Application of Artificial Neural Network Model to Forecast Runoff for Waikato River Catchment

MR. ZHIMIN ZHENG[1], DR. BABAR MAHMOOD[2]

[1] Student, [2] Staff at Unitec Institute of Technology, Auckland, New Zealand

ABSTRACT

As we know that over the past decade or so the Artificial Intelligence (AI) techniques (e.g. ANN - Artificial Neural Network & FIS - Fuzzy Interference System) have been used as an alternative modelling tools in water resources management studies. Runoff generated from a catchment as a result of a rainfall event is a very complex hydrological process as it depends on climatological (i.e. rainfall depth, duration and intensity, etc.) and geographical (i.e. soil type, infiltration rate, evapotranspiration, etc.) factors of the catchment. The present study is about the application of Artificial Neural Network (ANN) model to forecast runoff from the Waikato River catchment areas of New Zealand. Similar to other modelling approaches, successful application of ANN is also dependant on the selection of appropriate input factors. To investigate this, the study applied three different approaches for the selection of appropriate input vectors to be used for the ANN model. The study demonstrated that ANN can successfully forecast the runoff generated from a catchment using antecedent rainfall and runoff data series identified on the basis of cross-correlation and auto-correlation coefficients. The ANN models developed using three approaches (i.e. *sequential*, *pruned* and *non-sequential* time series) were able to predict runoff generated from the Waikato River catchment using antecedent rainfall/runoff data. The study showed that the ANN models were sensitive to the selection of appropriate input vector. The ANN model developed using the *non-sequence* approach performed well, and gave the highest R^2 and NSE values (i.e. 97-98 %) during the validation and testing phases of this modelling exercise.

Keywords
Artificial Neural Network (ANN), Artificial Intelligence (AI), Catchment runoff modelling.

PRESENTER PROFILE

The presenter is an Environmental Engineer by profession, and currently working as a Senior Lecturer in the Civil Engineering Department of Unitec. The presenter has 27 years of experience working with water industry (including working at World Bank's international projects & supervising local stormwater & wastewater projects). For 10 years, the presenter has also performed the duties of a Programme Leader for Environmental Engineering programmes at Unitec.

1 INTRODUCTION

Reliable and accurate estimation of runoff generated from a catchment/watershed as a result of the rainfall incident is an important area of research in hydrology. Therefore, computer models (for catchment hydrology) and AI techniques are being used extensively for more accurate runoff quantity estimation. The rainfall-runoff relationship is considered to be one of the most complex hydrological processes to be modelled because of the involvement of a number of variables in the modelling process and enormous spatial and temporal variability of watershed characteristics. To model this non-linear complex process of rainfall-runoff, numerous hydrological models have been proposed. Broadly speaking, these models can be divided into two categories i.e. knowledge driven (physics-based) models, and data driven (system theoretic) models (Dooge, 1977). Knowledge-driven models make use of different mathematical equations, which may be either empirical or partial differential equations to model each and every physical process (of the hydrological cycle) including evapotranspiration, infiltration, surface and groundwater flow etc. Examples of these models include Standford Watershed Model – SWM (Linsley & Crawford, 1960), Tank Model (Sugawara, 1995), the Soil Moisture Accounting and Routing - SMAR Model (O'Connell et al., 1970; Tan & O'Connor, 1996), Sacramento Model (Burnash et al., 1973), Xinanjiang Model (Ren-Jun, 1992). Although the results of these models are considered satisfactory, but these models are data hungry and therefore have a limited use so far.

The black-box data driven models, on the other hand, requires less data (as compared to physics-based models) to model this complex non-linear rainfall-runoff relationship (without considering various physical processes of the hydrological cycle). Examples of the conventional black-box data driven models includes the Auto Regressive Integrated Moving Average (ARIMA) or Seasonal ARIMA with exogenous input (SAIMAX) and Multiple Linear Regression (MLR). The ARIMA, SAIMAX and MLR models are a form of regression analysis models that use time series data to predict future trends. The details of these models can be found in many statistics books such as Rawlings at al. (1998); Wang et al. (2003), and Kleinbaum et al. (2013). These models have been successfully used in many previous hydrological forecasting studies (Sala et al., 1980; Cleaveland & Stahle, 1989; Nourani et al., 2011; Zhang et al., 2011; Adamowski et al., 2012). The ARIMA, SAIMAX and MLR models are simple to develop and use. However, in these models, the future value of a variable is assumed to be a linear function of several past observations and random errors. It is well known that these linear models are described by a linear equation which is of the form $y = ax + b$ where *a* and *b* are the constants and y and x are the dependant and independent variables, respectively. Therefore, it would be inappropriate to use linear functions or relationships if the underlying process under investigation is non-linear i.e. rainfall-runoff relationship.

With the advent in knowledge, a special type of black-box driven data models called ANN emerged and received great attention. The ANN was originally developed as a model of information storage and computing by neuronal processes found in nature (McCulloch & Pitts, 1943). Details of emergent computational properties of the ANN are also discussed in Hopfield (1982). The learning aspects of neural networks are described in (Rumelhart et al., 1986).

One of the features of the ANN technique/method is that it provides a computational or mathematical technique, which is powerful for modelling systems where the explicit form of the relationship between the variables involved is unknown (Fausett, 1994).

The ANN technique has been successfully employed for resolving various problems in numerous branches of science and engineering. In the field of hydrology, ANN method was used for the first time for forecasting rainfall by French et al. (1992), also Shamseldin (1997) pioneered ANN application in modelling the rainfall-runoff relationship. The ANN based rainfall-runoff models have been developed using the observed input and output rainfall and runoff data, respectively without considering the detailed understanding of the complex physical laws governing the rainfall-runoff processes. The advantages of ANN models over the other modelling approaches include less data requirements, and short development time to model. Furthermore, no great expertise is required to develop and apply ANN. Further, ANN has been successfully applied in many previous hydrological studies such as Hsu et al. (1995); Dawson and Wilby (1998); Sajikumar and Thandaveswara (1999); Tokar and Johnson (1999); Sudheer et al. (2002); Lallahem and Mania (2003); Jain et al. (2004); Senthil et al. (2005); Antar et al. (2006); and Nourani et al. (2009, 2011). A comprehensive review of ANN in hydrology can be found in the ASCE Task Committee on Application of the ANNs in hydrology (Committee, 2000a, 2000b).

To the author's knowledge, a limited work has been undertaken in terms of application of ANN to model the rainfall-runoff relationship in New Zealand. From water resources' management point of view, it is important to correctly estimate runoff /flow rate generated from a catchment area. The main goal of this study was to develop an ANN model to predict runoff using antecedent rainfall and runoff data collected from Wangamarino control structure located in Waikato river catchment areas. Therefore, this paper provides a brief introduction to ANN method and then presents the application of this intelligent model (i.e. ANN) having self-learning ability to predict runoff using past rainfall-runoff data for the studied catchments. The specific objectives of this study were to: (i) learn the fundamentals of the Neural Network Tool box of Matlab software; (ii) investigate different approaches available for the purpose of determining optimum input vectors for the ANN model, (iii) check the sensitivity of ANN models performance to the selection of appropriate input vectors, and (iv) identify the best ANN model by comparing the flow duration curves.

2 METHODOLOGY

2.1 ANN MODEL

The fundamentals of ANN can be found in Priddy and Keller (2005); Graupe (2007). However, a brief introduction of ANN is given here from reader's point of view. The ANN is a computing paradigm that is inspired by the working of the human brain and nervous system. The neurons in the biological neural networks receive information from the senses situated at various locations in the network. These neurons are linked to each other by a connection called synapse. These neurons produce a proper response to the information received by releasing chemicals which cause a synapse to conduct an electric current. The neuron which receives information can either pass this information to the other neuron in the network or neglect its input. This causes damping of the information. The ANN can be formally defined as follows *"A neural network is a parallel, distributed information processing structure consisting of processing elements (which can possess a local memory and can carry out localized information processing operations) interconnected via unidirectional signal channels called branches ('fans out') into as many collateral connections as desired; each carries the same signal – the processing element output signal. The processing element output signal can be of any mathematical type desired. The information processing that goes in within each processing element can be defined arbitrarily with the restriction that it must be completely local; that is, it must depend only on the current values of the input signals arriving at the processing element via impinging connections and on values stored in the processing element's local memory"* (Hecht-Nielsen, 1988).

Neurons in the ANN are denoted to as the computational elements and also considered as the basic-building blocks of ANN. Neurons in the ANN are arranged in layers. Neurons in each layer are linked to neurons in the next layers through connections called weights. The activation state of the neurons of a network is the state of the system at a certain point in time. It is the pattern of connectivity that governs how a network will respond to an arbitrary input. The propagation rules define the way net input to a neuron is calculated from numerous outputs of adjacent neurons. Typically, net input is the weighted sum of the inputs to the neurons. The activation rule also called transfer function calculates the new activation value of a neuron based on the net input. Based on the data presented to the ANN, an ANN tries to learn the relationships that are contained within the data by adjusting its connection weights and biases. The algorithm used to optimize these weights and biases is called training or learning algorithm. These training algorithms may be supervised learning or unsupervised learning. Different parameters of the ANN are varied based on the type of the ANN used Beale et al. (2015).

2.1.1 FEED-FORWARD NEURAL NETWORK (FFNN)

There are numerous types of Artificial Neural Networks (ANNs) such as FFNN, Generalized Feed-Forward Neural Network (GFNN), Radial Basis Function Neural Network (RBFNN), Adaptive Neuro-Fuzzy Inference System (ANFIS), but the FFNN with back-propagation is considered to be the most widely used neural network (Principe et al., 2000; Mutlu et al., 2008; Eluyode & Akomolafe, 2013; Amirhossein et al., 2015), and therefore FFNN was used in this study. FFNN is also called as Multilayer Perceptron Neural Network (MLPNN). The FFNN consists of a number of neurons organized in numerous layers. Normally, it is comprised of three layers i.e. an input layer, a hidden layer and an output layer. The number of neurons in each layer varies. Each neuron in all layers is linked to the neuron in the next layer through the connections called weights as shown in Figure 1. The symbols $\mathbf{x_1, x_2, \ldots x_n}$ refer to the external inputs to the neural network, which is the rainfall data that was used in this study. The $\mathbf{w^H}_{\mathbf{12}}$ refers to the connection weight between the first neuron in the hidden layer and the second neuron in the input layer. Likewise, $\mathbf{w^O}_{\mathbf{12}}$ in Figure 1 represents the connection weight between first neuron in the output layer and the second neuron in the hidden layer. Similar weight notation is assigned to each connection weight in Figure 1. Each neuron obtains an array of inputs and yields an output. The output of a neuron in the input layer will be input for the neuron in the next hidden layer. Similarly, the output of the neuron in the hidden layer will be input for the next output layer. Each neuron in all layers processes its input by a function known as the neuron transfer function. The neurons in the input layer have a connection with the neuron in the hidden layer while the neuron in the output layer is also only connected to the neuron in the hidden layer. There is no direct connection between the neuron in the input layer with those neurons in the output layer. The neurons in the input layer just perform an identity map between its input and output activity as follows (Principe et al., 2000):

$$f(x_i, w_i) = x_i \qquad (1)$$

The inputs to the neurons in the hidden layer (each of the output of neuron in the input layer) and the output layer (each of the output of the neuron in the hidden layer) are multiplied by their corresponding weight and the bias added according to the following equation (Principe et al., 2000):

$$Y_{net} = \sum_{i=1}^{N} Y_i w_i + w_0 \qquad (2)$$

Where N is the total number of neurons in the previous layer, Yi is the output of the neuron in the previous layer and w_o is the bias value added to the neuron.

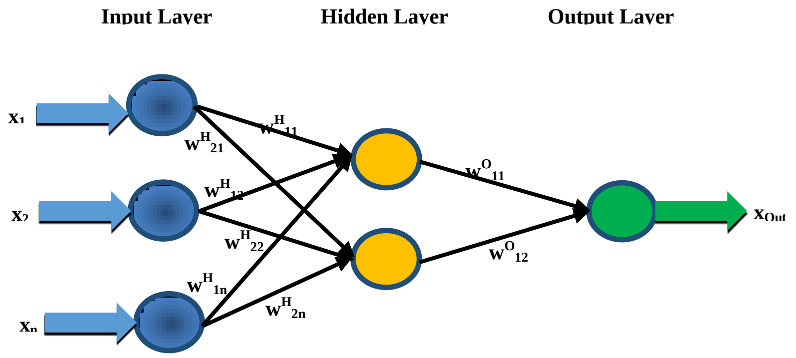

Figure 1. Feed-Forward Neural Network (FFNN).

2.2 STUDY AREA

The Waikato catchment is the largest catchment in the north island of New Zealand with a total area of about 14500 km^2. The Waikato River is about 425 Km in length and it is the longest river in New Zealand. It originates from the Lake Taupo, to discharge into the Tasman Sea at Port Waikato, which is approximately 30 km south of Auckland. The Waikato River is an important source of water supply, recreation and hydro-power generation in New Zealand (Khan et al., 2014). The existence of eight dams and nine hydroelectric stations on the river makes it economically vital for the country.

2.3 INPUT DATA

Daily rainfall and discharge data of gauging stations located in Waikato River catchment was obtained from the Environment Waikato Department, New

Zealand. Ten years daily rainfall-runoff data starting from 24th May 2002 of Waikato River catchment was used in this study. Data from sixteen rainfall stations was used to calculate the average depth of rainfall in the catchment. The arithmetic mean method was employed to compute the average in the present study because of its simplicity. The concurrent discharge data used in the study was the daily averaged data measured at the Whangamarino Control Structure on the Waikato River. The station was selected as it covers a maximum catchment area of 13500 km^2 of the Waikato River.

2.4.1 SELECTION OF NEURAL NETWORK TYPE

The Matlab neural network tool was used in this study to develop a runoff forecasting model. The neural network type used in this study was the **FFNN** with the back-propagation algorithm as it is considered to be the most common neural network used in hydrological applications (Committee, 2000a, 2002b; Mutlu et al., 2008; Eluyode & Akomolafe, 2013; Amirhossein et al., 2015), also stated earlier.

2.4.2 SELECTION OF TRANSFER FUNCTION

One of the important parameters of ANN is the selection of the transfer function which controls the generation of output in a neuron. The transfer function should be differentiable as most multilayer neural networks are based on optimization methods that use first and second order derivatives Caudill (1990). One of the most common activation functions for back-propagation neural networks is sigmoid nonlinearity (Haykin & Lippmann, 1994). The present study therefore used the default hyperbolic tangent function for the neurons of the hidden layer. This transfer function is continuous, differentiable and monotonically increasing and is given by the following equation (Beale et al., 2015):

$$f(x) = \frac{1-e^x}{1+e^x} \tag{3}$$

This transfer function takes the input, which can be in the range of plus infinity to minus infinity, and converts the output into the range -1 to 1. The default linear transfer function was used for the neurons of the output layer which yields the output into the range minus infinity to plus infinity.

2.4.3 SELECTION OF HIDDEN LAYER NEURONS

Performance of ANN model is also very sensitive to the selection of a number of neurons in the hidden layer. The selection of the number of hidden neurons is important in obtaining reliable results. The number of neurons in the hidden layer can be from one neuron to infinity. The selection of a small number of neurons in the hidden layer may decrease performance of the network as the network will have few degrees of freedom. However, the use of too many hidden neurons may lead to over-fitting. Over fitting is the case when the model gives good results during training but is unable to re-produce the similar results during testing. Over fitting means that the ANN model learns to reproduce the noise of the data or the data pairs itself rather than trends in the data set as a whole. Over fitting can be avoided by changing the number of neurons in the hidden layer. In the present study, the selection of an appropriate number of hidden neurons was done by a trial and error procedure in a similar manner as reported in previous studies (e.g. Nourani et al., 2011; Shamseldin, 1997; Wang & Ding, 2003; Tiwari & Chatterjee, 2010; Adamowski & Sun, 2010). The trial and error procedure involves training the network and evaluating its performance over a range of different increasing values of hidden layer neurons in order to obtain near maximum efficiency with the smallest number of neurons as necessary (Hammerstrom, 1993). The number of neurons in the hidden layer was varied in the range of 5 to 40 to find the best results by the trial and error method in this study. This range was selected in order to cover a wide range of neurons in the hidden layer.

2.4.4 SELECTION OF INITIAL WEIGHTS AND STOPPING CRITERIA

The learning of the neural networks from the observed data also is very much dependent on the selection of initial weights and also on the stopping criteria of the learning (Beale et al., 2015). The initial weights were randomly generated between -1 to 1 in the software ANN tool box while the cross validation method was used as the criteria for the early stopping of the training. This technique is based on dividing the data into three subsets, namely, training, validation and testing. The training sub-set of the data was used for the calculation of the gradient and updating the network weights and biases. The second validation data sub-set was used to determine the stopping criteria, and training was stopped when the mean square error reached a minimum in the validation phase. The network weights and biases were frozen at that point. The third data sub-set was the testing data set which was used to verify the network performance.

2.4.5 SELECTION OF LEARNING ALGORITHM

The purpose of the learning function/rule is to modify the variable connection weights on the inputs of each processing element (neuron) according to some neural based algorithm. Many learning algorithms are in common use such as Hebb’s rule, Hopfield law, the delta rule, the gradient descent rule, Kohonen’s learning law and the Levenberg-Marquardt rule. The details of these learning algorithms can be found in (Karayiannis & Venetsanopoulos, 2013; Anthony & Bartlett, 2009; Yegnanarayana, 2009). A back-propagation algorithm, which is essentially a gradient-descent technique that minimizes the network error function (Rumelhart et al., 1986; Haykin & Lippmann, 1994). The gradient descent is the process of making changes to weights and biases, where the changes are proportional to the derivatives of network error with respect to those weights and biases. The Levenberg-Marquardt Algorithm (LMA) available in the neural network tool box of MATLAB (Beale et al., 2015) was selected as the training algorithm in this study because it is quicker and more reliable than any other back-propagation method (Jeong & Kim, 2005).

***Note**: Selection of input vectors is covered in section 3.1 (later).*

2.5 DEVELOPMENT OF FLOW DURATION CURVES (FDC)

In order to evaluate the ability of ANN models (i.e. validation of the developed models) to capture low, medium and high flow regimes of the observed hydrographs, FDC were prepared. For preparation of the FDC, the discharges were sorted from largest to smallest and then ranks were assigned to each value of discharge. The rank one was assigned to the maximum discharge while the last rank was given to the minimum discharge in the record. The exceedance probability of all the discharges was then calculated by the following most common formula (Klingeman, 2005):

$$P = \frac{M}{n+1} \times 100 \qquad (4)$$

Where P is the exceedance probability, M is the rank, from highest to lowest and n is the total number of records under consideration. A graph was then prepared with exceedance probability versus discharge which is known as the flow duration curve. Further details on how to prepare FDC's can be found in (Klingeman, 2005).

2.6 PERFORMANCE PARAMETERS

The performance of the ANN model using different inputs was also evaluated in terms of statistical measurements (using the following three statistical measures).

Correlation Coefficient (R^2): The correlation coefficient is used in statistics in order to determine the strength of relationship between two variables (actual and predicted values). The formula to calculate the (R^2) is given by the following equation:

$$R = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}} \qquad (5)$$

Where X_i and Y_i are actual and computed values respectively while $\bar{X}$ and $\bar{Y}$ are the average values. The R^2 is called as coefficient of determination and will be used in this study. This R^2 value represents fraction of the variation in one variable that may be explained by the other variable. Its value varies between +1 to -1 with -1, 0 and +1 values showing negative relation, no relation and positive relation respectively between two variables.

Root Mean Square Error (RMSE): The RMSE is a common measure of the difference between the values predicted by a model and the values actually observed. The RMSE values ranges between 0 and ∞ with 0 values showing the perfect match and ∞ value shows no match between predicted and observed values. The formula to calculate RMSE is given by the following equation:

$$RMSE = \sqrt{\frac{1}{n}\left(\sum_{i=1}^{n}(X_i - Y_i)^2\right)} \qquad (6)$$

Where X_i and Y_i are actual and computed discharges (m^3/s), respectively, and n is the total number of observations. As RMSE is measure of error calculated between actual and computed discharges, so its unit will be m^3/s in this study.

Nash-Sutcliffe Efficiency (NSE): NSE criteria was also used (Nash & Sutcliffe, 1970) in order to evaluate the performance of a developed model. The Nash-Sutcliffe Efficiency range from -∞ to 1 with value 1 corresponds to a perfect match between model and actual values. The NSE value equal to zero indicates that the model predictions are as accurate as the mean of the observed data, while an NSE value less than zero shows that the observed value mean is a better predictor than the model values mean. The closer the model efficiency is to 1, the more accurate the model is. The formula to calculate the NSE is given by the following equation:

$$NSE = 1 - \frac{\sum_{i=1}^{n}(X_i - Y_i)^2}{\sum_{i=1}^{n}(X_i - \overline{Y_i})^2} \qquad (8)$$

Where X_i and Y_i are actual and computed values respectively while and $\overline{Y_i}$ are the mean of the computed values by the model.

3 RESULTS AND DISCUSSION

This study was aimed to develop a neural network model to forecast the runoff using antecedent rainfall and or runoff data.

3.1 SELECTION OF INPUT VECTOR

The performance of rainfall-runoff data driven models including FFNN type of ANN is very much dependent on the selection of appropriate input vector. Two common approaches for input selection of data driven models were found in literature (Talei et al., 2010). These two common approaches are *sequential* and *pruned* time series approaches.

3.1.1 APPROACH 1 - *SEQUENTIAL* TIME SERIES

The first approach was the selection of input vector comprising *sequential* time series data, which starts from containing only 1-day lagged time series data in the input vector, then modifying the external input vector by successively adding one more lagged time series into input vector and this continues up to a specific lag time. This approach has been applied in many studies (Furundzic, 1998; Tokar & Markus, 2000; Riad et al., 2004; Chua et al., 2008; Moosavi et al., 2013). This specific lag time can be determined either by trial and error or may be selected from present time to the time where the lagged/antecedent rainfall is most correlated with the observed discharge/runoff. This also depends on the time of concentration of the watershed, as Talei et al. (2010) denoted this approach as the *sequential* time series approach, and the same notation was used in this study. Ten input vectors i.e. I_1, I_2, …. I_9 and I_{10} are presented below to show an example of the input vector selection (also stated above in this section):

I_1 = r(t)
I_2 = r(t), r(t-1)
I_3 = r (t),r(t-1), r(t-2)
…..
I_{10} = r(t), r(t-1), r(t-2), r (t-3), r(t-4), r (t-5), r(t-6), r (t-7), r(t-8), r (t-9)

The first input vector contains only one day of antecedent rainfall data in the input vector, the second vector contains lagged 1 and lagged 2-day rainfall, the third input vector contains lagged 1, lagged 2 and lagged 3-day rainfall and so on to forecast the runoff. The first input vector I_1 contains only one variable which is the lagged 1-day rainfall data. The input vector I_2 contains two variables of lagged 1 and lagged 2-day rainfall data and so on.

3.1.2 APPROACH 2 - *PRUNED* TIME SERIES

The second approach was the selection of variables of the input vector around the most correlated lagged time series data. This approach was represented as *pruned* time series approach in this study as suggested by Talei et al. (2010. This approach has also been successfully applied in many studies (for example, Nayak et al., 2005a, 2005b, 2007). In order to identify appropriate inputs for the neural network with this second approach, a cross-correlation was conducted between the observed rainfall and runoff. Cross-correlation indicates the statistical dependence of two variables, as it is a measure of relationship between two variables data sets. The calculated value of the correlation coefficient explains the exactness between the predicted and actual values. Its value always lies between -1 to +1. If the value of the correlation coefficient is positive, it indicates a similar and identical relation between the two values.

Whereas negative values indicate the dissimilarity between the two values. However, it was clear from cross-correlation analysis that the lagged three-day rainfall had a maximum correlation value of 0.279 with the observed runoff while the lagged two-day rainfall has a correlation of about 0.275. Likewise, lagged five, six and seven days' correlation values were placed at third, fourth and fifth places, respectively. Based on the cross-correlation analysis results (not shown here), the following eight input vectors were established by including only the lagged rainfall data series which only have high correlation values:
I_1 = r(t-3)
I_2 = r(t-4)
I_3 = r(t-5)
I_4 = r (t-6)
I_5 = r (t-3), r(t-4)
I_6 = r (t-3), r(t-4), r (t-5)
I_7 = r(t-2), r (t-3), r(t-4), r (t-5)
I_8 = r(t-2), r (t-3), r(t-4), r (t-5), r(t-6)

With the *pruned* time series approach, the input vectors I_1, I_2 and I_3 contain only the three, four and five days lagged rainfall data series, respectively. These are the lagged data series which were found to have maximum correlation coefficient value with the observed runoff at the current time. Likewise, I_5, I_6, I_7 and I_8 input vectors were formed by adding different lagged time rainfall data series containing the highest correlation coefficient values.

3.1.3 APPROACH 3 - *NON-SEQUENTIAL* TIME SERIES

Another method was preseneted by Sudheer et al. (2002) for the selection of the external input vector for ANN models on the basis of cross-correlation and auto-correlation properties of the data series under consideration. This approach was denoted as the *non-sequential* time series approach in this study. Auto-correlation is the cross-correlation of discharge with its lagged values. The results of the auto-correlation analysis of discharge showed that lagged one-day discharge has the maximum relation with the current discharge. The following five input vectors were formed based on the cross-correlation and auto-correlation analysis results (i.e. by including the data series which have high values only):

I_1 = r(t-3), Q(t-1)
I_2 = r(t-3), r (t-4), Q(t-1)
I_3 = r(t-3), Q(t-1), Q(t-2)
I_4 = r(t-3), r (t-4), Q(t-1), Q(t-2)
I_5 = r(t-3), r (t-4), r (t-5), Q(t-1), Q(t-2)

The input vector one I_1 contains the rainfall and runoff data series which have high values of cross & auto correlations, respectively. Similarly, input vector I_2 contains two rainfall data series having high correlation values and one runoff data series. The input vectors $I_{3,}$ I_4 and I_5 were also formed using a similar method (as stated above in this current section).

3.2 ANN MODEL DEVELOPMENT

3.2.1 DATA DIVISION

The ANN model was developed using average daily rainfall and flow data. First, all data was divided into three parts: training, validation and testing. Different studies used different percentages for data division into training, validation and testing. The most common data division that was used by others (Moosavi et al., 2013; Lohani et al., 2012; Chang and Hong, 2012) was 70% (for training purposes) and 15% each for validation and testing. Therefore, the default setting of the ANN tool box of 70% was used for training and 15% was used for validation and testing (each) in this study. The Matlab function divide-block was used for this purpose of data division.

3.2.2 APPROACH 1 - *SEQUENTIAL* TIME SERIES RESULTS

The results of the best developed models using approach 1, 2, and 3 (i.e. *sequential*, *pruned*, and *non-sequential* time series approaches, respectively) are presented in Table 1. The R^2 (%), RMSE (m^3/s), and NSE (%) values for the input vector 1 (I_1) and input vector 10 (I_{10}) during training, validation, and testing are also given in Table 1. It is clear from Table 1 that a minimum value of R^2 i.e. 5% was obtained with I_1 as compared to I_{10} (which contained the lagged rainfall data from 1 to 10 days) that yields a maximum value of 36% for both R^2 and NSE. The RMSE value reduced from 195 (for I_1) to 161 (m^3/s) for I_{10}. The RMSE values corresponds to the error between the actual and computed discharges. It was obvious from the data that I_{10} has the minimum RMSE values among the ten input vectors tested for the *sequential* time series approach. On the basis of all these three figures, it was revealed that as more and more lagged rainfall data was added to the input vector, the accuracy of the models continued to increasing in terms of higher values of R^2 (%) and NSE (%) and lower values of RMSE. The ANN model with I_{10} was considered to be the best model with the *sequential* time series approach.

Furthermore, the observed and the predicted hydrographs for the least performing model I_1 and best model I_{10} with approach one (*sequential* time series approach) are presented in Figure 2. It is apparent from Figure 2 that the predicted discharge with I_1 was only able to track the average flows of the observed hydrographs. It was unable to trail the high and low flow trails of the observed hydrographs. Whereas, Figure 3 showed the observed and predicted hydrographs for I_{10}. It can be seen from Figure 3 that the predicted hydrograph was able to capture low and high flows features of the observed hydrograph better than I_1.

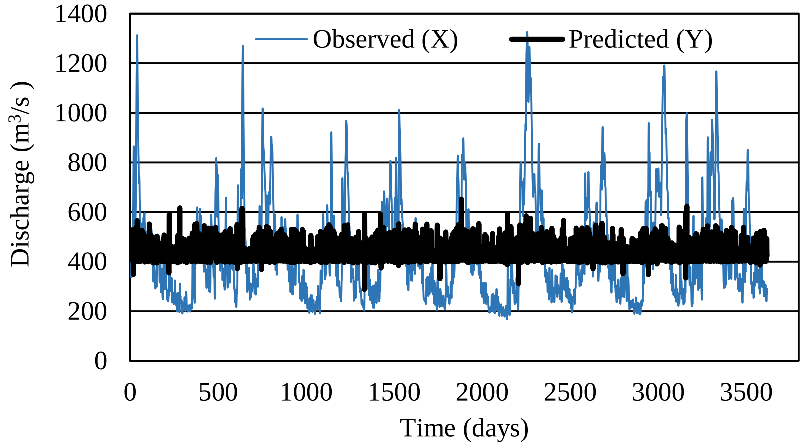

Figure 2. The performance of the ANN model (i.e. observed vs predicted flow rates) for I_1 with sequential time series approach.

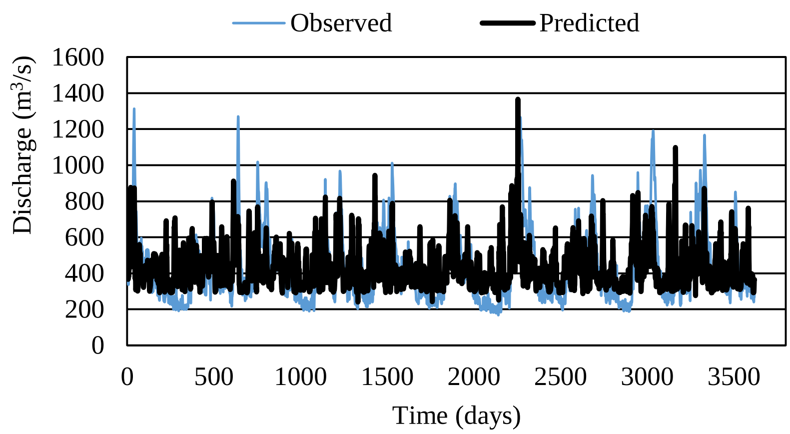

Figure 3. The performance of ANN model (i.e. observed vs predicted) for I_{10} with sequential time series approach.

3.2.3 APPROACH 2 - *PRUNED* TIME SERIES RESULTS

The ANN models using eight input vectors selected on the basis of correlation analysis were developed using a similar procedure stated above in section 3.1. The results of the developed models in terms of performance parameters of R^2 (%), NSE (%), and RMSE are presented in Table 1. It is apparent from Table 1 that input vector (I_1) which contains lagged three-day rainfall data series as input and present day discharge as output yields the R^2 and NSE values ranging between 8 and 11% (for the three parts i.e. training, validation, and testing). The RMSE values ranged between 194 and 215 m^3/s was found for the I_1 (for the three parts). However, results showed that for the I_8 i.e. input vector 8, the R^2 and NSE values were increased ranging between 20 and 29% (for the training, validation and testing of ANN models), and RMSE decreased to 169 m^3/s (Table

1), as the number of lagged rainfall data series was added in the input vector. The best performance was found with I_8 containing lagged two, three, four, five and six days' rainfall data series in the input vector.

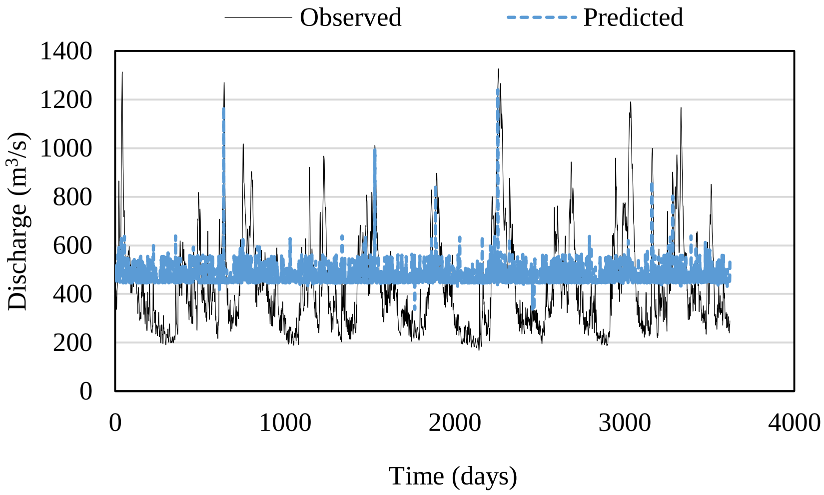

Figure 4: The performance of ANN model (i.e. observed vs predicted daily flow rates) for I_1 with pruned time series approach.

In other words, R^2 and NSE (%) values during training, validation and testing phases were increased from I_1 to I_8. The RMSE was found to be smaller with I_8 as compared with I_1 during the training, validation and testing phases (refer to **Table 1**). Similar to *sequential* time series approach, the hydrographs of the observed and predicted discharges for the least performing model with I_1 and the best performing model with I_8 are presented in Figures 4 & 5. The performances of the predicted discharges of the ANN model developed with I_8 was found to be better as compared to I_1. The predicted hydrograph by I_1 was not able to track the low and high flow trails of the observed hydrograph (Figure 4).

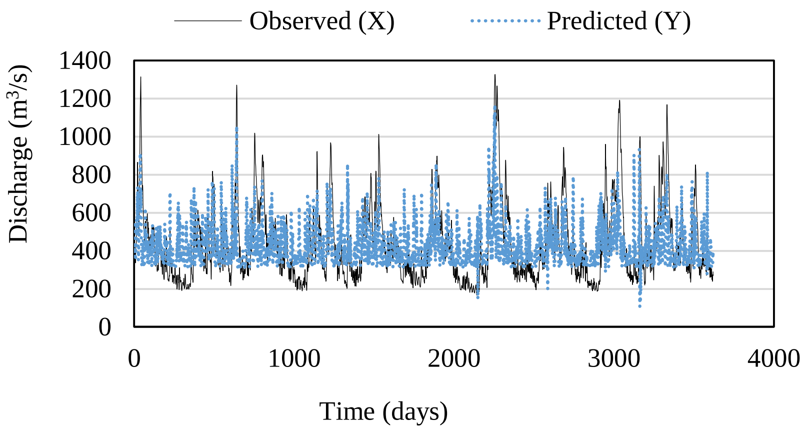

Figure 5: The performance of ANN model (i.e. observed vs predicted daily flows) for I_8 with pruned time series approach.

3.2.4 APPROACH 3 - *NON-SEQUENTIAL* TIME SERIES RESULTS

The values of performance parameters of R^2, NSE, and RMSE for all five input vectors during training, validation and testing (using input vectors identified on the basis of the *non-sequential* time series approach) are presented in Table 1.

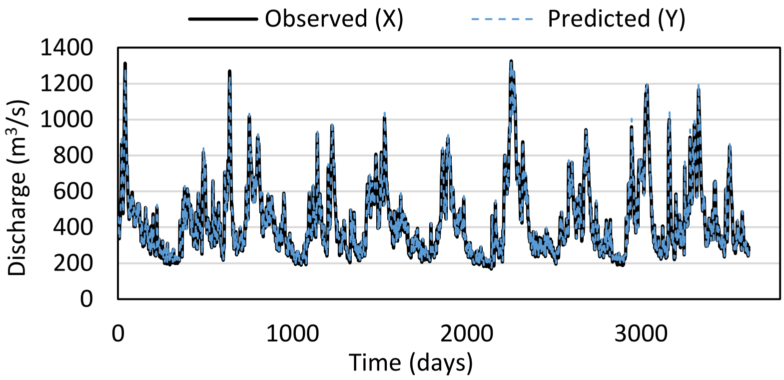

Figure 6: The performance of the ANN model for I_1 with non-sequential approach.

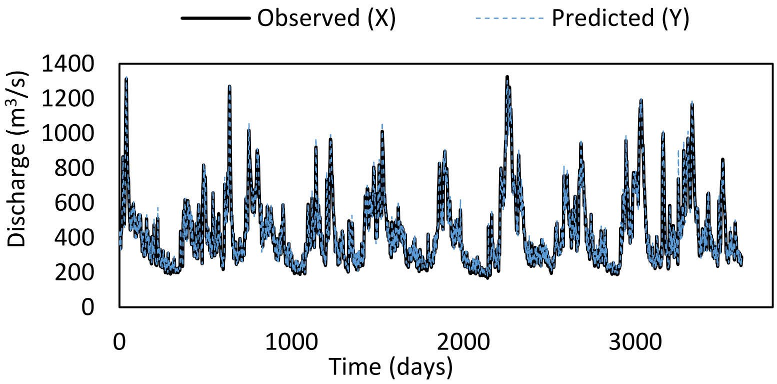

Figure 7: The performance of the ANN model developed using I_5 with non-sequential approach.

The ANN models developed using input vectors identified on the basis of cross and auto correlations analysis was found to produce significantly better results with all five input vectors relative to input vectors tested with the *sequential* and *pruned* time series approaches. The performance of the ANN models (in terms of R^2 & NSE values) was improved slightly (refer to Table 1) from I_1 to I_2 and then to I_3 and thereafter it remained almost constant. It can be seen from Table 1 that the input vector I_5 performed better (highest values of R^2, NSE and lowest values of RMSE) than I_1. It can be seen from Figures 6 & 7 that predicted discharges yielded by the ANN models developed using I_1 and I_5 were able to successfully capture the lower, medium and high flow features of the observed hydrographs.

3.2.5 PERFORMANCE COMPARISON FOR ANN MODELS

In summary, it is evident from Table 1 that overall the ANN models developed using the input vectors identified in the *non-sequential* time series approach outperformed as compared to the ANN models developed with the *sequential* and *pruned* time series approaches. The ANN model with the *non-sequential* time series approach yielded the average R^2 and NSE value of about 96% as compared to average value of 25% and 35% with the *pruned* and *sequential* time series approaches, respectively (for both R^2 and NSE). Similarly, the error in terms of RMSE (for testing phase) was found 37 m^3/s for the *non-sequential* time series approach as compared to 180 and 162 m^3/s for the *pruned* and *sequential* time series approaches (Table 1).

Table 1: Performance parameters (i.e. R^2, RMSE and NSE) of ANN models (with different input vectors) for three approaches (i.e. sequential, pruned, and non-sequential time series, respectively).

Approach	Input Vectors	Training			Validation			Testing		
		R^2 (%)	**RMSE (m^3/s)**	**NSE (%)**	**R^2 (%)**	**RMSE (m^3/s)**	**NSE (%)**	**R^2 (%)**	**RMSE (m^3/s)**	**NSE (%)**
1	I_1	5	195	5	10	220	8	6	195	5
	I_{10}	36	161	36	39	179	39	35	162	35
2	I_1	11	197	9	10	215	7	8	194	8
	I_8	29	169	29	27	199	26	22	180	20
3	I_1	97	35	97	97	38	97	95	43	95
	I_2	97	35	97	97	37	97	97	37	97
	I_3	98	31	98	98	32	98	95	43	95
	I_4	98	31	98	98	32	98	97	36	97
	I_5	98	31	98	98	32	98	97	37	97

3.3 FLOW DURATION CURVES COMPARISON

Finally, FDC's for the best models with all three approaches were prepared and are presented in Figure 8. The FDC demonstrated the percentage of time a given flow was equalled or exceeded during a specified period of time. The 10 percentile flow can be considered as a high flow percentile in the FDC. The 10 percentile of the flow represents the flow that was equalled or exceeded by 10 per cent of the period of record under consideration. Likewise, 11 to 89 percentile flow was considered as medium flow percentile while 90 percentile flows was considered as low flow percentile.

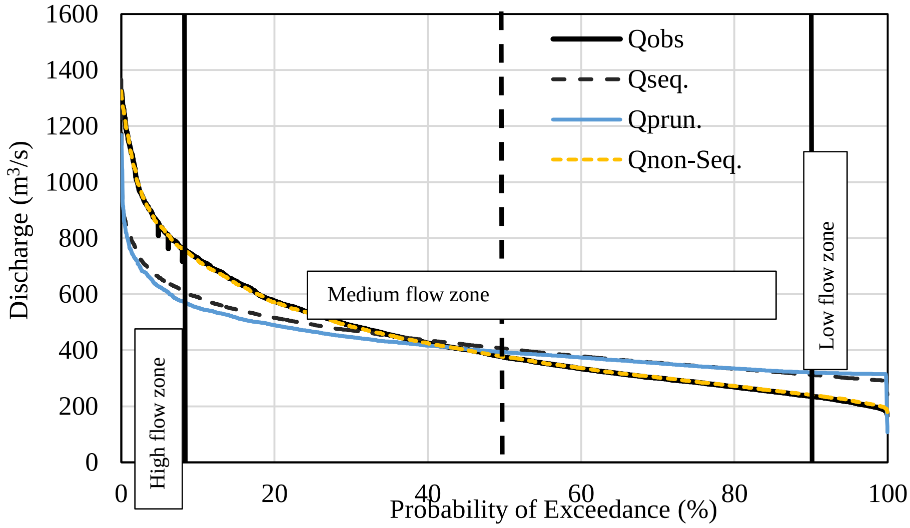

Figure 8: Observed and predicted flow duration curves for three approaches (i.e. sequential, pruned and non-sequence time series).

It can be seen from Figure 8 that the predicted discharges of the ANN models developed with *sequential time series* approach ($Q_{seq.}$) and *pruned time series* approach (Q_{prun}.) were found below the observed discharge for high flows and medium high flows ranges (i.e. under estimating the flows) of the FDCs.

However, the predicted discharge of the ANN models with *sequential time series* approach ($Q_{seq.}$) and *pruned time series* approach ($Q_{prun.}$) were over estimating for medium-low and low flow trails of the observed hydrographs as these curves were found above the observed discharge curves.

Furthermore, the FDC yielded by the ANN model developed with *non-sequential time series* approach ($Q_{non\text{-}seq.}$) well captured all flows i.e. the high, medium and low flow features of the observe discharges and thus can be considered as the best model.

4 SUMMARY AND CONCLUSIONS

As stated by Amirhossein et al. (2015) that *"….. artificial neural network is probably the most successful learning machine technique with flexible mathematical structure which is capable of identifying complex non-linear relationships between input and output data without attempting to reach the understanding of the nature of the phenomena. Statistical approach depending on cross-, auto- and partial-autocorrelation of the observed data is used as a good alternative to the trial and error method in identifying model inputs……"*.

The study was conducted to demonstrate the ability of the ANN model for forecasting runoff using antecedent rainfall-runoff data for the Waikato River catchment. Thus, the following conclusions can be drawn from this study:

1. The ANN models developed using three approaches (i.e. *sequential*, *pruned* and *non-sequential* time series) were able to predict runoff generated from the Waikato River catchment using antecedent rainfall and runoff data.
2. It is clear from the results that the ANN model developed with I_{10} in *sequential time series* approach yielded the best results among the ten input vectors tested. Likewise, I_8 and I_5 produced the best results with the *pruned time series* approach and the *non-sequential* time series approach, respectively.
3. However, the performance of ANN models was found very sensitive to the selection of appropriate input vector(s). As the input vector values increases the model performance increases.
4. This study showed that the input vectors selected (approach 3 i.e. *non-sequential* time series) on the basis of cross correlation (i.e. statistical dependence of two variables such as rainfall and runoff in this case) and auto correlation (i.e. cross relation of discharge with its lagged values) yielded the best results.
5. The ANN model developed using approach 3 performed well, and gave, on average, 96% value for both R^2 & NSE during the validation and testing phases of modelling exercise.

REFERENCES

1 Adamowski, J., Chan, H. F., Prasher, S. O. and Sharda, V. N. (2012) 'Comparison of multivariate adaptive regression splines with coupled wavelet transform artificial neural networks for runoff forecasting in Himalayan micro-watersheds with limited data' *Journal of Hydroinformatics,* 14(3), 731-744.

2 Adamowski, J. and Sun, K. (2010) 'Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds' *Journal of Hydrology,* 390(1–2), 85-91.

3 Amirhossein, F., Alireza, F., Kazem, J. and Mohammedbagher, S. (2015) 'A comparison of ANN and HSPF models for runoff simulations in Balkhichai river watershed, Iran' *American Journal of Climate Change*, 4, 203-216. Available at http://www.scirp.org/journal/ajcc [Accessed in June 2015].

4 Antar, M. A., Elassiouti, I. and Allam, M. N. (2006) 'Rainfall-runoff modelling using artificial neural networks technique: a Blue Nile catchment case study' *Hydrological Processes,* 20(5), 1201-1216.

5 Anthony, M. and Bartlett, P. L. (2009) 'Neural network learning: Theoretical foundations' Cambridge University Press.

6 Beale, M. H., Hagan, M.T. and Demuth, H.B. (2015) 'Neural Network Toolbox Getting Started Guide'. Available at http://au.mathworks.com/help/releases/R2015a/pdf_doc/nnet/nnet_gs.pdf [Accessed during May-August 2015].

7 Burnash, R. J. C., Ferral, R. L. and McGuire, R. A. (1973) 'A generalized streamflow simulation system, conceptual modeling for digital computers'.

8 Caudill, M. (1990) 'Neural Networks Primer' San Francisco: AI EXPERT.

9 Chang, N.B. and Hong, Y. (2012) 'Multiscale Hydrologic Remote Sensing: Perspectives and Applications' CRC Press.

10 Chua, L. H. C., Wong, T.S.W., Sriramula, L.K., . (2008) 'Comparison between kinematic wave and artificial neural network models in event-based runoff simulation for a n overland plane' *Journal of Hydrology, 357*, 337-348.

11 Cleaveland, M. K. and Stahle, D. W. (1989) 'Tree ring analysis of surplus and deficit runoff in the White River, Arkansas' *Water Resources Research,* 25(6), 1391-1401.

12 Committee, A. T. (2000a) 'Artificial neural networks in hydrology-I: Preliminary concepts' *Journal of Hydrologic Engineering, ASCE, 5*(2), 115-123.

13 Committee, A. T. (2000b) 'Artificial neural networks in hydrology-II: Hydrologic applications' *Journal of Hydrologic Engineering, ASCE, 5*(2), 124-137.

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

15 Dooge, J. C. I. (1977) 'Problems and methods of rainfall-runoff modelling: Mathematical models for surface water hydrology' Wiley New York, NY' 71-108).

16 Eluyode, O.S. and Akomolafe, D.T. (2013) 'Comparative study of biological and artificial neural networks' *European Journal of Applied Engineering and Scientific Research*, 2, 36-46.

17 Fausett, L. (1994) 'Fundamentals of neural networks: architectures, algorithms, and applications' Prentice-Hall, Inc.

18 French, M. N., Krajewski, W. F. and Cuykendall, R. R. (1992) 'Rainfall forecasting in space and time using a neural network' *Journal of Hydrology, 137*(1), 1-31.
19 Furundzic, D. (1998) 'Application example of neural netwroks for time series analysis: rainfall-runoff modeling' *Signal Processing,* 64, 383-396.
20 Graupe, D. (2007) 'Principles of artificial neural networks', 6, World Scientific.
21 Hammerstrom, D. (1993) 'Working with neural networks' *Spectrum, IEEE, 30*(7), 46-53.
22 Haykin, S. and Lippmann, R. (1994) 'Neural Networks: A Comprehensive Foundation' *International Journal of Neural Systems,* 5(4), 363-364.
23 Hecht-Nielsen, R. (1988) 'Neurocomputing: picking the human brain' *IEEE spectrum,* 25, (3), 36-41.
24 Hopfield, J. J. (1982) 'Neural networks and physical systems with emergent collective computational abilities' *Proceedings of the national academy of sciences,* 79(8), 2554-2558.
25 Hsu, K. l., Gupta, H. V. and Sorooshian, S. (1995) 'Artificial Neural Network Modeling of the Rainfall-Runoff Process' *Water resources research,* 31(10), 2517-2530.
26 Jain, A., Sudheer, K. P. and Srinivasulu, S. (2004) 'Identification of physical processes inherent in artificial neural network rainfall runoff models' *Hydrological Processes,* 18(3), 571-581.
27 Jeong, D.-I. and Kim, Y.O. (2005) 'Rainfall-runoff models using artificial neural networks for ensemble streamflow prediction' *Hydrological processes,* 19(19), 3819-3835.
28 Karayiannis, N. and Venetsanopoulos, A. N. (2013) 'Artificial neural networks: learning algorithms, performance evaluation, and applications' 209, Springer Science & Business Media.
29 Khan, M. M., Shamseldin, A. Y., Melville, B. W. and Shoaib, M. (2014) 'Impact of Ensemble Size on TIGGE Precipitation Forecasts: An End-User Perspective' *Journal of Hydrologic Engineering,* 20(2).
30 Kleinbaum, D., Kupper, L., Nizam, A. and Rosenberg, E. (2013) 'Applied regression analysis and other multivariable methods' Cengage Learning.
31 Klingeman, P. (2005) 'Streamflow Evaluations for Watershed Restoration Planninga and Design'. Available at http://streamflow.engr.oregonstate.edu/analysis/flow/tutorial.htm [Accessed in July 2015].
32 Lallahem, S. and Mania, J. (2003) 'A nonlinear rainfall-runoff model using neural network technique: example in fractured porous media' *Mathematical and Computer Modelling,* 37(9), 1047-1061.
33 Linsley, R.K. and Crawford, N.H. (1960) 'Computation of a synthetic streamflow record on a digital computer' Int. Assoc. Sci. Hydrol. Publication 51, 526–538.
34 Lohani, A., Kumar, R. and Singh, R. (2012) 'Hydrological time series modeling: a comparison between adaptive neuro fuzzy, neural network and auto regressive techniques' *Journal of Hydrology*, 442-443. Available at https://doi.org/10.1016/j.jhydrol.2012.03.031 [Accessed in August, 2015].
35 McCulloch, W. S. and Pitts, W. (1943) 'A logical calculus of the ideas immanent in nervous activity' *The bulletin of mathematical biophysics,* 5(4), 115-133.
36 Moosavi, V., Vafakhah, M., Shirmohammadi, B. and Behnia, N. (2013) 'A wavelet-anfis hybrid model for groundwater level forecasting for different prediction periods' *Water Resources Management*, 1-21.

37 Mutlu, E., Chaubey, I., Hexmoor, H. and Bajwa, S.G. (2008) 'Comparison of artificial neural network models for hydrologic predictions at multiple gauging stations in an agricultural watershed' *Hydrological Processes*, 22, 5097-5106. Available at http://dx.doi.org/10.1002/hyp.7136 [Accessed during June, 2015].
38 [4] O'Connell, P.E., Nash, J.E., & Farrell, J.P. (2007) 'River flow forecasting through conceptual models part II-The Brosna catchment at Ferbane. *Journal of Hydrology,* 10(4), 317-329.
39 Nash, J.E. and Sutcliffe, J.V. (1970) 'River flow forecasting through conceptual models part 1 – A discussion of principles' *Journal of Hydrology*, 10(3), 282–290.
40 Nayak, P. C., Sudheer, K. P. and Jain, S. K. (2007) 'Rainfall-runoff modeling through hybrid intelligent system' *Water Resources Research,* 43(7).
41 Nayak, P. C., Sudheer, K. P. and Ramasastri, K. S. (2005a) 'Fuzzy computing based rainfall–runoff model for real time flood forecasting' *Hydrological Processes,* 19(4), 955-968.
42 Nayak, P. C., Sudheer, K. P., Rangan, D. M. and Ramasastri, K. S. (2005b) 'Short-term flood forecasting with a neurofuzzy model' *Water Resources Research,* 41(4).
43 Nourani, V., Kisi, Ö. And Komasi, M. (2011) 'Two hybrid Artificial Intelligence approaches for modeling rainfall–runoff process' *Journal of Hydrology,* 402(1), 41-59.
44 Nourani, V., Komasi, M. and Mano, A. (2009) 'A multivariate ANN-wavelet approach for rainfall–runoff modeling' *Water Resources Management,* 23(14), 2877-2894.
45 O'Connell, P. E., Nash, J. E. and Farrell, J. P. (1970) 'River flow forecasting through conceptual models part II-The Brosna catchment at Ferbane' *Journal of Hydrology, 10*(4), 317-329.
46 Priddy, K. L. and Keller, P. E. (2005) 'Artificial Neural Networks: an Introduction', 68, SPIE Press.
47 Principe, J. C., Euliano, N.R. and Lefebvre, W.C. (2000) 'Neural and Adaptive Systems' John Wiley & Sons, Inc.
48 Rawlings, J. O., Pantula, S. G. and Dickey, D. A. (1998) 'Applied Regression Analysis: A Research Tool' Springer Science & Business Media.
49 Ren-Jun, Z. (1992) 'The Xinianjian model applied in china' *Journal of Hydrology,*135 (1-4), 371-381.
50 Riad, S., Mania, J., Bouchaou, L. and Najjar, Y. (2004) 'Predicting catchment flow in a semi-arid region via an artificial neural netwrok technique' *Hydrological Processes,* 18, 2387-2393.
51 Rumelhart, D. E., McClelland, J. L. and Group, P. D. P. R. (1986) 'Parallel distributed processing: Explorations in the microstructures of cognition' Volume 1: Foundations, *MIT Press,* Cambridge, MA, 2, 560-567.
52 Sajikumar, N. and Thandaveswara, B. (1999) 'A non-linear rainfall–runoff model using an artificial neural network' *Journal of Hydrology,* 216(1), 32-55.
53 Salas, J. D., Delleur, J. W., Yevjevich, V. and Lane, W. L. (1980) 'Applied Modeling of Hydrologic Time Series' Water Resources Publication. ISBN: 0-918334-37-3.
54 Senthil Kumar, A. R., Sudheer, K. P., Jain, S. K. and Agarwal, P. K. (2005) 'Rainfall-runoff modelling using artificial neural networks: comparison of network types' *Hydrological Processes,* 19(6), 1277-1291.
55 Shamseldin, A. Y. (1997) 'Application of a neural network technique to rainfall-runoff modelling' *Journal of Hydrology,* 199(3–4), 272-294.

56 Sudheer, K., Gosain, A. and Ramasastri, K. (2002) 'A data-driven algorithm for constructing artificial neural network rainfall-runoff models' *Hydrological Processes,* 16(6), 1325-1330.
57 Sugawara, M. (1995) Tank model. In V. P. Singh (Ed.), 'Computer Models of Watershed Hydrology' Water Resources Research Publications, LA, USE, 165-214.
58 Talei, A., Chua, L. H. C. and Wong, T. S. W. (2010) 'Evaluation of rainfall and discharge inputs used by Adaptive Network-based Fuzzy Inference Systems (ANFIS) in rainfall–runoff modeling' *Journal of Hydrology,* 391(3), 248-262.
59 Tan, B. Q. and O'Connor, K. M. (1996) 'Application of an empirical infiltration equation in the SMAR conceptual model' *Journal of Hydrology,* 185(1), 275-295.
60 Tiwari, M. K. and Chatterjee, C. (2010) 'Development of an accurate and reliable hourly flood forecasting model using Wavelet–Bootstrap–ANN (WBANN) hybrid approach' *Journal of Hydrology,* 394(3–4), 458-470.
61 Tokar, A. S. and Johnson, P. A. (1999) 'Rainfall-runoff modeling using artificial neural networks' *Journal of Hydrologic Engineering,* 4(3), 232-239.
62 Tokar, A. S. and Markus, M. (2000) 'Precipitation-runoff modling using artificial neural networks and conceptual models' *Journal of Hydrologic Engineering,* 5, 156-161.
63 Wang, G. C. S. and Jain, C. L. (2003) 'Regression Analysis: Modeling & Forecasting' Institute of Business Forec.
64 Wang, W. and Ding, J. (2003) 'Wavelet network model and its application to the prediction of hydrology' *Nature and Science,* 1(1), 67-71.
65 Yegnanarayana, B. (2009) 'Artificial Neural Networks' PHI Learning Pvt. Ltd.
66 Zhang, G., Patuwo, B. E. and Hu, M. Y. (1998) 'Forecasting with artificial neural networks: The state of the art' *International Journal of Forecasting,* 14(1), 35-62.
67 Zhang, Q., Wang, B.D., He, B., Peng, Y. and Ren, M.-L. (2011) 'Singular spectrum analysis and ARIMA hybrid model for annual runoff forecasting'. *Water Resources Management,* 25(11), 2683-2703.