Rainfall-Runoff Modeling using Artificial Neural Networks (A Case study of Khodiyar Catchment Area)

Mr. Mahesh B. Shrivastav[1] Prof. Haresh M. Gandhi[2] Prof. Kashyap B. Gohil[3] Prof. Nirav D. Acharya[4] Mr. Jignesh A. Joshi[5]

[1]PG Scholar [2,3,4]Associate Professor [5]Assistant Engineer

[1,2,3,4]Civil Engineering Department

[1,2,3,4]Shantilal Shah Engg. College, Bhavnagar, Gujarat [5]Salinity Control Sub-division, Bhavnagar, Gujarat

Abstract— An Artificial Neural Network (ANN) methodology has been employed to develop Rainfall-Runoff Model as a function of rainfall, temperature, evaporation losses, infiltration losses and humidity for the Khodiyar catchment located in Amreli district, Gujarat, India,. The investigation of sensitivity of the modeling accuracy to the content and length of training data has been carried out. The comparison of ANN rainfall-runoff model was done favorably by obtaining results using existing techniques including statistical regression model. The ANN model provides a more systematic approach, reduces the length of calibration data, and shortens the time spent in calibration of the models, at the same time, it represents an improvement upon the prediction accuracy and flexibility of current methods.

Keywords: rainfall-runoff, modeling, ANN, algorithm, simulation, prediction

I. Introduction

In many parts of the world, the demand for water has increased because of rapid population growth, urbanization and industrialization which have resulted in altered watersheds and river systems, contributing to a greater loss of life and property damages due to flooding. It is becoming increasingly critical to plan, design, and manage water resources systems carefully and intelligently. Hydrologists have attempted for many years to understand the transformation of rainfall to runoff, in order to predict runoff for purposes such as water supply, flood control, irrigation, drainage, water quality, power generation, recreation and fish and wildlife propagation. Due to the tremendous spatial and temporal variability of watershed characteristics and rainfall patterns and the number of variables involved in the modeling of the physical processes, the rainfall-runoff relationship is one of the most complex hydrologic phenomena to comprehend. Since 1930s, numerous rainfall-runoff (R-R) models have been developed to predict runoff. Conceptual models provide daily, daily or seasonal estimates of runoff for long term prediction on a continuous basis. The entire physical process in the hydrologic cycle is mathematically formulated in conceptual models which are composed of a large number of parameters. For example, the Stanford Watershed Model is defined by 20 to 30 parameters. The optimization of model parameters is usually accomplished by a trial-and-error procedure because there are numerous model parameters and the interaction of these parameters is highly complicated. The accuracy of model predictions is very subjective and highly dependent on the user's ability, knowledge and understanding of the model and of the watershed characteristics. In conceptual models, continuous rainfall data are usually employed as input data. In addition, runoff data is required for calibration of the model. Calibration of the models require 8 to 20 years of continuous rainfall and runoff data. Though the conceptual models provide reasonable accuracy, their use is limited because of the difficulties discussed. ANNs have been first developed in the 1940s, and, in recent decades, because the current algorithms overcome the limitations of early networks considerable interest has been raised over their practical applications. There is a wide variety of ANN algorithms. However, the main function of all ANN algorithms is to map a set of inputs to a set of outputs. An ANN is described as an information-processing system that is composed of many nonlinear and densely interconnected processing elements or neurons. ANNs are proven to provide better solutions when applied to: (1) problems that deal with noise or involve pattern recognition, diagnosis, abstraction, and generalization; (2) complex systems that may be poorly described or understood; and (3) situations where input is incomplete or ambiguous by nature. It has been reported that the ANN has the ability to extract the patterns in phenomena and overcome the difficulties due to the selection of model form, such as linear, power, or polynomial. Due to its ability to generalize patterns in noisy and ambiguous input data and to synthesize a complex model without a prior knowledge or probability distributions, an ANN algorithm is capable of modeling the rainfall-runoff relationship. As an ANN model is calibrated using automatic calibration techniques, it eliminates subjectivity and lengthy calibration cycles. Some specific applications of ANN to hydrology include modeling daily rainfall-runoff process, assessment of stream's hydrologic and ecological response to climate change, rainfall prediction, sediment transport prediction, and groundwater remediation. In this study, an **ANN** algorithm has been used to model the daily rainfall-runoff relationship in the Khodiyar Catchment located at Dhari, in Amreli district in Gujarat, India. The sensitivity of the network performance to the content and length of the calibration data has been examined using various training data sets. The networks have been trained and tested using data that represent different characteristics of the watershed and rainfall patterns. The performance of the ANN model has been compared with that of existing methods.

An ANN is an information-processing system composed of many nonlinear and densely interconnected processing elements or neurons which are analogous to the biological neurons in the human brain. Neurons in an ANN are arranged in groups called layers. Each neuron in a layer operates in logical parallelism. Information is transmitted from one layer to others in serial operations. A network can be composed of one to many layers. The basic structure of a network usually consists of three layers: the input layer, where the data are introduced to the network; the hidden layer, where data are processed; and the output layer, where the results for given inputs are produced. The architecture of an ANN is designed by weights between neurons, a transfer function that controls the generation of output in a neuron, and learning laws that define the relative importance of weights for input to a neuron. In this study, the training of ANNs have been accomplished by aback-propagation algorithm. In the multilayer feed-forward networks, back-propagation is the most commonly used training algorithm.

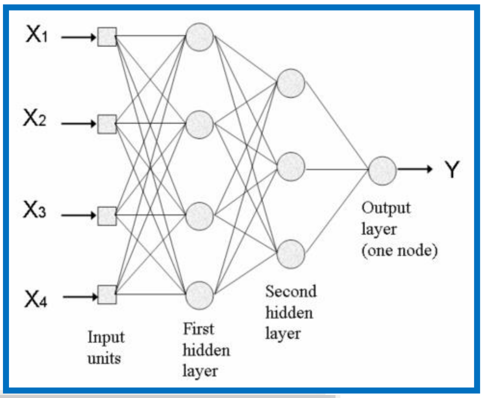

Fig. 1: Structure of a feed-forward ANN Model

II. ARTIFICIAL NEURAL NETWORKS

The network weights are modified with the development of a back-propagation algorithm, by minimizing the error between a target and computed outputs. The information is processed in the forward direction from the input layer to the hidden layers in back-propagation networks, and then to the output layer. The objective of a back-propagation network is to find the weight that approximate target values of output with a selected accuracy. Along with the generalized-delta rule, the least-mean square error method, is used to optimize the network weights in back-propagation networks. Training is composed of two major phases: forward pass and reverse pass. In the forward pass, first the input data are multiplied by the initial weights, then the weighted inputs are added by simply summation to yield the net to each neuron. The net of a neuron is passed through an activation or transfer function to produce the output of a neuron. The modification of the network weights is accomplished with the derivative of the activation function in the back-propagation networks. Therefore, continuous-transfer functions are desirable. The most commonly used continuous-transfer functions in the back-propagation networks are the sigmoid and hyperbolic-tangent functions. This procedure is repeated until the output layer is reached after the output of the neuron is transmitted to the next layer as an input. The error between the output of the network and the target outputs are computed at the end of each forward pass. If an error is higher than as elected value, the procedure continues with a reverse pass; otherwise, training is stopped. In the reverse pass, the weights in the network are modified by using the error value. The modification of weights in the output layer is different from the modification of weights in the hidden layers. The target outputs are provided in the output layer, whereas in the intermediate layers, target values do not exist. Therefore, back-propagation uses the derivatives of the objective function with respect to the weights in the entire network to distribute the error to neurons in each layer in the entire network.

In this study, input dimension includes daily rainfall, average air temperature, humidity, evaporation losses & infiltration losses data. Output dimension is the predicted runoff. Only one hidden layer was used. The appropriate number of neurons in the hidden layer is determined by using the constructive algorithm, by increasing the number of neurons from 4 to 18. There is a use of log-sigmoid, tangent-hyperbolic and linear activation functions. The ANN model for stream flow evaluation was written in the MATLAB environment, version 2013a & “nntool” (Neural Network toolbox) as well as “nftool” (Neural Network fitting tool) were made use of. The L-M algorithms were evaluated for network training so that the algorithm with better achieved accuracy and convergence speed could be selected. All commonly used algorithms for network training in hydrology, i.e. BP, CG and L-M algorithms apply a function minimization routine, which can back propagate error into the network layers as a means of improving the calculated output. The L-M algorithm is viewed as a very efficient algorithm with a high convergence speed.

III. STUDY AREA

Khodiyar Catchment which is located at Dhari in Amreli district, in Gujarat, India has been selected to demonstrate the methodology for modeling daily rainfall-runoff relationship using an ANN. The watershed has a catchment area of 393 km^2. The runoff has been recorded as average and extreme, minimum and maximum values of daily flow for the period between 1983 and 2010 have been used. The rainfall, temperature, humidity, evaporation losses, infiltration losses & runoff data are available as daily averages as well as extreme values at the rain gauge. During the modeling process, it has been assumed that the rainfall, temperature, humidity, evaporation losses, infiltration losses & runoff data at the gauging stations represented the average characteristics of these variables in the River basin. The max annual precipitation is about 1374 mm, max monthly precipitation 847 mm observed in August-2006 and max. daily precipitation is as high as 240 mm observed on 3rd July 2008. Duration of these recorded data was 28 years from 1983 to 2010. A number of ANN models were designed, evaluated & Computational efficiencies of the BP, CG and L-M algorithms and the effect of enabling/disabling of input parameters were also evaluated.

IV. METHODOLOGY

As compared with measurements of soil characteristics, initial soil moisture, and groundwater characteristics measurements of rainfall (P), air temperature (T), humidity, evaporation losses, infiltration losses & runoff and stream discharge (Q) can be obtained easily and cost effectively. Therefore, a model that uses available real-time data would be more easily applied in the operational predict systems. In order to predict Q variables P, T, H, E_{loss} & I_{loss} have been selected to describe the physical phenomena of the rainfall-runoff process. The selection of training data which represents the characteristics of a watershed and meteorological patterns is extremely important in modeling. The training data should be large enough to accommodate the characteristics of the watershed and to contain the requirements of the ANN architecture. An increase in the complexity of a network (i.e., an increase in the number of neurons or layers in a network) will not enable the network to generalize the patterns in the physical phenomena if the information included in the training data set is insufficient. Contrarily, an increase in the complexity of the models might mislead the modeler to over fit the training data and lead to poor predicts. For each one of the developed models, available data were separated as 70% for training, 15% for validation & 15% for testing. Data usage by an ANN model typically requires data scaling which is also known as normalization. In the present paper, the data were scaled in the range of 0 to +1, based on the foll. Equation

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

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

A. *Evaluation criteria for ANN prediction :*

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

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

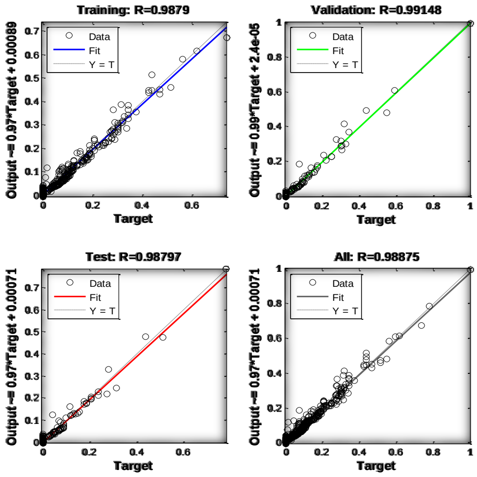

Fig.2: Regression chart of observed v/s predicted Runoff For best Model Architecture 7-14-1

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

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

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

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

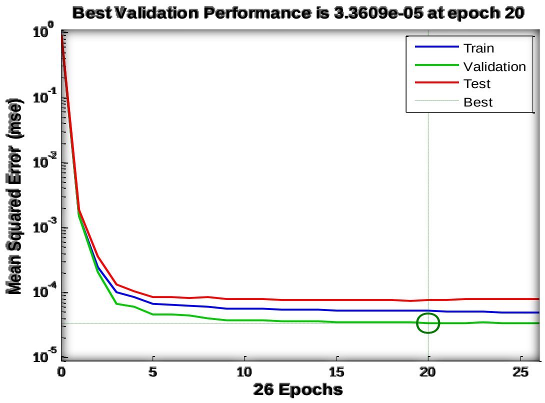

Fig.3: Mean Square Error chart showing best validation performance for Model Architecture 7-14-1

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

In this study, based on the annual average rainfall average- (*A*), dry-year (*D*) and wet-year (*W*) data has been selected for training and testing. Recent data has been used whenever possible, since they reflect the current land-use conditions in the watershed. to illustrate the capability of model in predicting future occurrences of runoff the most current data has been used in the test set, without directly including the land-use characteristics of the watersheds. Networks with various numbers of neurons (e.g., networks with a number of neurons varying from one to as many as 20) have been trained using 40 years data. Combinations that have been used included wet and dry years, average and dry years, and average and wet years (i.e., WA, WD, and AD). Forty years data have been formed as a combination of wet, dry, and average years (i.e., WDA). Using this data, in order to illustrate the effect of input variables, characteristics of data contained in the training set, and the length of training data on the model prediction accuracy several networks have been trained and tested for various combinations of daily P, T, H, E_{loss}, I_{loss} and Q. Daily rainfall- runoff process has been modeled using networks with one hidden layer. Addition of each input variable is based on high predictor-criterion correlations and low inter correlations between the predictors. The procedure that has been used to model rainfall-runoff relationship is summarized in the following steps:

(1) A simple model has been selected by representing runoff at the present time, *t*, as a function of rainfall at time *t* (i.e., *Q*(*t*) = *f* {*P*(*t*)}). Various ANN configurations have been trained and tested using this model (i.e., networks with number of neurons varying from one to as many as 18 in a hidden layer based on the number of observations in the training set). The goodness-of-fit statistics have been computed for both training and testing for each ANN architecture. The best-fit network has been selected among the networks trained, as a representative of this model based on the goodness-of-fit statistics of training and testing.

(2) The rainfall at time *t* - 1 has been added as an additional input variable to the model. Hence, runoff has been expressed as a function of rainfall at time *t* and *t* - 1 (i.e., *Q*(*t*) = *f* {*P*(*t*), *P*(*t* - 1)}). For training and testing procedures and comparision with those for the best-fit model at the previous step, the goodness-of-fit statistics for the present model have been computed. If the goodness-of-fit statistics of the present model have been significantly different from the previous model, then rainfall at time *t* - 2 has been added as another input variable to the present model (i.e., *Q*(*t*) = *f* {*P*(*t*), *P*(*t* - 1), *P*(*t* - 2)}). Until there has been no significant change in model training and testing accuracy based on the goodness-of-fit statistics. This procedure has been repeated by adding rainfall at previous time periods as input variables.

(3) Another input variable, such as temperature, evaporation, or runoff at previous time periods, have been added to the best-fit model obtained from step after step 2 has been completed. Until all the available input variables have been exhausted (i.e., *Q*(*t*) = *f* {*P*(*t*), ….. *P*(*t* - *n*), *T*(*t*), ….. *T*(*t* - *n*), *E*(*t*), ….. *E*(*t* - *n*), *S*(*t*), ….. *S*(*t* - *n*), *Q*(*t* - 1), …. *Q*(*t* - *n*)}) the procedure at step 2 has been repeated for each of these variables. 72 daily rainfall-runoff models have been developed based on the procedure outlined in steps 1–3. The networks have been trained in the development of daily rainfall-runoff models, using a back-propagation algorithm and one hidden layer. Training of the networks has been accomplished using the Neural Network 2013a software. Based on various statistical goodness-of-fit indices the best-fit network for each model has been selected.

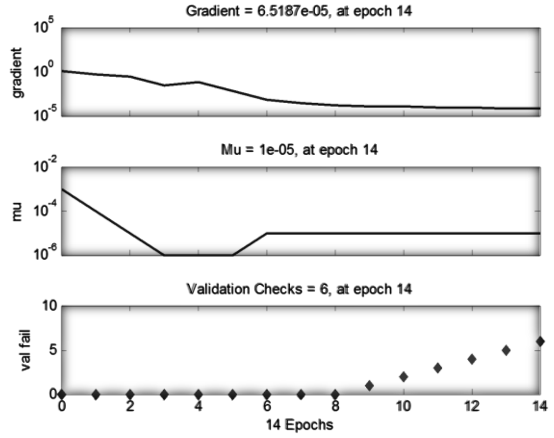

Fig. 4: Gradient, Mu, Validation checks chart for Model Architecture 7-14-1

V. Results and Discussion

A. *Model structures:*

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

- Model 1 is enabled for 4 input variables i.e. rainfall, minimum & maximum temperature & evaporation losses data as input dimension.
- Model 2 is enabled for 5 input variables i.e. rainfall, min & max temperature, evaporation losses & infiltration losses as input dimension.
- Model 3 is enabled for 6 input variables i.e. rainfall, max & min temperature, humidity, evaporation losses & infiltration losses as input dimension.
- Model 4 is enabled for 7 input variables i.e. rainfall $P_{(t)}$, $P_{(t\ -1)}$, max & min temperature, humidity, evaporation losses & infiltration losses as input dimension.

Equations 1 to 4 represent Model 1 to Model 4 respectively.

$Q_{(t)} = f\ \{ P_{(t)}, T_{max}, T_{min,} E_{loss} \}$ ……………………(1)

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

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

$Q_{(t)} = f\ \{ P_{(t)}, P_{(t\,-1)}, T_{max}, T_{min}, E_{loss}, I_{loss}, H \}$ …....(4)

where; $Q_{(t)}$ is simulated modeled run off; {Q} is daily runoff data; {P} is daily rainfall data, $T_{(t)}$ is average daily air temperature data, E_{loss} is Evaporation loss, I_{loss} is the Infiltration loss and H is the humidity.

Model	R-values				M.S.E.
Architec-Ture	Training	Validation	Testing	Total	values
D-7-14-1	0.987	0.991	0.987	0.988	0.000033
D-6-12-1	0.977	0.981	0.970	0.976	0.000056
D-5-10-1	0.966	0.982	0.924	0.957	0.000046
D-4-8-1	0.936	0.885	0.989	0.935	0.00013

Table.1 : R-value & MSE values of 4 Best Models

B. *Model performance levels:*

Table 1 shows individual model performance levels as measured by MSE and R and individual model architecture as represented by the number of neurons in the input, output and hidden layers. Furthermore, computed rainfall-runoff by individual models are compared with the corresponding observed values and illustrated by their graph (Fig. 2, 3, 4 & 5) which is indicated by the results. It can be concluded that Model 1 resulted with the lowest achieved performance levels. Model 2 & 3 resulted in a considerable improvement of the performance levels & Model 4 has highest achieved performance levels with R = 0.98875 & least MSE of value 0.0000336.

The goodness-of-fit statistics for the networks trained using 72 models are presented in Fig. 2. Due to its high training and testing accuracy a network with 14 neurons that has been trained using WDA data and Model 4 (M4-WDA-10) based on the results of training Models, has been selected as the best-fit network to model the rainfall-runoff relationship. Therefore, addition of *P* at *t* - 1 improved both training and testing accuracy, as well as the estimation of peak discharges. As compared with other Models both the testing and training accuracy and percent peak discharge estimates for the best-fit network for Model 4 significantly improved. This result indicates that the addition of runoff and rainfall at time *t* - 1 did not improve the training or testing accuracy.

VI. CONCLUSION

Where input data are incomplete and ambiguous by nature the ANN methodology has been reported to provide reasonably good solutions for circumstances where there are complex systems that may be poorly defined or understood using mathematical equations, problems that deal with noise or involve pattern recognition and situations. It has been believed that ANN could be applied to model the daily rainfall-runoff relationship because of these characteristics. In previous discussions, it has been demonstrated that the ANN rainfall-runoff models exhibit the ability to extract patterns in the training data. Based on the ratio of standard error to standard deviation and percent prediction of peak discharges, supports this conclusion favorably comparable with the training accuracy. For Khodiyar Catchment, an ANN model provided higher training and testing accuracy when compared with the regression and simple conceptual models. The accuracy of ANN compared favorably with the model accuracy of existing techniques based on the goodness-of-fit statistics. The selection of training or calibration data has a very large impact on the model prediction accuracy. The model will not provide reliable future predictions if the calibration data do not represent the characteristics of a watershed and the climate. In this study, networks have been trained and tested using wet-year, dry-year and average-year data to illustrate the impact of the content of training data on network prediction accuracy. The networks have been able to recognize the patterns in test data that contains low and average flow conditions when the input data include the high flow extremes. The networks have been able to distinguish patterns in the test data that have been different from the training data since the wet-year data includes information on both high and low-flow conditions. Compared with the networks trained using a combination of dry and average year data based on the ratio of the standard error of the estimate to the standard deviation, networks trained using data that include wet and dry or wet and average years had the highest prediction accuracy. In addition, the former predicted peak discharges closer to their observed values as compared with the latter. In the ANNs presented in this study, it has been observed that the effect of the length of training data on the network accuracy has been not as dramatic as the effect of the content of the training data. Therefore, the length of data required is correlated to the complexity of the model. When the complexity of a model increases, the amount of data that is needed increases assuming that the complexity of a network is described by the number of input variables. In this study, the length of the data record has been sufficient for the selected ANN architecture.

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

In this study, the results show clearly that the artificial neural networks are capable of modeling rainfall-runoff relationship. In this research, the influences of training algorithm efficiencies and enabling/disabling of input dimension on rainfall-runoff modeling/simulation capability of the artificial neural networks was applied. A watershed system of Khodiyar catchment located in Amreli district, Gujarat, India was selected as case study. The used data in ANN were daily hydrometric and climatic data with 28 years duration from 1983 to 2010. For the mentioned model, 70 % data were used for its training but for the validation/testing of the model remaining 30 % data were applied. Four types of model structures were developed to investigate the probability impacts of enabling/disabling rainfall-runoff, temperature, humidity, evaporation losses & infiltration losses input data. Computational efficiencies i.e. Better achieved accuracy and convergence speed, were evaluated for the Back-Propagation (BP), Conjugate Gradient (CG) and Levenberg-Marquardt (L-M) training algorithms. So under each Model category, by applying above 3 algorithms & by changing number of nodes in the hidden layer from 4 to 18, 24 Models were developed. Totally 72 Models were prepared for selecting the Best Model for this catchment. The L-M algorithm proved to be more efficient than the CG and BP algorithm. Based on the results, coefficient of determination (R) & validation stage of Mean Square Error (MSE) measures were: 0.935, 0.0013 (Model 1); 0.957, 0.000046 (Model 2); 0.976, 0.000056 (Model 3); 0.988, 0.000033 (Model 4) as indicated in Table-1. As indicated by the results, model 4 provided the highest performance.

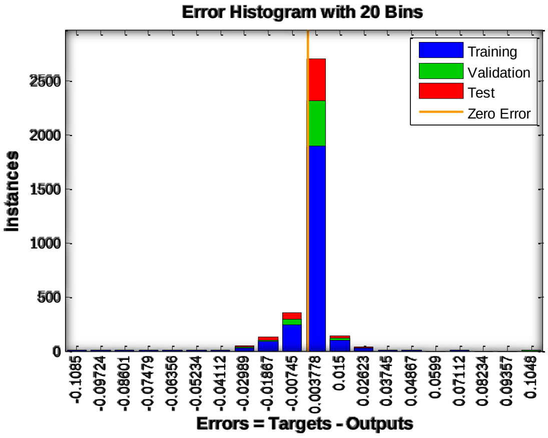

Fig. 5: Error Histogram chart for Model Architecture 7-14-1

The charts for the best Model having Model Architecture 7-14-1 are mentioned in fig. 2, 3, 4 & 5. This was due to enabling of the rainfall, average temperature, resulting in improved training and thus improved prediction. The results of this study has shown that, with combination of computational efficiency measures and ability of input parameters which describe physical behavior of hydro-climatologic variables, improvement of the model predictability is possible in artificial neural network environment.

ACKNOWLEDGEMENT

The writer wish to express gratitude to SWDC, Gandhinagar for providing data of the study area.

REFERENCES

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