Discharge Forecasting By Applying Artificial Neural Networks At The Jinsha River Basin, China

Muhammad Tayyab
Jianzhong Zhou
Xiaofan Zeng
Rana Adnan
School of Hydropower & Information Engineering,
Huazhong University of Science & Technology, Wuhan, China

Abstract
       Flood prediction methods play an important role in providing early warnings to government offices. The ability to predict future river flows helps people anticipate and plan for upcoming flooding, preventing deaths and decreasing property destruction. Different hydrological models supporting these predictions have different characteristics, driven by available data and the research area. This study applied three different types of Artificial Neural Networks (ANN) and an autoregressive model to study the Jinsha river basin (JRB), in the upper part of the Yangtze River in China. The three ANN techniques include feedforward back propagation neural networks (FFBPNN), generalized regression neural networks (GRNN), and the radial basis function neural networks (RBFNN). Artificial Neural Networks (ANN) has shown Great deal of accuracy as compared to statistical autoregressive (AR) model because statistical model cannot able to simulate the non-linear pattern. The results varied across the cases used in the study; based on available data and the study area, FFBPNN showed the best applicability, compared to other techniques.

Keywords: Feedforward back propagation neural networks (FFBPNN); generalized regression neural networks (GRNN); radial basis function neural networks (RBFNN); Jinsha River Basin (JRB)

Introduction
       Many hydrologists are currently researching the ability to generate timely and accurate stream flow predictions. Correctly estimating streamflow is vital for effectively managing water resource systems. Dependable discharge predictions can allow water establishments to optimally provision

water to support a range of water uses, such as domestic use, hydropower generation, agricultural use and ecological flow (Kişi, 2007).Revising predictions generally requires a series of past data; upcoming events can be forecasted based on historical activity (Sulaiman, El-Shafie, Karim, & Basri, 2011). Reliable discharge conditions require that local organizations, such as water consultants, maintain accurate discharge data. For example, water authorities record daily precipitation, humidity, temperature, water level, rainfall, river discharge, evaporation, and snow melt data (Mishra, Gupta, K Pandey, & P Shukla, 2014). Forecasting relies on these historical hydrological data.

The need to create additional and accurate time series forecast models has motivated investigators to develop advanced approaches to model time series, solving non-linearity problems. By monitoring the relationship between rainfall and runoff, researchers have developed numerous methods to forecast upcoming events. Two types of mathematical methods are typically used for discharge forecasting: streamflow models and rainfall–runoff models.

Streamflow models use only hydrological data, whereas rainfall-runoff models use both hydrological data and climatic data. The mathematical methods can be either physically based, or based on system black box models. Physical models are based on physical laws; these models imitate hydrological processes in a representative or physical way. Physically based models (PBM) can be further divided based on the following levels of increasing complexity: from conceptual, to lumped, to fully distributed models. Physically based deterministic processes require large amounts of data for calibration and verification; the extensive computational techniques require significant time to complete. The key benefit of PBM is that these models explain watershed and physical processes in detail, and as such, can be used for developmental processes.

Black-box models try to improve associations between the input and output variables involved in a physical process, without seeing the fundamental physical process. Such models are called "data driven models." These models are driven by links among the system state variables, with only limited data needed to explain the system's physical action. Data driven models are usually easy to develop and implement. This method is very effective for real-time stream flow prediction, where the purpose is to reliably predict discharge at a desired location at a specific time.

Recently, artificial intelligence methods, such as artificial neural networks (ANN), have been applied to efficiently address hydrological applications, such as flood forecasting, precipitation estimations, rainfall-runoff modeling, evaporation estimations, water quality modeling, and developing groundwater models of stage-discharge relationships. ANN

models are a kind of data-driven model, and have been used extensively to predict discharge or floods. Many unique characteristics of ANN models make them effective for prediction purposes (Maier & Dandy, 2000). ANN discharge flow prediction models contain several characteristics, which support using ANN techniques in developing countries. ANN models are not difficult to develop, as they do not need extensive knowledge about catchment physical operations. Physical variables simply serve as external inputs to the model. After establishing parameters, little implementation time is required.

Several investigators have studied conventional approaches for time series examinations, modeling, and predictions. Examples include Box–Jenkins methods of auto regression (AR), auto-regressive moving average (ARMA), auto-regressive integrated moving average (ARIMA), and autoregressive moving average with exogenous inputs (ARMAX). There is extensive literature on applying the AR model to forecast hydrologic time series. These models are important in hydrology, because these models can generate fresh time series orders with the same statistical parameters as the observed order. However, these models do not try to represent the non-linear dynamics inherent in the transformation of rainfall to runoff, and therefore, may not always perform well (Hsu, Gao, Sorooshian, & Gupta, 1997). Although these time series modeling methods have helped scientists and are important in the own right, they deliver just basic correctness and they suffer from the expectations of stationary states and linearity. In other words, these models can only provide effective results for linear time series data.

Over the past 20 years, many researchers have applied different ANN techniques and compared them with other statistical techniques. For the most part, they have found that ANN has great accuracy compared to other techniques. Discharge forecasting was done by ___ (Raman & Sunilkumar, 1995) using ANN's at two reservoir sites; results were compared to an AR(2) model. Jayawardena and Fernando (1996) used multiple layer perceptron (MLP) and RBF neural network methods for forecasting, and compared results with an ARMAX model. The authors found that the ANN method performed well. The radial basis function (RBF) and feedforward ANN (FFNN) models were applied in a day-to-day discharge assessment by ___ (Sudheer & Jain, 2003). Applying ANN's in forecasting mean monthly discharge flows was also verified by __ (Kisi, 2004), and ___ [9] compared soft computing techniques for flood prediction.

These outcomes suggest that the ANN method may be a good substitute for the AR model to evolve input-output simulations and prediction methods. The accuracy of FFNN and RBF models were examined by __ (Sahoo & Ray, 2006), and ___ [11] applied three different ANN techniques for two different stations on the Canakdere and Isakoy Rivers, in

the eastern Black Sea region of Turkey. In Northern Vermont in the U.S., two ANN's models were tested at ungauged basins by ___ (Besaw, Rizzo, Bierman, & Hackett, 2010). At Awash, ____ [13] conducted long term forecasting using ANN models, with a great deal of accuracy.

No research has been done to apply the three ANN models to predict discharge at the Panzhihua gauging station. The JRB is the primary source of water for the Yangtze River; as such, researchers have recently focused on discharge fluctuations in this area. As such, there is value in studying discharge changes in the JRB using ANN's. In this study, three ANN models and an AR model were applied to forecast discharge flow from the JRB River at the Panzhihua gauging station. The ANN techniques studied include Feedforward Back Propagation Neural Networks (FFBP), Generalized Regression Neural Network (GRNN), and Radial Basis Neural Networks (RBF). This study then compares the results of the FFBP, GRNN, and RBF networks, with each other and with the AR model statistical technique.

Materials and Methods
Study Area

The JRB (Figure 1) is known as the upper stream of Yangtze River in China. The drainage area of the JRB is 473.2×103 km^2, accounting for approximately 26% of the total drainage area of the Yangtze River basin. The water cycle of this area has great significance to national projects, such as the Three-Gorges-Dam. The JRB River flows over five main landforms: the Qinghai-Tibet plateau, the western Sichuan plateau, the Hengduan Mountains, the Yunnan-Guizhou plateau, and the mountainous area of SW Sichuan. Due to complicated geographical conditions, the basin experiences many climate patterns, including a typical plateau climate, the Hengduan Mountain climate, a vertical climate, and a monsoon climate. The Jinsha River is the main stream of the JRB (Figure 2); "Jinsha" literally means "gold sand" or "golden sands." The JRB is approximately 3,464 km long, with a mean annual runoff of approximately 152×10^6 m^3. JRB discharge rises considerably as the Yalong River merges into the Jinsha River at Panzhihua City in Sichuan Province. Based on this, the Panzhihua station was selected as the forecast station for this study. The city has a hydropower capacity of 700 million kilowatts, an installed capacity of 3.474 million kilowatts has been developed. This includes two beach locations at 330 million kilowatts, a salt water station at 88,000 kilowatts, and a small hydropower Miyi at 86,000 kilowatts. There are 3.5 million kilowatts of installed capacity of hydropower to develop.

The Jinsha River is the most western of the Yangtze River's major headwater streams. It flows from the north to the south, and forms the world's deepest gorge. The JRB has plentiful hydropower resources. The

total exploitable installed capacity for power generation in the JRB makes up 40% of the total capacity of the Yangtze River, and represents one sixth of the capacity of China. The JRB also contributes to irrigation, water supply, flood control, wood drift, and tourism. Clearly, the JRB plays a very important role in regional and national economic development.

Data

Due to the vast hydropower energy in the upper Yangtze River basin, predicting stream flow is an important aspect of water resource management. Recent research has shown that rainfall increases in spring and summer and significantly decreases in winter and autumn. Table 1 shows the rainfall variations across the past 45 years. Because of the increased precipitation in spring and summer, there may be more floods and geological disasters, such as earthquake disaster, landslide, debris flow, and collapses. Considering fluctuations in regional climate variables supports streamflow level predictions at the Panzhihua hydrological station. As such, we used monthly river discharge data collected at Panzhihua hydrological station between 1961 and 2005. These data were provided by the Changjiang Water Resources Commission of the Ministry of Water Resources.

TABLE 1: Average annual and seasonal precipitation in JRB (mm)

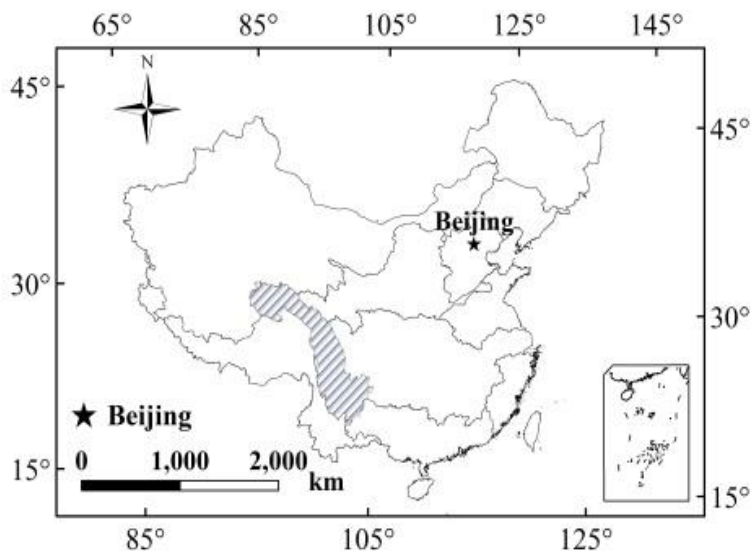| Time	1961-69	1970-79	1980-89	1990-99	2000-05
Annual	652.33	726.34	737.95	761.58	750.02
Spring	46.57	109.78	101.59	105.89	124.17
Summer	443.20	422.97	436.23	459.10	437.57
Winter	148.45	170.41	176.43	169.83	164.25
Autumn	12.69	17.41	18.33	22.22	18.93

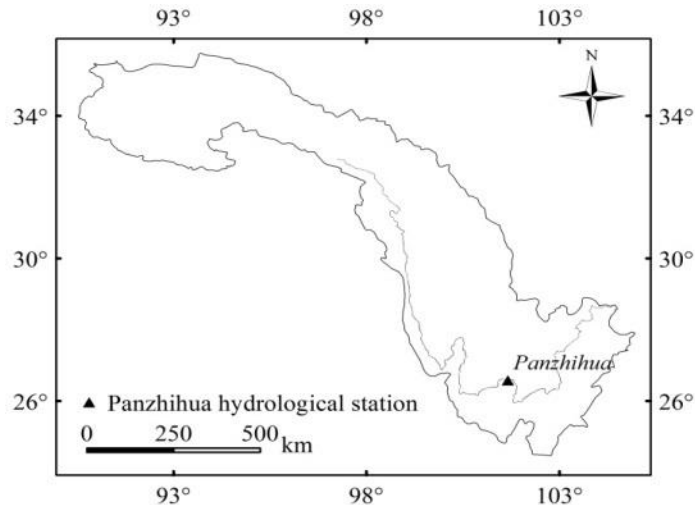FIGURE 1: Location of the JRB in China

FIGURE 2: Location of the Panzhihua hydrological station in JRB

Matlab Application

MATLAB is software developed by the Mathworks Company in the United States of America. The MATLAB computer language is primarily used to develop algorithms, mathematical calculations, and for data analysis; however, it also offers a range of matrix operations. Researchers have applied MATLAB extensively in the water science field. For example, Lei Xiaoyun and his team used Wulasita River in the Tacheng area of Xinjiang as a case study, using the MATLAB neural network toolbox to develop an annual flow forecasting model, with very accurate results (LEI, ZHANG, & LIANG, 2008). Sheng Zhongbiao used the MATLAB language for neural networks, also gaining effective results (Sheng, 2008); neural network functions have also been used for planning, training, and simulation. This research has demonstrated that MATLAB toolboxes can be used to build a BP neural network. Using the annual flow forecasting model for the upper Yangtze river as a case study may demonstrate the significance of the forecasting results (Xiang, Liang, Lin, & Liang, 2012). All of these applications and research demonstrate that researchers can use the MATLAB neural network toolbox to design, train, and simulate neural networks. The neural network toolbox concept, based on artificial neural network theory, is one of the most distinctive MATLAB toolboxes. In addition, based on rules for revising weights of various typical networks and network training processes, researchers have developed network weight training programs using MATLAB languages; these can directly improve research efficiency and quality (Wen, Qiao, Li, & Shao, 2001). In this study, we used the MATLAB neural network toolboxes to predict discharge at the Panzhihua gauging station.

Feed-Forward Back Propagation Neural Network

FFBPNN is most commonly used in engineering applications (Cloke & Pappenberger, 2009). The FFBP contains three layers: the input, the hidden layer, and output layer. The input layer gets information about the neurons, which it then passes to the invisible middle layer. This layer is responsible for all midway calculations; the intermediate information is then passed to the outer layer, with a single neuron and generating internal calculated results. (Figure 3) shows the FFBPNN structure with the three neuron layers. Finding an appropriate number of hidden layers, number of neurons, and the type of transform function greatly influences FFBP model construction (Dawson & Wilby, 1998). Selecting the number of hidden layers depends on the complexity of both input and target parameters, and has a significant role in the quality of learning. FFBP generally has fewer layers; increasing the number of layers raises the number of calculations. Previous research has shown only one hidden layer is enough (Govindaraju, 2000); as such, only one was used in this research. Also, for the hidden nodes, transfer functions are needed to present nonlinearity into the network. As an activation function of the hidden neurons, the sigmoid was selected, while a linear activation function was used in the output neurons.After selecting the hidden layer, the next step involves obtaining the best number of neurons for the hidden layer. If there are insufficient neurons, the network will not be able to compute complex data, and outputs will not match the desired points. More hidden neurons improve training, and the network gives more suitable and desire output points (Maier & Dandy, 1997). In this study, the optimal number in the hidden layer was selected using trial and error, based on the least mean square error (MSE). Optimizing joining weights (w*) is the objective of training FFBP; the calculated output should tie the output values. Generally, this is a non-linear optimization problem:

$$w *= \mathrm{argmin} E(w) \qquad (1)$$

In this expression, w is the weight matrix; E(w) is an objective function of w, and should be minimized. The E(w) is evaluated at any point of w based on Equation (2):

$$E(w) = \sum_{P} E_{P}(w) \qquad (2)$$

In this expression, p is the number of examples in the training set and Ep(w) is the output error for each example p. Ep(w) is expressed by Equation (3):

$$E_{P}(w) = \frac{1}{2} \sum_{j} (d_{pj} - y_{pj}(w))^{2} \qquad (3)$$

In this expression, ypj(w) and dpj are the calculated and desired network outputs of the jth output neuron for the pth example, respectively. The objective function to be minimized is represented by Equation (4):

$$E(w) = \frac{1}{2} \sum_{p} \sum_{j} (d_{pj} - y_{pj}(w))^2$$

(4)

For every learning (training) procedure, the network calculated output value is compared to the desired output value. Based on the differences between these two, the synaptic weights (which play a role in producing significant errors) are altered, controlling the weight to achieve an acceptable marginal error. Weight alteration starts with the output neurons and then extends toward the input data. Numerous algorithms exist to achieve weight selection and modification (Box, 1976). The most common is the gradient descent, which has slow convergence times and can acquire a local minimum within the vector space of weight during the learning process. This helps evolve the model in a more accurate direction.
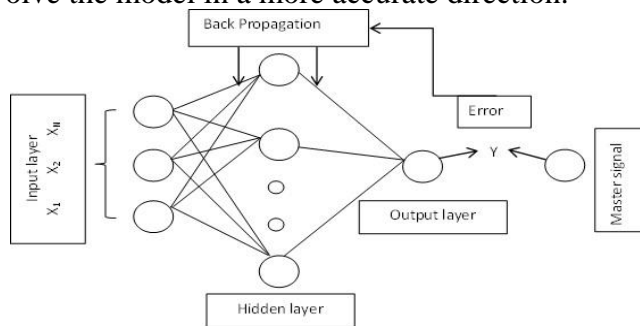

FIGURE 3: FFBPNN structure with three neuron layers

Generalized Regression Neural Network

The requirement for wide planning and research through open limits is meaningfully controlled using GRNNs. This kind of neural network has more structural significance compared to standard neural networks (Theodosiou, 2011). The Generalized Regression Network contains four layers of nodes, shown in (Figure 4): the input layer, the radial basis layer, the summation layer, and the output layer with completely diverse functions. The input layer takes the data, which then passes the data to the second layer for processing. The radial basis layer joins and processes the data in an orderly way to generate a 'best fit" relationship among the input and output variables, using the Gaussian transfer function as shown in Equation (5):

$$\varnothing(X) = \exp\left(-\frac{\|X - \mu\|^2}{2\sigma^2}\right) \sigma > 0, X \in R$$
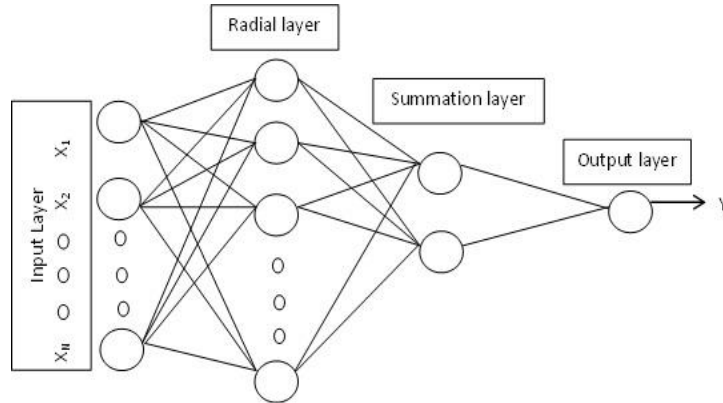
(5)

115

FIGURE 4: GRNN structure with four neuron layers

The data are then passed to outline/summation neurons, where the output is amplified and passed to the output neurons. Instead of input and output layers, the only free factor, the smoothing factor, plays an important role in network design. This factor modifies the grade of GRNN generalization. If a factor's value approaches 1, it indicates a high smoothing factor, strengthening the forecasting line's path. A 0 value will generate a dot-to-dot map. The smoothing factor has a direct relationship with the network's ability to generalize. This means that if the smoothing factor has a greater value, it degrades the prediction error (Popescu, Constantinou, Nafornita, & Nafornita, 2004). On the other hand, a low smoothing factor can degrade the network's ability to generalize and may generate ineffective predictions.

GRNNs have a simple design, but learn quickly, and consistently achieve the best regression point. Like many other neural networks, the need many repetitions and extensive computational time to achieve the preferred functional point. Given these properties, GRNNs do not face general problems such as the local minima problem, which occur on other neural networks. Further, they do not generate ambiguous predictions. GRNNs are able to bear a higher noise level in the input, due to the design's simplicity and toughness, and have quick calibration and verification properties.

Radial Basis Function Network

The back-propagation algorithm of a multi-layer feed-forward ANN suffers from a local optimum problem, as well as extended training time. This problem is resolved using Radial-Basis Function (RBF) networks. First, the RBNN was presented into the neural network texts (Sudheer & Jain, 2003). Radial basis functions (RBF) are influential methods for interpolating multidimensional fields.

An RBF is a function that effectively filters data from noise, and can replace the sigmoidal transfer function. These networks are considered to be three-layer networks: the input layer, the hidden layer (which executes a static nonlinear transformation with no modifiable parameters), and a linear output layer. For nonlinearity, the utmost standard choice is the Gaussian function, where the activation level Oj of hidden unit j is calculated using Equation (6):

$$O_j = \exp\left(\frac{-(X - W_j) \times (X - W_j)}{2\sigma_j^2} \right)$$

(6)

In this equation, X is the input vector; Wj is the weight vector associated with hidden unit j (i.e., the Center of the Gaussian function); and σ2j is the normalization factor. The purpose of the hidden layer is to generate a significant non-zero response (Alvisi, Mascellani, Franchini, & Bardossy, 2006). The hidden layer outputs have values between 0 and 1; the closer the input is to the center of the Gaussian function, the larger the node's response. Because the node produces an identical output for inputs with an equational distance from the center of the Gaussian function, it is called a radial basis function. The activation level Oj of an output unit is determined by Equation (7):

$$O_j = \sum_{t=1}^{L} W_{ji} O_i$$

(7)

In this expression, Wji is the weight from hidden unit i to output unit j, and L is the number of hidden units. The output units form a linear combination of the nonlinear basis functions. As such, the overall network performs a nonlinear transformation of the input. (Figure 5) shows the process involved in the training:
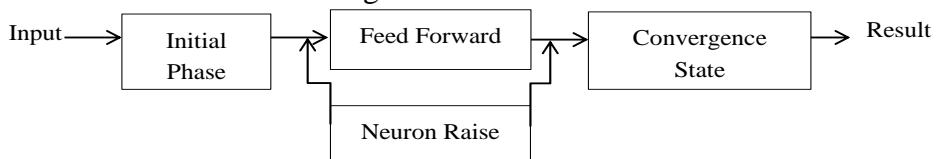
Input → | Initial Phase | ⤴ | Feed Forward | ⤴ | Convergence State | → Result

| Neuron Raise |

FIGURE 5: Training process for the radial basis function neural network

Autoregressive Model

In a multiple regression model, we predict the variable of concern using a linear combination of forecasts. In an autoregression model, we predict the variable of interest using a linear combination of previous variable values. The term autoregression indicates that it is a regression of the variable against itself. Time series models are used to forecast stream flows in hydrology. The general equation of an AR (autoregressive) model is as follows:

$$y_i = \sum_{j=1}^{p} \emptyset_j Y_{j-1} + \varepsilon_i$$

(8)

In this expression, Φ is the regression coefficients of the model, ε is an independent variable. The AR (p) model is given in the following matrix:

$$\begin{bmatrix} \rho_1 \\ \rho_1 \\ . \\ . \\ . \\ \rho_p \end{bmatrix} = \begin{bmatrix} 1 & \rho_1 & . & . & . & \rho_{p-1} \\ \rho_1 & 1 & . & . & . & \rho_{p-2} \\ . & & 1 & . & . & . \\ . & & & 1 & . & . \\ . & & & & 1 & \rho_1 \\ \rho_{p-1} & . & . & . & . & 1 \end{bmatrix} x \begin{bmatrix} \emptyset_1 \\ \emptyset_2 \\ \\ \\ \\ \emptyset_{p-1} \end{bmatrix}$$

$$[\rho] = [\quad\quad \rho \quad\quad] x [\emptyset \quad]$$

$$\emptyset = P^{-1}\rho$$

$$y_i = \sum_{j=1}^{p} \emptyset_j Y_{i-j} + \varepsilon_i = \emptyset_1 Y_{i-1} + \emptyset_2 Y_{i-2} + \dots\dots + \emptyset_p Y_{i-p} + \varepsilon_i$$

(9)

Data Conversion

To obtain more suitable and effective results for ANN's, data are scaled to remain within specific ranges. In this research, data were ranged between negative one to positive one or [-1 to +1] by using equation (10):

$$Z_t = \frac{2(Q_t - Q_{\min})}{(Q_{\max} - Q_{\min})} - 1$$

(10)

In this expression, Z_t represents inflows with values ranging between $[-1$ to $+1]$; Q_t is the monthly flow; Q_{max} and Q_{min} are maximum and minimum flow values, respectively (Demuth et al. 2008).

Calibration and Verification

Past research provides evidence to help calibrate and verify models, to calculate the catchment features and boundary scenarios. Calibration is a process of examining and regulating results against an identified standard for instrument correctness.

The result is well-defined criteria with a stated degree of assurance. Continued research on these approaches lead to better models and model application. Verification is the process used to certify that the instrument is properly set to constantly execute, accordingly to the predefined conditions appropriate for future use. In short, calibration is the process of adjusting model parameters, whereas verification checks model performance on the basis of adjustments done during the calibration.

The JRB was selected to calibrate and verify all models used in this research. Monthly discharge data over 45 years were used in this study; 80% of the data were used for calibration, and the remaining 20% of the data were used for verification. The ANN inputs included the previous months' discharges; the output was the discharge of present month t. Six different input combinations were observed:

(i) Q_{t-1}
(ii) Q_{t-1} and Q_{t-2}
(iii) Q_{t-1}, Q_{t-2} and Q_{t-3}
(iv) Q_{t-1},Q_{t-2}, Q_{t-3} and Q_{t-4}
(v) Q_{t-1},Q_{t-2}, Q_{t-3},Q_{t-4} and Q_{t-5}
(vi) Q_{t-1},Q_{t-2}, Q_{t-3},Q_{t-4}, Q_{t-5} and Q_{t-6}

Performance Criteria

Many goodness-of-fit measures have been applied to evaluate model performance. Appropriate evaluation criteria are important when using a multi-criteria analysis to validate model performance. In this study, the following four popular statistical measures were used as evaluation criteria for evaluating ANN model performance: correlation coefficient (R), mean square error (MSE), relative error (RE), and the Nash Sutcliffe model efficiency coefficient (E). These measures are shown in Equations (11), (12), (13), and (14), respectively. These variables describe the degree to which results are likely to be accurately forecasted by the model.

$$R = \frac{\sum_{t=1}^{N}(Q_{t(com)} - \overline{Q}_{(com)})(Q_{t(obs)} - \overline{Q}_{(obs)})}{\sqrt{\left[\sum_{t=1}^{N}(Q_{t(com)} - \overline{Q}_{(com)})^2\right]\left[\sum_{t=1}^{N}(Q_{t(obs)} - \overline{Q}_{(obs)})^2\right]}}$$

(11)

$$MSE = \frac{1}{N}\sum_{t=1}^{N}(Q_{t(obs)} - \overline{Q}_{t(com)})^2$$

(12)

$$RE = \frac{1}{N}\sum_{t=1}^{N}\frac{\left|Q_{t(com)} - Q_{t(obs)}\right|}{Q_{t(obs)}}$$

(13)

$$E = 1 - \frac{\sum_{t=1}^{T}(Q_{t(obs)} - Q_{t(com)})^2}{\sum_{t=1}^{T}(Q_{t(obs)} - \overline{Q}_{t(obs)})^2}$$

(14)

Results and Discussion

The purpose of this research was to obtain suitable discharge outputs and more accurately predict discharge at Panzhihua gauging station of JRB. Different ANN methods were applied in this research, including a feed

forward back propagation neural network, a generalized regression network, and a radial basis function network. Results with each ANN were compared with each other and to an autoregressive model. Based on established performance criteria, the study found that the feed forward back propagation model was more accurate than the others.

Application of Feed Forward Back Propagation Neural Network

It was important to have a well calibrated model to generate suitable simulations for FFBPNN. The most important part was to find an appropriate number of hidden layers, number of neurons, and the type of transform function. A literature review and experiments showed that one hidden layer provided more suitable results than many; as such, in this study, results are generated using a single hidden layer. The next step involves finding the suitable number of neurons. The model cannot provide desired outputs without finding the best fit number of neurons for the hidden layer.

In this study, a trial and error method was used to determine the best output results on the basis of least mean square error (MSE). Six different input combinations (i) Q_{t-1} (ii) Q_{t-1} and Q_{t-2} (iii) Q_{t-1} Q_{t-2} and Q_{t-3} (iv) Q_{t-1} Q_{t-2} Q_{t-3} and Q_{t-4} (v) Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} and Q_{t-5} (vi) Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} Q_{t-5} and Q_{t-6} were observed by choosing different number of neurons with single hidden layer. With trial and error, we found suitable results at specific neurons, such as 3, 4, 5, 6, 8, 10, 12, 14, 16, 18, and 20.

Flow hydrograph, as well as other performance criteria such as R, E, and MSE show the best FFBPNN results for 1 hidden layer, with 5 hidden neurons with a sigmoid function at input Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} Q_{t-5} and Q_{t-6}. Table 2 lists results for FFBPNN at each input; (Figure 6) shows the FFBPNN hydrograph.

Application of Generalized Regression Neural Network

For GRNNs, it was important to specify the spread value for each case, to produce the minimum MSE value. In this study, spread values are ranged from 0.001-1.0 and were examined using trial and error. The best results were seen for JRB when the spread values ranged from 0.1-0.2. For all inputs, GRNN resulted in better results at a spread value of 0.1 compared to other used values. Table 3 lists GRNN performance. (Figure 7) shows the GRNN hydrograph.

Radial Basis Function Network

To obtain the most precise RBFNN outcomes, trial and error was used to implement an appropriate node number and spread value for each studied case. For JRB, the best node was number 20 and the best spread value was 1.0 for the studied cases. Similarly, to obtain the most precise

outcomes of RBFs, the same approach was used: for JRB, the best node was number 20 and the best spread value was 1.0 for the studied cases. Table 4 lists RBFN performance with each input. (Figure 8) shows the RBFNN hydrograph.

Comparing Different Methods

On an individual basis, all neural networks (FFBPNN, GRNN, RBFN) led to significant results. All neural network results were also compared with the statistical auto regressive model. When comparing the validation results of FFBPNN with other the neural networks and the AR model, FFBPNN generated more significant results compared to others. The validation hydrograph for FFBPNN was the best at picking and simulating the lower peak, high peaks, and middle range values.

The GRNN validation hydrograph simulated the lower peaks and middle range values quite well compared to RBFNN and the AR model. RBFNN picked middle range values effectively, but did not pick high and lower peaks. Compared with the neural networks, the AR model did not pick the lower peak, high peak, and middle range values smoothly. In the lower peak section, the AR model showed irregular behavior in simulated results. The AR model is a statistical tool; as such, it cannot simulate a nonlinear pattern. (Figure 9) provides validation hydrographs for all models.

Table 5 lists the model performance results, including R, MSE, R.E, and E. In all cases, ANN model results are better than the AR model. The AR model resulted in values of R, MSE, R.E and E of $0.84, 14.14 \times 10^5 m^6/s^2$, 0.45, and 0.54 respectively. The FFBPNN, GRNN, and RBFNN resulted in more significant values of R, MSE, R.E, and E than the AR model. For FFBPNN, significant values for R, MSE, R.E and E were 0.99, $5.55 \times 10^5 m^6/s^2$, 0.11, respectively. For GRNN, significant values of R, MSE, R.E and E were 0.92, $5.19 \times 10^5 m^6/s^2$, 0.15 and 0.18, respectively. For RBFNN, significant values of R, MSE, R.E and E were 0.89, $7.89 \times 10^5 m^6/s^2$, 0.39 and 0.75, respectively.

TABLE 2: FFBPNN Validation MSE ($\times 10^5 m^6/s^2$), R and E values

INPUT	R	MSE	E
Q_{t-1}	0.79	8.95	0.61
Q_{t-1} and Q_{t-2}	0.87	8.94	0.76
Q_{t-1} Q_{t-2} and Q_{t-3}	0.90	7.34	0.79
Q_{t-1} Q_{t-2} Q_{t-3} and Q_{t-4}	0.92	6.08	0.82
Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} and Q_{t-5}	0.91	6.39	0.81
Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} Q_{t-5} and Q_{t-6}	0.99	5.55	0.97

TABLE 3: GRNN Validation MSE ($\times 10^5 m^6/s^2$), R and E values

INPUT	R	MSE	E
Q_{t-1}	0.77	9.03	0.58
Q_{t-1} and Q_{t-2}	0.85	8.88	0.71
Q_{t-1} Q_{t-2} and Q_{t-3}	0.90	7.22	0.79
Q_{t-1} Q_{t-2} Q_{t-3} and Q_{t-4}	0.92	5.91	0.82
Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} and Q_{t-5}	0.91	6.21	0.80
Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} Q_{t-5} and Q_{t-6}	0.91	7.11	0.81

TABLE 4: RBFNN Validation MSE ($\times 10^5 m^6/s^2$), R and E values

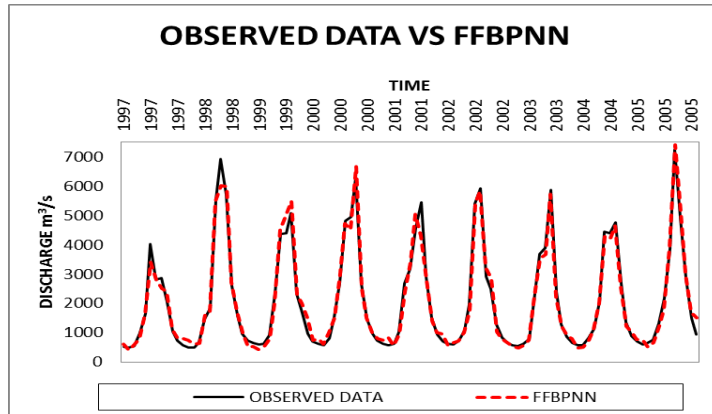| INPUT	R	MSE	E
Q_{t-1}	0.76	9.21	0.57
Q_{t-1} and Q_{t-2}	0.84	9.00	0.70
Q_{t-1} Q_{t-2} and Q_{t-3}	0.85	8.85	0.70
Q_{t-1} Q_{t-2} Q_{t-3} and Q_{t-4}	0.87	8.11	0.71
Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} and Q_{t-5}	0.88	7.94	0.74
Q_{t-1} Q_{t-2} Q_{t-3} Q_{t-4} Q_{t-5} and Q_{t-6}	0.84	9.10	0.64

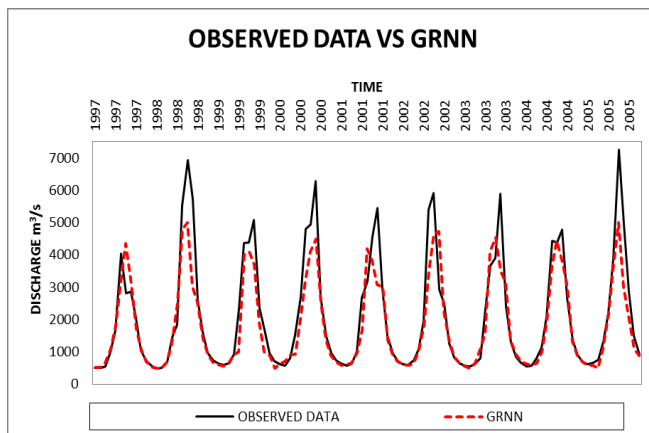FIGURE 6: Validation phase, observed and forecasted monthly flows for FFBPNN

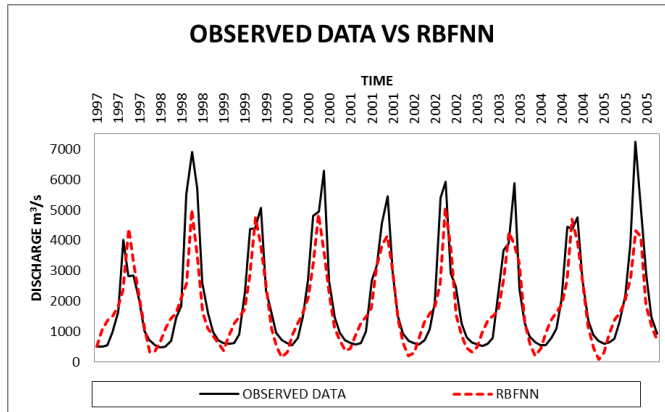FIGURE 7: Validation phase, observed and forecasted monthly flows for GRNN

FIGURE 8: Validation phase, observed and forecasted monthly flows for RBFNN

TABLE 5: ANN's and AR model Validation MSE ($\times 10^5 m^6/s^2$), R, R.E and E.

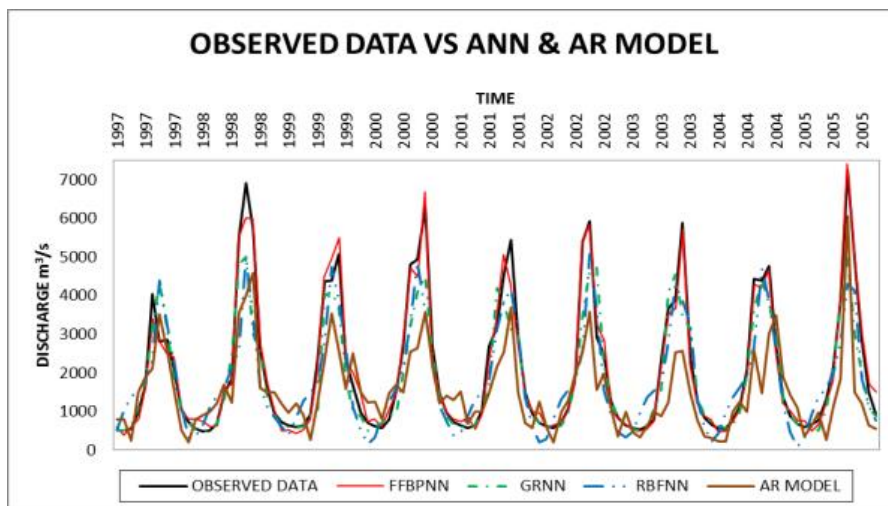| MODEL	R	MSE	R.E	E
FFBPNN	0.99	5.55	0.11	0.98
GRNN	0.92	5.91	0.15	0.81
RBFNN	0.89	7.89	0.39	0.75
AR	0.84	14.14	0.45	0.54

FIGURE 9: Validation phase, observed and forecasted monthly flows of ANN & AR models.

Conclusion

The Jinsha River basin has many water resources. Water resource fluctuations due to climate change make integrated water management vital. This study used data from Panzhihua, the main controlling hydrological station, to better understand river discharge projections in Jinsha river basin. Accurate river flow forecasts are a vital component of sustainable water

resource planning and management. Precise and well-timed predictions of high and low flow events provide information to inform deliberate decisions.

In this study, three types of artificial neural networks, FFBPNN, GRNN, and RBFNN, were investigated to develop river flow forecasts in an area of the Upper Yangtze River called JRB. This study demonstrated the feasibility of adopting the ANN's as a river flow forecasting tool, as the ANN's results were suitable and accurate for the upper Yangtze River. All ANN results were compared with each other and with other autoregressive models.

The ANN's performed better than the AR in all studied cases. Model performance was assessed using correlation coefficient (R), mean square error (MSE), relative error (R.E) and the Nash Sutcliffe model efficiency coefficient (E). Statistical model is not good enough to produce simulation for the non-linear pattern that is why all ANN's have generated better simulations in all cases. Comparing ANN techniques, in most studied cases, the FFBPNN performance was the best; GRNN performance was better than RBFNN for most cases. In conclusion, ANN's can be used to predict river flows by using the flow data from other rivers; this can be a vital tool in mitigating missing flow data records.

Acknowledgments

Author Contributions

All authors were involved in preparing the model and getting suitable outputs according to available data and study area. Muhammad Tayyab and Xiaofan Zeng wrote the research article. Jianzhong Zhou and Rana Adnan provided important data sets.

Refrences:

Alvisi, S., Mascellani, G., Franchini, M., & Bardossy, A. (2006). Water level forecasting through fuzzy logic and artificial neural network approaches. Hydrology and Earth System Sciences Discussions, 10(1), 1-17.

Besaw, L. E., Rizzo, D. M., Bierman, P. R., & Hackett, W. R. (2010). Advances in ungaued streamflow prediction using artificial neural networks. Journal of Hydrology, 386(1), 27-37.

Box, G. E. (1976). P, and Jenkins, GM,"Time Series Analysis: Forecasting and Control,". Time Series and Digital Processing.

Cloke, H., & Pappenberger, F. (2009). Ensemble flood forecasting: a review. Journal of Hydrology, 375(3), 613-626.

Dawson, C. W., & Wilby, R. (1998). An artificial neural network approach to rainfall-runoff modelling. Hydrological Sciences Journal, 43(1), 47-66.

Govindaraju, R. S. (2000). Artificial neural networks in hydrology. II: hydrologic applications. Journal of Hydrologic Engineering, 5(2), 124-137.

Hsu, K.-l., Gao, X., Sorooshian, S., & Gupta, H. V. (1997). Precipitation estimation from remotely sensed information using artificial neural networks. Journal of Applied Meteorology, 36(9), 1176-1190.

Kisi, Ö. (2004). River flow modeling using artificial neural networks. Journal of Hydrologic Engineering, 9(1), 60-63.

Kişi, Ö. (2007). Streamflow forecasting using different artificial neural network algorithms. Journal of Hydrologic Engineering, 12(5), 532-539.

LEI, X.-y., ZHANG, L.-x., & LIANG, X.-p. (2008). Study on the Forecast Model of Annual Runoff Base on An Annual Runoff Forecasting Model of BP Neural Network Based on MATLAB Toolbox [J]. Journal of China Hydrology, 1, 009.

Maier, H. R., & Dandy, G. C. (1997). Determining inputs for neural network models of multivariate time series. Computer-Aided Civil and Infrastructure Engineering, 12(5), 353-368.

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. Environmental modelling & software, 15(1), 101-124.

Mishra, S., Gupta, P., K Pandey, S., & P Shukla, J. (2014). An Efficient Approach of Artificial Neural Network in Runoff Forecasting. International Journal of Computer Applications, 92(5), 9-15.

Popescu, I., Constantinou, P., Nafornita, M., & Nafornita, I. (2004). Generalized regression neural network prediction model for indoor environment. Paper presented at the iscc.

Raman, H., & Sunilkumar, N. (1995). Multivariate modelling of water resources time series using artificial neural networks. Hydrological Sciences Journal, 40(2), 145-163.

Sahoo, G., & Ray, C. (2006). Flow forecasting for a Hawaii stream using rating curves and neural networks. Journal of hydrology, 317(1), 63-80.

Sheng, Z. (2008). BP Neural Network Principles and Matlab Simulation. Journal of Weinan Teachers University, 23(5), 65-67.

Sudheer, K., & Jain, S. (2003). Radial basis function neural network for modeling rating curves. Journal of Hydrologic Engineering, 8(3), 161-164.

Sulaiman, M., El-Shafie, A., Karim, O., & Basri, H. (2011). Improved water level forecasting performance by using optimal steepness coefficients in an artificial neural network. Water resources management, 25(10), 2525-2541.

Theodosiou, M. (2011). Disaggregation & aggregation of time series components: A hybrid forecasting approach using generalized regression neural networks and the theta method. Neurocomputing, 74(6), 896-905.

Wen, Y., Qiao, B., Li, Z., & Shao, X. (2001). Application of neural network based on MATLAB toolbox to earthquake prediction. Seismological Research of Northeast China, 17(3), 36-41.

Xiang, Y., Liang, C., Lin, Y., & Liang, Y. (2012). The Application of MATLAB Neural Network Algorithm in Short-Term Hydrological Forecasting. Paper presented at the Civil Engineering and Urban Planning 2012

Alvisi, S., Mascellani, G., Franchini, M., & Bardossy, A. (2006). Water level forecasting through fuzzy logic and artificial neural network approaches. *Hydrology and Earth System Sciences Discussions, 10*(1), 1-17.

Besaw, L. E., Rizzo, D. M., Bierman, P. R., & Hackett, W. R. (2010). Advances in ungauged streamflow prediction using artificial neural networks. *Journal of Hydrology, 386*(1), 27-37.

Box, G. E. (1976). P, and Jenkins, GM,"Time Series Analysis: Forecasting and Control,". *Time Series and Digital Processing*.

Cloke, H., & Pappenberger, F. (2009). Ensemble flood forecasting: a review. *Journal of Hydrology, 375*(3), 613-626.

Dawson, C. W., & Wilby, R. (1998). An artificial neural network approach to rainfall-runoff modelling. *Hydrological Sciences Journal, 43*(1), 47-66.

Govindaraju, R. S. (2000). Artificial neural networks in hydrology. II: hydrologic applications. *Journal of Hydrologic Engineering, 5*(2), 124-137.

Hsu, K.-l., Gao, X., Sorooshian, S., & Gupta, H. V. (1997). Precipitation estimation from remotely sensed information using artificial neural networks. *Journal of Applied Meteorology, 36*(9), 1176-1190.

Kisi, Ö. (2004). River flow modeling using artificial neural networks. *Journal of Hydrologic Engineering, 9*(1), 60-63.

Kişi, Ö. (2007). Streamflow forecasting using different artificial neural network algorithms. *Journal of Hydrologic Engineering, 12*(5), 532-539.

LEI, X.-y., ZHANG, L.-x., & LIANG, X.-p. (2008). Study on the Forecast Model of Annual Runoff Base on An Annual Runoff Forecasting Model of BP Neural Network Based on MATLAB Toolbox [J]. *Journal of China Hydrology, 1*, 009.

Maier, H. R., & Dandy, G. C. (1997). Determining inputs for neural network models of multivariate time series. *Computer-Aided Civil and Infrastructure Engineering, 12*(5), 353-368.

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. *Environmental modelling & software, 15*(1), 101-124.

Mishra, S., Gupta, P., K Pandey, S., & P Shukla, J. (2014). An Efficient Approach of Artificial Neural Network in Runoff Forecasting. *International Journal of Computer Applications, 92*(5), 9-15.

Popescu, I., Constantinou, P., Nafornita, M., & Nafornita, I. (2004). *Generalized regression neural network prediction model for indoor environment.* Paper presented at the iscc.

Raman, H., & Sunilkumar, N. (1995). Multivariate modelling of water resources time series using artificial neural networks. *Hydrological Sciences Journal, 40*(2), 145-163.

Sahoo, G., & Ray, C. (2006). Flow forecasting for a Hawaii stream using rating curves and neural networks. *Journal of hydrology, 317*(1), 63-80.

Sheng, Z. (2008). BP Neural Network Principles and Matlab Simulation. *Journal of Weinan Teachers University, 23*(5), 65-67.

Sudheer, K., & Jain, S. (2003). Radial basis function neural network for modeling rating curves. *Journal of Hydrologic Engineering, 8*(3), 161-164.

Sulaiman, M., El-Shafie, A., Karim, O., & Basri, H. (2011). Improved water level forecasting performance by using optimal steepness coefficients in an artificial neural network. *Water resources management, 25*(10), 2525-2541.

Theodosiou, M. (2011). Disaggregation & aggregation of time series components: A hybrid forecasting approach using generalized regression neural networks and the theta method. *Neurocomputing, 74*(6), 896-905.

Wen, Y., Qiao, B., Li, Z., & Shao, X. (2001). Application of neural network based on MATLAB toolbox to earthquake prediction. *Seismological Research of Northeast China, 17*(3), 36-41.

Xiang, Y., Liang, C., Lin, Y., & Liang, Y. (2012). *The Application of MATLAB Neural Network Algorithm in Short-Term Hydrological Forecasting.* Paper presented at the Civil Engineering and Urban Planning 2012.