2010 International Conference on Computational and Information Sciences

Rainfall Prediction Using Generalized Regression Neural Network: Case study Zhengzhou

Zhi-liang Wang, Hui-hua Sheng
Institute of hydrology and water resource
North China University of Water Conservancy and Electric Power
Zhengzhou, China
Email: wzl@ncwu.edu.cn

***Abstract*—Although many models have been developed for prediction and forecasting of time series in various engineering fields, there is no perfect model to forecast hydrologic time series. In recent decades, Artificial Neural Networks (ANNs) have been very common for prediction and forecasting of hydrologic time series because of their practicality in applications. In this paper, we propose the application of generalized regression neural network (GRNN) model to predict annual rainfall in Zhengzhou. The results prove that GRNN has more advantage in fitting and prediction compared with back propagation (BP) neural network and stepwise regression analysis methods. The simulation results of GRNN for annual rainfall is better than that of BP neural network. And the accuracy of the prediction results is higher than that of BP neural network. The stepwise regression method is inferior to both of them in the accuracy of simulation and prediction results. In short, GRNN network structure is simple, calculate rapidly and stability. Compared with the traditional linear model and BP neural networks, the GRNN has smaller prediction error.**

Keywords-*rainfall; generalized regression neural network; BP neural network; prediction*

Ⅰ. INTRODUCTION

Long-term annual rainfall prediction is a challenging task especially in the modern world where we are facing the major environmental problem of global warming. In general, climate and annual rainfall are highly non-linear phenomena in nature exhibiting what is known as the "butterfly effect"[1]. The global nature of this phenomenon is very complicated and requires sophisticated computer modeling and simulation to predict accurately. The parameters that are required to predict rainfall are enormously complex and subtle even for a short time period. The period over which a prediction may be made is generally termed the event horizon and in best results, this is not more than a week's time. Now, BP neural network is the most commonly used one of the artificial neural networks, but when used in function approximation, it exists slow convergence speed , local minimum and so on, which limits its hydrological forecast and large sample data real-time situation of application[2].

In this paper, we use GRNN model for predicting the rainfall time series and train on the rainfall data corresponding to a certain period in the past and predictions are made over some other period. Meanwhile, we compare the result of simulation and forecast with BP neural network and stepwise regression analysis methods. The results show that GRNN has a better result than those two, which provides a new way for the annual rainfall prediction.

Ⅱ. ARTIFICIAL NEURAL NETWORKS

Currently ANNs are used for a very wide variety of tasks in many different fields of business, industry and science. Neural networks provide a random mapping between an input and an output vector by mimicking the biological cognition process of our brain. The network "learns" by adjusting the interconnections (called weights) between layers. When the network is adequately trained, it is able to generalize relevant output for a set of input data. A valuable property of neural networks is that of generalization, whereby a trained neural network is able to provide a correct matching in the form of output data for a set of previously unseen input data. Learning typically occurs by example through training, where the training algorithm iteratively adjusts the connection weights (synapses).Network represents in the Fig. 1 includes the input layer with three nodes (independent variables), four nodes in the hidden layer and one output node (dependent variable).

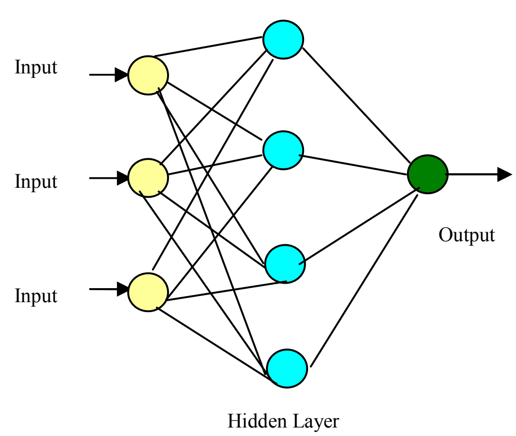

Figure1. A typical feed-forward ANN

978-0-7695-4270-6/10 $26.00 © 2010 IEEE
DOI 10.1109/ICCIS.2010.312

A. BP neural network

Back propagation is the generalization of the Widrow-Hoff learning rule to multiple-layer networks and nonlinear differentiable transfer functions. Input vectors and the corresponding target vectors are used to train a network until it can approximate a function, associate input vectors with specific output vectors, or classify input vectors in an appropriate way as defined by you. Networks with biases, a sigmoid layer, and a linear output layer are capable of approximating any function with a finite number of discontinuities. Standard back propagation is a gradient descent algorithm, as is the Widrow-Hoff learning rule, in which the network weights are moved along the negative of the gradient of the performance function. The term back propagation refers to the manner in which the gradient is computed for nonlinear multilayer networks. There are a number of variations on the basic algorithm that are based on other standard optimization techniques, such as conjugate gradient and Newton methods.

The multilayer perceptron is trained with error-correction learning, which means that the desired response for the system must be known. The error correction learning works in the following way: from the system response $d_i(n)$ at node i at iteration n, and the desired response $y_i(n)$ for a given input pattern, an instantaneous error $e_i(n)$ is defined by

$$e_i(n) = d_i(n) - y_i(n) \tag{1}$$

Using the theory of gradient-descent learning, each weight in the network can be adapted by correcting the present value of the weight with a term that is proportional to the present in put and error at the weight, i.e. the weight from node j to node $i(w_{ij})$ can be calculated by:

$$w_{ij}(n+1) = w_{ij}(n) + \eta\delta_i(n)x_j(n) \tag{2}$$

Where, x_j is a transform function at node j, i and j indicate different layers.

The local error $\delta_i(n)$ can be directly computed $e_i(n)$ at the output node or can be computed as a weighted sum of errors at the internal nodes. The constant η is called the step size. Most gradient search procedures require the selection of step size. The larger step size, the faster the minimum can be reached. However, if the step size is too large, then the algorithm will diverge and the error will increase in stead of decrease. If the step size is too small then it will take too long to reach the minimum, which also increases the probability of getting caught in local minima. Momentum learning is an improvement to the straight gradient descent in the sense that a memory term (the past increment to the weight) is used to speed up and stabilize convergence. In momentum learning the equation to update the weights becomes:

$$w_{ij}(n+1) = w_{ij}(n) + \eta\delta_i(n)x_j(n) + \alpha(w_{ij}(n) - w_{ij}(n-1)) \tag{3}$$

Where α is the momentum. Normally α should be set between 0.1 and 0.9. The black propagation algorithm is applied as follow:

1) Initialize all weights and bias (normally a small random value) and normalize the training data.

2) Compute the output of neurons in the hidden layer and in the output layer (net) using

$$net_i = \sum w_{ij}x_i + \theta_i \tag{4}$$

Where θ_i is a bias for PE_i.

Compute the error and weight update. Update all weights, bias and repeat steps2 and 3 for all raining data. Repeat steps2 to 4 until the error converges to an acceptable level.

B. Generalized regression neural network (GRNN)

GRNN is often used for function approximation. GRNN can be considered as a normalized Radial Basis Function (RBF) network which has a radial basis layer and a special linear layer. These RBF units are called kernels and are usually probability density functions such as the Gaussian. The hidden-to-output weights are just the target values, so the output is simply a weighted average of the target values of training cases close to the given input case. The first layer is just like of a RBF network with as many neurons as there are input/target vectors. Choosing the spread parameter of the radial basis layer determines the width of an area in the input space, to which each neuron responds. Spread should be large enough that neurons respond strongly to overlapping regions of the input space. If the spread is small the RBF is very steep so that the neuron with the weight vector closest to the input will have a much larger output than other neurons. The network will tend to respond with the target vector associated with the nearest design input vector. As the spread gets larger the RBF slope gets smoother and several neurons may respond to an input vector. The network then acts like it is taking a weighted average between target vectors whose design input vectors are closest to the new input vector.

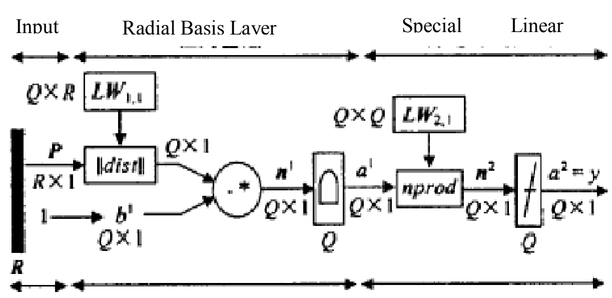

Figure2. GRNN structure

Where R=number of elements in input vector; Q=number of neurons in layer 1; Q=number of neurons in layer 2; Q=number of input/target pairs; a^2=*purelin* (n^2); $a_i^1 = radbas(\|_i LW^{1,1} - P\| b_i^1)$; a_i^1 is i^{th} element of

a^1 where ${}_iLW^{1,1}$ is a vector made of the i^{th} row of $LW_{1,1}$. The ‖dist‖ box in this figure accepts the input vector P and the input weight matrix $LW_{1,1}$, and produces a vector having Q elements. Here the *nprod* box shown above produces Q elements in vector n^2. Each element is the dot product of a row of $LW^{2,1}$ and the input vector a^1, all normalized by the sum of the elements of a^1.

III. A CASE STUDY ZHENGZHOU

A. Study area and data

Zhengzhou, the capital and commercial city of Henan province, is one of the highly developed cities in China. Having a land area of 7446.2 km^2, it is located in the north central part of the Henan on southeastern plain of Huanghuai, with longitude 112.23 E and latitude 34.1 N. The area has a tropical type of climate with long hours of sunshine, temperate, monsoonal and four distinct seasons. The average annual temperature is 14.4℃, with the average minimum temperature of 0.2℃ in January and the average maximum temperature of 27.3℃ in July. Zhengzhou receives an average annual rainfall of 640.9 mm. The city is affected by the influenced by the flood on regular basis due to rainfall. Some of the immediate consequences of a heavy rainfall in Zhengzhou are: water clogging in the streets, heavy traffic jams, blackouts and direct or indirect economic losses.

Historical rainfall data is collected from China's meteorological science data sharing network for the period from 1955 to 2009.

B. Modeling process

1) The data processing

Split-sample training is a common method to train ANN models; in this method, collected data are divided into training and testing set. However, recent works have found that the better-trained model is not always coupled with better performance in the testing. A practical way to find a point of better generalization and avoid over training is to set aside a small percentage of the training data set, which then can be used for cross validation. Monitoring the Errors in the training set and the validation set should be carried out during the training process. When the error in the validation set increases, the training should stop because the point of best generalization has reached. This cross validation approach was adopted for the training of ANN models in this study.

Select the precipitation in statistical data, as all the study sample GRNN network. According, 55 years data (1955 to 2009) with annual rainfall records are selected to train ANN models, in which 75% of the data is set as a training set, 15% of the data is set aside to use for cross validation and 15% of the data is set as a testing data. The training set is used to calibrate the weights of the network; the cross validation is used to monitor the progress of training process. The testing data set have no effect on training and so provide an independent measure of network performance during and after training.

In order to avoid the level difference between the sample data and eliminate errors by different dimension and unit, we need to normalize the data. That is making the input and output data between 0 and 1. Generally, normalize the data by the low formal (5):

$$x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \tag{5}$$

Where x_i 、 x_i' are variables before and after normalization respectively; $x_{\max}$ and $x_{\min}$ are maximum and minimum variable of x respectively.

And further more, to enhance the network convergence speed and computing speed, this paper presents (6) to normalize data.

$$z_i = z_{\min} + \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \times (z_{\max} - z_{\min}) \tag{6}$$

Where x_i 、 z_i are variables before and after normalization respectively; $x_{\max}$ and $x_{\min}$ are maximum and minimum variable of x respectively; the value of $t_{\min}$ is between 0.1 and 0.2 and the value of $t_{\max}$ is between 0.8 and 0.9.

2) Network training

Input the value of smoothing parameter σ , and then do GRNN network training. In this paper, we choose different value of σ to compare mean square error (MSE) in order to judge whether the selected values is the optimal parameters. Meanwhile determine the optimal GRNN network structure and the weights of connections between neurons to obtain the optimum regression estimation results. Finally, enter a new sample data to predict.

Therefore, the essence of GRNN is that it fits firstly based on the existing annual rainfall sample data. Input new influence factors into trained prediction model, and the output is the prediction.

We use MATLAB neural tool to execute, and test data is presented to the network and the output from the network is compared with the actual rainfall data in the time series.

Ⅳ. RESULT AND DISCUSSION

For comparative analysis, we compare fitting and prediction results of GRNN with BP neural network and stepwise regression analysis methods' results. It shows that the simulation results of GRNN for annual rainfall is better than that of later methods.

Fig.3 shows the comparison of observed rainfall and predicted rainfall by different methods from 1955 to 2009 in Zhengzhou.

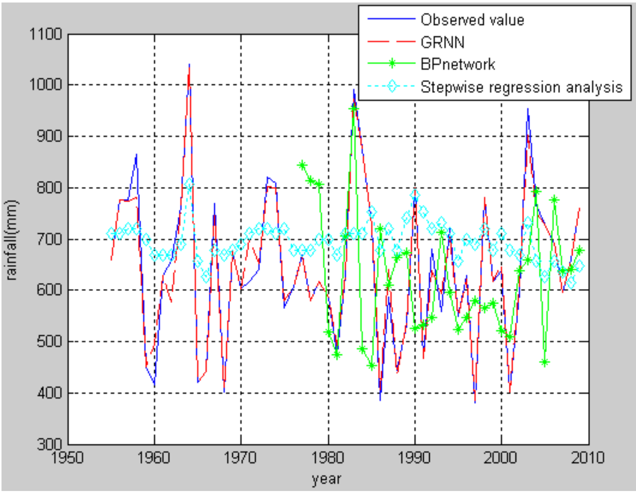

Figure3. Comparison of observed rainfall and predicted rainfall by different methods from 1955 to 2009 in Zhengzhou

We can see from Fig.3 that the simulation results very close to the measured sequence. And the accuracy of the prediction results is higher than that of BP neural network. The stepwise regression method is inferior to both of them in the accuracy of simulation and prediction results.

Moreover, table1 shows the relative error of prediction comparison of different models.

Table1. The relative error of prediction comparison of different models

Year \ MSE(%) \ Methods	GRNN	BP NN	Stepwise regression
2000	-1.87	4.71	11.45
2001	-0.057	-20.89	68.80
2002	-2.89	-3.79	11.41
2003	5.33	-12.81	-23.33
2004	2.36	25.58	-14.37
2005	0.01	30.99	-14.20
2006	0.03	-9.52	-5.12
2007	-0.03	-5.69	6.62
2008	-0.01	4.81	-6.61
2009	-0.05	14.33	-15.21

The absolute value of the maximal relative of GRNN is 5.33%, and the average absolute value of the relative is 1.26%; the absolute value of the maximal relative of BP neural network is 30.99%, and the average absolute value of the relative is 13.32%; the absolute value of the maximal relative of Stepwise regression is 68.80%, and the average absolute value of the relative is 17.72%.

Ⅴ. CONCLUSION AND FUTURE OUTLOOK

A reasonable long-term annual precipitation prediction has important significance in the flood control, water resources planning and management and utilization of Zhengzhou. Combined with the geography and climate characteristics, the corresponding GRNN model was established for the annual rainfall in Zhengzhou. The results are proved that GRNN had more advantage in fitting and prediction compared with BP neural network and stepwise regression analysis methods. The simulation results of GRNN for annual rainfall is better than that of BP neural network. And the accuracy of the prediction results is higher than that of BP neural network. The stepwise regression method is inferior to both of them in the accuracy of simulation and prediction results.

Since the choice of inertia factor and learning factor will affect calculation accuracy, further work is needed.

REFERNCES

[1] “Weather Patterns are influenced by the Butterfly Effect”. http://www.unexplainable.net/artman/publish/article_1037.shtml.
[2] Bishop, C, M., “Neural networks and their application”. Rev. Sci. Instrum, 65, 1803-1832, 1994.
[3] Ahmad, S. and Simonovic, S. P., “An artificial neural network mode for generating hydrograph from hydro meteorological parameters”, J. Hydrol, 315(1-4), 236-251, 2005.
[4] Akhtar M. K., Corzo G. A., van Andel S. J., and Jonoski A., “River flow forecasting with artificial neural networks using satellite observed precipitation pre-processed with flow length and travel time information: case study of the Ganges river basin”, Hydrol. Earth Syst. Sci., (13), 1607–1618, 2009.
[5] David, C. H, D. R. Maidment, G.-Y. Niu, Z.-L. Yang and F. Habets., “River network routing in the Guadalupe and San Antonio River Basins”, submitted to Water Resources Research on Sept 25 2009.
[6] N.Q. Hung, M.S. Babel, S. Weesakul, and N.K. Tripath, “An artificial neural network model for rainfall forecasting in Bangkok, Thailand”, Hydrol. Earth Syst. Sci.,(13), 1413-1425, 2009.