forecasting is useful to water resources temporal and spatial planning and distributions. River flow forecasting has been studied by various scientists during the past few decades. Generally, river flow models can be classed into the two main groups, physical based models and data driven models. Typically, physically based models are complex and require sophisticated mathematical tools, a significant amount of calibration data, and some degree of expertise and experience with the models (Aqil et al., 2007). While data driven models do not provide any information on the physics of the hydrologic processes, they are very useful for river flow forecasting where the main concern is accurate predictions of runoff (Nayak et al., 2005; Chau et al., 2005; Wu et al., 2009). Recently, three data driven methods that have been gained popularity as an emerging and challenging computational technology such as artificial neutral networks (ANNs), adaptive neuro fuzzy inference system (ANFIS) and support vector machine (SVM). These methods offer advantages over conventional modeling, including the ability to handle large amounts of noisy data from dynamic and nonlinear systems, especially when the underlying physical relationships are not fully understood.

In the past few decades, ANNs and ANFIS methods have been extensively used in a wide range of engineering applications including hydrology, such as for rainfall–runoff simulation (Nourani et al., 2009; Talei et al., 2010; Wu and Chau, 2011), groundwater modeling (Kuo et al., 2004; Daliakopoulos et al., 2005; Sahoo et al., 2005; Ghose et al., 2010; Taormina et al.,

* Corresponding author at: Linze Inland River Basin Research Station, Chinese Ecosystem Research Network, China. Tel.: +86 931 4967165.
E-mail address: hzbmail@lzb.ac.cn (Z. He).

0022-1694/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jhydrol.2013.11.054

2012), river flow forecasting(El-Shafie et al., 2006; Shu and Ouarda, 2008) and water quality modeling (Singh et al., 2009; Yan et al., 2010). Recently, SVMs are gaining recognition in hydrology (Moradkhani et al., 2004; Yu et al., 2006; Lin et al., 2006; Wu et al., 2008; Lin et al., 2009; Chen et al., 2010; Yoon et al., 2011). But for some catchments where have a very few meteorological observatories and have complicated topography, it is not clear whether these data driven models still have a good performance.

In this study, the ANN, ANFIS and SVM were used to forecast river flow in a smaller catchment in the Qilian Mountains of northwestern China and the results obtained are compared to each other. The purpose of this study is to investigate the accuracy of three different data driven models ANN, ANFIS and SVM in modeling daily river flow, and evaluate the performance of three data driven models in the small river basin of semiarid mountain regions where have complicated topography.

2. Methodology

2.1. Artificial neural network (ANN)

ANN is a massively parallel distributed information processing system that has certain performance characteristics resembling biological neural networks of the human brain (Haykin, 1999). A neural network is characterized by its architecture that represents the pattern of connection between nodes, its method of determining the connection weights and the activation function. The most commonly used neural network structure is the feed forward hierarchical architecture. A typical three-layered feed-forward neural network is comprised of a multiple elements also called nodes, and connection pathways that link them. The nodes are processing elements of the network and are normally known as neurons, reflecting the fact the neural network method model is based on the biological neural network of the human brain. A neuron receives an input signal, processes it, and transmits an output signal to other interconnected neurons.

In the hidden and output layers, the net input to unit i is of the form

$$Z = \sum_{j=1}^{k} w_{ji} y_j + \theta_i \tag{1}$$

where w_{ji} is the weight vector of unit i and k is the number of neurons in the layer above the layer that includes unit i. y_j is the output from unit j, and y_i is the bias of unit i. This weighted sum Z; which is called the incoming signal of unit i, is then passed through a transfer function f to yield the estimates $\hat{y}_i$ for unit i. The sigmoid function is continuous, differentiable everywhere, and monotonically increasing. The sigmoid transfer function, f_i, of unit i, is of the form

$$\hat{y}_i = \frac{1}{1+e^{-Z}}. \tag{2}$$

A training algorithm is needed to solve a neural network problem. Since there are so many types of algorithms available for training a network, selection of an algorithm that provides the best fit to the data is required. Levenberg–Marquardt learning algorithm was used increasingly due to the better performance and learning speed with a simple structure.

2.2. Levenberg–Marquardt algorithm

The Levenberg–Marquardt algorithm (LMA), is similar to the quasi-Newton method in which a simplified form of the Hessian matrix (second derivative) is used. The Hessian matrix can be approximated as:

$$H = J^T J \tag{3}$$

and the gradient can be computed as

$$g = J^T e \tag{4}$$

in which J is the Jacobian matrix which contains first derivatives of the network errors with respect to the weights and biases, and e is a vector of network errors. An iteration of this algorithm can be written as

$$x_{k+1} = x_k - \left[J^T J + \mu I\right]^{-1} J^T e \tag{5}$$

where μ is the learning rate and I is the identity matrix (Dedecker et al., 2004). During training the learning rate μ is incremented or decremented by a scale at weight updates. When μ is zero, this is just Newton's method, using the approximate Hessain matrix. When μ is large, this becomes gradient descent with a small step size.

2.3. Adaptive neuro fuzzy inference system (ANFIS)

ANFIS, first introduced by Jang (1993), is a universal approximator and as such is capable of approximating any real continuous function on a compact set to any degree of accuracy (Jang et al., 1997). ANFIS is functionally equivalent to fuzzy inference systems. Specifically the ANFIS system of interest here is functionally equivalent to the Sugeno first-order fuzzy model (Drake, 2000). Below, the hybrid learning algorithm, which combines gradient descent and the least-squares method, is introduced.

As a simple example we assume a fuzzy inference system with two inputs x and y and one output z. The first-order Sugeno fuzzy model, a typical rule set with two fuzzy If-Then rules can be expressed as

$$\text{Rule 1}: \text{If } x \text{ is } A_1 \text{ and } y \text{ is } B_1; \text{then } f_1 = p_1 x + q_1 y + r_1 \tag{6}$$

$$\text{Rule 2}: \text{If } x \text{ is } A_2 \text{ and } y \text{ is } B_2; \text{then } f_2 = p_2 x + q_2 y + r_2 \tag{7}$$

where p_1, q_1, r_1 and p_2, q_2, r_2 are the parameters in the then-part (consequent part) of the first-order Sugeno fuzzy model. The architecture of ANFIS consists of five layers (Fig. 1), and a brief introduction of the model is as follows.

Layer 1: Each node of this layer generates membership grades to which they belong to each of the appropriate fuzzy sets using membership functions.

$$O_{1,i} = \mu_{\mathrm{Ai}}(x) \quad \text{for } i = 1,2 \tag{8}$$

$$O_{1,i} = \mu_{\mathrm{Bi-2}}(y) \quad \text{for } i = 3,4 \tag{9}$$

where x, y are the crisp input to the node i; A_i and B_i are the fuzzy set associated with this node, characterized by the shape of the membership functions (MFs) in this node and can be any appropriate functions that are continuous and piecewise differentiable such as Gaussian, generalized bell shaped, trapezoidal shaped and triangular shaped functions. The membership functions for A and B are generally described by generalized bell functions, e.g.

$$\mu_{Ai}(x) = \frac{1}{1 + [(x-c_i)/a_i]^{2b_i}} \tag{10}$$

where $\{a_i, b_i, c_i\}$ is the parameter set that changes the shapes of the MFs with maximum equal to 1 and minimum equal to 0.

Layer 2: This layer consists of the nodes labeled $\prod$ which multiply incoming signals and sending the product out. For instance,

$$O_{2,i} = w_i = \mu_{\mathrm{Ai}}(x)\mu_{\mathrm{Bi-2}}(y) \quad i = 1,2 \tag{11}$$

Layer 3: Every node in this layer is a fixed node labeled N. The ith node calculates the ratio between the ith rule's firing strength to the sum of all rules' firing strengths: