Prediction of Runoff using Artificial Neural Networks (A Case study of Khodiyar Catchment Area)

Mr. Mahesh B. Shrivastav[1] Prof. Haresh M. Gandhi[2] Prof. Pinak S. Ramanuj[3] Prof. Milan K. Chudasama[4] Mr. Jignesh A. Joshi[5]
[1]P.G. Student [2,3]Associate Professor [4]Assistant Professor [5]Assistant Engineer
[1,2,3,4]Department of Civil Engineering
[1,2,3,4] Shantilal Shah Engg. College, Bhavnagar, Gujarat, India.
[5]Salinity Control Sub-division, Bhavnagar, Gujarat, India.

Abstract— the present study aims to utilize an Artificial Neural Network (ANN) for modeling the rainfall runoff relationship of Khodiyar catchment area located in Amreli district, Gujarat, India. An Artificial Neural Network (ANN) methodology was employed to predict monthly runoff as a function of precipitation, temperature, evaporation losses, infiltration losses and humidity. The paper illustrates the applications of the feed forward network for the Runoff prediction with various algorithms and accordingly, different structures of ANNs were used and their efficiencies in terms of the mean squared error 'MSE', training and validation determination coefficients 'R' to select better predicted Runoff data were examined. The monthly hydrometric and climatic data of Khodiyar Watershed in ANN were ranged from 1971 to 2010 and analyzed in order to calibrate the given models. Efficiencies of the Back-Propagation (BP), conjugate gradient (CG) and Levenberg-Marquardt (L-M) training algorithms are compared to improving the computed performances and 72 models were prepared to select a best model having mean square error 'MSE' nearer to zero and co-relation factor 'R' nearer to unity. The results revealed that the best model is composed of the feed-forward networks, trained by the Levenberg-Marquardt algorithm and considering only one hidden layer. The results extracted from the comparative study indicated that the Artificial Neural Network method is more appropriate and efficient to predict the river runoff than classical regression model. The ANN model provides a more systematic approach, reduces the length of calibration data, and shortens the time spent in calibration of the models.

Key words*:* Rainfall-runoff Model, ANN, algorithm, simulation, prediction, time-series

I. INTRODUCTION

In many parts of the world, rapid population growth, urbanization, and industrialization have increased the demand for water which has resulted in altered watersheds and river systems, which have contributed to a greater loss of life and property damages due to flooding. It is becoming increasingly critical to plan, design and manage water resources systems carefully and intelligently. For many years, hydrologists have attempted to understand the transformation of precipitation to runoff, in order to forecast Runoff for purposes such as water supply, flood control, irrigation, drainage, water quality, power generation, recreation, and fish and wildlife propagation. The rainfall-runoff relationship is one of the most complex hydrologic phenomena to comprehend due to the tremendous spatial and temporal variability of watershed characteristics and precipitation patterns and the number of variables involved in the modeling of the physical processes. Since the 1930s, numerous rainfall-runoff (R-R) models have been developed to predict Runoff. Rainfall-runoff models play an important role in water resource management planning and therefore, different types of models with various degrees of complexity have been developed for this purpose. These models, regardless to their structural diversity generally fall into three broad categories; namely, black box or system theoretical models, conceptual models and physically-based models. Black box models normally contain no physically-based input and output transfer functions and therefore are considered to be purely empirical models. Conceptual rainfall-runoff models usually incorporate interconnected physical elements with simplified forms and each element is used to represent a significant or dominant constituent hydrologic process of the rainfall-runoff transformation. There has been a tremendous growth in the interest of application of ANNs in rainfall-runoff modeling in the 1990s. ANNs were usually assumed to be powerful tools for functional relationship establishment or nonlinear mapping in various applications. Cannon and Whitfield, found ANNs to be superior to linear regression procedures. Shamseldin examined the effectiveness of rainfall-runoff modeling with ANNs by comparing their results with the Simple Linear Model (SLM), the seasonally based Linear Perturbation Model (LPM) and the Nearest Neighbor Linear Perturbation Model (NNLPM) and concluded that ANNs could provide more accurate discharge forecasts than some of the traditional models. The ability of ANNs as a universal approximate has been demonstrated when applied to complex systems that may be poorly described or understood using mathematical equations; problems that deal with noise or involve pattern recognition, diagnosis and generation; and situations where input is incomplete or ambiguous by nature. The formation of ANN model inputs usually consists of meteorological variables, such as rainfall, temperatures, evaporation, infiltration, humidity and geomorphological properties of the catchment, such as topography, vegetation cover and antecedent soil moisture conditions. The frequently used inputs to ANNs also include observed runoff at nearby sites or neighboring catchments. In many cases, network inputs with or without time lags may also be considered in scenario analysis. The nonlinear ANN model approach is capable of providing a better representation of the rainfall-runoff relationship than the conceptual model. Therefore, instead of using ANNs as simple black box models, the development of hybrid neural networks has received considerable attention. The hybrid neural networks has shown the potential of obtaining more accurate predictions of process dynamics by combining mechanistic and neural network models in such a way that the neural network model properly accounts for unknown and nonlinear parts of the mechanistic model.

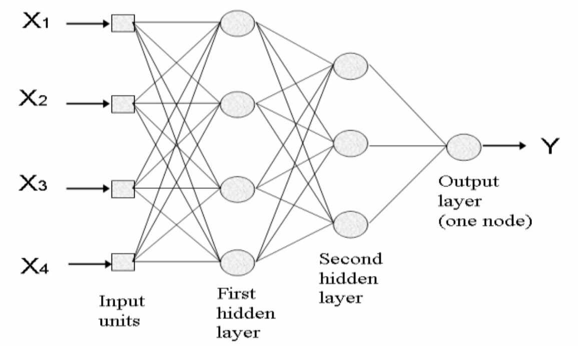

Fig. 1: Structure of a feed-forward ANN Model

II. ARTIFICIAL NEURAL NETWORKS

A. Overview

An ANN is an information-processing system composed of many nonlinear and densely interconnected processing elements or neurons which are analogous to the biological neurons in the human brain. Neurons in an ANN are arranged in groups called layers. Each neuron in a layer operates in logical parallelism. Information is transmitted from one layer to others in serial operations. A network can be composed of one to many layers. The basic structure of a network usually consists of three layers: the input layer, where the data are introduced to the network; the hidden layer or layers, where data are processed; and the output layer, where the results for given inputs are produced (see Fig. 1). The architecture of an ANN is designed by weights between neurons, a transfer function that controls the generation of output in a neuron, and learning laws that define the relative importance of weights for input to a neuron. In a feed-forward network, the weighted connections feed activations only in the forward direction from an input layer to the output layer. On the other hand, in a recurrent network additional weighted connections are used to feed previous activations back to the network. The structure of a feed-forward ANN is shown in Fig. 1. An important step in developing an ANN model is the determination of its weight matrix through training. Since the ANNs do not consider the physics of the problem, they are treated as black-box models; however, some researchers have recently reported that it is possible to detect physical processes in trained ANN hydrological models.

B. The artificial neural network structure

Network structure includes input and output dimensions, the number of hidden neurons and model efficiency calculations. In this study, input dimension includes monthly rainfall, average air temperature, humidity, evaporation losses & infiltration losses data for time step t. Output dimension is the predicted runoff at time t+1. Only one hidden layer was used. The appropriate number of neurons in the hidden layer is determined by using the constructive algorithm, by increasing the number of neurons from 6 to 20. There is a use of log-sigmoid, tangent-hyperbolic and linear activation functions. The ANN model for stream flow evaluation was written in the MATLAB environment, version 2013a, & “nntool” (Neural Network toolbox), “nftool” (Neural Network fitting tool) as well as “ntstool” (Neural Network time series tool) were made use of. The L-M algorithms were evaluated for network training so that the algorithm with better achieved accuracy and convergence speed could be selected. In order to provide adequate training, network efficiency was evaluated during the training and validation stages. In this case, if the calculated errors of both stages continue to decrease, the training period is increased. This is continued to the point of the training stage error starting to decrease, but the validation stage error starting to increase. At this point training is stopped to avoid overtraining and optimal weights and biases are determined.

Model	Architecture	R	MSE
Model-1	3-6-1	0.869	0.0248
Model-2	4-10-1	0.917	0.0052
Model-3	7-14-1	0.931	0.0019
Where MSE is mean squared error & R is co-relation coefficient			

Table. 1 : Result of Model Performance level

C. Network training algorithms

The Back-Propagation (BP) algorithm, has been the most commonly used training algorithm. The CG algorithm has also been used to train ANNs by several researchers. In a study by Chiang *et al*., the CG algorithm was found to be superior when compared with the BP algorithm in terms of the efficiency and effectiveness of the constructed network. In more recent studies the L-M algorithm is also being used due to its superior efficiency and high convergence speed. All commonly used algorithms for network training in hydrology, i.e. BP, CG and L-M algorithms apply a function minimization routine, which can back propagate error into the network layers as a means of improving the calculated output. The L-M algorithm is viewed as a very efficient algorithm with a high convergence speed.

III. STUDY AREA

The proposed methodologies were applied to the Khodiyar Catchment for the prediction of runoff. The Khodiyar Catchment is located in Amreli district in Gujarat, India. The max annual precipitation is about 1374 mm and max monthly precipitation is as high as 847 mm observed in August-2006. Duration of these recorded data was 40 years from 1971 to 2010. A number of ANN models were designed and evaluated for their capability on Runoff prediction. Computational efficiencies of the BP, CG and L-M algorithms and the effect of enabling/disabling of input parameters were also evaluated.

IV. METHODOLOGY

For each one of the developed models, available data were separated as 70% for training, 15% for validation & 15% for testing. Data usage by an ANN model typically requires data scaling which is also known as normalization. This may be due to the particular data behavior or limitations of the transfer functions. In the present paper, the data were scaled in the range of 0 to +1, based on the foll. Equation

$$p_n = \frac{p_o - p_{min}}{p_{max} - p_{min}}$$

in which p_n is the normalized value, p_o is the observed value and p_{max} & p_{min} are the maximum and minimum observed values. Finally, the network output was un-normalized and a regression analysis was carried out between the measured data and their corresponding un-normalized predicted data.

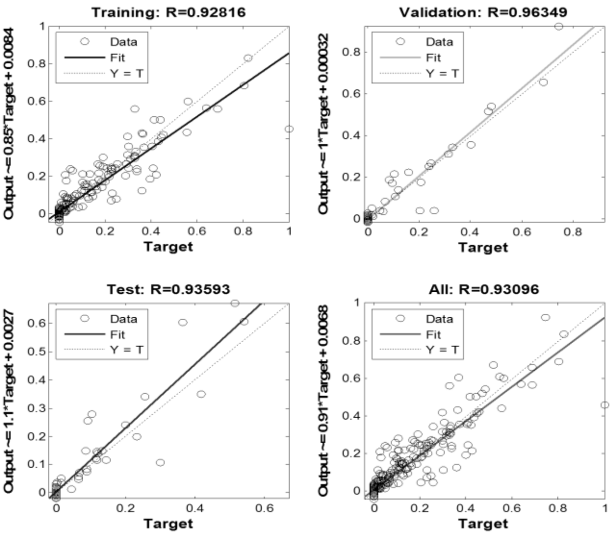

Fig. 2: Regression chart of observed v/s predicted Runoff for Model No. 3 having Architecture 7-14-1

A. Evaluation criteria for ANN prediction

The performances of the ANN are measured with two efficiency terms. Each term is estimated from the predicted values (outputs) of the ANN and the measured discharges (targets) as follows:

1) The correlation coefficient (R-value) has been widely used to evaluate the goodness-of-fit of hydrologic and hydrodynamic models. This is obtained by performing a linear regression between the ANN-predicted values and the targets. A case with R equal to 1 refers to a perfect correlation and the predicted values are either equal or very close to the target values whereas, Intermediate values closer to 1 indicate better agreement between targets and predicted values.

$$R = \frac{\sum_{i=1}^{n} t_i p_i}{\sqrt{\sum_{i=1}^{n} t_i^2}\sqrt{\sum_{i=1}^{n} p_i^2}}$$

Where R is the correlation coefficient, n is the number of samples, $t_i = T_i - T$; $p_i = P_i - P$ in which T_i & P_i are the target and predicted values for i = 1 to n and T & P are the mean values of target and predicted data set respectively.

1) The ability of the ANN-predicted values to match measured data is evaluated by the Root Mean Square Error (RMSE).

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}^{n}(T_i - P_i)^2}$$

Overall, the ANN responses are more precise if R, MSE and RMSE are found to be close to 1, 0 and 0, respectively. In the present study, MSE is used for network training, whereas R and RMSE are used in the network-validation phase.

V. RESULTS AND DISCUSSION

A. Model structures

Three model structures were developed to investigate the impact of variable enabling/disabling of input dimension on model performance.

1) Model 1 is enabled for rainfall, minimum & maximum temperature data as input dimension.
2) Model 2 is enabled for rainfall, min & max temperature, evaporation losses & stream flow data as input dimension
3) Model 3 is enabled for rainfall $P_{(t)}$, $P_{(t+1)}$, max & min temperature, humidity, evaporation losses, infiltration losses and stream flow data as input dimension. Equations 1 to 3 represent Model 1 to Model 3, respectively.

$Q_{(t+1)} = f\{ P_{(t)}, T_{max}, T_{min} \}$ ……………………………...(1)
$Q_{(t+1)} = f\{ P_{(t)}, T_{max}, T_{min}, E_{loss}, Q_{(t)} \}$ …………………(2)
$Q_{(t+1)} = f\{ P_{(t)}, P_{(t+1)}, T_{max}, T_{min}, E_{loss}, I_{loss}, H, Q_{(t)} \}$ ….(3)

where; $Q_{(t+1)}$ is predicted run off, for the time step of t+1; {Q} is monthly runoff data; {P} is monthly rainfall data for the time step of t, $T_{(t)}$ is average monthly air temperature data, E_{loss} is Evaporation loss, I_{loss} is the Infiltration loss and H is the humidity.

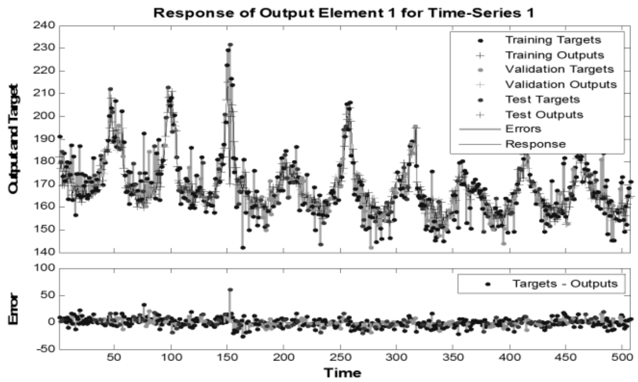

Fig. 3: Time-series Prediction Chart)

B. Model performance levels

Table 1 shows individual model performance levels as measured by MSE and R and individual model architecture as represented by the number of neurons in the input, output and hidden layers. Furthermore, computed rainfall-runoff by individual models are compared with the corresponding observed values and illustrated by their graph (Fig. 2, 3, 4, 5 & 6) which is indicated by the results. It can be concluded that Model 1 resulted with the lowest achieved performance levels. Model 2 resulted in a considerable improvement of the performance levels & Model 3 has highest achieved performance levels with R = 0.931 & least MSE of value 0.0019.

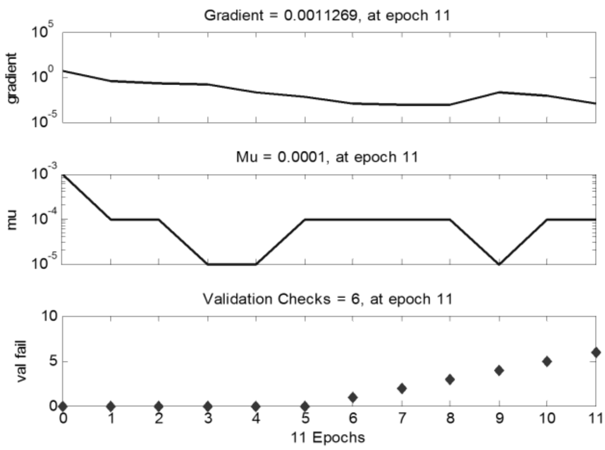

Fig. 4: Gradient, Momentum factor & Validation checks graph for Model No. 3 having Architecture 7-14-1

Also it's clearly observed by using time-series tool (ntstool) of MATLAB that when short prediction intervals of t+1, t+2

& t+3 are considered, the Runoff simulated by this network has a high accuracy. The correlation coefficients that indicate the strength of the relationship between observed and predicted data are higher than 0.9 for the first 3 delays i.e. t+1, t+2 & t+3, but the prediction is less reliable thereafter from t+4 onwards.

VI. CONCLUSION

The Artificial Neural Network (ANN) models show an appropriate capability to model hydrological process. They are useful and powerful tools to handle complex problems.

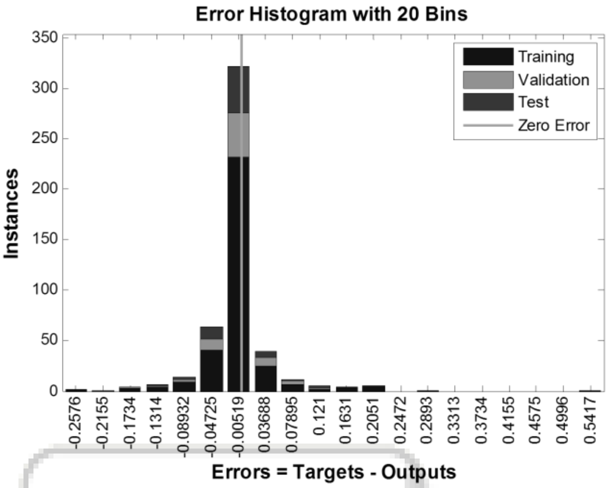

Fig. 5: Error Histogram chart for Model No. 3 having Architecture 7-14-1

In this study, the results show clearly that the artificial neural networks are capable of modeling rainfall-runoff relationship. In this research, the influences of training algorithm efficiencies and enabling/disabling of input dimension on rainfall-runoff prediction capability of the artificial neural networks was applied. A watershed system of Khodiyar catchment located in Amreli district, Gujarat, India was selected as case study. The used data in ANN were monthly hydrometric and climatic data with 40 years duration from 1971 to 2010. For the mentioned model, 28 year's i.e. 70 % data were used for its training but for the validation/testing of the model 12 years i.e. 30 % data were applied. Three types of model structures were developed to investigate the probability impacts of enabling/disabling rainfall-runoff, temperature, humidity, evaporation losses & infiltration losses input data.

Efficiency of model 1 is enabled for Rainfall, Min & Max temperature data as input dimension, model 2 for rainfall minimum & maximum temperature, evaporation losses & stream flow data as input data and model 3 for rainfall, max. & min. temperature, humidity, evaporation & infiltration losses and stream flow data as input dimension. Computational efficiencies, i.e. better achieved accuracy and convergence speed, were evaluated for the Back-Propagation (BP), Conjugate Gradient (CG) and Levenberg-Marquardt (L-M) training algorithms. So under each Model category, by applying above 3 algorithms & by changing number of nodes in the hidden layer from 6 to 20, 24 Models were developed. Totally 72 Models were prepared for selecting the Best Model for this catchment. The L-M algorithm proved to be more efficient than the CG and BP algorithm. Based on the results, validation stage of Mean Square Error (MSE) and coefficient of determination (R) measures were: 0.0248, 0.869 (Model 1); 0.0052, 0.917 (Model 2); 0.0019, 0.931 (Model 3) as indicated in Table-1. As indicated by the results, model 3 provided the highest performance.

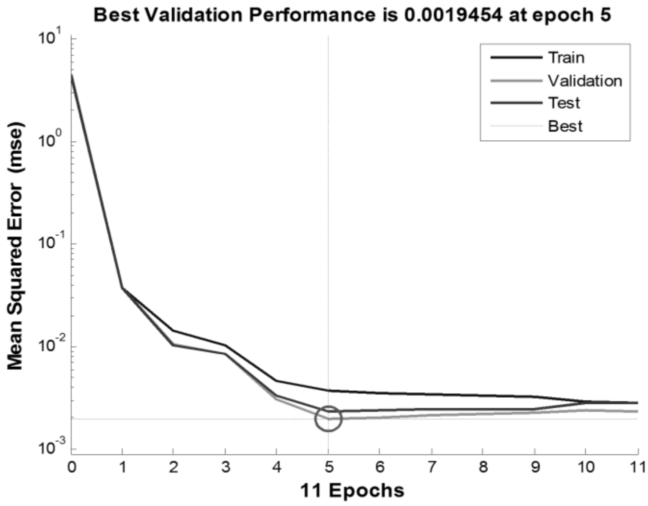

Fig. 6: Mean Square Error M.S.E. graph for Model No. 3 having Architecture 7-14-1

The charts for the best Model having Model Architecture 7-14-1 are mentioned in fig. 2, 3, 4, 5 & 6. This was due to enabling of the rainfall, average temperature and stream flow data, resulting in improved training and thus improved prediction. Also it's clearly observed by using time-series tool (ntstool) of MATLAB that the neural network provides a high accuracy of prediction for the next 3 delays i.e. t+1, t+2 & t+3, but then onwards the accuracy of prediction decreases from t+4 onwards. The results of this study has shown that, with combination of computational efficiency measures and ability of input parameters which describe physical behavior of hydro-climatologic variables, improvement of the model predictability is possible in artificial neural network environment.

ACKNOWLEDGEMENT

The authors would like to thank the State Water Data Centre, Gandhinagar for their support.

REFERENCES

[1] N.Q. Hung and N.K. Tripathi, "An Artificial Neural Network model for rainfall forecasting in Bangkok, Thailand" Hydrology Earth Syst. Sci. Discuss., 5, 183-218, 2009.
[2] Toshihisa Egawa and Tatsuya Iizaka, "A water flow forecasting system for dams using Neural Networks and regression models" Power and Energy Society General Meeting, IEEE, 2011.
[3] Asaad Y. Shamseldin, "Artificial neural network model for river flow forecasting in a developing country", 2010.
[4] Debbaram Sentu, "In flow Prediction by Different Neural Network Architectures: A Case Study", 2011.
[5] George Gabitsinashvili, "Evaluation of Artificial Neural Network Techniques for River Flow Forecasting", 2007.
[6] Dawson C W; Wilby R L "Hydrological modeling using artificial neural network" 2001.

