(14) 25

Thesis Report

On

REAL TIME FLOOD FORECASTING USING ARTIFICIAL NEURAL NETWORK

Conducted at National Institute of Hydrology, Roorkee

Uttarakhand

Gautam budha university, Greater Noida

Uttar pradesh

Department of Civil Engineering

Submitted By

MISS. SHENU TONGER

11ICE097

Under the guidance of

Dr. A. R. Senthil Kumar

Scientist 'E'

SWH Division

CERTIFICATE

This is to certify that **Miss. Shenu Tonger** has undergone a project work on "**Real time flood forecasting using ANN**" from 1st november, 2014 to 30th April, 2016 as a six months training submitted to the Research Management and Outreach Division, National Institute of Hydrology, Roorkee, in partial fulfillment of the requirement for the award of degree of "**Master of Technology"** in **Civil Engineering** specialization in **Environmental Engineering** is the original work carried out by her under our supervision and guidance.

A. R. Senthilkumar 04/05/16

Dr. A. R. Senthil Kumar

Scientist 'E'

SWH Division

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provided us the possibility to complete this report. It is my great pleasure to acknowledge those whose active help and support make this report possible in the present form.

First of all I am highly grateful to ***Er. R. D. SINGH****, honorable Director, National Institute of Hydrology, Roorkee, Uttarakhand, for accepting me as a trainee.*

I express my sincere gratitude to ***Dr. V. C. Goyal****, Head, Research Coordination and Management Unit, for allowing me to pursue six months training programme at the National Institute of Hydrology, Roorkee.*

My diction would be inadequate to express my deepest sense of gratitude and heartfelt thanks to my co-guide ***Dr. A. R. SENTHIL KUMAR****, Scientist E, Surface Water Hydrology Division, National Institute of Hydrology, for spending out his precious and valuable time and consistently guiding me throughout my training period. This would not have been possible without his guidance, support and encouragement. Under his guidance I successfully overcame many difficulties and learned a lot.*

My diction would be inadequate to express my deepest sense of gratitude and heartfelt thanks to my guide ***Dr. Shobha Ram,*** *for his* support and *for spending out his precious and valuable time and consistently guiding me throughout my training period.* suggestion and blessings.

I thank almighty, my parents for their constant encouragement and financial support. I equally thankful to my ***friends*** *for their co-operation and helping hand at the times of adversities.*

SHENU TONGER

ABSTRACT

Forecasting is the making predictions about the uncertainty of the future by using the historical data. Forecasting provide advance warning about any danger which is going to happen in future. Forecasting is broadly considered as a technique or a method for estimating or finding many future aspects of any operation. Planning for the future is a critical aspect. The main goal of forecasting is to give an accurate picture of future as possible as to. ANNs are a form of computing inspired by the functioning of the brain and nervous system. Recently, another class of black box models in the form of Artificial Neural Network (ANN) has been popularized in modeling real time problems wherein the non- linear relationship between the rainfall and runoff process is modeled.

In this report, the application of Artificial Neural Network model (ANN) and a model combining the multiple layer regression (MLR) is investigated for modeling the real time flood prediction using rainfall-runoff data of Hamp River, Chattisgarh. . The rainfall in the catchment area Chirapani, Bodla, and Panadariya and the hourly discharge data is used to carry out this research work. The duration of data used is from 1981 to 2009. In this study the ANN and MLR model results are compared with each other. The ANN model performs better for real time flood prediction than that of MLR model during calibration and validation. In addition, the comparison of the scatter plots of ANN model is more precise than that of MLR. The result of all lead times of calibration and validation are compared.

The RMSE of ANN model during calibration and validation for seven lead time were found to be 1.5881 and 1.0554, 1.8957 and 1.3638, 2.0828 and 1.5195, 2.2288 and 1.5826, 2.3039 and 1.6345, 2.3689 and 1.7120, 2.4547 and 1.7868 respectively, whereas for the MLR model, RMSE value during calibration and validation were 1.7244 and 1.1617, 2.0635 and 1.4664, 2.2671 and 1.6222, 2.3963 and 1.6742, 2.4826 and 1.7120, 2.5393 and 1.7745, 2.5935 and 1.860 respectively, and also the ANN model efficiency during calibration and validation were 0.8682 and 0.8946, 0.8122 and 0.8241, 0.7733 and 0.7817, 0.7405 and 0.7633, 0.7227 and 0.7476, 0.7068 and 0.7232, 0.6852 and 0.6986 respectively, whereas the MLR model efficiency during calibration and validation were

CONTENTS

LIST OF TABLES

CHAPTER – 1

INTRODUCTION

1.1 GENERAL INTRODUCTION

Floods are the most dangerous natural disaster. From the last few decades the damage due to flood is growing exponentially. This is due to the change in the upstream land use, heavy rain and increase in the population. Floods occur at irregular intervals and vary in size, duration and area of extent. Floods occur suddenly and move away quickly, or it may take time to happen like days or even months. Flood is natural phenomena which damage the property, loss of human life, damage to the crops, loss of livestock, and cause waterborne diseases.

To control all these consequences forecasting is done. Forecasting is the making predictions about the uncertainty of the future by using the historical data. Flood Forecasting is an important tool in reducing flood risk. To reduce the negative effects of the flood we do flood forecasting. It can be done by many methods. Forecasting the flow of a river provides a warning about the stage of a flood and also helps in regulating the reservoir outflow when the flow of water is low for water resource management. Flood forecasting can be done by making rainfall- runoff models. These are deterministic models, stochastic and statistical models and more recently Artificial Neural Network (ANN) and fuzzy logic techniques. These models can be done by many methods like black box models, gauge to gauge correlation method and artificial neural network models. In last few years ANN models have been used successfully for flood forecasting. ANN gives improved performance than other black box models. The fuzzy logic and ANN based models, having the potential for the real time flood forecasting, are capable of considering the inherent non-linearity in the rainfall-runoff process.

The main advantage of ANN models over other traditional models is that it does not require information about the complex nature of the underlying process under consideration to be explicitly described in mathematical form (Senthil kumar, 2005).

1.2. FLOOD FORECASTING

1.2.1. Flood Problems in India

The main problems in India due to floods are inundation, bank erosion and drainage due to urbanization. The problems of flood depend on the topography of the place, river system, and flow phenomenon. The flood problems in India are visualized on regional basis. However, for the sake of simplicity, India may be broadly divided into four zones of flooding which are as follow:

- Brahmaputra River Basin,
- Ganga River Basin
- Central India and Deccan Rivers Basin
- North-West Rivers Basin

1.2.2. Forecasting

Forecasting is the making predictions about the uncertainty of the future by using the historical data. Forecasting provide advance warning about any danger which is going to happen in future. Forecasting is broadly considered as a technique or a method for estimating or finding many future aspects of any operation. Planning for the future is a critical aspect. The main goal of forecasting is to give an accurate picture of future as possible as to. But, it can never be possible to be fully accurate as in other forms of fortune telling. There are simply too many interactive variables. A small change in any of these can cause the change in the forecasted scenario. But, with the help of forecasting we are able to prepare for the future danger and also be able to reduce the effect of any danger going to happen in the future.

1.2.3. Flood Forecasting

Flood forecasting is a technique which uses the known characteristics of a river basin to predict the timing, discharge, and height of flood peaks resulting from a measured rainfall, usually with the objective of warning populations who may be endangered by the flood.

1.2.3.1. Development of flood forecasting in India

In 1969, the Government of India created a Central Flood Forecasting Directorate headed by a Superintending Engineer. In 1970, under Member (Floods), six flood forecasting divisions were set up on inter-state river basins. These covered the flood prone basin/sub-basins of the Ganga. The Brahmaputra, the Narmada, the Tapti, the Teesta and coastal all are rivers of Orissa. By the year 1977, the Central Flood Forecasting organization comprised of one Chief Engineer's Office, 3 circles and 11 divisions. Now, in most of the States there are arrangements for the issue of flood warning from the upstream stations to the downstream stations (Wikipedia). These warnings include:

- Whether the river is rising above a certain specified level, known as danger level or not
- Whether the river is or falling rising
- Whether the stage of the river is high, medium or low.

The above warnings, issued by telegrams, telephone or wireless systems are of purely qualitative in nature and they give only an indication of the nature of the flood. Such procedures are at present being followed in West Bengal, Andhra Pradesh and Bihar states.

1.2.3.2. Methodology employed for flood forecasts

The various steps involved in the operation before issue of forecasts and warning are as follow:

- Observation and collection of hydrological and meteorological data
- Transmission or Communication of data to the forecasting Centers
- Analysis of data and formulation of forecasts
- Distribution of forecasts and warning to the Administration and Engineering Authorities of the States.

1.2.3.3. Need for flood forecasting

- To evacuate the affected people to the safer places,
- To make an intense monitoring of the flood protection works such as embankments so as to save them from failures, etc.
- To control the floods through the reservoirs and barrages, so that the safety of these structures can be taken care of against the higher return period floods.

- To operate the multi-purpose reservoirs in such a way that an encroachment into the power and water conservation storage can be made to control the incoming flood.
- To operate the city drains (out falling into the river) to prevent bank flow and flooding of the areas drained by them.

1.3. REAL TIME FLOOD FORECASTING

The real time flood forecasting is one of the most effective non- structural measures for flood management. For determining the flood forecast in the real time, firstly the observed meteorological and flow data are transferred to the forecasting station by the different modes of data communication like telephone, wireless and network of telemetry stations etc. Then in real time collected meteorological and flow data are used into the real time flood forecasting model for forecasting the flood flow and also the water levels for different lead periods varying from few days few hours to depending on the size of catchment and purpose of the forecast. The model structure should be simple and also should not require excessive input, but at the same time the forecasted flood must be as accurate as possible.

Many types of flood forecasting techniques in real time and models have been suggested and used by many investigators. Those are classified as:

- deterministic models,
- stochastic and statistical models,
- Artificial Neural Network (ANN) and fuzzy logic techniques.

In India the various flood forecasting centers are using different forecasting models, which are based on the availability of data of hydrological and hydro-meteorological, basin characteristic, the availability of computational facilities at the forecasting stations, warning time required and purpose of forecast. Some of the commonly used methods by various forecasting techniques include:

- simple correlation which is based on stage-discharge data,
- co-axial correlation which is based on stage, discharge and also on rainfall data etc,

- net work model wherein the lateral flows from different sub-basins are determined using unit hydrograph and successive routing through different sub-reaches which is carried out by using river flood routing techniques, and
- Hydrologic models (at selected places).

The recent techniques like fuzzy logic and ANN are commonly used by the academicians and researchers for the development and testing. Real time flood forecasting systems are formulated for issuing the flood warning in real time in order to prepare the eviction plan during the flood. Thus mostly statistical approach is used to formulate the real time flood forecast in India. Event based network model and multi-parameter hydrological models are applied for some of the trial projects during the execution of the International projects.

1.3.1. Methods For Formulating The Real Time Flood Forecasting

The methods for formulating the real time flood forecast may be categorized as:

- Statistical methods and
- Deterministic methods
- Computational techniques like Artificial Neural Network (ANN) and fuzzy logic models.

1.3.1.1. Statistical methods

Methods base on statistical approach makes use of the statistical techniques to analyze the historical data with an objective to develop methods for the establishment of flood forecasts. The methods thus developed can be presented either in the form of mathematical equations or graphical relations. A large number of data, covering a wide range conditions are analyzed to derive the relationships which interracial include rainfall peak stage relationships and gauge to gauge relationship with or without additional parameter. These methods are more commonly used in India.

1.3.1.2. Deterministic methods

One of the important areas in hydrology pertains to the study of the transformation of the time distribution of rainfall on the catchment to the time distribution of runoff.

This transformation is studied by first relating the volume of rainfall to the volume of direct surface runoff, thus determining the time distribution of rainfall excess and then transforming it to the time distribution of direct runoff though a discrete or continuous mathematical model. The first step decides the volume of the input to the catchment and therefore any error in its determination is directly transmitted through the second step to the time distribution of direct runoff. A number of watershed conceptual models find this component for each time step through a number of stores representing various processes on the catchment. The parameters of these models including those in the functional relationship are determined from the historical record and their performance is tested by simulating some of the rainfall-runoff events which have not been used in the parameter estimating process. The models need to be run continuously so that the status of various stores is available at all times. One of the operational uses of these models is in the area of real time flood forecasting required for real time operation of the reservoir. In such a situation these models are run by inputting the rainfall and forecasts are issued assuming no rainfall beyond the time of forecast value of the rainfall in the future.

The infiltration part of these models and their context decide the volume of input. At the time of calculation the catchment is also performing the transformation operation to produce the direct runoff at the gauging station. Since the model is simulating the action of the catchment it would be appropriate to make use of this information in finding out the contribution which the rainfall is going to make to the direct runoff on the catchment. However, the complexity of these models does not lend itself to this exercise during the event. SSARR (Stream flow Synthesis and Reservoir Regulation) model, Sacramento model and NAM-System 11 FF model are some of the watershed conceptual models for formulating the real time flood forecast. In India, the real time flood forecast has been formulated in some pilot projects using these models. However, these conceptual models are not being utilized because of inadequacy of data and problems associated with the proper calibration of the models (Singh 2001).

1.3.2. Advantage of Real Time Flood Forecasting

- It provides more accurate flood warning.
- It provides timely warnings.

- It can be used for the complex nature data.
- Enabling efficient and targeted emergency response.

1.4. ARTIFICIAL NEURAL NETWORK (ANN)

1.4.1. ANN – An Overview

Artificial Neural Networks (ANN) is computational approaches imitate the ability of the biological neural network by interconnecting many artificial neurons. ANN can identify the correlated patterns between input data sets and corresponding target values. It can also process problems having high nonlinear and complex data even if the data are imprecise and noisy. ANN method is ideally suited for the prediction of disordered time series such as water resources data, which are known to be nonlinear and very complex. One of the most popular neural networks is the layered feed-forward neural network with a back-propagation (BP) least-mean-square learning algorithm (Wenrui Huang, 2004).

ANNs are a form of computing inspired by the functioning of the brain and nervous system. Recently, another class of black box models in the form of Artificial Neural Network (ANN) has been popularized in modeling real time problems wherein the non- linear relationship between the rainfall and runoff process is modeled. The ANN model has wide applicability in Civil Engineering applications and many research papers have been published on its application. The use of ANN in real time flood forecasting is of very recent origin and is still in the evolution stage.

1.4.2. Basic Structure of ANN

Neurons commonly have multi-layer structures connected in parallel which make it possible that input signals are transmitted feed-forward. It is called a MFN and the network is composed of input layer, output layer, and hidden layer (Shin et al. 2004).

The architecture of a feed forward ANN can have many layers where a layer represents a set of parallel neurons. The basic structure of ANN usually consists of three layers:

- The input layer: where the data is feed to the network.

- The hidden layer or layers: where data are processed.
- The output layer: where the results of given outputs are produced.

The neurons in the layers are interconnected by strength called weights (Senthil kumar, 2005). ANN of three layered feed forward figure is shown in fig. 1

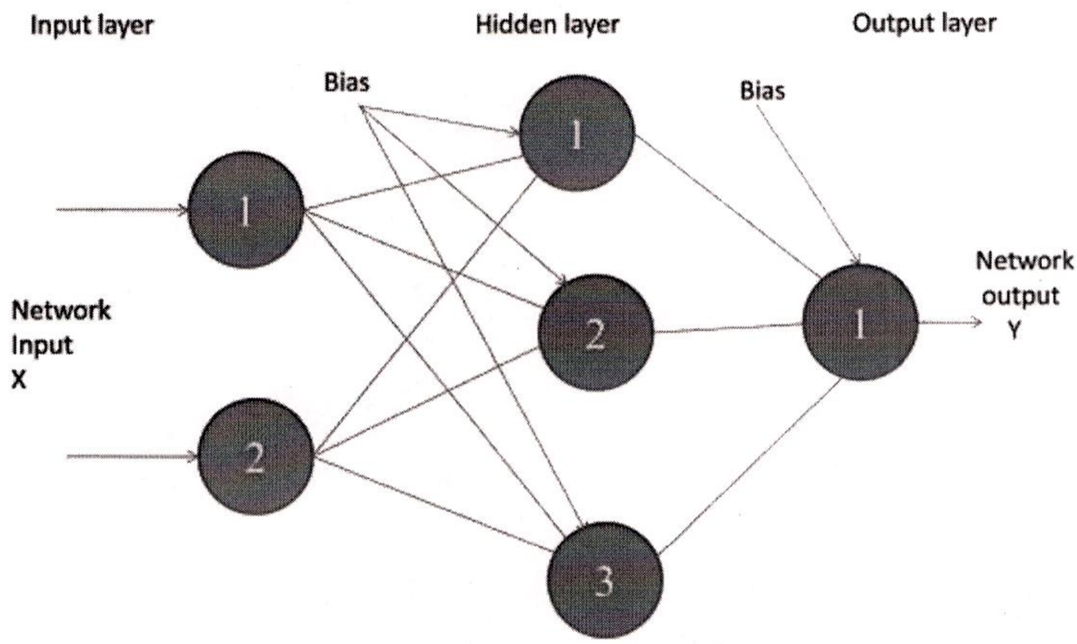


Fig.1. A three layer feed forward ANN (Senthil kumar).

1.4.3. ANN Training Algorithm

1.4.3.1. Back propogation algorithm

This algorithm involves minimization of the total error by using the gradient descent method. The network weights and biases are adjusted by moving a small step in the direction of the negative gradient of the error function during iterations. The iterations are repeated until either a specified convergence is reached or a number of iterations are over. One iteration is over when all training patterns were finished. The iterations are continued until the overall mean squared error for all output nodes and training patterns reaches to a minimum (jeong, 2000).

1.4.3.1. Cascade- correlation algorithm

The Cascade-Correlation algorithm is an efficient constructive training algorithm developed by Fahlman and Lebiere(1990). Unlike back propagation algorithm, here the network architecture is not fixed. Hidden nodes are added one by one starting from zero during the training until the training termination criterion is reached. This algorithm does not involve learning by descending down the error gradient, but by maximizing the correlation of the new hidden node's output on the residual error. It also does not involve transmission of the error backward as in the back propagation algorithm (jeong, 2000).

1.5. SCOPE OF THE STUDY

A flood prediction model can play a key role in providing relevant information of possible impending floods in populated locations. The development of ANN models can reduce the damage in areas by decreasing the economic and environmental impacts of floods.

Artificial neural network (ANN) models provide sufficiently accurate forecasts, even one day ahead, the lead time for flood warning can be extended and the subsequent flood emergency measures can be better planned and executed. ANN has a wide scope in hydrology and in other field also. It processes the complex data into the simple form and gives accurate prediction of flood forecasting. ANN has the ability to represent any non- linear data into simple form by sufficient complexity of trained network. The computational techniques used for ANNs have relatively low demands and can easily be combined with other techniques. The neural network as non-linear model is a promising approach compared to linear models for flood forecasting.

CHAPTER – 2

LITERATURE REVIEW

2.1 INTRODUCTION

ANN has many applications in real world such as speech processing, medicine and image processing. ANN has flexibility to solve a complex problem and also have robustness in noisy environment. In many area of the water resource neural network approach is used. ANN has the ability to represent any non- linear data into simple form by sufficient complexity of trained network. Many applications of ANN in flood forecasting modeling are reported. Some of them are reviewed in detail.

2.2 LITERATURE REVIEW ON FLOOD FORECASTING BY ANN

Thirumalaiah and Deo (1998) used ANN to highlight its use in real time forecasting and forecasted the water levels at Jagdalpur. Error back propagation, cascade - correlation and conjugate gradient algorithms were used to train the network. The results of training were compared and with untrained data network was verified. The following conclusions were made:

- The more correctly trained values were the lower values.
- The small fraction of time was taken by cascade correlation algorithm.
- Lesser accuracy in stage forecasting was not indicating larger warning time.
- To produce more satisfactory stage forecast a dynamically adaptive network was found.

Jeong et al. (2000) developed a rainfall runoff model and a real time inflow forecast system by neural network theory. In this a basin around Soyangang River has been studied. Back propagation algorithm was used to train a neural network model. For finding optimum neural network structure Cascade- correlation algorithm was applied. Training ability was increased by increasing the hidden nodes. The following conclusion was made:

- It was found that more precise result was shown by Cascade-correlation algorithm for forecasted inflow of 1-hr, 3-hr by neural network model.
- For 6-hr flood inflow forecasting cascade correlation algorithm forecasted better peak discharge than back propagation.

- Model's goodness of fit and accuracy increased by applying filtering method to the neural network model with Cascade -correlation algorithm.

Thirumalaiah and Deo (2000) used ANN to forecast the water level in real time and throughout the year with a continuous discharge at a given site based on the same levels at the same site using the stage time history recorded at some upstream gauging station. To model the river stage forecasting system feed forward neural network structure was used. Error back propagation, cascade-correlation and conjugate gradient algorithms were used to train the network. Then the result was compared. The untrained data were used to verify the trained network. It was concluded that the use of neural network make the possibility for the continuous forecasting in real time of a river stage.

Liong et al. (2000) used ANN a relatively new approach which was a highly suitable flow prediction tool. In this paper water level prediction accuracy for up to 7 lead days at Dhaka, Bangladesh were demonstrated. To verify the importance of each of the input neuron a sensitivity analysis has been introduced. Less sensitive input neurons was eliminated which reduces the degree of accuracy. The following conclusions were made:

- The study gave the computational time required by the conventional rainfall runoff models for the suitability of a neural network for flow prediction with high accuracy.
- It was a technique to reduce unnecessary data collection and operational cost in less sensitive input neurons.

Huang (2004) developed the ANN to forecast the change in the river of Apalachicola in real time. A three layer, non linear, back propagation network were developed for modeling the river flow. Feed- forward neural network was used to model the change in the flow of river. The network was trained by gradient descent and conjugate gradient algorithm. It was concluded that ANN model provide adequate predictions of river flow. The water resource management activities such as monitoring drought for navigation and low freshwater input in downstream estuary the flow forecasting capability was beneficial. To predict the circulation and salinity in the estuary of the Apalachicola bay for its hydrodynamic models, the prediction of the river flow can also be used as the fresh water input.

Shin et al. (2004) developed a real time flood forecasting model to verify applicability of neural network for large river basin and to predict flood runoff of non linearity nature. Three NRDFM (neural river discharge-stage forecasting model) was developed in Nakdong river of Korea on Waekwan station for flood discharge. For simulation of flood runoff in Nakdong River that has a MFN with a back propagation algorithm was applied. The following conclusion was made:

- In forecasting without converting from rainfall to runoff the performance of NRDFM-II shows the best result. For short term flood forecasting NRDFM-III shows sufficient availability.
- All the three models show overestimated or underestimated result for long term forecasting than those of short term forecasting.
- NRDFM-I shows possibility for long term forecasting with correlation coefficient of more than 0.95.
- So these models can be effectively used for real time flood forecasting if they are extensively applied to flood warning station.

Senthil kumar et al. (2005) developed ANN model of Narmada up to Mandla site for flood forecasting. To select the input vector of the model the statistical parameters such as partial correlation, cross correlation and auto correlation of a series has been computed. Statistical indices like percentage error, root mean squared error, Nash-Sutcliffe model efficiency, coefficient of correlation were used to analyze the result of ANN models. The ANN model have been developed and compared with MLR model. It was concluded that ANN model performed better than MLR models from the validation and calibration results.

Kim et al. (2009) developed ANN with RDAPS (regional data assimilation and prediction system) and without RDAPS for inflow forecasting of a reservoir up to 12hr. Then the comparison was made between both the models. It was concluded that the performance of both the model was good comparing with the observed records. The ANN model with recorded rainfall data has more accuracy than that of ANN model using only RDAPS.

CHAPTER – 3

STUDY AREA AND DATA COLLECTION

3.1 THE STUDY AREA

3.1.1 The Mahanadi river basin

The Mahanadi basin is one of the major rivers of India. It is flowing from east part and draining into the Bay of Bengal. The Mahanadi is a peninsular river. It ranks second to the Godavari River in flood producing capacity and in water potential. The Mahanadi situated in east central India. It lies between north latitudes 19° 8' to 23° 32' and east longitudes 80° 28' to 86° 43'. The origin of the river source is situated 6 km from the Pharsiya village to the south of Nagri town in Dhamtari district of Chhattisgarh. Over sea surface at an elevation of 442 meters the headwaters of the Mahanadi River are situated. The main tributaries of Mahanadi River are Jonk, Seonath, Hasdeo, lb, Tel, Ong, and Mand.

The Mahanadi basin covers an area of 141,589 km^2 which is nearly about 4.3% of the total country geographical area. It is bounded by Eastern Ghats on the south and east, by the Central India hills on the north and by the Maikala range on the west. The river passes through the state of Orissa, and Chhattisgarh and small portion from Jharkhand, Madhya Pradesh, and Maharashtra. The river length is 851km from origin to the outfall and approximately 132,100 km^2 catchment basins. The 4.45% of the part of the basin is covered by water bodies and 54.27% is covered with agricultural land. The Chhattisgarh basin is saucer shape. It is a circular shape basin with diameter of 400 km and an exit passage of 16 km breadth and 160 km length. The basin physical regions of basin are the Eastern Ghats, the erosion plains of central table land, the coastal plains, and the northern plateau. Northern plateau and Eastern Ghats are hilly regions. The central table land is the central inerior region and is traverse by the river and its tributaries. The fertile delta area is the coastal plain. The main types of soils found in the basin are mixed red and black soils, red and yellow soils, deltaic soils, and laterite soils. The annual rainfall of the basin varies from 1143 mm to 2032 mm and average is 1438.1 mm. Temperature in the basin in December is ranging from 10°c to 13.7°c and in May it is 38°c over the hill region and 43°c in the plains.

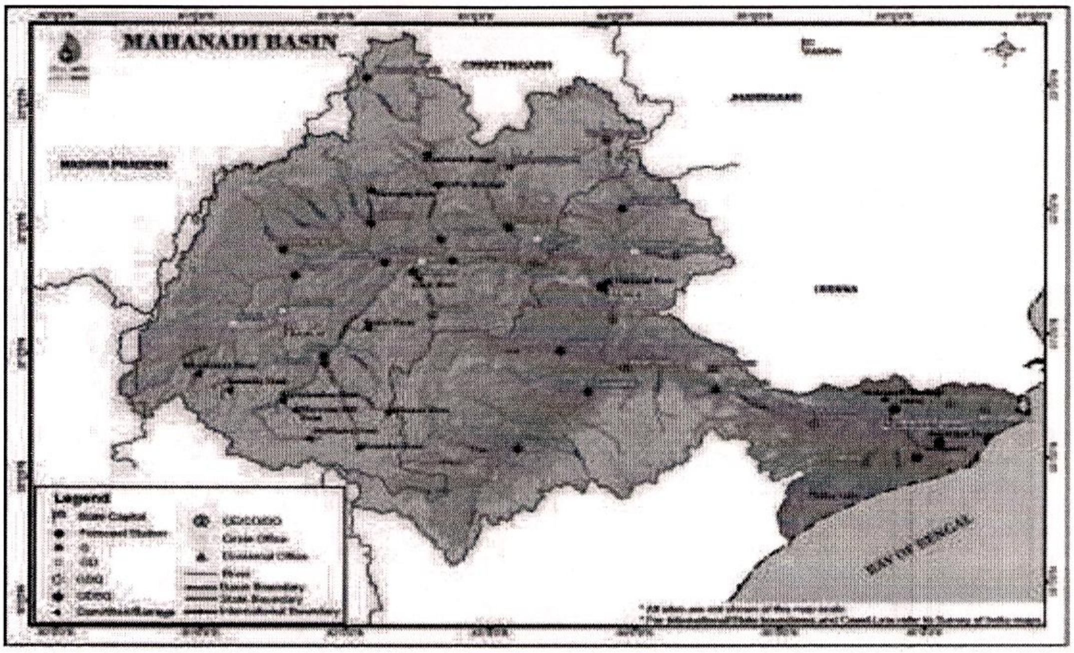


Fig: 3.1 Map of Mahanadi River

The highest relative humidity during July/August varies from 68% and 87% and average highest relative humidity is 82%. The lowest relative humidity during April/May varies from 9% to 45% and the average is 31.6%.

3.1.2 The Seonath sub basin

The Mahanadi basin longest tributary is Seonath River. Seonath river arises in village Kotgai, Durg district Chhattisgarh. It flows from three districts Durg, Bilaspur and Rajandgaon of Chhattisgarh. The Seonath River emanate near village Panabaras in district of Rajandgaon. The basin is established between 80^0 25' E to $82^0$35' E longitude and 20^0 16' N to 22^0 41' N latitude. The basin drainage area of the river is 30,860 km^2 up to meeting with the Mahanadi River. The river travels a length of 380 km. The main tributaries of Seonath River basin are Kharun, Tandula, Arpa, Agar, Hamp, and Maniyari rivers.

The districts covered by the Seonath sub-basin are Durg, parts of Bastara and Mandla of Madhya Pradesh, Raipur, Rajnandgaon, and Chandrapur district. The Sheonath and Mahanadi rivers consist of 58.48% of the state's water resources. The basin mean annual rainfall varies from 1005 mm to 1255 mm. Seonath river basin contains 25% of the Mahanadi basin of the catchment. Black soil and red soil are mainly found in the Seonath basin. Durg and Raipur are the two main significant urban centres in the basin.

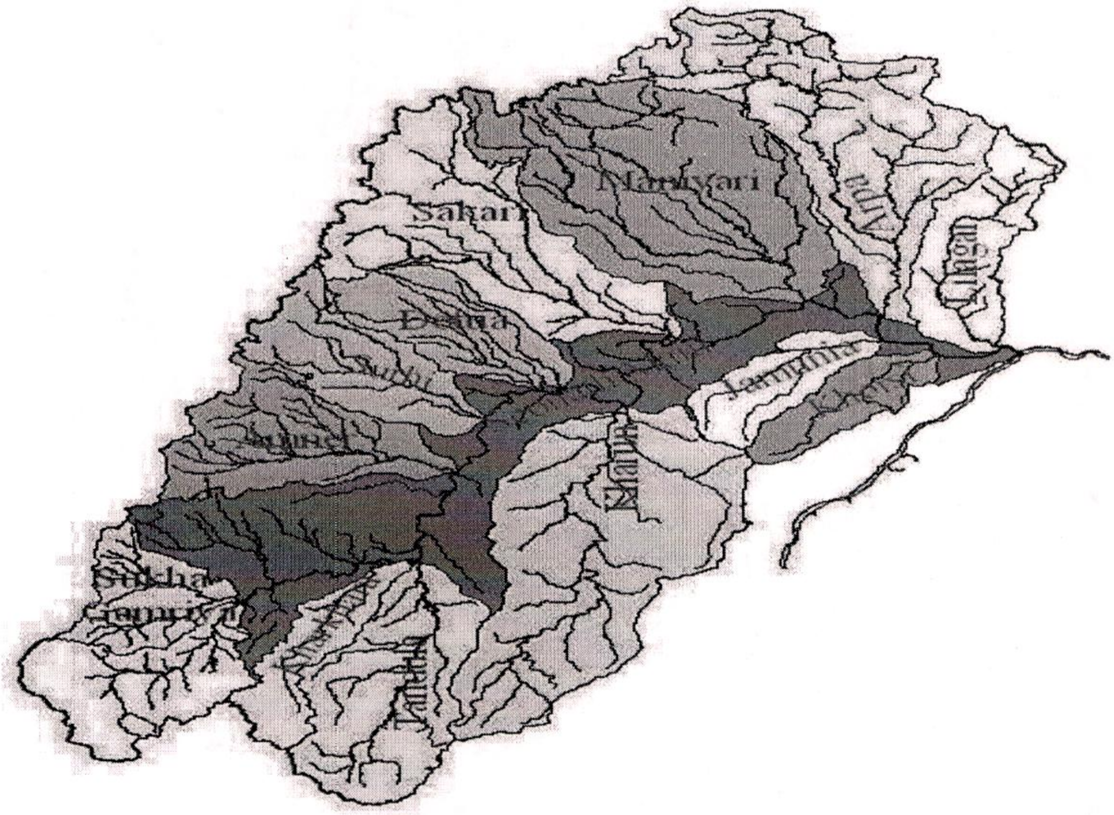


Fig: 3.2 Map of Seonath River

3.1.3 Hamp tributary

Hamp is a tributary of Seonath Sub-basin of Mahanadi River. The basin is established between 22^0 12' 24" N to $22^0$30'16" N latitude and $81^0$6'0" E to $81^0$30' E longitude in Kabirdham district of Chhattisgarh. The basin drainage area of the river is 2,210km^2. The stations selected for the research work are Chirapani, Bodla, and Pandariya. The drainage and thiessen's polygon map of Hamp are given in fig

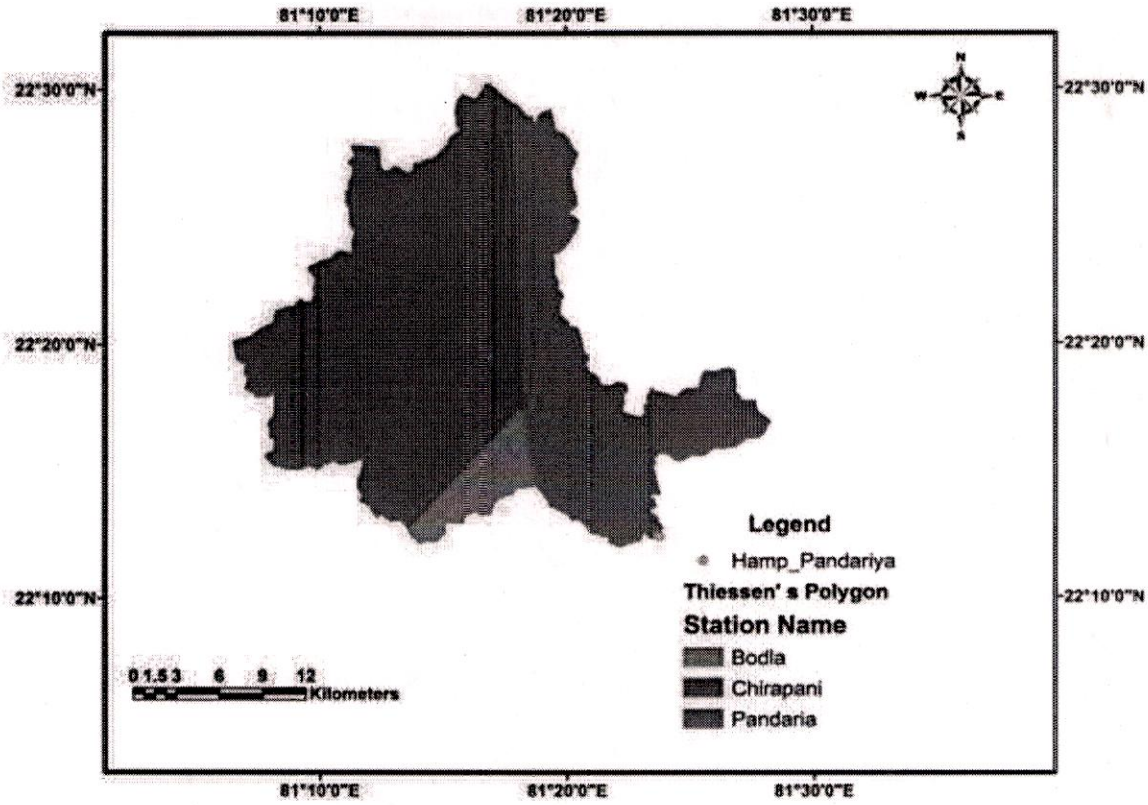


Fig 3.3: Hamp pandariya stations map

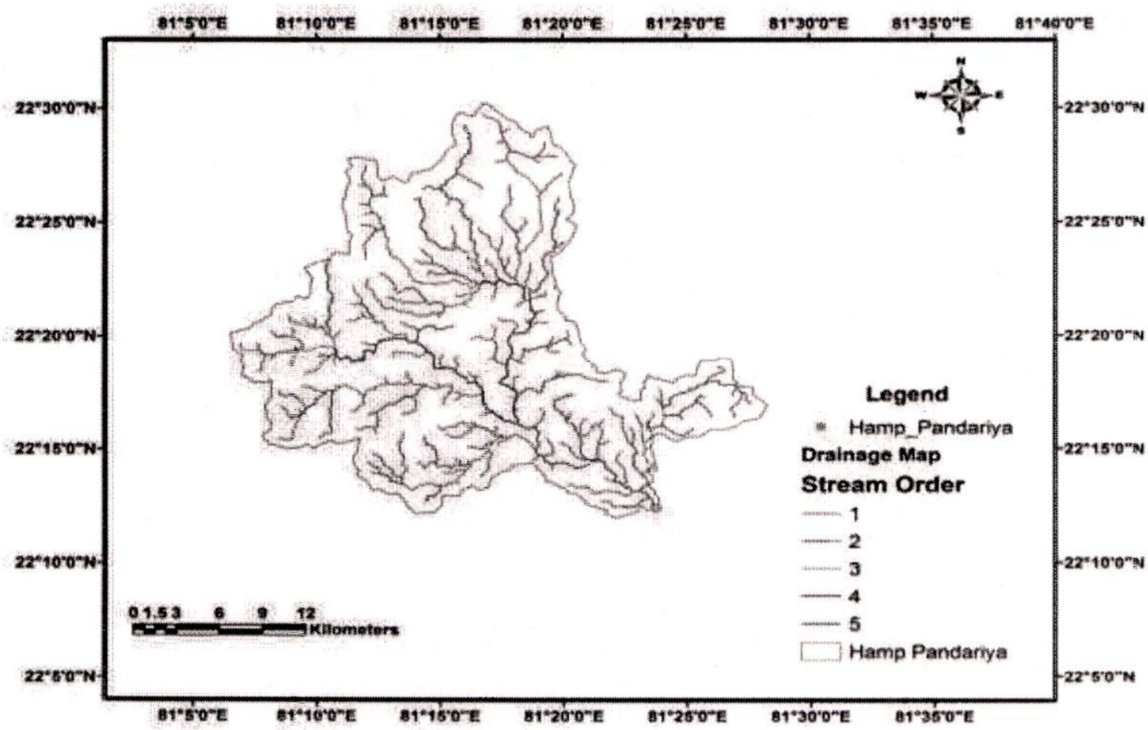


Fig: 3.4 Drainage map of Hamp River

3.2 DATA USED

The data used for this study comes mainly in the state of Chhattisgarh and Orissa. The Hamp river data is used for real time flood forecasting. The Hamp is a tributary of Seonath sub-basin of Mahanadi River. The rainfall in the catchment area Chirapani, Bodla, and Panadariya and the hourly discharge data is used to carry out this research work. The duration of data used is from 1891 to 2009. The data are collected from central water resource department of Chhattisgarh (CWC, Chhattisgarh) and then checked for the errors and processed according to the requirement of the research work. The data is very bulky that's why it is represented in graphical form. The prediction of river flow for seven lead hours uses the hourly rainfall and discharge. The hourly rainfall at Chirapani, Bodla, Pandariya, and discharge at Pandariya is given in figure below

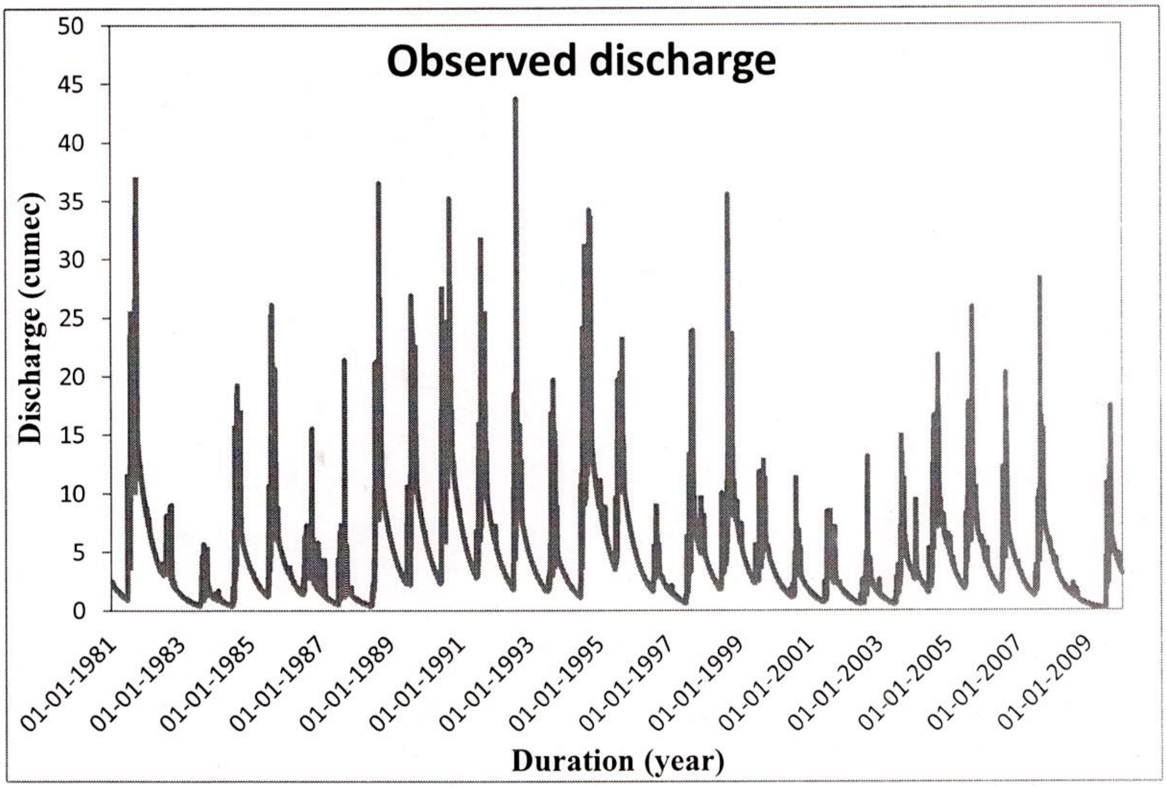


Fig: 3.5 The hourly observed discharge data at Pandariya

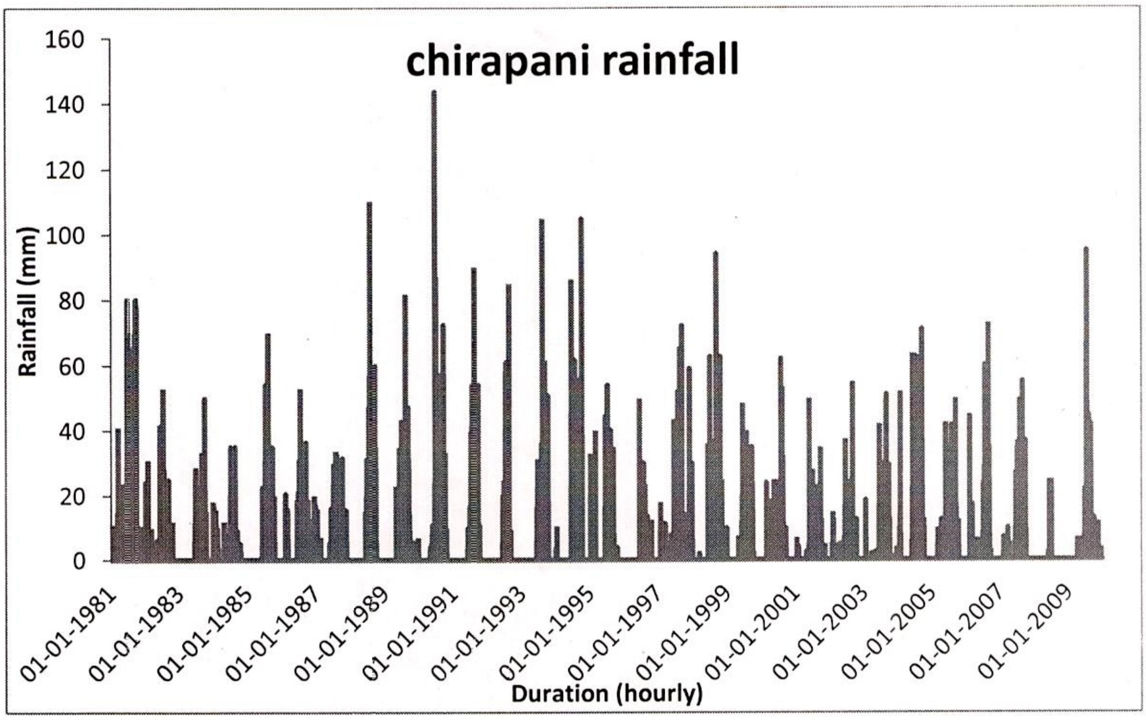


Fig:3.6 the hourly rainfall data at Chirapani

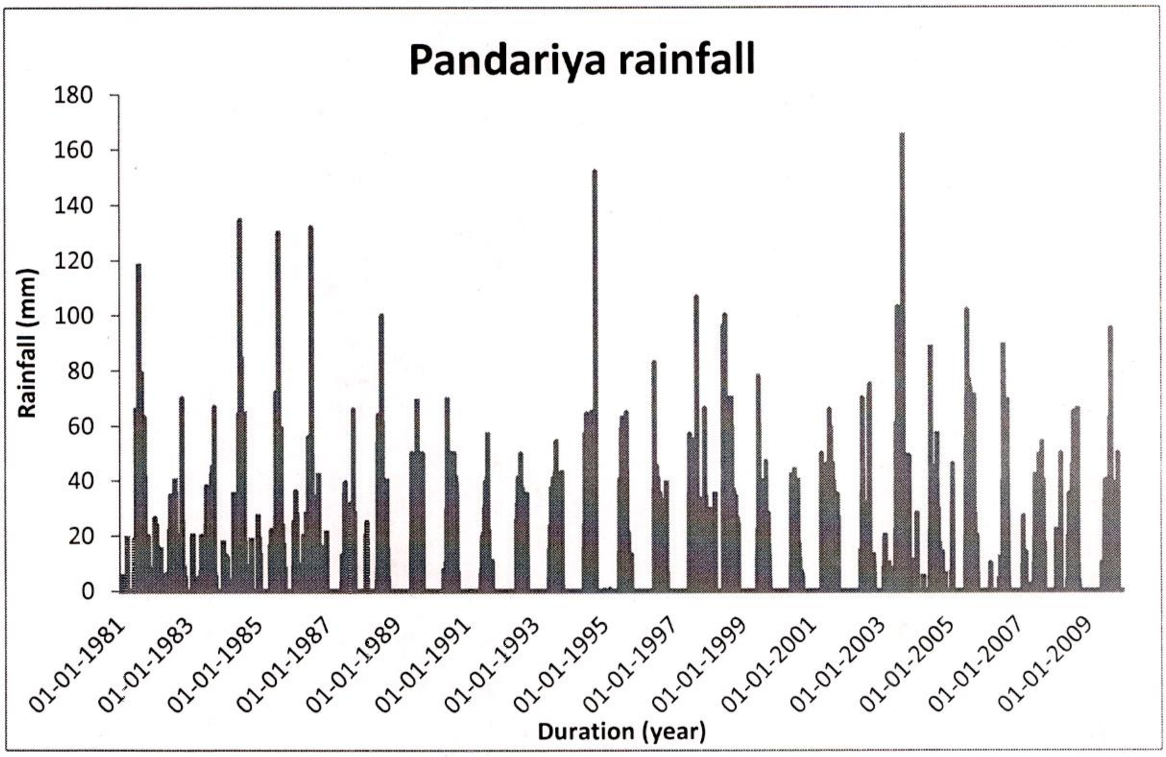


Fig:3.7 the hourly rainfall at Pandariya

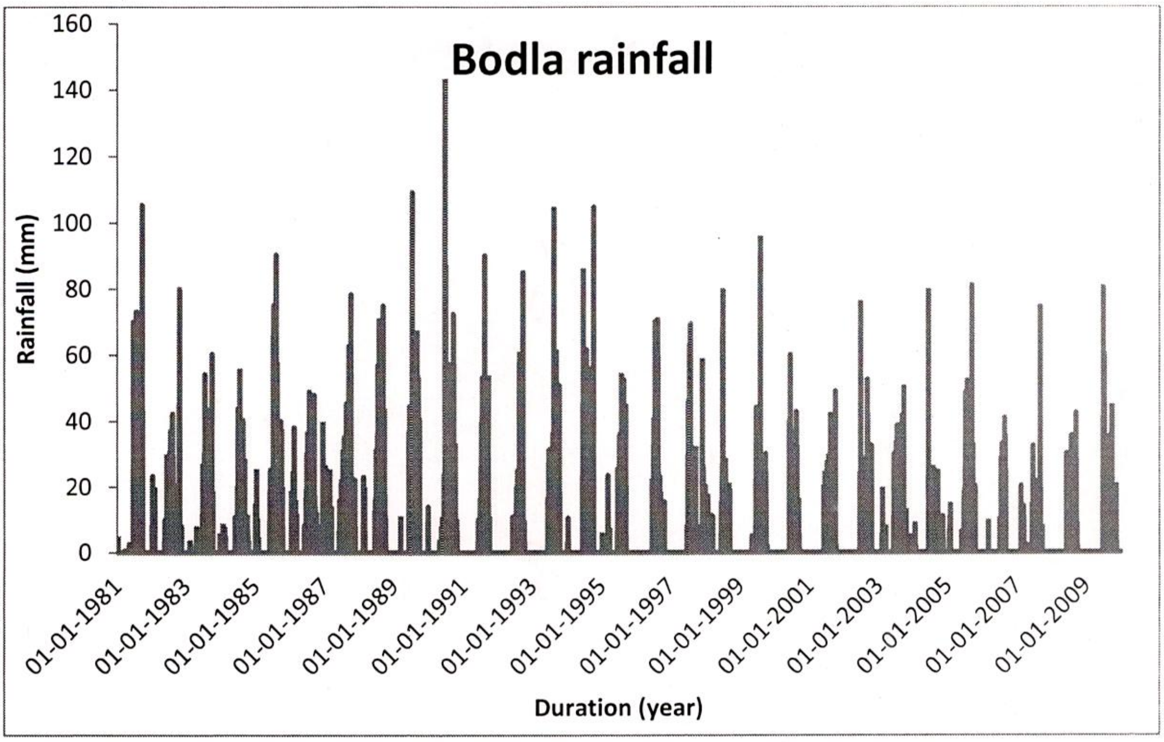


Fig: 3.8 the hourly rainfall at Bodla

3.3 DATA PROCESSING

Data processing is organizing the data into a feasible form. To produce the meaningful information data processing is done. Data processing includes all the function like storing data, updating data, rearranging data etc. it is a very broad term which includes all the activities from receiving of observed raw data and make them available in usable form. Many gaps and inconsistencies are found in the raw data when it is observed and recorded. To overcome these inconsistencies and gaps the raw data is processed. Many series of operations such as filling of missing values, making necessary validation checks, processing of raw data to estimate the required value and analysis of data for commonly required statistics are applied for processing the data.

CHAPTER – 4

METHODOLOGY

4.1 INTRODUCTION

Over the last two decades the use of artificial neural network has been increased in the hydrologic processes. Flood Forecasting is an important tool in reducing flood risk. To reduce the negative effects of the flood we do flood forecasting. It can be done by many methods. The technique used for flood forecasting model these days are ANN and Fuzzy logic using the catchment data available. . In last few years ANN models have been used successfully for flood forecasting. ANN gives improved performance than other black box models. The fuzzy logic and ANN based models, having the potential for the real time flood forecasting, are capable of considering the inherent non-linearties in the rainfall-runoff process. The main advantage of ANN models over other traditional models is that it does not require information about the complex nature of the underlying process under consideration to be explicitly described in mathematical form (Senthil kumar, 2005).

The following section gives the detailed description of the methods used for the development and simulation of ANN model for flood forecasting using the Hamp reservoir data.

4.2 ABOUT ANN

Artificial neurons (AN) mimic the functioning of a human brain by gaining knowledge through a process of learning which involves the determination of an optimal set of threshold values for the nodes and weights for the connections. Artificial neural networks (ANNs) becomes very popular for prediction and forecasting in many areas including power generation, environmental science, medicine, finance, and water resources (Maier and Dandy 2000). At present ANNs have proved to be an adequate alternative to traditional methods for hydrological modeling, such as rainfall – runoff modeling, stream flow forecasting and are discussed in detail in hydrologic papers (Minns and Hall, 1996; Danh et al., 1998; Zealand et al., 1999; ASCE, 2000 a,b; Maier and Dandy, 2000; Tokar and Momcilo 2000; Sudheer et al., 2002; Hu et al. 2005).

In the recent years an alternative method is developed for flow forecasting which is based on ANN. Recent studies have reported that for hydrological flow forecasting of stream ANN may offer a good alternative. ANN is a computer program which is design to model and is based on human brain and its ability to learn tasks. As in expert system ANN is not rule based like the other form of computer intelligence. An ANN is trained to generalized and recognize the relationship between outputs and inputs.

Early ANN was inspired by the perception of human brain. In the recent years ANN technological developments have been achieved. Now it is more of mathematical technique with some similarities to the human brain. ANNs contain mainly two characteristics of the human brain the ability to learn and the ability to generalize from limited information as a primary features.

Both biological and artificial neural networks use neurons or interconnected simple processing elements. The knowledge stored in ANN as the strength of the interconnecting weights and is modified using a learning algorithm through a process called learning. To modify the weights of ANN in an orderly fashion this algorithmic function in a combination with learning rule or back-propagation is used.

ANN is not programmed like other computer applications it is taught and to a particular problem it give an acceptable answer. In ANN input and output values are sent then initial weights in the architecture of the ANN are assigned. The ANN frequently adjusts these interconnecting weights until it will get output values which match the original output values. The neural network uses this weighted matrix of interconnection to learn and remember.

When ANN is used to solve a problem the ANN is trained to learn the relationship between the output and input values. This is done by giving the network the known values of input and output in conjunction with a learning algorithm. Then the ANN plots the relationship between input and output and determines the best relationship by modifying its internal functions. The inner processing and working of ANN are usually thought of as a black box model with output and inputs.

The ANN consists of three main layers the input layer, the hidden layer, and the output layer as shown in figure 4.1.

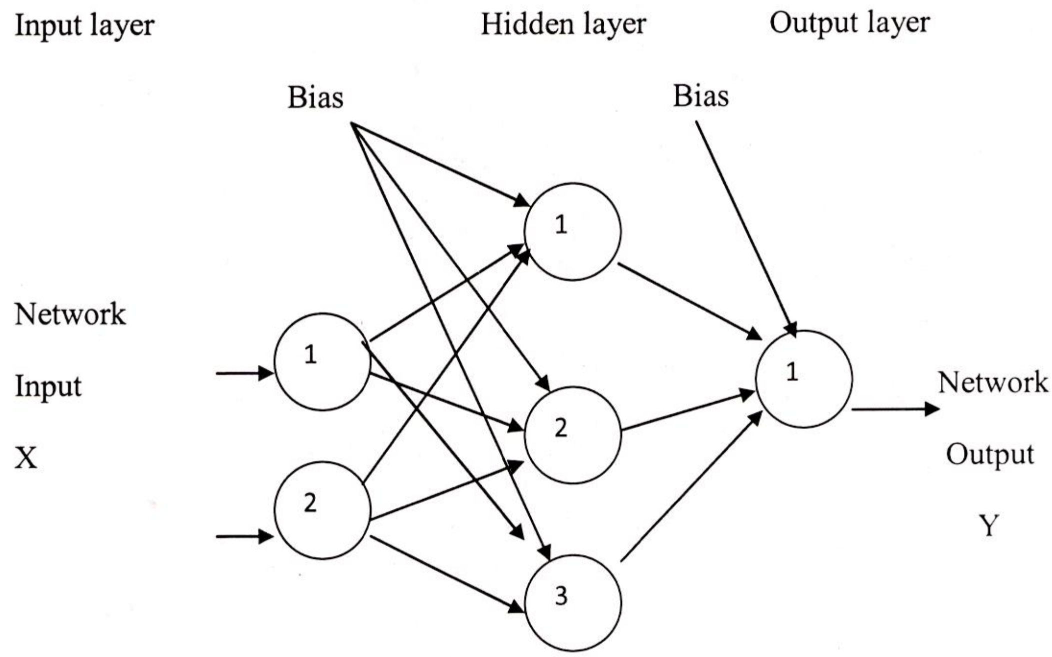


Figure: 4.1 ANN network structure

The input layer consist the raw information which is feed into the network. The hidden layers are where data are processed. In the output layer the results of given outputs are produced. The output layer behavior is depending on the hidden layer activity and the weights between output and hidden layers.

In general, a neuron can have n number of input which can be labeled from 1 to n. As in given figure 4.1 neuron 3 have n=2 in the hidden layer. Each of the neuron has an input which is known as bias and is equal to 1.0. In the previous layer each neuron (j) collects the information from every node (i). With each input (x_i) a weight (w) is associated to the node (j). To the node (j) the effective incoming information (NET_j) is the weighted sum of all the incoming information and also said as the net input computed as:

$$NET_j = \sum_{i=0}^{n} w_{ji} x_i \tag{4.1}$$

Equation (4.1) is implemented to the nodes in the hidden layers and output layers. To obtain the output from the neuron, the weighted sum of the input layer is passed through an activation function which is also called as transfer function. The transfer function helps in determining the nonlinearity presented in the function which is going to be mapped by introducing some nonlinearity in the network. The most commonly used transfer function is the sigmoid function (ASCE, 2000a) which is given as follow:

$$OUT_j = \frac{1}{1+e^{-NET_j}} \tag{4.2}$$

The interconnected weights are adjusted by using a learning algorithm such that the output from the ANN model is very close to the observed values by minimizing the error through a mathematically formulated procedure and is called as training of network (ASCE, 2000a).

To learn the relationship between input and output pairs ANN model can be trained for a given problem. The applicability to a variety of different problems the feed forward ANN is generally used in all studies (Hsu et al., 1995). However, there are no guidelines are given to develop a good architecture of ANN although some suggestions are given by the researchers which can be implemented to develop an ANN model. It is suggested that not more than one hidden layer is required in feed forward networks because a three-layer network can generate arbitrarily complex decision regions (Maier and Dandy, 2000). Also, the appropriate input vector to the ANN model can be identified according to the procedure of (Sudheer et al., 2002).

4.3 ANN MECHANISM

ANN architecture has mainly two basic components connections and neurons. The main objective of ANN is to find out the connection weights and neurons arrangements. As discussed above neurons have three layers: input, hidden, and output layers as shown in figure 4.1. The neurons are not only for receiving input signal but also get output information through connection weights with a selective strength to the input access of other neurons. By using their output function all the neurons determine their outputs and then this result may be used for next step of processing by putting this to the neighbor neurons. The intermediate value for each neuron which include the weighted sum of input values

$$I = \sum_{i=0}^{n} w_{ij} x_i \tag{4.3}$$

Where, x_i is input value, w is weight of input values, i is the input source number, and j is the target neuron number. Then this value is processed through a transfer function f(i). To determine the activation level of a neuron transfer function is used which operate a non-linear squashing operation. The neural network is designed to find out the weights of the connection and the procedure by which the information passes through the neural network. It is the most difficult task to find the appropriate neural network for the model building. The neural network developed in this study is feed forward network which is commonly used for flood prediction.

The neuron microstructure processing is given in figure 4.2. User can select the transfer function as per their requirement whereas most of the software have certain common options available like linear, logistic (sigmoid), hyperbolic tangent, and Gaussian transfer functions. The information given to the neural in the form of input is processed by it into the output form. All of these units are simplified neuron models which transform the input information into the output. This process cannot be done without the activation of neuron which is the weighted sum of all the inputs. This activation process is done by using an activation transfer function.

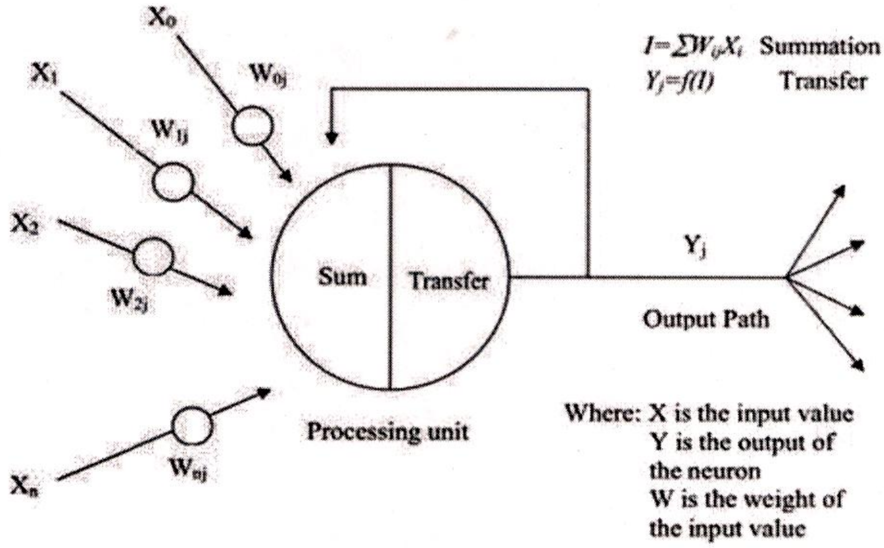


Figure 4.2 the microstructure of neurons in network (Haijie Cai, 2010)

4.3.1 Transfer functions

4.3.1.1 Logistic (sigmoid)

For most of the NN application this function is very useful. The values map by this function is in the (0, 1) range. When outputs are categories this function is used.

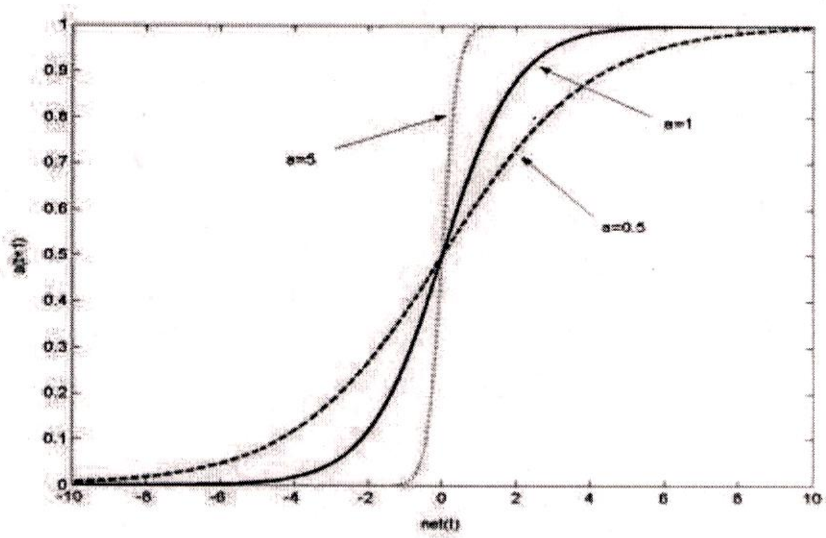


Figure 4.3 sigmoid activation functions (google images)

4.3.1.1 Linear function

This function is used only for the output layer. If output is a continuous variable then this function is useful. For complex network modeling this function is not powerful, it sometime restrict the network from generating the output with numerous error near the maximum or minimum of output scale that means the result is more consistent. For this function it is more suitable to use small momentum, learning rates, and initial weight sizes. Otherwise, the more and more errors will be produced by the network. This function will not work if on the output layer there are large number of connections because the sum of the total weight generated will be more. It is given as follow:

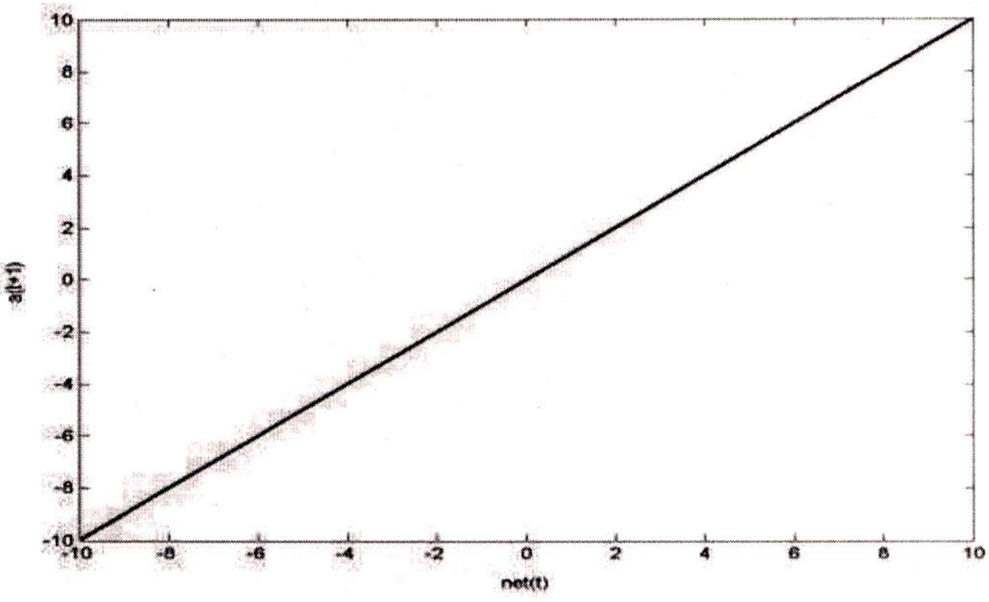


Figure 4.4 linear activation function (N.J. de Vos, 2009)

4.3.1.3 Hyperbolic tangent function

Hyperbolic tangent function can be used almost exclusively. It can be used for continuous output values especially if on the output layer linear function is used. This function scale of input is [-1, 1]. Good results can be obtained if hyperbolic tangent is used for the hidden layer and for the output layer logistic or linear function is used.

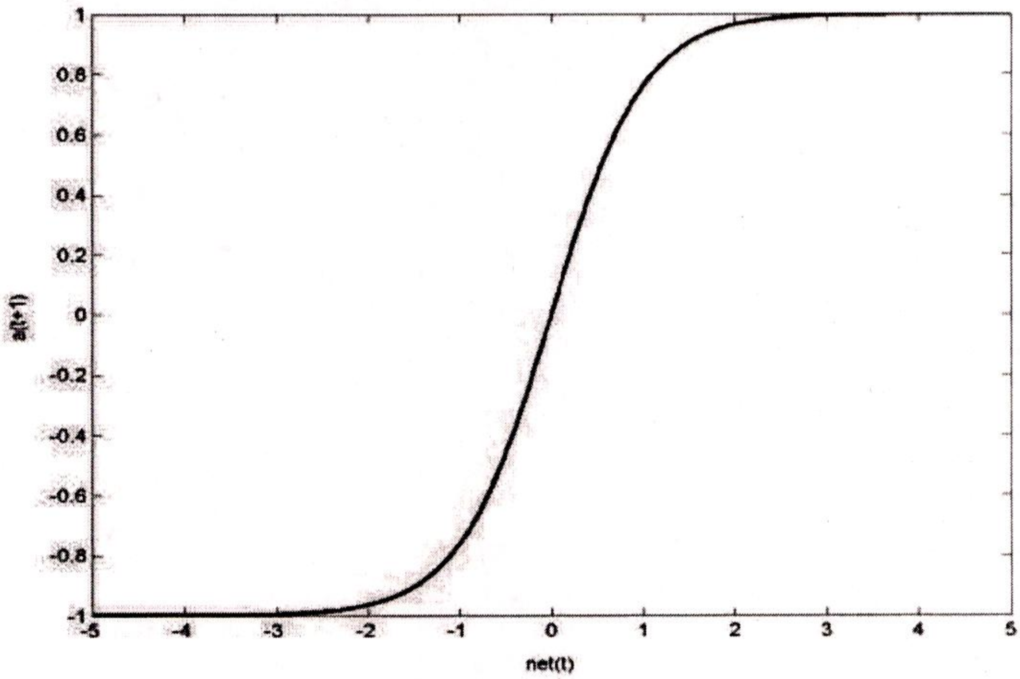


Figure 4.5 hyperbolic tangent activation function (N.J. de Vos, 2009)

4.3.1.4 Gaussian function

This function is not like other function it is unique function. It is not an increasing function like others. Its curve shape is classic bell like shape. In this mid range values can be mapped into high ones and high can be in low ones. It is useful in small set of problems. The output produce by this function is in the range of [0, 1].

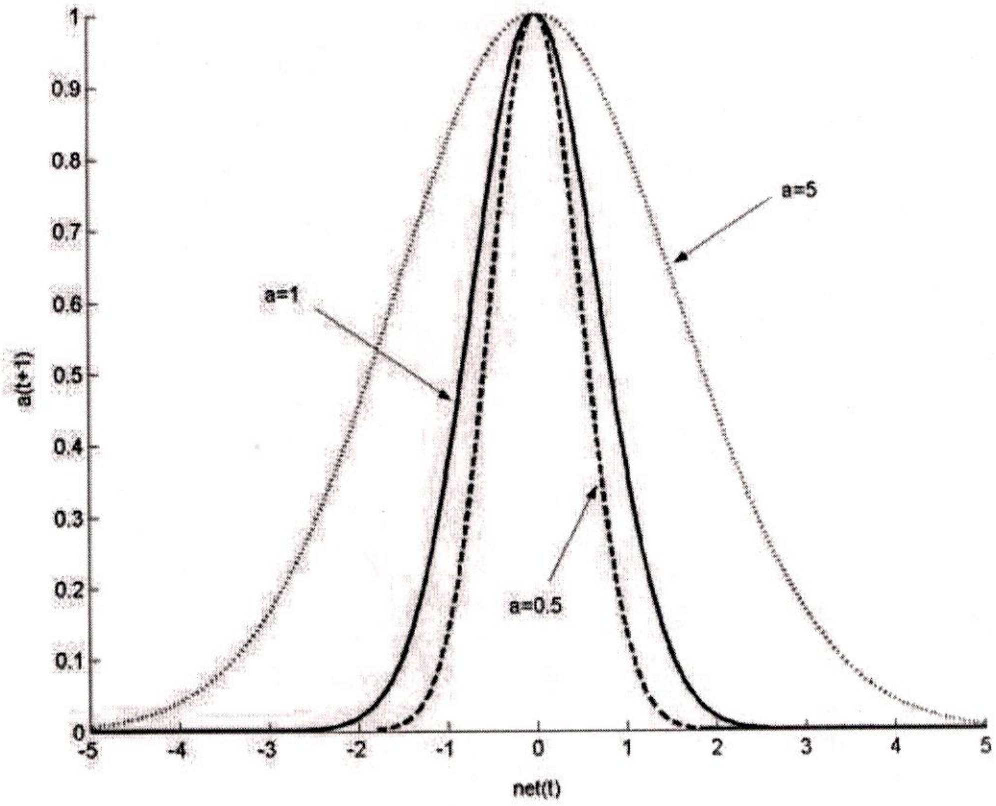


Figure 4.6 Gaussian activation function (N.J. de Vos, 2009)

4.3.2 Normalization of Input Data

Before passing the input values into a neural network it should be normalized to the range from 0 to 1 because the output is bound from 0 to 1 of the sigmoid function (Minns and Hall, 1996). Many researchers have given the importance of the normalization of data Minns and Hall (1996), Dawson and Wilby (1998). To get the meaningful results the output from the neural network should be renormalized.

The equation used for normalising the set of data is given below:

$$N_i = \frac{R_i - Min_i}{Max_i - Min_i} \tag{4.4}$$

where R_i is the real value applied to neuron i; N_i is the subsequent normalized value calculated for neuron i; Min_i is the minimum value of all values applied to neuron i; Max_i is the maximum value of all values applied to neuron i. (Senthil kumar)

4.3.3 Training of ANN

Training is a process in which the connection weights are adjusted according to the given algorithm to generate output. There are mainly two type of training methods supervised and unsupervised. (Jain and Deo 2006)

In supervised method external guidance is required for training process which signifies that a large number of input output patterns are needed. Training process minimize the error functions by selecting a set of threshold values and connection strengths which help the ANN to give the output value which are near or equal to target value. After the process of training the ANN is capable to generate the required results. Whereas in an unsupervised training the external guidance is not required. In this the input data set is given to the network which automatically change the connection weights (ASCE, 2000a).

In training the input output data change its weight values, according to algorithm determined and the environment in which network is embedded. The main focus of training a network is to accomplish an output value $Y=(y_1,y_2,y_3....,y_p)$ which is close to the target value $T=(t_1,t_2,t_3....,t_p)$ when an input value $X=(x_1,x_2,x_3,....,x_p)$ is given to the ANN. In training process the bias vector V and the weight matrices W is determined by reducing the predetermined error function as given below:

$$E=\sum_{P}\sum_{p}(y_i-t_i)^2 \tag{4.5}$$

here t_i is the desired output component T; y_i is the ANN output; p is the number of output nodes; and P is the number of training patterns.

4.3.4 Back propagation algorithm

This algorithm involves minimization of the total error by using the gradient descent method. The network weights and biases are adjusted by moving a small step in the direction of the negative gradient of the error function during iterations. The iterations are repeated until either a specified convergence is reached or a number of iterations are over. Single iteration is over when all training patterns are finished. The iterations are continued until the overall mean squared error for all output nodes and training patterns reaches to a minimum (jeong, 2000).

While using this gradient descent method two problems may occur. First, the convergence may show oscillatory behavior and work slowly. Second, the prefixing of network architecture is needed by trial.

Back propagation is the most commonly used algorithm for the training of the feed forward ANNs (Hsu et al., 1995; Dawson and Wilby, 1998; Tokar and Jhonson, 1999; Zealand et al., 1999; Thirumalaiah and Deo, 2000; ASCE, 2000a). From the input layer each input pattern of the training data is passed through the network to output layer. Then the comparison is made between the network output and the desired target output, and an error is determined based on above equation (4.5). Then the determining error is back propagated through the neural network and the weights are revised using the given equation

$$\Delta w_{ij}(n) = -\varepsilon * \frac{\partial E}{\partial w_{ij}} + \alpha * \Delta w_{ij}(n-1) \tag{4.6}$$

here $\Delta w_{ij}(n)$ and $\Delta w_{ij}(n-1)$ are weight increments between node i and j during nth and *(n-1)*th time, or epoch (ASCE, 2000a). For correction of bias value similar equation is used. In the equation (4.6), ε is learning rate and α is momentum. In the flat region of error surface the momentum factor speed up training. It also help in avoiding the oscillations in the weights. A learning rate is the rate at which weight is changed after the completion of first iterations. To increase the progress a high value is required at the starting point. It also increases the chance to prevent the training process being captured in local minima instead of global minima.

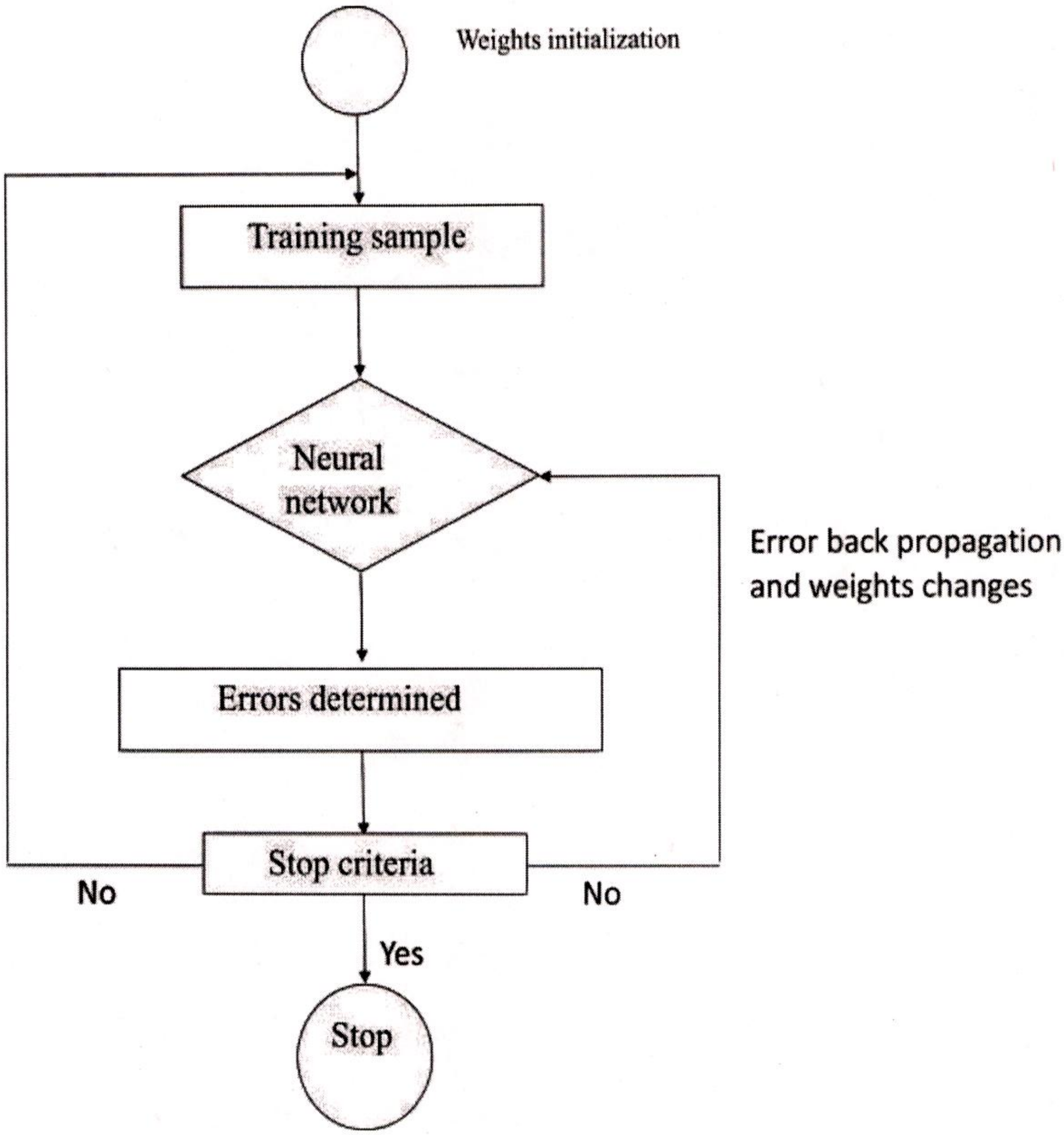


Figure 4.7 Flow chart representation of error back propagation

4.3.5 Performance Evaluation of ANN Model

the entire data is divided into two parts one is for calibration and one is for validation of ANN model which is based on time series statistical properties like mean and standard deviation. The result of calibration and validation are checked by the performance indicator which are RMSE (root mean square error), EFF (model efficiency) and CORR (correlation coefficient). (Senthil kumar,) they are given as follows:

RMSE (Root mean squared error) $= \sqrt{\frac{\sum_{k=1}^{K} (t-y)^2}{K}}$ (4.7)

EFF (Efficiency) $= 1 - \frac{\sum (t-y)^2}{\sum (t-\bar{t})^2}$ (4.8)

CORR (Coefficient of Correlation) $= \frac{\sum TY}{\sqrt{\sum T^2 \sum Y^2}}$ (4.9)

Here, K represents the number of observations; t represents the observed data; y represents the computed data; $T = t - \bar{t}$ where $\bar{t}$ is the observed data mean; and $Y = y - \bar{y}$ where $\bar{y}$ is the computed data mean.

4.4 MODEL CATEGORIES

Feed forward network with back propagation algorithm and multiple linear regression (MLR) methods are used for modeling.

4.4.1 Feed forward network

Neurons commonly have multi-layer structures connected in parallel which make it possible that input signals are transmitted feed-forward. It is called a MFN and the network is composed of input layer, output layer, and hidden layer (Shin et al. 2004). In a feed forward network firstly the input node is feed with the input quantities then these values are passed on to the hidden layer node after assigning a weight. Then the weighted input which is received from each input node on the hidden layer is associated with a bias and then the result is passed on to a nonlinear transfer function. Then this result is passed on the output node. The output node works same as hidden node. The network is trained before applying the network to any problem. By adjusting the bias and weights using training algorithm the error between the target output and the network output is minimized.

The architecture of a feed forward ANN can have many layers where a layer represents a set of parallel neurons. The basic structure of ANN usually consists of three layers:

- The input layer: where the data is feed to the network.

- The hidden layer or layers: where data are processed.
- The output layer: where the results of given outputs are produced.

The neurons in the layers are interconnected by strength called weights (Senthil kumar, 200). ANN of three layered feed forward figure is shown in figure 4.2

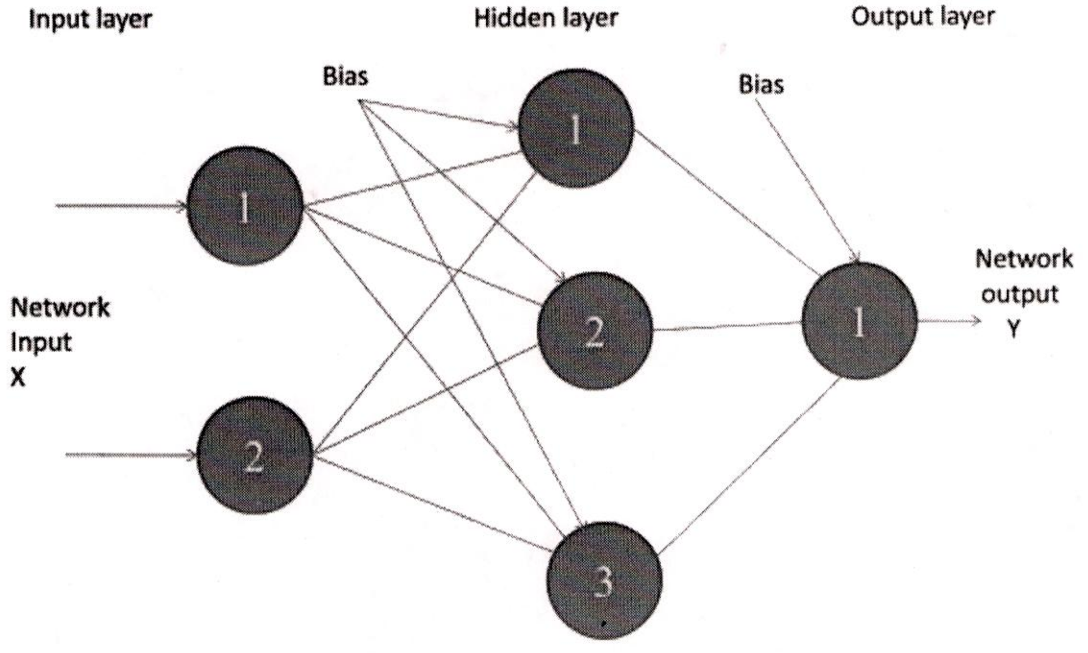


Figure.4.8 A three layer feed forward ANN (Senthil kumar).

4.4.2 Multiple linear regressions (MLRs)

An MLR model is the simplest and well-developed representation of a causal, time-invariant relationship between an input function of time and the corresponding output function. Hence, an MLR model is developed as the benchmark for comparison in flood forecasting.

4.5 ADVANTAGE AND LIMITATION OF ANN

ANN has the capability to process the data, ability to learn and establish the output. In this the networks have the collective power which allows the implementation of complex number of task and give efficient result. Some of the list of limitations and advantages of ANN in hydrological modeling are given below.

1. Non-linearity: In ANN non linear processes can be modeled. The hydrological problems which occur are non-linear in nature like rainfall. The interconnections

present in neurons of ANN structure develop the non-linear structure which is divided across the network. This feature allows the modeling or processing of non-linear data like conversion of rainfall into runoff.

2. Ability of modeling the relationship of input output: There is no need of mathematical equations in ANN to represent the relationship between input and output. So ANN can be used where the processes are not understood for developing the simple models.

3. Flexibility: one observation about ANN is that it is static in nature. However they are able to change or they are adjustable in nature. As new data is available they are able to adjust the weights. By using this feature ANN becomes more useful for the non-stationary process treatment. The model can be able to learn continuously if learning strategies are designed in real time.

4. Quick modeling: as compared to the physical and conceptual based model the ANNs construction process is very fast. However this will require a perfect choice like input variable to be used, the network architecture, the lag time, which training algorithm can be used, etc. whereas if clear guidance and procedures are not provided on ANN construction it will make it subjective process. In this case trial and error method can be used. This method is never exhausted.

5. Efficient performance: this technique is related to the training of ANN. Once ANN is trained it become very efficient and run very fast (ASCE, 2000a). Many of the training algorithms used are efficient or through research development have been made like second order method is the development of first order method as in backpropagation.

6. Noisy data sensitivity: ANN models are distributed in nature so they have the ability to handle the noise present in input data (Thirumalaiah and Deo 1998a, b; Zealand et al., 1999, ASCE, 2000a). A large number of data is needed for the training process of the network to get an effective ANN which handles the problem of noise. So this will totally depend on the data quality and quantity.

7. Modularity: ANN can be easily developed in the form of modular architecture to solve any task or subtask related to any problem efficiently. These tasks include pattern recognition procedure or functional approximation. The ANN establishment within a decision or expert support system is an example.

8. Black box nature: the black box nature of ANN is its main disadvantage which makes it less used as compared to conceptual and physical based models. All the research (ASCE 2000b; Maier and Dandy, 2000; Dawson and Wilby, 2001; Maier et al., 2010) have tried to solve the black box theory and trying to find the physical meaning of this in ANN in the research area where more attention is needed. In some of the research this is addressed (Wilby et al., 2003; Sudheer and Jain, 2004) but more research is required in this area.

The advantages explained above are taken from the application of ANN in the area of hydrology whereas the limitation gives the area for further research.

4.6 SUMMARY

This section gives the application of artificial neural network in hydrologic modeling with respect to the forecasting of river flow or rainfall runoff modeling. This chapter includes the brief review of the artificial neural network (ANN) in flow forecasting and use of ANN and the modeling approaches. This chapter also gives the explanation about the terms and methodology used in the modeling of flow forecasting. This section start with ANN detailed overview and followed by the terms, definitions used in ANN, structure of ANN, algorithm used for ANN modeling, training, advantages and disadvantages of ANN. This section also describes the working of ANN and how flood forecasting model is made in ANN. This section is concluded by the performance and modeling of ANN and also by the important themes which are arise from the flow forecasting of river using ANN literature and how the thesis work will be accomplished.

CHAPTER – 5

RESULTS AND CONCLUSION

5.1 INPUT VECTOR SELECTION

The selection of the important input variable for ANN model development is the most important step. To make a perfect choice of input vector it is very important for getting the good modeling result also it is a bit difficult task. Usually, not all of the potential input variables will be equally informative, because some may be correlated, noisy, or have no significant relationship with the output variable being modeled (Maier and Dandy, 2000). The input variables of ANN model are selected on the basis of autocorrelation, partial autocorrelation and cross-correlation between the given input variables. For real time flood forecasting the ANN model is generally developed using the rainfall and discharge value of the given stations as input vector.

The determination of number of antecedent rainfall and discharge values involve the determination of the lags of rainfall and discharge concentrations which influence the prediction of discharge. The different lags and bounds of these values can be very well established through statistical analysis of the data series as given in sudheer et al., 2002. The selection of input vector is based on trial and error method. The selection of good input vector for ANN model is determined by the simple correlation between the independent and dependent variables as the rainfall and discharge. For determining the input vector the autocorrelation, partial autocorrelation and cross-correlation of the following variable is identified.

- The autocorrelation of the discharge value at Pandariya
- The partial autocorrelation of the given discharge value at Pandariya
- The cross-correlation between discharge and Chirapani
- The cross-correlation between discharge and Boadla
- The cross-correlation between discharge and Pandariya

This correlation technique helps in finding the possible good input variable for forecasting the flow but it is not confirm that the values given by this are exact or not, so mostly trial and error technique is adopted for determining the significant lag values of input variable. Sudheer et al. (2002) have suggested a statistical procedure that avoids the trial and error procedure. They have reported that the statistical parameters such as auto-correlation function (ACF), partial auto-correlation function (PACF) and cross-correlation function (CCF) could be used to find out the significant lag values of input variables.

The auto-correlation coefficient (Salas *et al.*, 1980) is given as

$$r_k = \frac{\sum_{t=1}^{N-k}(x_t - \overline{x_t})(x_{t+k} - \overline{x_{t+k}})}{\left[\sum_{t=1}^{N-k}\left(x_t - \overline{x_t}\right)^2 \sum_{t=1}^{N-k}\left(x_{t+k} - \overline{x_{t+k}}\right)^2\right]^{1/2}} \tag{5.1}$$

Where r_k is called the *lag-k* correlation coefficient, the serial correlation coefficient or the auto-correlation function (ACF), x_t is the time series for *t = 1,..,N*, x_{t+K} is the lagged time series for *t = 1, ..., N-k*, $\overline{x_t}$ is the sample mean for *t = 1, ...,N*, $\overline{x_{t+k}}$ is the sample mean *for t = 1,...,N-k*, *N* is the sample size. The partial auto-correlation coefficient may be obtained by given equation below (Salas *et al.*, 1980)

$$\phi_1(1) = \rho_1, \phi_1(2) = \frac{\rho_1(1-\rho_2)}{(1-\rho_1^2)}, \phi_2(2) = \frac{\rho_2 - \rho_1^2}{(1-\rho_1^2)}$$

$$\phi_k(k) = \frac{\rho_k - \sum_{j=1}^{k-1}\phi_j(k-1)\rho_{k-j}}{1 - \sum_{j=1}^{k-1}\phi_j(k-1)\rho_j}$$

$$\phi_j(k) = \phi_j(k-1) - \phi_k(k)\phi_{k-j}(k-1) \tag{5.2}$$

To determine the partial auto-correlation function from a sample series $x_1, \ldots, x_N$, the sample autocorrelation the *ρ's* are replaced by *r's*. *ρ's* are auto-regression coefficients. The cross-correlation coefficient is given as (Salas *et al.*, 1980)

$$r_k^{ij} = \frac{\sum_{t=1}^{N-k}\left(x_t^{(i)} - \overline{x_t^{(i)}}\right)\left(x_{t+k}^{(j)} - \overline{x_{t+k}^{(j)}}\right)}{\left[\sum_{t=1}^{N-k}\left(x_t^{(i)} - \overline{x_t^{(i)}}\right)^2 \sum_{t=1}^{N-k}\left(x_{t+k}^{(j)} - \overline{x_{t+k}^{(j)}}\right)^2\right]^2} \tag{5.3}$$

where r_k^{ij} is the lag-k cross-correlation coefficient, $x_t^{(i)}$ is the time series values of series i, $x_t^{(j)}$ is the time series values of series j, $\overline{x_t^{(i)}}$ is the mean of the first N-k values of series i, and $\overline{x_{t+k}^{(j)}}$ is the mean of the last N-k values of series j.

The ACF and PACF of rainfall and discharge at Hamp are presented in Figures 5.1 and 5.2 respectively. The CCF between discharge and rainfall values at Chirapani, Bodla, and Pandariya are presented in Figures from 5.3 to 5.5 respectively. The PACF of the rainfall and discharge at Hamp with 95 % confidence levels and CCF of discharge values at Hamp between rainfall of all stations Chirapani, Bodla, and Pandariya suggest the most valuable input vector for ANN modeling for real time flood forecasting.

The auto-correlation function shows the smooth curve which indicate that the rainfall discharge value at Hamp is autoregressive. The partial auto correlation coefficient (PACC) of time series helps in determining the order of the auto-regression.

The partial auto-correlation coefficient of discharge value at Hamp for lag 1 is 0.94. The partial auto-correlation coefficient values for other lags are less than 0.12. The cross-correlation coefficients of discharge at Hamp with rainfall at Chirapani, Bodla and Pandariya for lag 0 are 0.36, 0.24 and 0.18 respectively. The cross-correlation coefficient of discharge at Hamp for rainfall at Chirapani for lag 0 that is 0.36 is higher than all other lagged cross-correlation coefficient values of other lags so for Chirapani input value selected is (t) whereas, for Bodla and Panadariya the cross-correlation for lag -1 is 0.25 and 0.21 which is higher than all other lagged cross-correlation coefficient values so the input variable selected is (t-1).

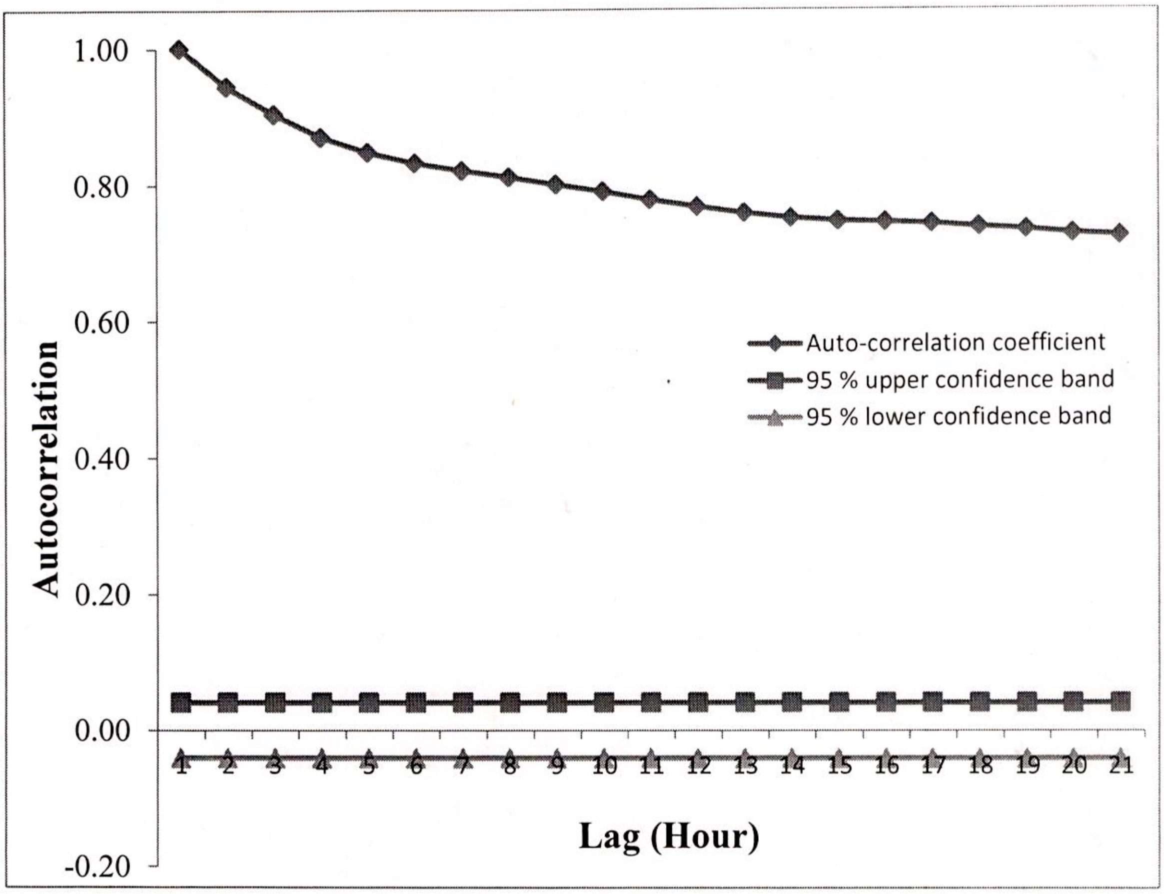


Figure 5.1 The autocorrelation of the runoff value at Hamp

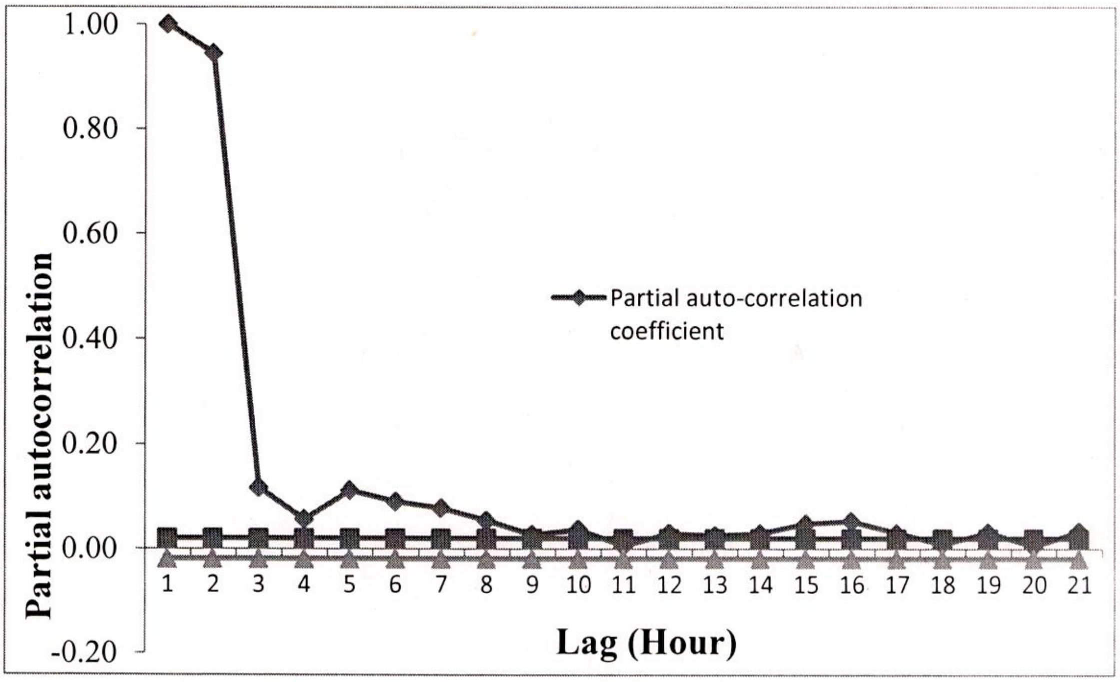


Figure 5.2 The partial autocorrelation of the runoff value at Hamp

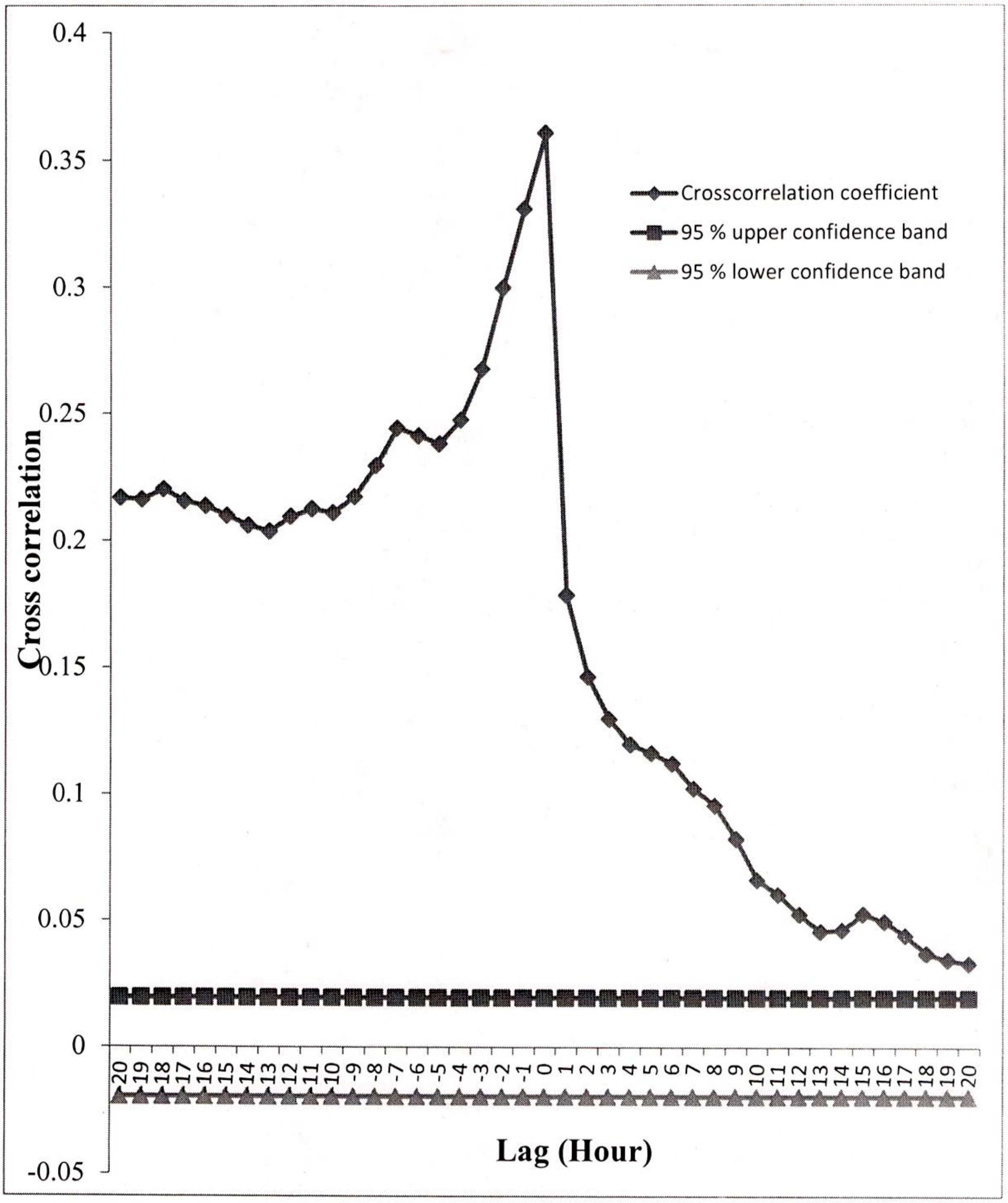


Figure 5.3 the cross correlation of discharge and rainfall at Chirapani

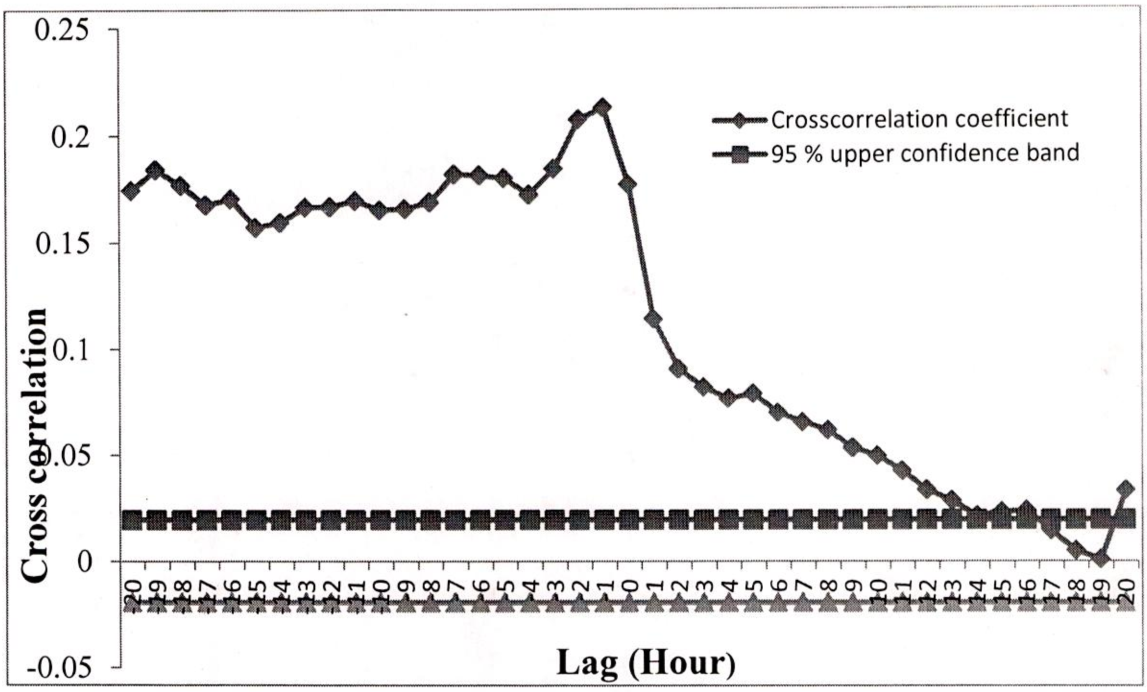


Figure 5.4 the cross correlation of discharge and rainfall at Bodla

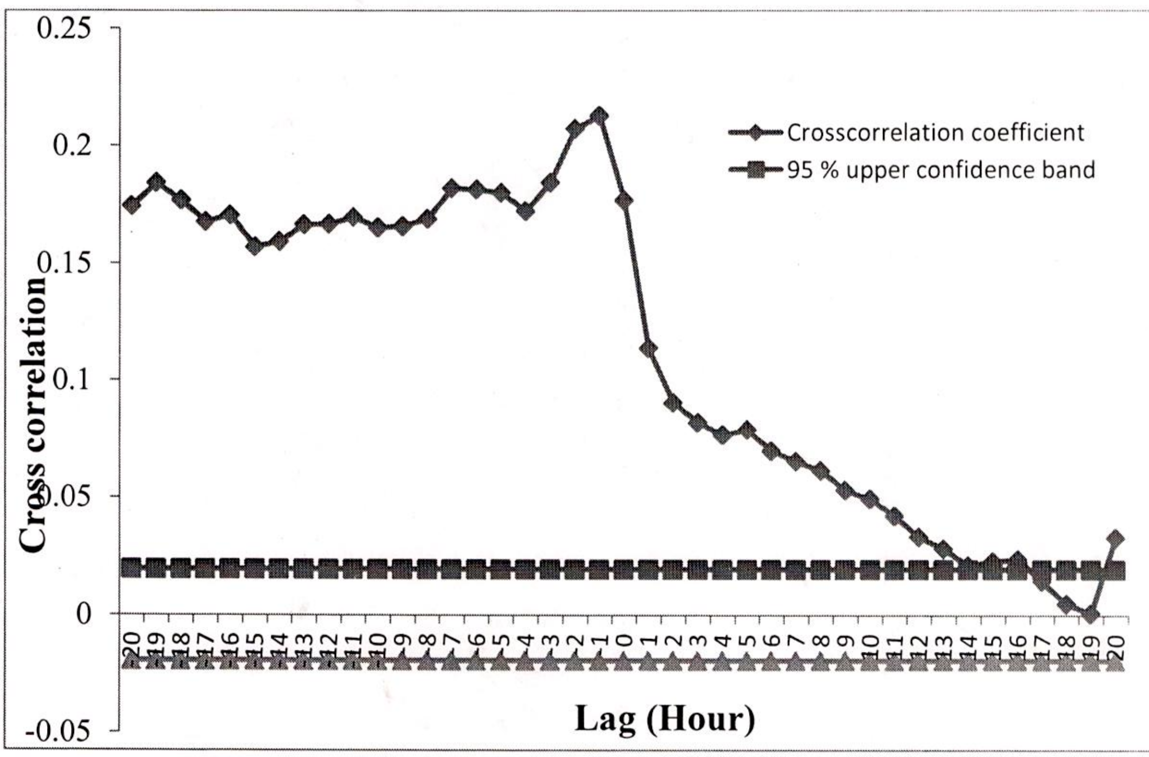


Figure 5.5 the cross correlation of discharge and rainfall at Pandariya

Based on the values of CCF and PACF of the data series as given above, the following input vectors are selected for neural network training for prediction of real time flood forecasting at Hamp. For Chirapani the input vector selected is (t) and for

Bodla and Pandariya the input vector selected is (t-1). The equation determined for the training of neural network is given below

Dis (t) = f (rchi(t), rpan(t-1), rbod(t-1), dis(t-1)) (5.4)

In this rchi, rpan, rbod are rainfall at Chirapani, Pandariya and Bodla and dis is discharge at Hamp.

5.2 TRAINING OF ANN MODEL

To train the ANN model back propagation algorithm is used. The data used is of Hamp River hourly data. For the calibration (training) and validation (testing) purpose the total data is divided into two sets one for training and one for testing. For real time flood forecasting the hourly data is used. The duration of hourly data used is from 1981 to 2009 (29 years). The data used for calibration of the model is from 1981 to 2003 (8400 hours) which includes the extreme values of discharge. The data used for validation is from 2004 to 2009 (2190 hours). The software used for the purpose of model training is MATLAB 7.8.0 (R2009a). For real time flood forecasting the performance of ANN model for calibration and validation is determined using statistical procedure given by Sudheer et al. (2002). To determine the significant lag values of the input variable this procedure is used. For determining the lag values ACF (auto correlation factor), PACF (partial auto correlation factor) and CCF (cross-correlation factor) is used by statistical procedure. The neurons number in hidden layer is determined by trial and error basis. The training simulation is used to find out the architecture of neural network which will give the best result for flood prediction.

In real time flood forecasting prediction of forward in time is made. For this two approaches are given. The first one is the same input data is used for prediction in single step ahead (t+1) or in n step ahead (t+n). In the second one the input structure will be same for predicting one step ahead but it will repeat consecutively.

For real time flood forecasting the ANN is calibrated and validated for seven lead hours. The network architecture for seven lead hours is 4-1-1. The architecture have four neurons in the input layer, one output neuron in the output layer and the hidden neurons change as the calibration process proceed. The results obtained from this process are given in tables below.

Table 5.1: result of ANN model for lead time one

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC11	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.9208	1.7134	0.8466	0.9264	1.2329	0.8562
ANNHC12	"	4-2-1	0.927	1.6448	0.8586	0.9411	1.1034	0.8848
ANNHC13	"	4-3-1	0.9266	1.6483	0.8581	0.9435	1.0806	0.8895
ANNHC14	"	4-4-1	0.931	1.6026	0.8658	0.9469	1.0511	0.8954
ANNHC15	"	4-5-1	0.9294	1.619	0.863	0.9466	1.0509	0.8955
ANNHC16	"	4-6-1	0.9324	1.5881	0.8682	0.9467	1.0554	0.8946
ANNHC17	"	4-7-1	0.9306	1.6068	0.8651	0.9468	1.0511	0.8954
ANNHC18	"	4-8-1	0.9324	1.5882	0.8682	0.9471	1.0512	0.8954
ANNCH19	"	4-9-1	0.9328	1.5844	0.8688	0.9477	1.0476	0.8961
ANNHC110	"	4-10-1	0.9329	1.5816	0.8693	0.9466	1.0574	0.8942
ANNHC111	"	4-11-1	0.9336	1.5752	0.8704	0.947	1.0537	0.8949
ANNHC115	"	4-15-1	0.934	1.5706	0.8711	0.9467	1.058	0.8941
ANNHC120	"	4-20-1	0.9352	1.5568	0.8734	0.9451	1.0761	0.8904

Table 5.2: result of ANN model for lead time two

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC21	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.8922	1.9873	0.7936	0.8966	1.4513	0.8008
ANNHC22	"	4-2-1	0.8959	1.9515	0.801	0.903	1.4045	0.8134
ANNHC23	"	4-3-1	0.8966	1.9504	0.8012	0.9021	1.4117	0.8115
ANNHC24	"	4-4-1	0.895	1.9578	0.7997	0.9075	1.3706	0.8223
ANNHC25	"	4-5-1	0.8961	1.9493	0.8015	0.9075	1.37	0.8225
ANNHC26	"	4-6-1	0.8969	1.9419	0.803	0.9085	1.3634	0.8242
ANNHC27	"	4-7-1	0.9025	1.8957	0.8122	0.9096	1.3638	0.8241
ANNHC28	"	4-8-1	0.9022	1.8982	0.8117	0.9085	1.3706	0.8223
ANNCH29	"	4-9-1	0.9032	1.8893	0.8135	0.9096	1.3636	0.8241
ANNHC210	"	4-10-1	0.9024	1.8953	0.8123	0.9098	1.3614	0.8247
ANNHC211	"	4-11-1	0.9034	1.8878	0.8138	0.9092	1.3681	0.8229
ANNHC215	"	4-15-1	0.9038	1.8843	0.8145	0.908	1.3793	0.82
ANNHC220	"	4-20-1	0.9057	1.8669	0.8179	0.9083	1.3797	0.8199

Table 5.3: result of ANN model for lead time three

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC31	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.8706	2.1678	0.7545	0.8795	1.5578	0.7705
ANNHC32	"	4-2-1	0.8758	2.1286	0.7633	0.8844	1.5265	0.7797
ANNHC33	"	4-3-1	0.8782	2.107	0.768	0.8869	1.5113	0.784
ANNHC34	"	4-4-1	0.8793	2.0987	0.7699	0.8856	1.5205	0.7814
ANNHC35	"	4-5-1	0.8792	2.0991	0.7698	0.8859	1.5185	0.782
ANNHC36	"	4-6-1	0.8799	2.0943	0.7708	0.8862	1.5162	0.7826
ANNHC37	"	4-7-1	0.8804	2.09	0.7718	0.886	1.5205	0.7814
ANNHC38	"	4-8-1	0.8763	2.1219	0.7647	0.8867	1.5085	0.7849
ANNCH39	"	4-9-1	0.8814	2.0828	0.7733	0.8859	1.5195	0.7817
ANNHC310	"	4-10-1	0.8818	2.0776	0.7745	0.8869	1.516	0.7827
ANNHC311	"	4-11-1	0.8818	2.0788	0.7742	0.8863	1.5201	0.7815
ANNHC315	"	4-15-1	0.884	2.0607	0.7781	0.8871	1.5172	0.7823
ANNHC320	"	4-20-1	0.8844	2.0577	0.7788	0.8858	1.5267	0.7796

Table 5.4: result of ANN model for lead time four

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC41	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.8554	2.2849	0.7272	0.8752	1.5808	0.7638
ANNHC42	"	4-2-1	0.8609	2.2455	0.7365	0.8755	1.5793	0.7643
ANNHC43	"	4-3-1	0.8629	2.2288	0.7405	0.875	1.5826	0.7633
ANNHC44	"	4-4-1	0.8564	2.2726	0.7301	0.8756	1.5749	0.7655
ANNHC45	"	4-5-1	0.8586	2.2583	0.7335	0.8749	1.5799	0.7641
ANNHC46	"	4-6-1	0.859	2.2554	0.7342	0.8765	1.5701	0.767
ANNHC47	"	4-7-1	0.8602	2.2468	0.7362	0.8771	1.5671	0.7679
ANNHC48	"	4-8-1	0.8623	2.2301	0.7401	0.8754	1.5825	0.7633
ANNCH49	"	4-9-1	0.8624	2.2313	0.7399	0.8776	1.5659	0.7682
ANNHC410	"	4-10-1	0.8655	2.2093	0.745	0.8746	1.5869	0.762
ANNHC411	"	4-11-1	0.8652	2.2108	0.7446	0.8741	1.5909	0.7608
ANNHC415	"	4-15-1	0.8671	2.1967	0.7479	0.8747	1.5894	0.7612
ANNHC420	"	4-20-1	0.87	2.1753	0.7528	0.8716	1.6131	0.754

Table 5.5: result of ANN model for lead time five

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC51	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.8437	2.3699	0.7065	0.8703	1.6081	0.7557
ANNHC52	,,	4-2-1	0.8451	2.3587	0.7093	0.8689	1.6153	0.7535
ANNHC53	,,	4-3-1	0.848	2.3395	0.714	0.8665	1.6301	0.7489
ANNHC54	,,	4-4-1	0.8447	2.3583	0.7094	0.8678	1.6208	0.7518
ANNHC55	,,	4-5-1	0.8442	2.3615	0.7086	0.8705	1.6052	0.7565
ANNHC56	,,	4-6-1	0.8465	2.3465	0.7123	0.8708	1.6037	0.757
ANNHC57	,,	4-7-1	0.8529	2.3039	0.7227	0.866	1.6345	0.7476
ANNHC58	,,	4-8-1	0.8533	2.3005	0.7235	0.8653	1.6408	0.7456
ANNCH59	,,	4-9-1	0.8519	2.3085	0.7216	0.8642	1.6455	0.7442
ANNHC510	,,	4-10-1	0.8534	2.2994	0.7237	0.8665	1.6328	0.7481
ANNHC511	,,	4-11-1	0.853	2.3019	0.7231	0.8657	1.6367	0.7469
ANNHC515	,,	4-15-1	0.8555	2.2851	0.7272	0.8666	1.6308	0.7487
ANNHC520	,,	4-20-1	0.859	2.2599	0.7331	0.8596	1.6791	0.7336

Table 5.6: result of ANN model for lead time six

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC61	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.8348	2.4323	0.6909	0.8569	1.6846	0.732
ANNHC62	,,	4-2-1	0.8332	2.4379	0.6895	0.854	1.6988	0.7274
ANNHC63	,,	4-3-1	0.8413	2.3881	0.702	0.8507	1.7187	0.721
ANNHC64	,,	4-4-1	0.8365	2.4161	0.695	0.8593	1.6694	0.7368
ANNHC65	,,	4-5-1	0.8438	2.3689	0.7068	0.8521	1.712	0.7332
ANNHC66	,,	4-6-1	0.836	2.4174	0.6946	0.8584	1.6745	0.7352
ANNHC67	,,	4-7-1	0.8396	2.3969	0.6998	0.8589	1.6718	0.736
ANNHC68	,,	4-8-1	0.8386	2.4018	0.6986	0.8578	1.6775	0.7342
ANNCH69	,,	4-9-1	0.8425	2.3769	0.7048	0.8578	1.6778	0.7341
ANNHC610	,,	4-10-1	0.8453	2.3589	0.7093	0.8532	1.7052	0.7254
ANNHC611	,,	4-11-1	0.8455	2.3568	0.7098	0.852	1.7123	0.7231
ANNHC615	,,	4-15-1	0.848	2.3403	0.7138	0.8534	1.7045	0.7256
ANNHC620	,,	4-20-1	0.8513	2.3164	0.7196	0.8487	1.7311	0.717

Table 5.7: result of ANN model for lead time seven

Model No	Input combinations	ANN Structure	Calibration			Validation		
			CORR	RMSE	EFF%	CORR	RMSE	EFF%
ANNHC71	rchi(t), rpan(t-1), rbod(t-1),dis(t), dis(t-1)	4-1-1	0.8251	2.4976	0.6741	0.8399	1.776	0.7022
ANNHC72	"	4-2-1	0.8266	2.4844	0.6775	0.8389	1.7789	0.7012
ANNHC73	"	4-3-1	0.8316	2.4547	0.6852	0.8377	1.7868	0.6986
ANNHC74	"	4-4-1	0.828	2.4725	0.6806	0.8436	1.7547	0.7093
ANNHC75	"	4-5-1	0.828	2.4713	0.6809	0.8435	1.7553	0.7091
ANNHC76	"	4-6-1	0.8304	2.4576	0.6844	0.8447	1.7479	0.7116
ANNHC77	"	4-7-1	0.831	2.4547	0.6851	0.8455	1.7438	0.7129
ANNHC78	"	4-8-1	0.8325	2.4462	0.6873	0.8466	1.7383	0.7147
ANNCH79	"	4-9-1	0.8377	2.4128	0.6958	0.8408	1.7702	0.7042
ANNHC710	"	4-10-1	0.8378	2.4117	0.6961	0.8402	1.7738	0.703
ANNHC711	"	4-11-1	0.8379	2.4114	0.6962	0.8398	1.7769	0.7019
ANNHC715	"	4-15-1	0.8401	2.3971	0.6998	0.8384	1.7828	0.6999
ANNHC720	"	4-20-1	0.8422	2.3826	0.7034	0.8383	1.7855	0.699

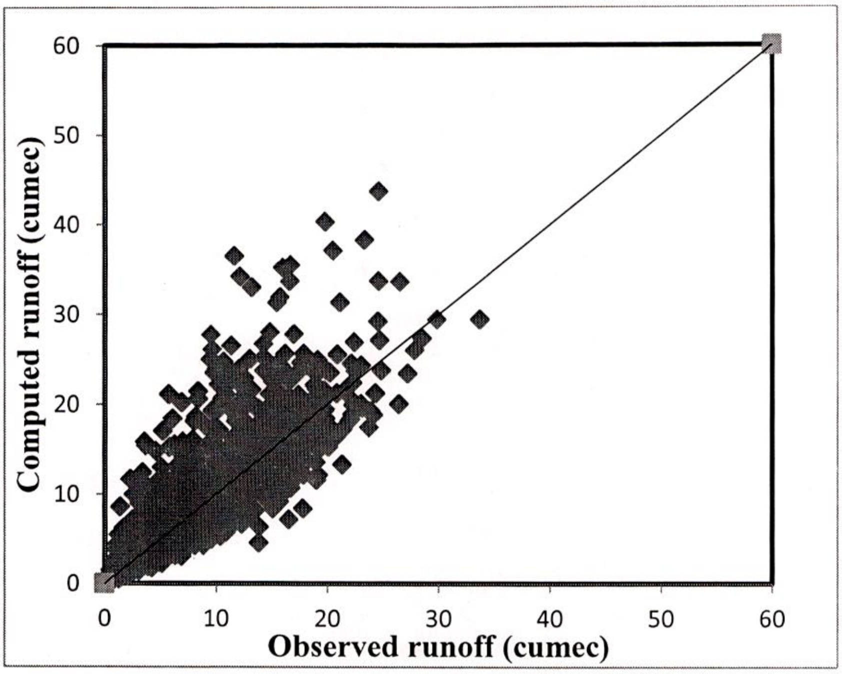
5.3 RESULTS OF ANN AND MLR

The performance of best ANN model for real time flood forecasting for all lead times during calibration and validation is presented in Figures below. respectively along with the corresponding observed runoff. The visual inspection of the plots clearly demonstrates the potential of the developed ANN model in prediction of the runoff at

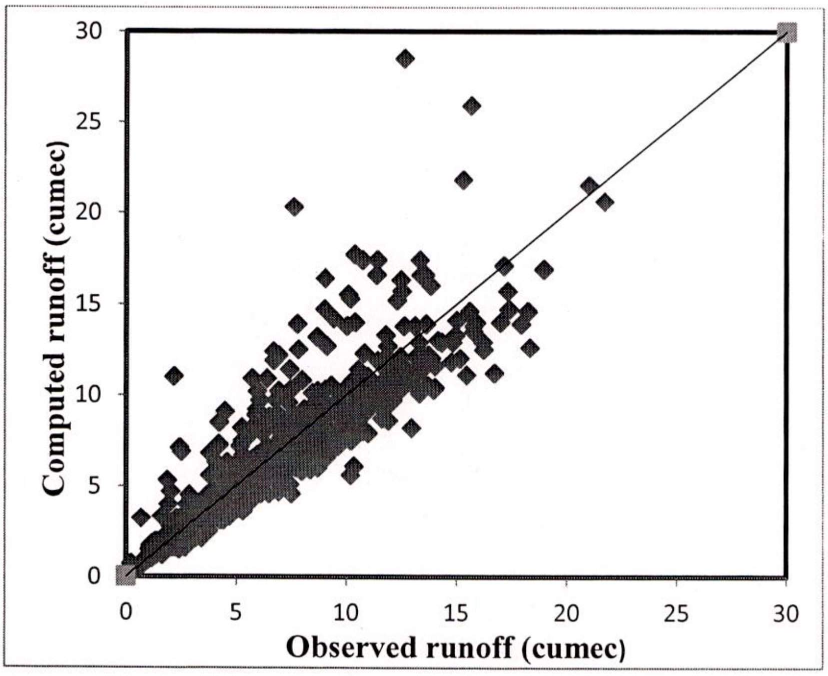
Hamp. The results of the calibration and validation of the best ANN models for seven lead times are presented below.

For lead 1

Calibration for lead 1

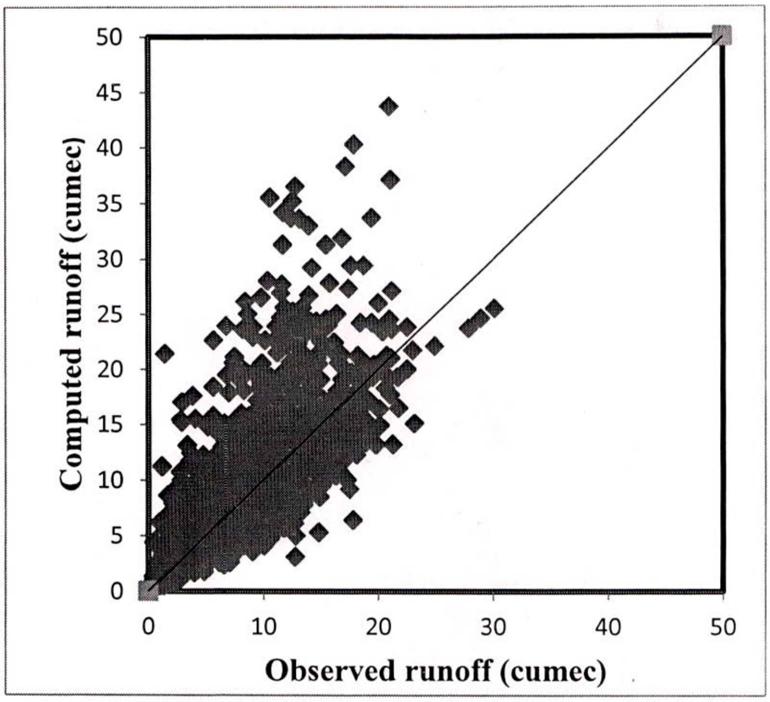


Validation for lead 1

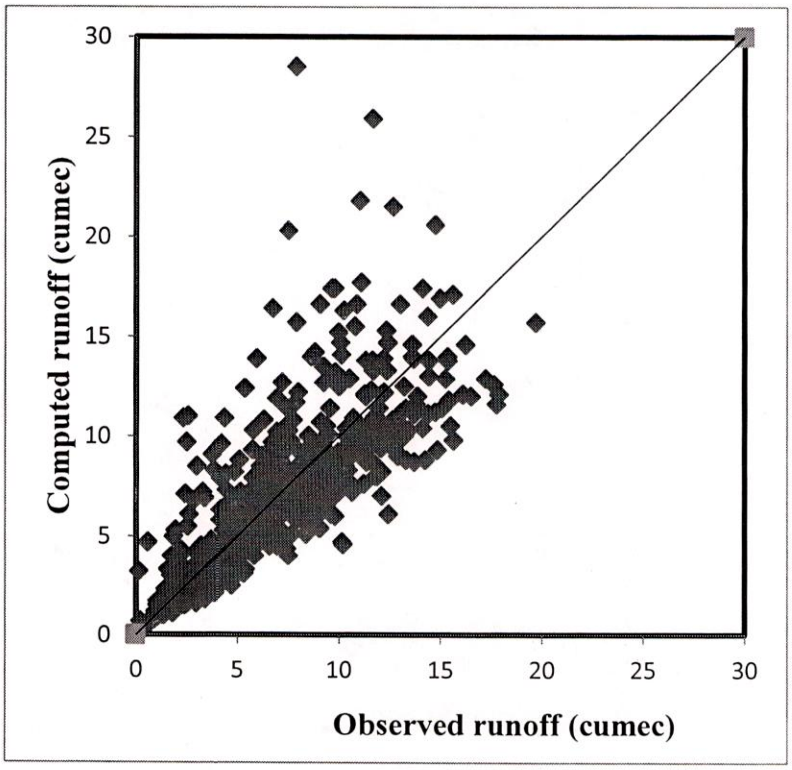


For lead two

Calibration for lead 2

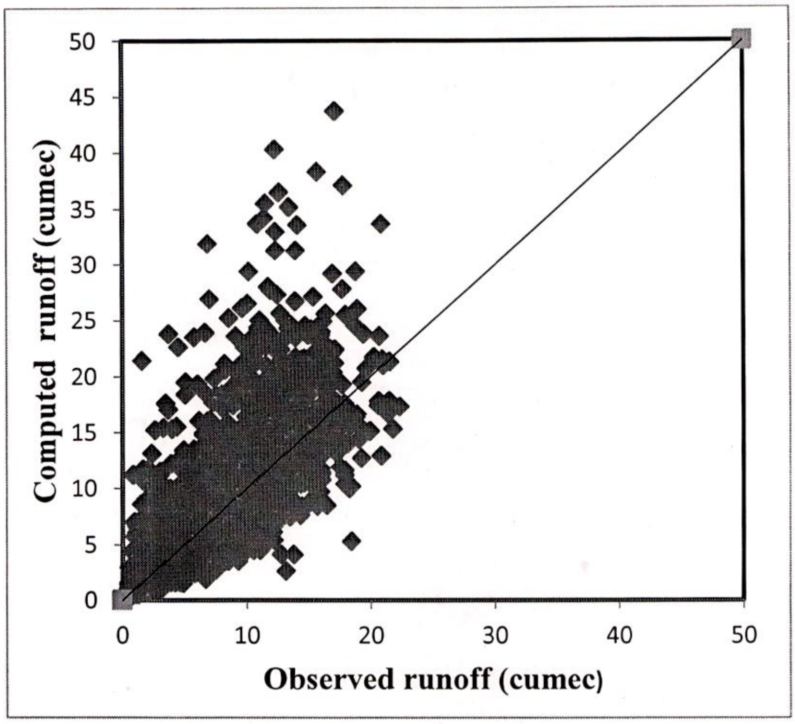


Validation for lead 2

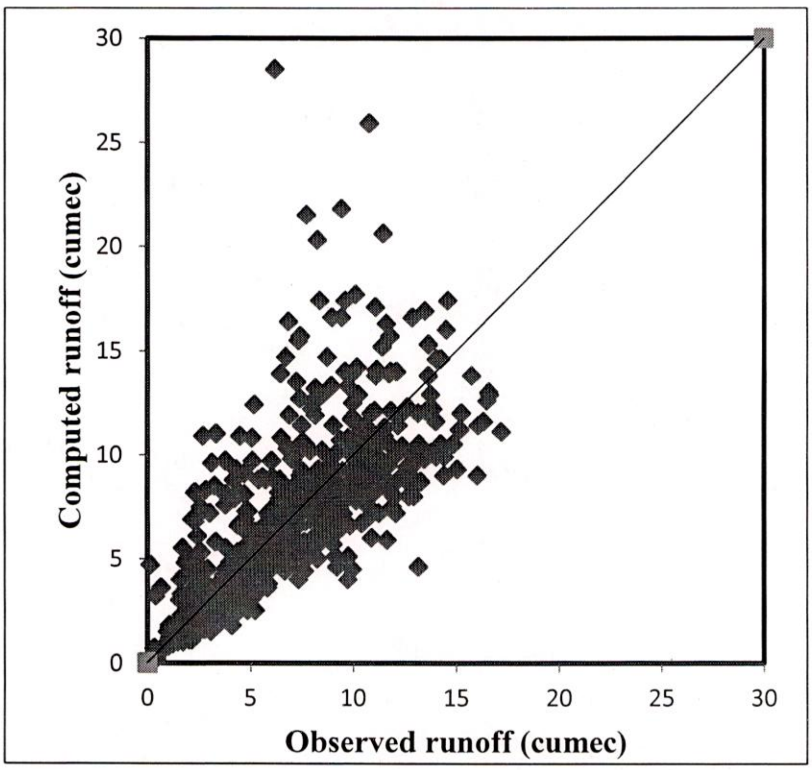


For lead 3

Calibration for lead 3

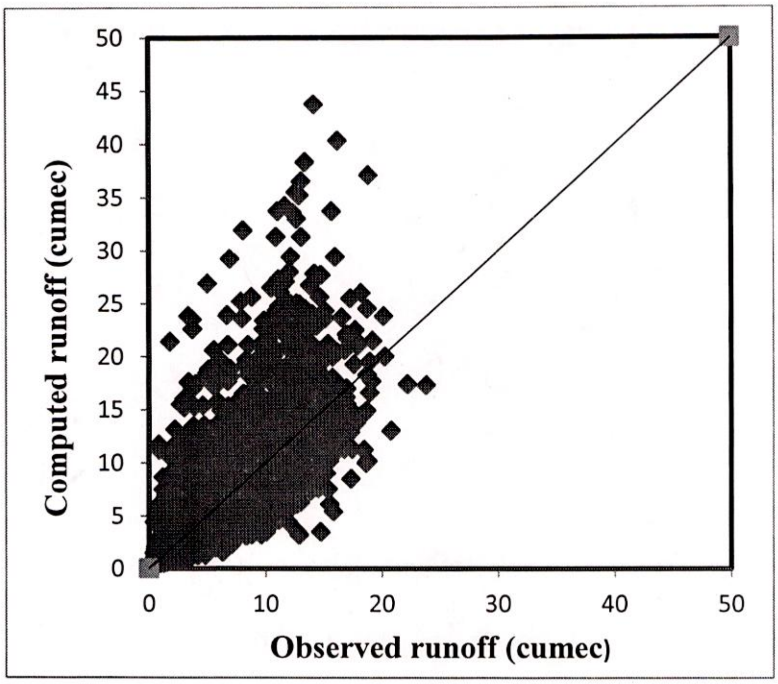


Validation for lead 3

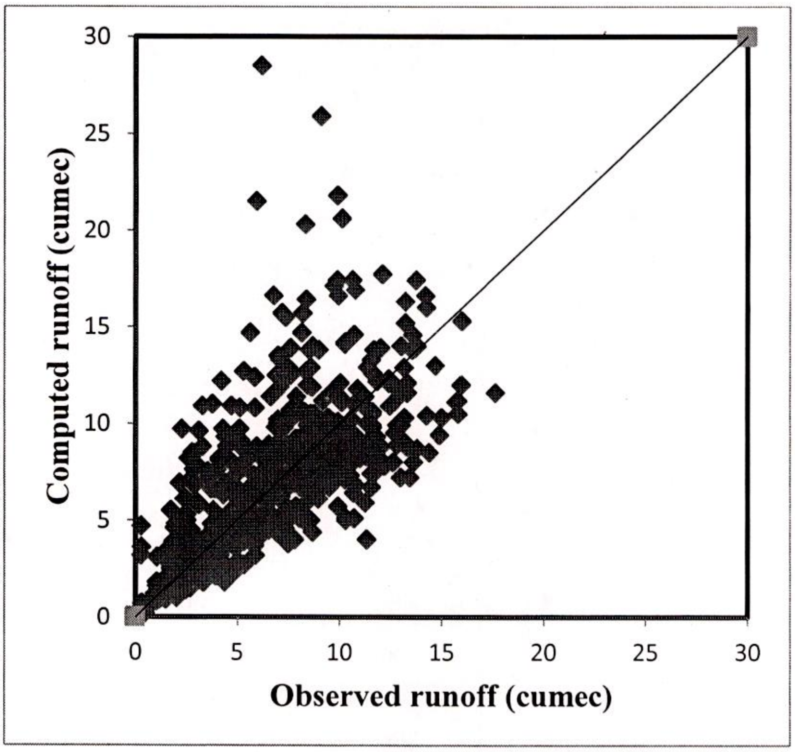


For lead time 4

Calibration for lead 4

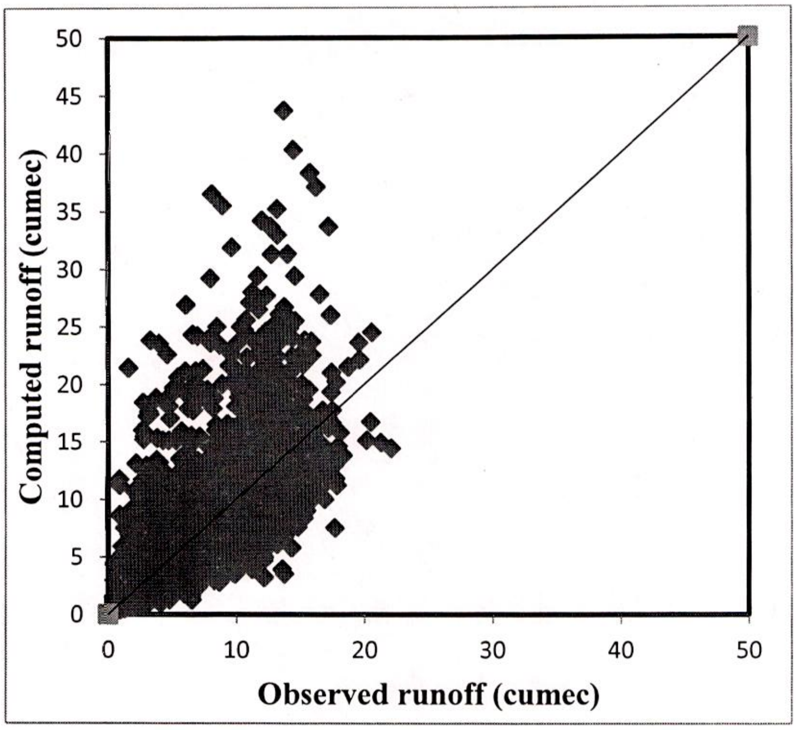


Validation for lead 4

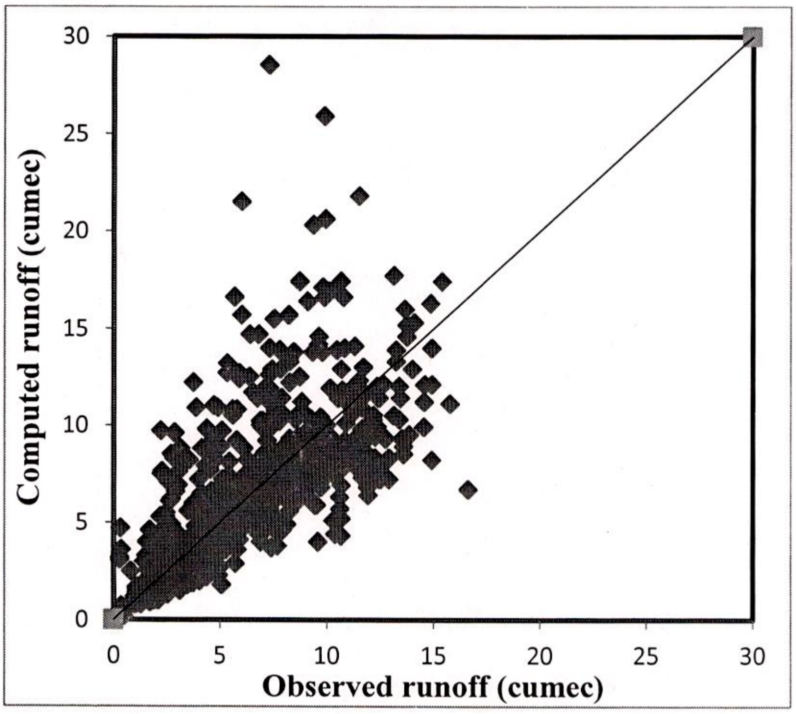


For lead 5

Calibration for lead 5

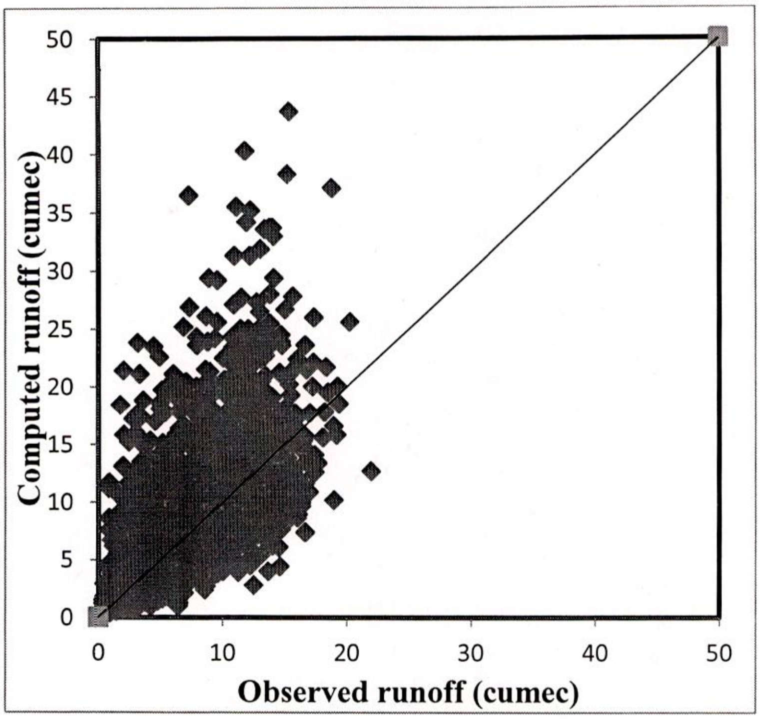


Validation for lead 5

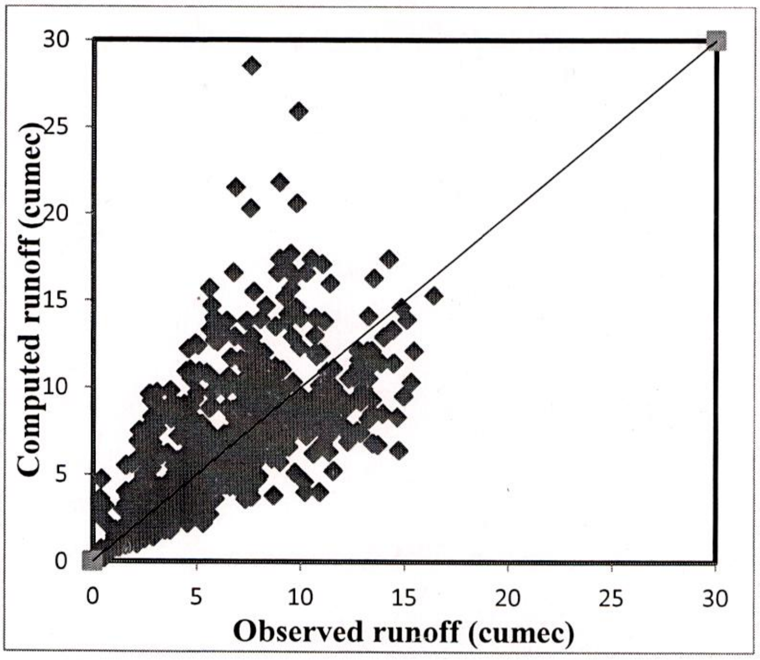


For lead 6

Calibration for lead 6

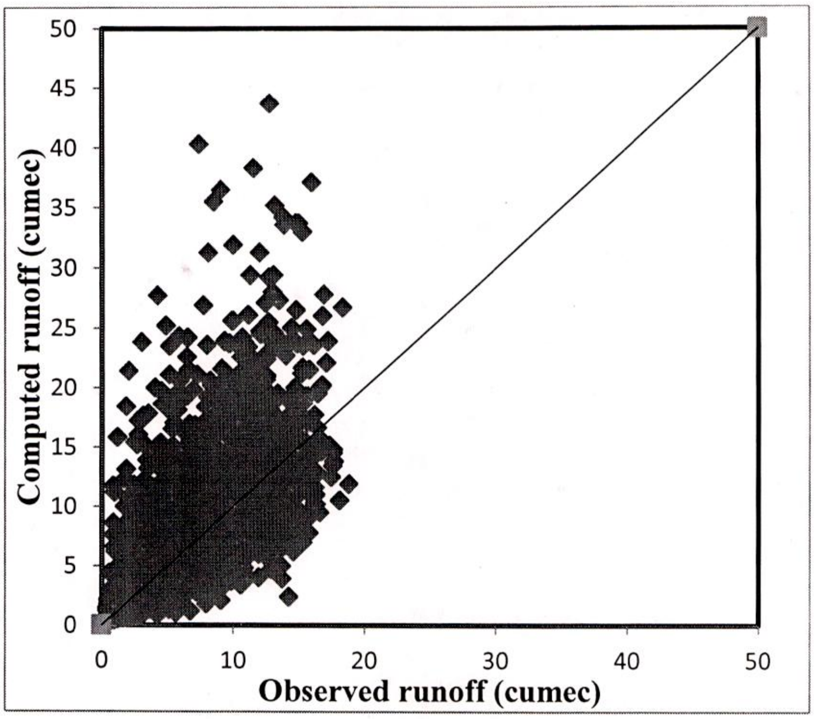


Validation for lead 6

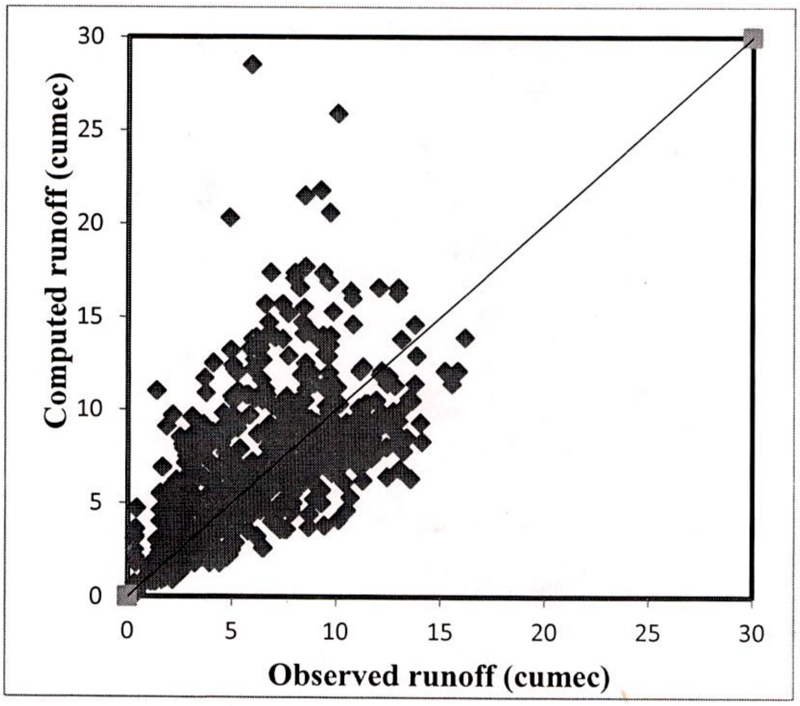


For lead 7

Calibration for lead 7



Validation for lead 7



5.4 RESULTS OF MLR (MULTIPLE LINEAR REGRESSIONS) MODEL

Multiple Linear Regression (MLR) model is also developed for flood prediction using the data and combination of input vector considered in the development of ANN model. The equation of MLR model for all the seven lead time are presented below

For lead 1

$$RQ_{hamp,\, t-1} = 0.0947R_{chirapai,t} + 0.0072R_{pandariya,t-1} - 0.0222R_{bodla,t-1} + 0.8808DIS_{hamp,t-1} + 0.3690$$

For lead 2

$$RQ_{hamp,\, t-1} = 0.0839R_{chirapai,t} + 0.0017R_{pandariya,t-1} - 0.0229R_{bodla,t-1} + 0.8528DIS_{hamp,t-1} + 0.5390$$

For lead 3

$$RQ_{hamp,\, t-1} = 0.0703R_{chirapai,t} + 0.0005R_{pandariya,t-1} - 0.0216R_{bodla,t-1} + 0.8340DIS_{hamp,t-1} + 0.6570$$

For lead 4

$$RQ_{hamp,\, t-1} = 0.0596R_{chirapai,t} + 0.0100R_{pandariya,t-1} - 0.0274R_{bodla,t-1} + 0.8196DIS_{hamp,t-1} + 0.7353$$

Lead 5

$$RQ_{hamp,\, t-1} = 0.0532R_{chirapai,t} + 0.0109R_{pandariya,t-1} - 0.0228R_{bodla,t-1} + 0.8070DIS_{hamp,t-1} + 0.7944$$

Lead 6

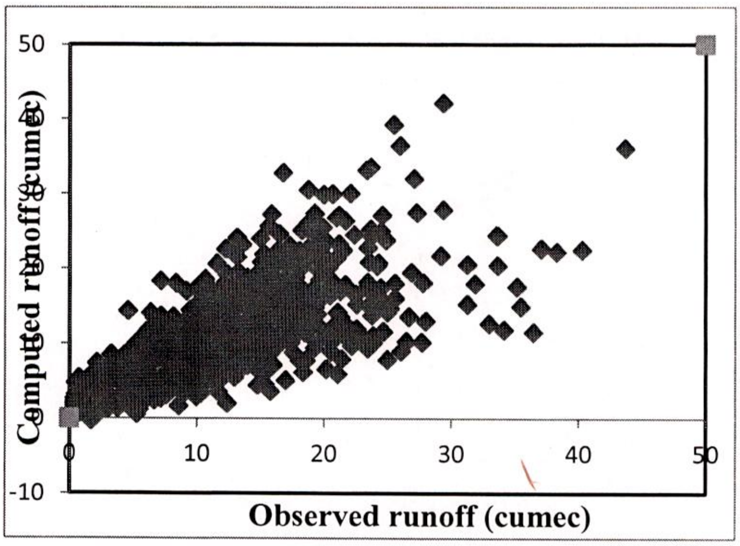
$$RQ_{hamp,\, t-1} = 0.0532R_{chirapai,t} + 0.0110R_{pandariya,t-1} - 0.0213R_{bodla,t-1} + 0.7972DIS_{hamp,t-1} + 0.8365$$

Lead 7

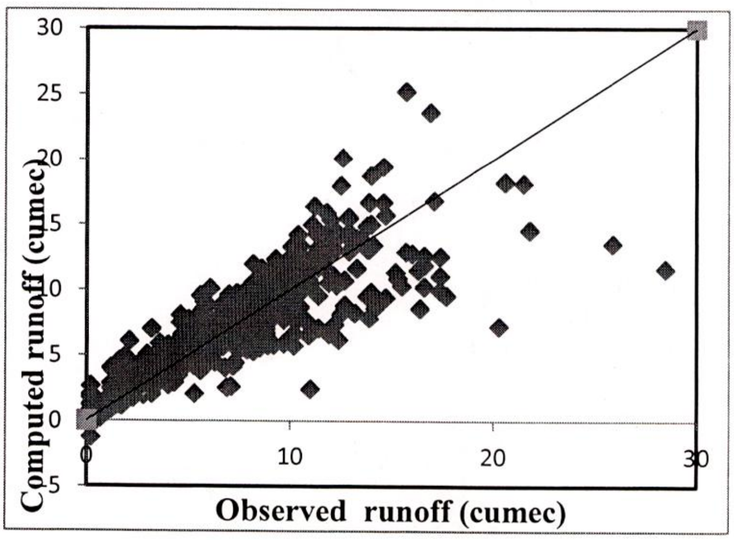
$$RQ_{hamp,\,t-1} = 0.0600R_{chirapai,t} + 0.0057R_{pandariya,t-1} - 0.0282R_{bodla,t-1} + 0.7901DIS_{hamp,t-1} + 0.8842$$

Where RQ and R are discharge and rainfall values respectively. The performance of ANN and MLR models is evaluated based on the performance indices. The results of ANN and MLR models are compared with each other. Using these equations the correlation, root mean square error and efficiency for the MLR model is determined. To determine correlation, root mean square error and efficiency for the MLR the coefficient of the equations are used with the given formulas. These calculations are done in Microsoft excel sheet. The performance of MLR model for the flood prediction during calibration and validation for seven lead times is presented below

Calibration for lead 1

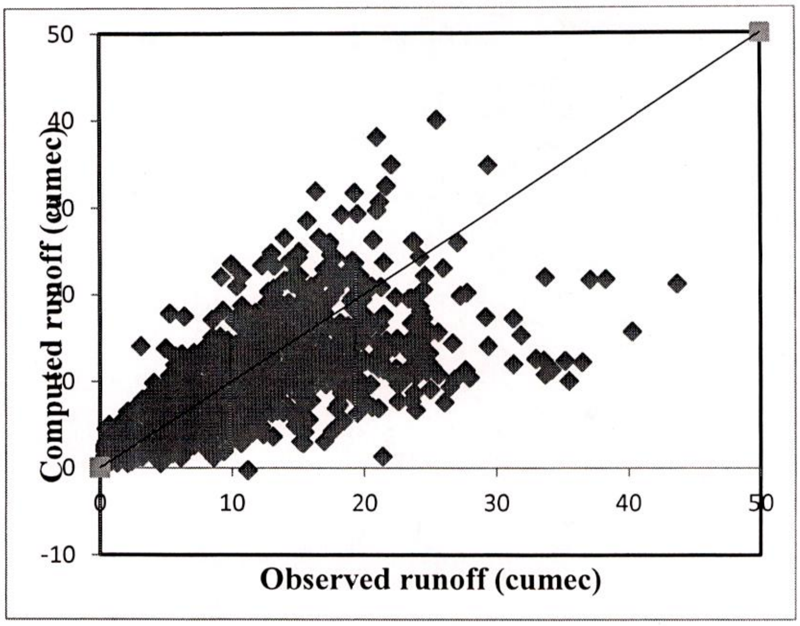


Validation for lead 2

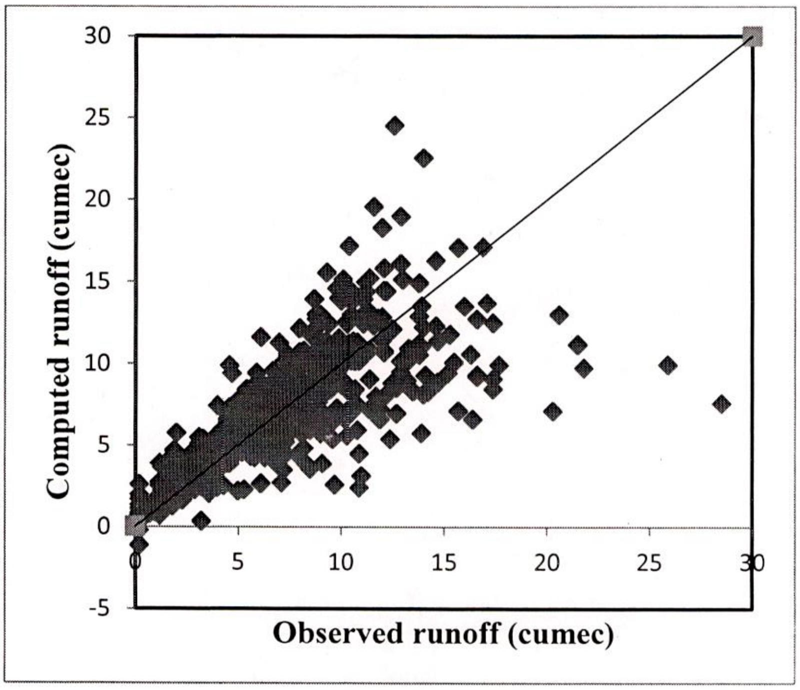


For lead 2

Calibration for lead 2

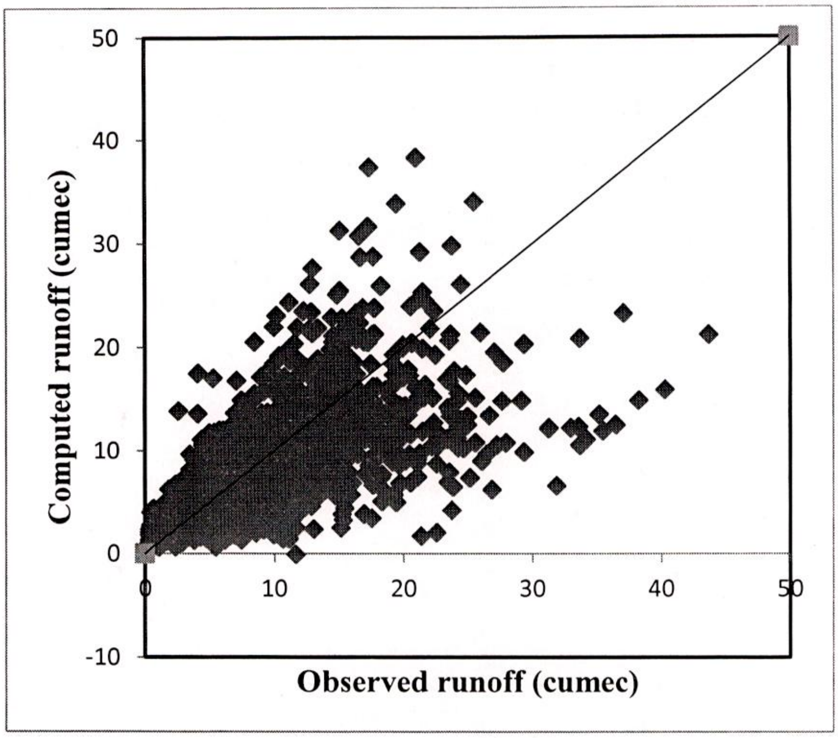


Validation for lead 2

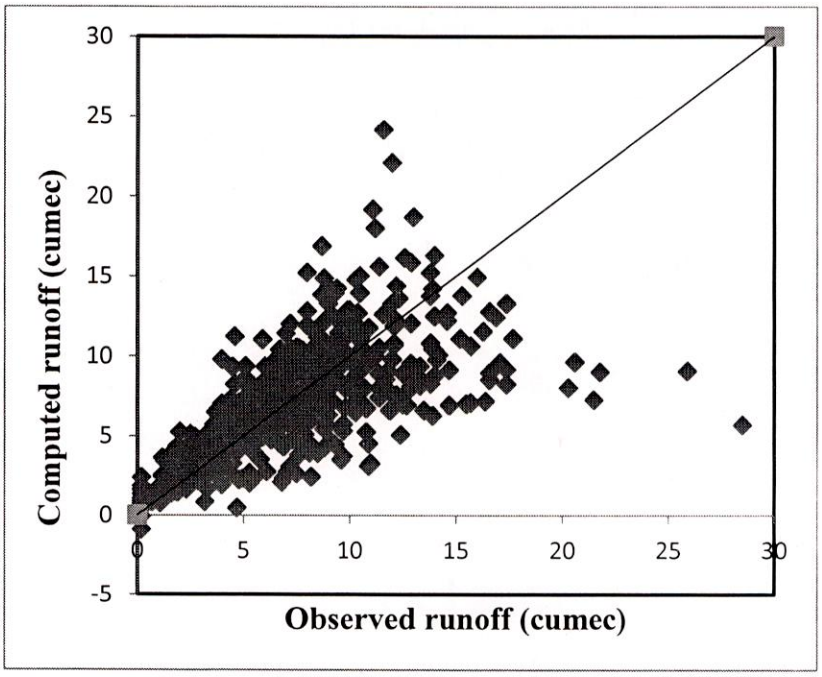


For lead 3

Calibration for lead 3

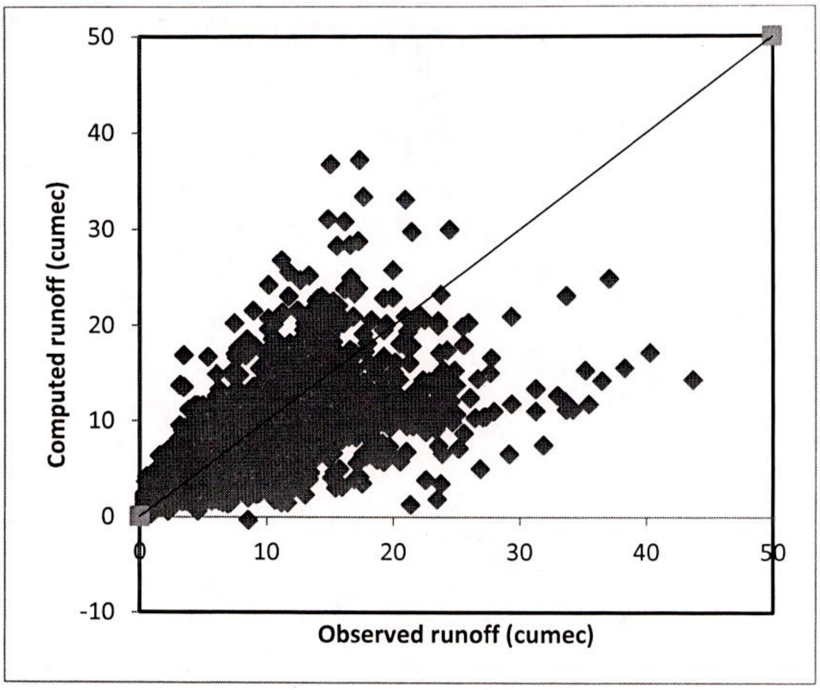


Validation for lead 3

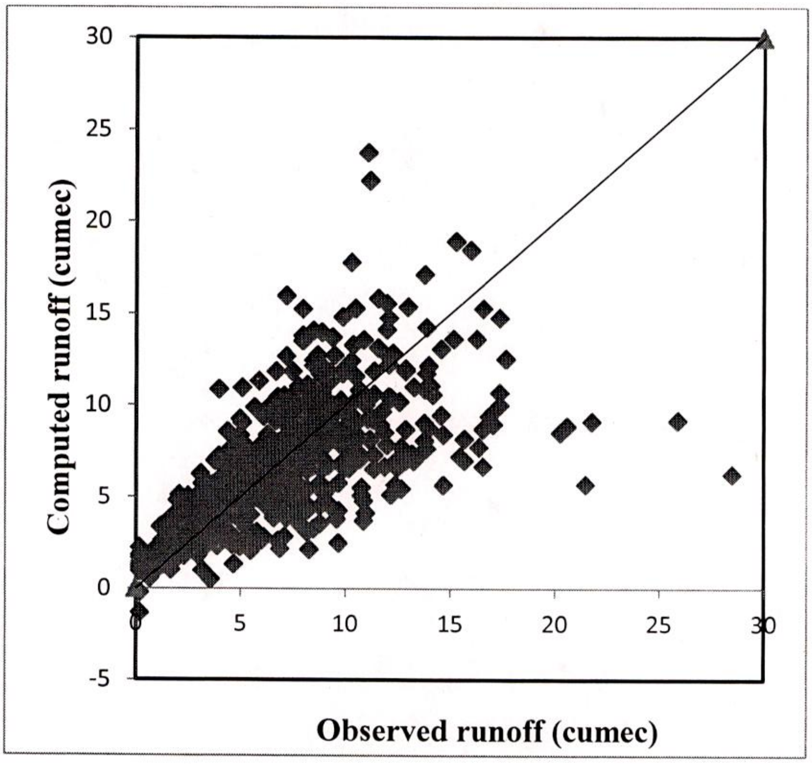


For lead 4

Calibration for lead 4

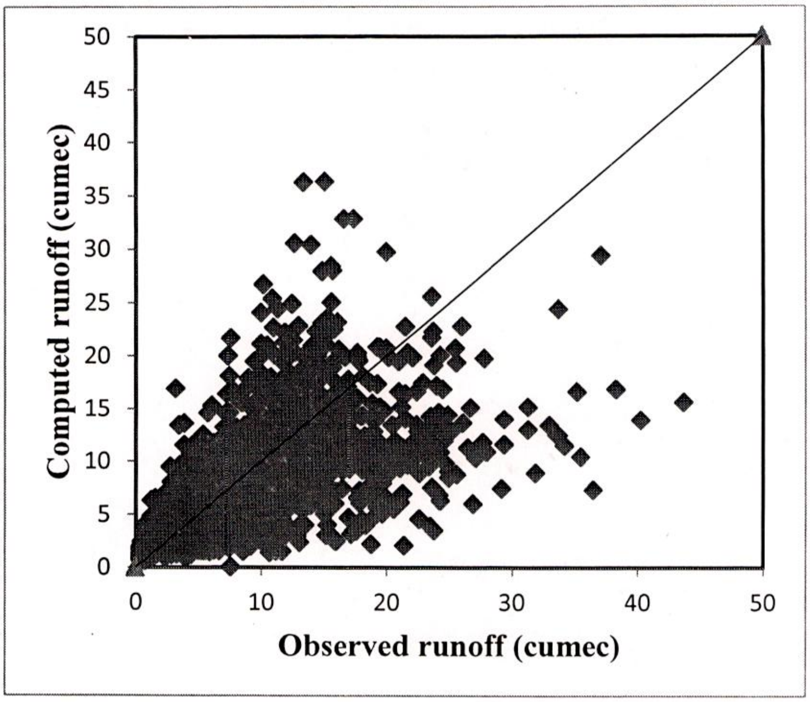


Validation for lead 4

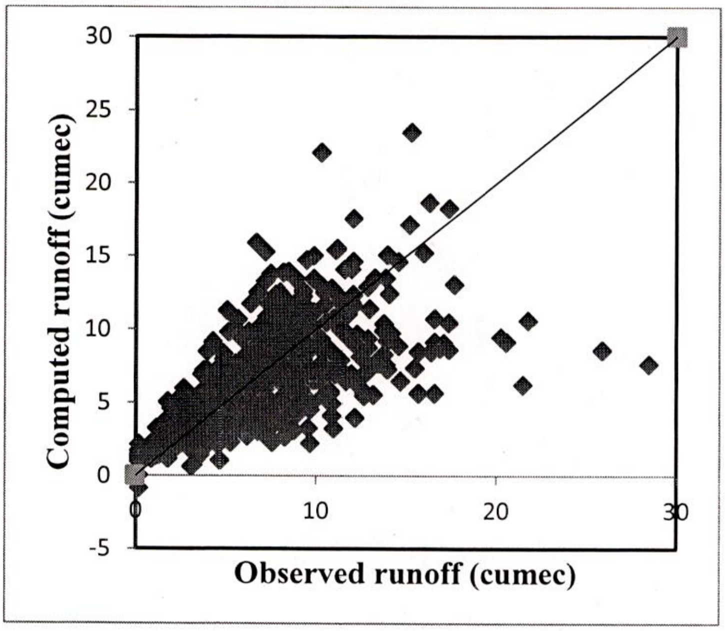


For lead 5

Calibration for lead 5

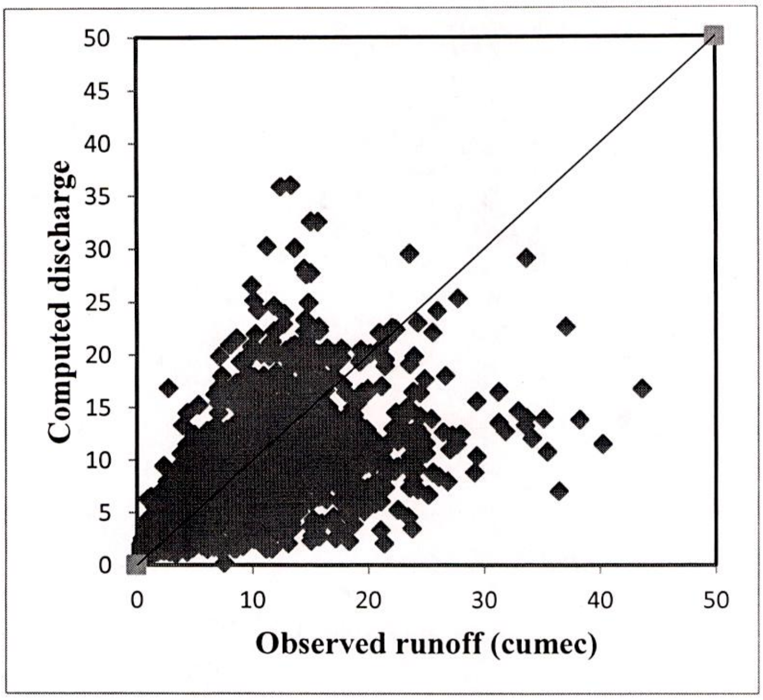


Validation for lead 5

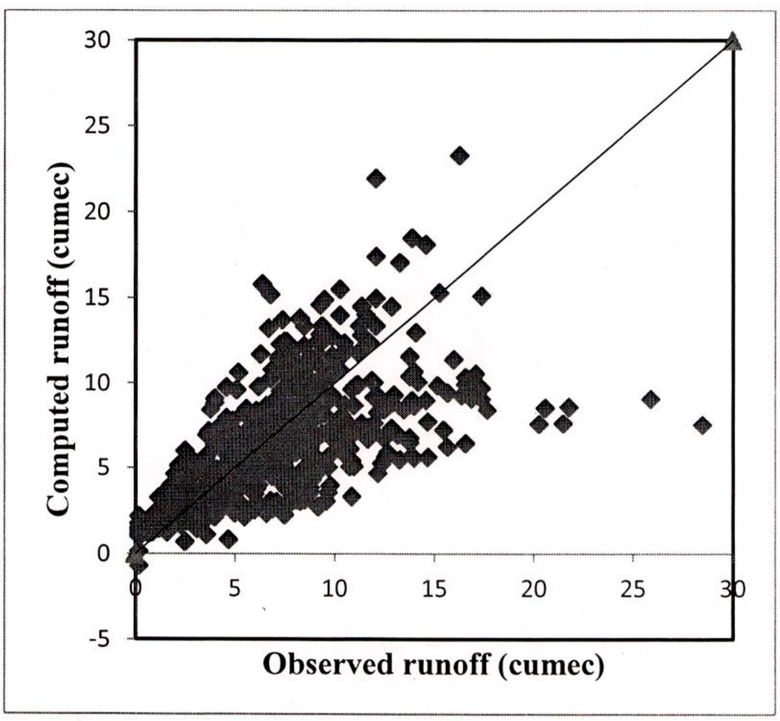


For lead 6

Calibration for lead 6

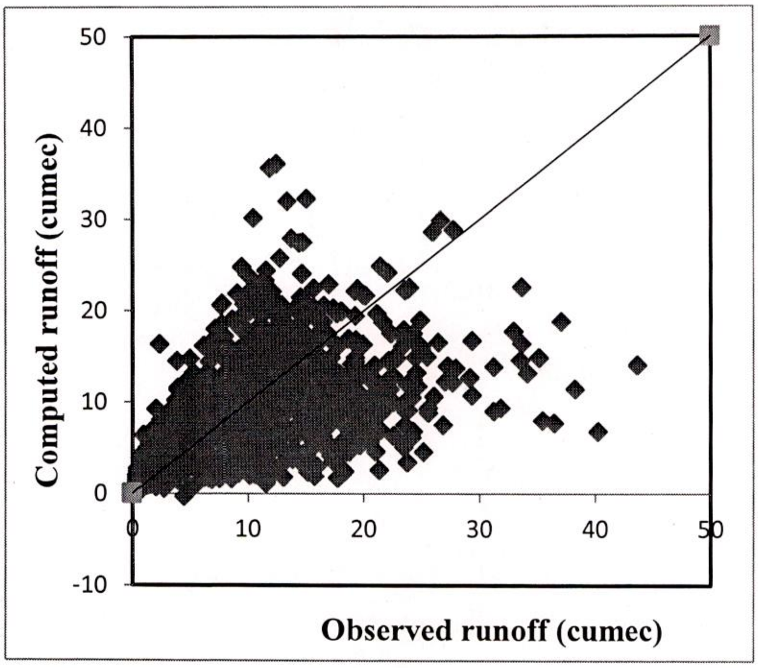


Validation for lead 6



For lead 7

Calibration for lead 7



Validation for lead 7

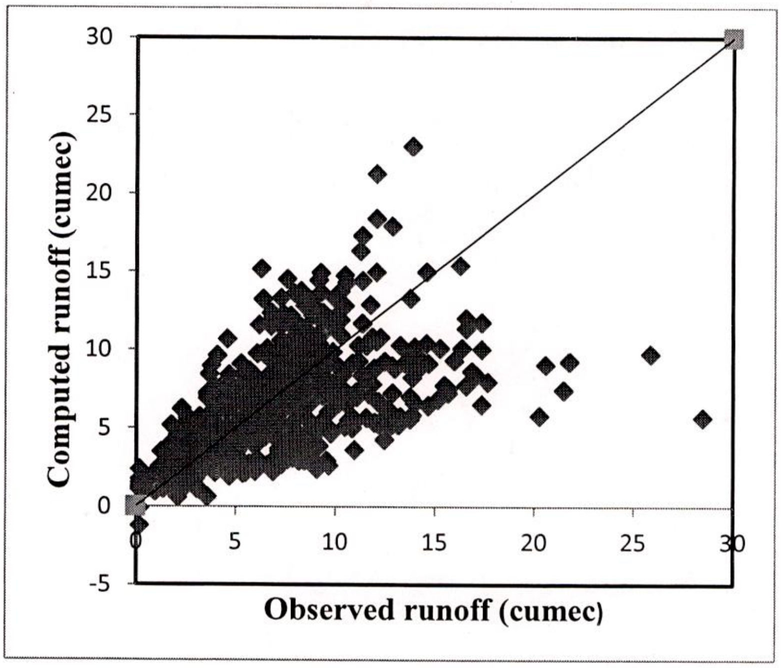


Table 5.8 Comparison of results between best ANN and MLR models for all lead times

	ANN model		MLR model	
	Calibration	Validation	Calibration	Validation
For lead 1				
Coefficient of Correlation	0.9324	0.9467	0.9191	0.9342
RMSE	1.5881	1.0554	1.7244	1.1617
Model efficiency	0.8682	0.8946	0.8446	0.8723
Percentage error	-29.5	-30.97	3.66	11.68
For lead 2				
Coefficient of Correlation	0.9025	0.9096	0.8817	0.8928
RMSE	1.8957	1.3638	2.0635	1.4664
Model efficiency	0.8122	0.8241	0.7775	0.7966
Percentage error	-44.89	-44.5	8.32	13.92
For lead 3				
Coefficient of Correlation	0.8814	0.8859	0.8552	0.8672
RMSE	2.0828	1.5195	2.2671	1.6222
Model efficiency	0.7733	0.7817	0.7315	0.7512
Percentage error	-95.4	-65.5	12.4	15.32
For lead 4				
Coefficient of Correlation	0.8629	0.8750	0.8366	0.8581
RMSE	2.2288	1.5826	2.3963	1.6742
Model efficiency	0.7405	0.7633	0.7099	0.7351
Percentage error	-83.22	-61.29	14.94	16.77
For lead 5				
Coefficient of Correlation	0.8529	0.8660	0.8234	0.8512
RMSE	2.3039	1.6345	2.4826	1.7120
Model efficiency	0.7227	0.7476	0.6779	0.7231

Percentage error	-97.64	-71.17	16.92	17.6
For lead 6				
Coefficient of Correlation	0.8438	0.8521	0.8142	0.8390
RMSE	2.3689	1.7120	2.5393	1.7745
Model efficiency	0.7068	0.7232	0.6631	0.7025
Percentage error	-98.54	-73.56	17.6	18.36
For lead 7				
Coefficient of Correlation	0.8316	0.8377	0.8053	0.8215
RMSE	2.4547	1.7868	2.5935	1.860
Model efficiency	0.6852	0.6986	0.6485	0.6733
Percentage error	-131.85	-75.70	17.5	19.23

The high coefficient of correlation during the calibration and validation for real time flood prediction indicates that the variance is high and the developed ANN model is good for estimating the runoff for real time flood with less error. The coefficient of correlation, RMSE, Model efficiency of MLR model during calibration are lower than the values of ANN model during calibration. The performance of MLR model during validation is only slightly lower than the ANN model performance. The RMSE of ANN, which is a measure of the residual variance during calibration, is high compared to the value of the validation. The same is with the performance of MLR model.

5. 5 CONCLUSIONS

Based on this study it is concluded that the results of ANN model for runoff and real time flood forecasting shows that the ANN model is good for modeling within limited data sets. It gives good results using only rainfall and discharge data for real time flood forecasting model. The performance of ANN model is compared with the performance of MLR model and it is found that the result of ANN was good than that of MLR model. The inputs for ANN model are selected by statistical procedure. Based on the performance evaluation of ANN and MLR for estimating the runoff for real time flood it is found that ANN does better than MLR models.

REFRENCES

1. ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. (2000a). "Artificial neural networks in hydrology-I: Preliminary concepts." *J. Hydrol. Engrg.*, 5(2), 115-123.
2. ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. (2000b). "Artificial neural networks in hydrology-II: Hydrologic applications." *J. Hydrol. Engrg.*, 5(2), 124-137.
3. Chen, X., Jinglu, W., and Hu, Q. (2008). "Simulation of climate change impacts on streamflow in the Bosten Lake basin using an artificial neural network model." *J. Hydrol. Engrg.*, 13(3), 180-183.
4. Fernando, A.K. and Jayawardena, A.W. (1998). "Runoff forecasting using RBF networks with OLS algorithm." *J. Hydrol. Engrg.*, 3(3), 203-209.
5. Hsu, K-L., Gupta, H.V., and Sorooshian, S. (1995). "Artificial neural network modeling of the rainfall-runoff process." *Water Resour. Res.*, 31(10), 2517-2530.
6. Maier, H.R., and Dandy, G.C. (2000). "Neural networks for the prediction and forecasting of water resources variables: A review of modelling Issues and applications." *Environmental Modelling & Software*, 15, 101-124.
7. Minns, A. W., and Hall, M. J. (1996). "Artificial neural networks as rainfall runoff models." *Hydrol. Sci. J.*, 41(3), 399-417.
8. Nash, J. E., and Sutcliffe, J. V. (1970). "River flow forecasting through conceptual models:1. A discussion of principles." *J. Hydrol.*, 10(3), 282-290.
9. Sudheer, K. P. Nayak, P. C., and Ramasastri, K. S. (2003). "Improving peak flow estimates in artificial neural network river flow models." *Hydrolog. Process*. 17(3), 677-686.
10. Thirumalaiah, K., and Deo, M.C. (2000). "Hydrological forecasting using neural networks." *J. Hydrol. Engrg.*, 5 (2), 180-189.
11. Liong Shie-Yui, Lim Wee-Han, and Paudyal Guna N. (200). "River stage forecasting in bangladesh neural network approach." J. Comput. Civ. Eng., 14(1), 1-8.
12. Thirumalaiah, K., and Deo, M.C. (1998). "River stage forecasting using artificial neural networks." J. Hydrol. Eng., 3(1), 26-32.