Research Article

EFFECT OF TRAINING DATASET LENGTH ON THE PERFORMANCE OF ARTIFICIAL NEURAL NETWORK MODEL

***Jitendra Sinha, Rajendra Kumar Netam and Rekha Bai**
Department of Agricultural Engineering, Indira Gandhi Krishi Vishwavidyalaya,
Raipur - 492006, Chhattisgarh, India
**Author for Correspondence*

ABSTRACT

Artificial intelligence such as artificial neural network (ANN) has been proved to be an excellent tool for rainfall runoff modeling with reasonable accuracy. Training is the most important phase in model development. A well trained network performs well on the unknown dataset during testing period. Many a times question arises that what percentage of dataset be used for training an artificial neural network model. In this study an attempt has been made to find out optimum percentage of dataset that is to be used for training an artificial neural network model. The present study has been carried out for the Upper Shivnath Catchment (6990 km^2) considering gauge discharge data of central water commission Gauging site Kotani. The daily rainfall and gauge – discharge data for past 16 year (1986-1990 and 1997-2007) were used. The ANN based analysis was carried out in MATLAB. ANN training was conducted using the Bayesian regularization (Trainbr) algorithm. The ‘logsig' activation function was used for both hidden and output layer nodes. The dataset was divided into 50:50, 60:40, 70:30 80:20 percentage ratios for training and testing. The performance of the models was tested through statistical tools such as mean absolute error (MAE), root mean square error (RMSE), coefficient of correlation (CC), and index of agreement (IA). The best performing models for different datasets, based on the performance in the Testing period (unknown output dataset) for ANN models was found to be 70:30 model with MAE: 681.01, RMSE: 1203.4, CC: 81.54, & IA: 85.86.

Keywords: *Artificial Neural Network, Training, Testing, Rainfall, Runoff, Shivnath River*

INTRODUCTION

Rainfall-runoff process is highly nonlinear and varies with space and time. A rainfall-runoff model simulates the hydrologic response of a catchment to rainfall as input. Estimation of runoff from a catchment is required for the purposes such as design of storage facilities, flood forecasting, assessment of water available for municipal, agricultural or industrial purposes, planning irrigation operations, estimating future dependable water supplies for power generation, wildlife protection etc. Catchments are gauged to provide continuous records of flow in a river.

However, due to high cost involved in setting up and maintenance of gauging station many small catchments remains ungauged and calls for modeling.

Artificial neutral networks (ANN), a black box modeling has gained significant attention in recent year due to its ability to provide better solution when applied to complex system that have been poorly described or understand, problem that deals with noise or involve in pattern recognition, diagnosis, abstraction and generalization and where input is incomplete or ambiguous by nature. It has become very popular for prediction and forecasting in a number of areas including finance, medicine, water resources, and environmental science. The main reason is that ANNs can satisfactorily represent any arbitrary nonlinear function if a sufficient and properly trained neural network is employed. ANNs are applied widely in various domains of hydrology such as rainfall-runoff modeling (Sinha *et al.*, 2013) stream flow forecasting, ground water modelling, water quality management, precipitation forecasting, and reservoir operations. ASCE (2000a, 2000b) and Maier and Dandy (2000) reviewed applications of ANN in hydrology and water resources.

A study has been carried out to simulate runoff at Kotani station of Shivnath River using artificial neural network models trained with different length of dataset.

MATERIALS AND METHODS

Study Area

The study has been carried out for the catchment of Shivnath River. The Shivnath River, major tributary of river Mahanadi, is a non - perennial river having its origin near village Panabaras in Rajnandgaon district. The basin of river Shivnath is located between 20^0 16' N to 22^0 41' N latitude and 80^0 25' E to $82^0$35' E longitude. The stream flow model has been developed based on the stream flow data recorded at Kotani gauging station (Fig. 1). Gauging site Kotani is located at Latitude $21^0$13'02" N and Longitude $81^0$14'19" E. The total catchment area upto Kotani outlet is 6990 sq.km, which is termed as Upper Shivnath Catchment (USC). General slope of the basin comes under Mahanadi River slope and is towards the north and north east and locally in some places towards east. The study area lies in the Chhattisgarh plains. There are three seasons, viz. winter (mid October to mid February), summer (mid February to mid June) and monsoon (mid June to mid October). Onset of monsoon is usually from 15th June and the monsoon season extends up to 15th October. The study area is underlain by diverse rock types of different geological ages from Pre-Cambrian to Recent and from Azoic to Quaternary. Topographical characteristics of the study area were analysed by using the combination of survey of India toposheets No. 64-G, 64-H and 64-D on 1:250,000 scale. The analysis reveals that USC drains from south to north and north-east. Upper portion of the study area comprises of light coloured soil and the areas adjoining the river valleys have smooth fertile soil. Rice is the major crop grown in the USC. The study area is a part of Chhattisgarh - popularly known as rice bowl of central India where large number of indigenous rice varieties is grown.

Collection of Data

Stream flow data of Kotani gauging station have been collected from O/o Chief Engineer, Central Water Commission, Odisha. Daily Stream flow data from the period 1st January to 31st December (1986-1990 and 1997-2007) has been used for the analysis and development of model. The daily rainfall data of Kotani for the period of 16 year (1986-1990 and 1997-2007) have been collected from the office of the State Data Centre, Department of Water Resources, and Govt. of Chhattisgarh. Weighted rainfall for the study area was then estimated by constructing the Thiessen polygons as shown in Fig. 2. The calculated weights of each rainguage station, starting from rainfall station one to five (Rajnandgaon, Ambagarh, Mohla, Balod and Durg) were found to be 0.15, 0.24, 0.24, 0.27 and 0.10 respectively. It can be seen from the rainfall data that the monsoon rainfall occurs from June to October, and this period was considered as the active rainfall period (June 11 to October 28) of the year. Therefore, in the present study the analysis was carried out on active period basis. Also from agricultural point of view water availability is seen on weekly bases therefore in the study weekly basis. Therefore, in this study weekly data of 16 year have been considered.

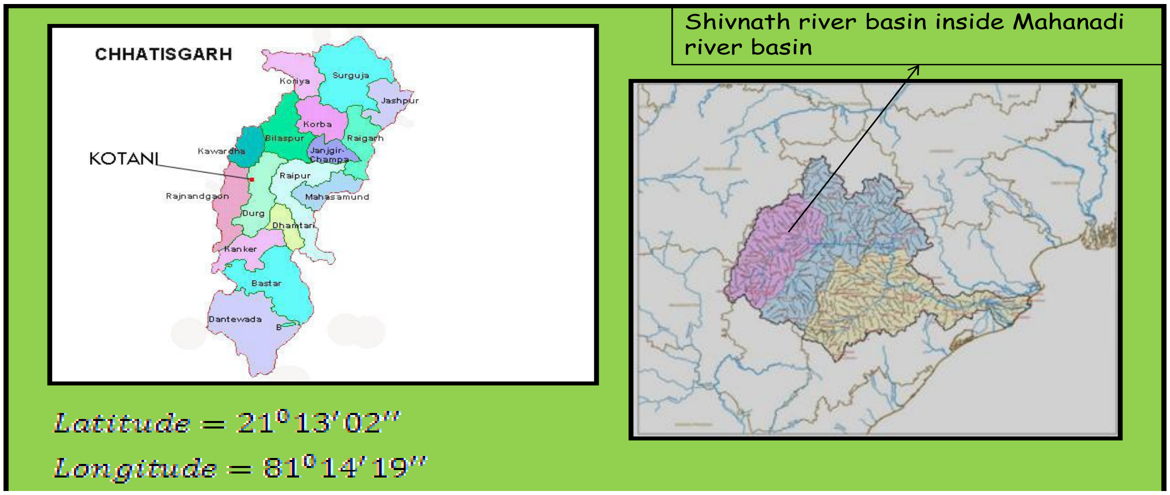

Figure 1: Location of CWC Gauging site Kotani & Shivnath basin

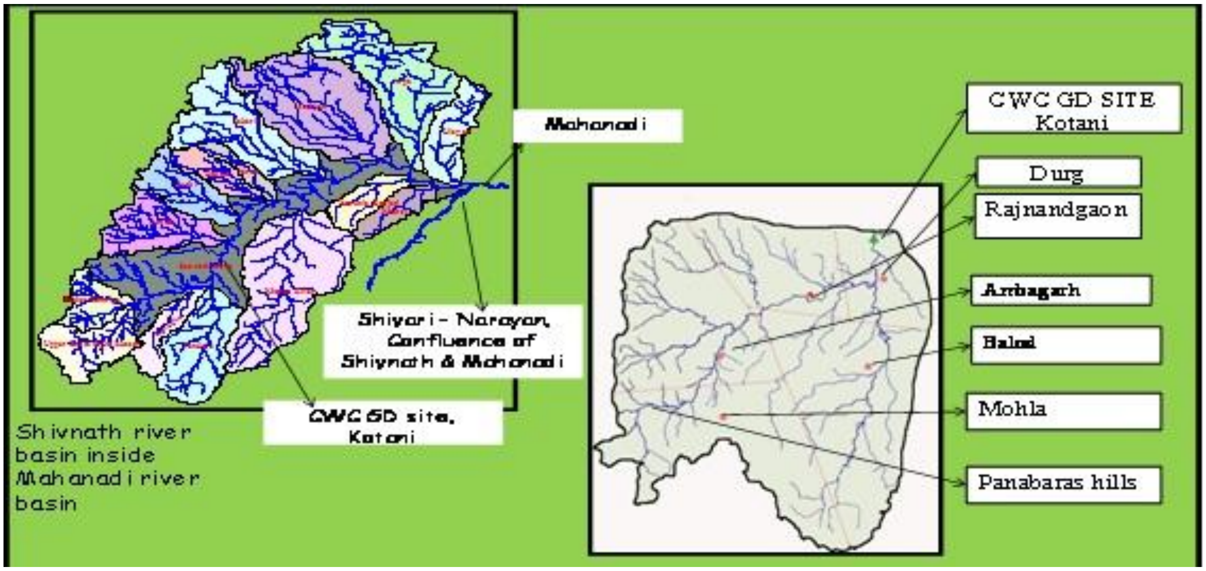
Mahanadi
Shiyari - Narayan, Confluence of Shivnath & Mahanadi
CWC GD site, Kotani
Shivnath river basin inside Mahanadi river basin
CWC GD SITE Kotani
Durg
Rajnandgaon
Ambagarh
Balod
Mohla
Panabaras hills

Figure 2: Drainage Network map of Upper Shivnath Catchment showing Thiessen polygon

Multiple Variable Models

Rainfall is the important input variable which varies with space and time in the catchment. Hence, the rainfall recorded at the Kotani (P_t) shall be considered as input variable for the stream flow (Q_t) being output variable. Since the catchment area is 6990 sq. km., time of concentration is more than a day. Also there is delayed ground water contribution hence previous rainfall (P_{t-1}), (P_{t-2}) and (P_{t-3}) have been also considered as input variables. Previous stream flow has been also considered as input variable to evaluate the model. The inputs selected in this way are subjected to stepwise regression for further refinements in input selection.

Artificial Neural Network (ANN)

An ANN is a massively parallel-distribution information processing system that has certain performance characteristics resembling biological neural network of the human brain (Haykin, 1994). 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 (Caudill, 1987).

The important step in ANN modeling involves the determination of the ANN architecture and selection of a training algorithm. ANN optimal architecture may be considered the one yielding the best performance in the term of error minimization while retaining a simple and compact structure.

In this study, network-growing technique (Gallant, 1986; Kwok and Yeung, 1995) is used to arrive at the final architecture of network. This typically starts with a small network and adds node when a suitable chosen measure stops decreasing. The initial number of neurons is taken equal to number of input vectors and the same numbers have been added for growing the network keeping maximum neurons equal to ten times of the vector. A maximum 1000 epochs/cycles is allowed to judge the convergence of the network. An optimal architecture is the one yielding the best performance in terms of error minimization while retaining a simple and compact structure. The selection of the number of hidden layers and the number of nodes within a layer is quite arbitrary. There are no fixed rules as to how many nodes should be included in a hidden layer. If there are less number of nodes in the hidden layer the network may have difficulty in generalization. On the other hand, the network may take a longer time to learn if there are too many nodes in the hidden layer and it may tend the network to memorize instead of learning and generalization (Vemuri, 1992).

Distribution and Format of Training Data

Rainfall and runoff data which are collected are used to train the network and to calibrate the network. This is normally considered as a satisfactory method selection of the training data. To assess the effect of training data on the performance of network, training is performed with different number of experimental dataset. Firstly a relatively small set of data including all ranges of input variable is used to train the network. The performance of the network during testing is monitored and observation made on the level of accuracy obtained during testing. Subsequently the same above mentioned procedure is repeated using higher percent of data during training and the performance of the network during testing monitored. In this way the sensitivity of the performance of the network to the size of training data is evaluated. Different combinations are used to train the model, according to total length of data as shown in table 1.

Training Algorithms

There are different types of ANNs. Some of the most popular types include the multi-layer Perceptron, which is generally trained with the back propagation of error algorithm, radial basis function. ANNs can also be classified as feed forward or recurrent (feedback) depending on how data is processed.

Table 1: ANN Model trained and tested with varying length of records.

S. No.	Model	training with % of the total length of	testing with %of total length of data
1	M 50-50	50	50
2	M 60-40	60	40
3	M 70-30	70	30
4	M 80-20	80	20

In the present study the ANN models were designed by using MATLAB codes. A programme was written, edited, debugged and run in MATLAB. The programme was suitably modified to accommodate different input patterns. The programme is flexible to accommodate different activation functions (logsig), performance functions (mse, sse and msereg), training algorithm (trainbr, trainlm etc.) and a preset number of iterations. In this study ‘logsig' activation function was used. The training algorithm chosen was Bayesian regularization (Trainbr) and the performance function chosen is sum squared error (sse).

Design of ANN Model for Runoff Simulation

The neural network utility file is edited in MATLAB. The input variable selection, input data source file, network option, training function, setting for the data for training, testing is created and run in MATLAB software. The model was first implemented into an Excel spreadsheet to serve as the guideline to the MALAB program defining the model. In this study, models with various input variables were developed and their performance was evaluated. Various models developed with varying inputs are given in table 2. The variables were selected according to the model M-50, M-60, M-70, M-80 for developing and evaluating the ANN models. The ANN model architecture is single layer feed forward network, which is the most commonly used neural network for the prediction of the non-linear process like rainfall-runoff relationship. The number of the hidden layer is one. The transfer function chosen is log-Sigmoid Transfer Function. The back propagation training function has been selected, which is the most common and accurate as reported by many workers. The performance function for training and testing the networks used are MSE (Mean Squared Error). The various combinations of hidden nodes and training function are done to arrive at optimum combination to give less error.

RESULTS AND DISCUSSION

Stepwise Regression Analysis

Increasing input variable indicate that inclusion of each input variable has increase the correlation coefficient. In some cases the increase in correlation coefficient (R) is very small or insignificant. To find out that which input variables makes significant change in correlation coefficient (R) and should remain in the model and to discard those variables having insignificant influence on correlation coefficient, a stepwise regression analysis was carried out. Seven input variable (P_t, Pt_{-1}, P_{t-2}, P_{t-3}, Q_{t-1}, Q_{t-2}, Q_{t-3}) have been considered for regression analysis and it can be seen that the highest correlation (61.1%) is between

P_t –Qt. After step wise regression the P_t & Q_{t-1} have been found significant Hence only P_t & Q_{t-1} have been considered as input for modeling.

Performance of ANN Models

Learning process forms the interconnections between the neurons and is accomplished by using known inputs and outputs, and presenting them to the network in some ordered manner. The strength of these connections is adjusted using an error convergence technique so that the desired outputs will be produced for a given input. Once the network is formed, the interconnection weights and biases are fixed and the programme was saved as MATLAB file, which can accessed in a MATLAB environment to carry out the intended task with different inputs. Selecting a successful network geometry is highly problem dependent. The number of input and output layer neurons is fixed depending on the number of inputs and outputs used in the model. As only runoff was simulated in this study, only one neuron existed in the output layer. Performance of the different models during training and testing is presented in table 2.

Table 2: Performance of different multiple layer Perceptron (MLP) with varying length of data set during training & testing.

Model Name	Model Structure	Training				Testing			
	(NOL-NOIN-NOHN- NOON-NOI)	MAE	RMSE	CC	IA	MAE	RMSE	CC	IA
50:50	3-2-2-1-65	610.98	1066.3	81.61	88.6	**677.04**	1229.9	72.1176	83.71
60:40	3-2-2-1-45	607.31	1058.9	81.16	88.41	692.45	1210.2	81.32	85.46
70:30	3-2-2-1-25	**549.63**	**989.8**	**81.45**	**88.71**	681.01	**1203.4**	81.54	**85.86**
80:20	3-2-4-1-60	604.84	1055.0	77.57	85.5	687.14	1242.4	**81.87**	83.88

Comparison of Models

Comparison of models for best generalization showed that a best fit network model could be obtained without going for higher number of iterations. A higher number of iterations over learned the training process and memorized the given patterns. As a result of which the model gave poor performance during Testing period.

Comparison of ANN models on the basis of MAE, RMSE, CC and EV showed that the 70:30 models performed better than the respective models. It can be seen from table 2 that the values of MAE, RMSE, CC and IA for model 70:30 are 549.63, 989.8, 81.45, and 88.71 during training and 681.01, 1203.4, 81.54, & 85.86 respectively during testing.

Conclusion

Models were formed by considering the standard meteorological week (SMW) number 24 to 43 data (June 11 to October 28). From the above findings it could be inferred that rainfall-runoff modelling of the study area can be described using ANN. However, in one or two cases, the model is felt to be reinvestigated at other places to enrich the present findings in similar catchments.

REFERENCES

ASCE (2000a). Task committee on Application of Artificial Neural Networks in Hydrology, Artificial neural networks in Hydrology. I: Preliminary Concepts. *Journal of Hydrologic Engineering* **5**(2) 115-123.

ASCE (2000b). Task Committee on Application of Artificial Neural Networks in Hydrology, Artificial neural networks in Hydrology. II: Hydrology applications. *Journal of Hydrologic_Engineering* **5**(2) 124-137.

Caudill M (1987). *Neural Networks Primer,* Part-I. AI Expert, Netherlands.

Haykin S (1994). *Neural Networks: A Comprehensive Foundation* (IEEE press).

Gallant SI (1986). Three constructive algorithms for network learning. *Proceedings of 8th Annual Conference of Cognition Science Society, Michigan* 652-660.

Kwok T and Yeung D (1995). Constructive feed-forward neural networks for regression problems: a survey, Technical Report, Hong Kong. University of Science and Technology, Hong Kong.

Maier HE and Dandy GC (1996). The use of artificial neural networks for the prediction of water quality parameters. *Water Resources Research* **32**(4) 1013-1022.

Sinha J, Sahu RK, Agarwal A, Senthil AR and Sinha BL (2013). Levengerg – Marquardt algorithm based ANN approach to rainfall – runoff modeling. *Journal of Soil and Water Conservation, SCSI* **12**(1) 48-54.

Vemuri V (1992). *Artificial Neural Networks: Concepts and Control Application*. Los Alamitos, California (IEEE Computer Society Press) 509.