Modelling Reference Evapotranspiration Using Artificial Neural Network: A Case Study of Ameleke watershed, Ethiopia

Pakhale G.K.[*], Nale J.P. [**], Temesgen W.B.[*] and Muluneh W.D. [***]

[*] Dept. of Hydraulics and Water Resources Engineering, College of Engineering and Technology, Dilla University, Ethiopia
[**] SWCE, Dr. A.S. College of Agricultural Engineering, Rahuri, Maharashtra, India
[***] Dept. of Civil Engineering, College of Engineering and Technology, Dilla University, Ethiopia

Abstract- This study investigates the utility of artificial neural networks (ANNs) for estimation of daily grass reference crop evapotranspiration (ETo) and compares the performance of ANNs with the conventional method (Penman–Monteith) used to estimate ETo. Several issues associated with the use of ANNs are examined, including different learning methods, number of processing elements in the hidden layer(s), and the number of hidden layers. The input parameters and the ANN model architecture was decided based on use of MARS tool and trial and error approach leading to optimal error statistics. Different ANN architectures namely BPNN, RBFNN and GRNN were used. Model performance show that BPNN architecture suits for prediction of reference crop evapotranspiration and can be used for future scenario in the Ameleke watershed.

Index Terms- Artificial neural network, Reference crop evapotranspiration, Normalization, Weights

I. Introduction

A common practice for estimating evapotranspiration (ET) from a well irrigated agricultural crop is to estimate reference evapotranspiration (ET_o) from a standard surface and to apply an appropriate empirical crop coefficient, which accounts for difference between the standard surface ET and crop ET. In the past few decades, several studies (Hill et al. 1983; Allen, 1983; Allen et al. 1994 a, b; Allen et al. 1998) have focused on the development of accurate methods for ET_o estimation and improving the performance of existing methods due to wide application of ET_o data. ET_o can either be estimated using lysimeter or water balance approach or estimated indirectly using the meteorological data. The combination approach links evaporation dynamics with fluxes of net radiation and aerodynamic transport characteristics of natural surface. Based on the observation that biotic factors are not the only factors for latent heat transfer in plants, Monteith (1981) introduced a surface conductance term that accounts for the response of leaf stomata to its hydrologic environment. The Penman-Monteith method was ranked as the best method was for all climatic conditions (Jensen et al. 1990; Allen et al. 1998), however, ranking of other methods varied depending on their adoption to local calibrations and conditions.

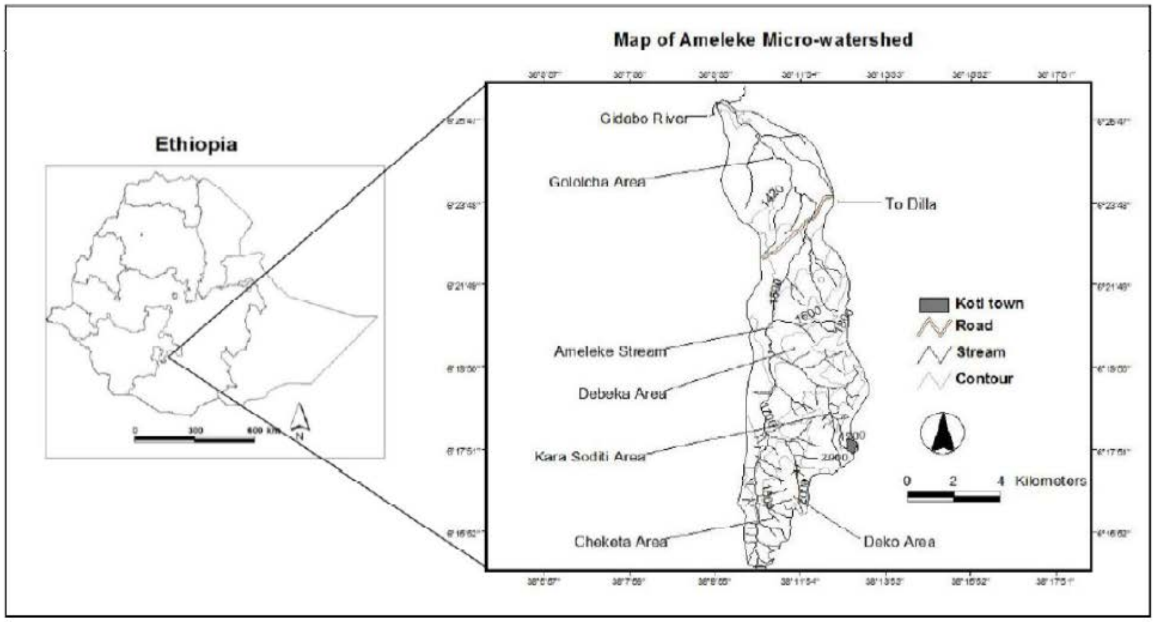
Therefore, there is a need for developing an appropriate method for estimating ET_o on a daily scale, using simpler and fewer input data for temperate hilly region. Artificial neural network (ANN) modelling allows easier translation between human and computers for decision making and better way to handle imprecise and uncertain information. ANN are data processing systems comprising a large number of simple, highly interconnected processing elements (artificial neurons) in an architecture inspired by the structure of the cerebral cortex (Tsoukalas and Uherg, 1996). It has been demonstrated that neural networks are a competitive substitute to conventional classifiers for many practical classification problems (Zhang, 2000). ANNs have the gain over deterministic models, as the ANNs require lesser data and are capable of long term forecasting. There has been a rising drift of application of ANNs in water resources and hydrologic modeling (ASCE, 2000; Sudheer et al., 2002; Yitian and Gu, 2003; Zhang and Govindaraju, 2003) and land drainage engineering (Shukla et al., 1996; Yang et al., 1996). The ANN has been successfully employed in the studies of rainfall-runoff processes (Hus et al. 1995; Smith et al. 1995). The problem of rainfall-runoff modelling has perhaps received the maximum attention of ANN modellers. Besides this, ANN has been applied in the field of stream flow (Markus et al. 1995), ground water modelling (Patil et al. 2006) and in estimation of precipitation (Kuligowski, et al. 1998). The ANN has been successfully employed in the studies of evapotranspiration modelling by various researchers for different places (Jyothiprakash et al. 2002; Kumar et al. 2002). However, application of ANN to model ET_o, especially for the temperate hilly region, has not been reported. This study is an attempt to model evapotranspiration using artificial neural network for a solani watershed in the hilly region of uttrakhand.

However, the disadvantages of ANN is that it is based on a ‘black box’ approach and the result is obtained through a series of experiments (Sharma et al., 2003). Thus enclosure of system parameters as processing elements (PEs) or as mathematical association with the PEs in the input layer will reorient the ANNs from a complete ‘black box’ to a ‘gray box’ approximation (Sarangi and Bhattacharya, 2005). The Multivariate Adaptive Regression Splines (MARS) can be used as a tool to recognize unneeded parameters in predictive analysis and select the sensitive parameters for consideration in model development (Abraham and Steinberg, 2001). The use of MARS tool in selection of the sensitive input parameters and development of ANN models to predict reference evapotranspiration in this study is a new approach for estimation of reference evapotranspiration in the study area.

II. MATHERIALS AND METHODS

2.1 Study Area and Data Acquisition

The Ameleke watershed is located between 6°15'N to 6°26'N latitude and 38°10'E to 38°12'E longitude. The watershed has an area of 70 km^2. Ameleke watershed is one of the tributary of Gidabo River. The mean monthly temperature of this watershed ranges from 23.2 °C in July to 30.2 °C in January. Regarding rainfall, mean annual rainfall ranges from 800 to 1400 mm in the upper part of the watershed while it is 105.5 mm in the lower stream of the watershed. The rain has bimodal pattern where March to June and September to November are rainy seasons. The rainfall variability is very high. Seasonally *Bega* has high rainfall variability than *Belg* and *kiremt*. In Ameleke watershed altitude ranges from 1200 to 2000 masl. In The upper stream of the watershed coffee, *Enset*, maize and teff are major crops where as in the lower and middle stream of the watershed livestock production is the main stay of the community.



2.2 Penmen-Monteith method

Actual evapotranspiration estimation from the field requires Lysimeter data. However, it is not always possible to estimate ET$_o$ using lysimeter, because it is time-consuming, requires large infrastructure and precise and carefully planned experiments. The Food and Agriculture Organisation (FAO) of United Nations suggested use of the FAO Penman-Monteith method for estimation of ET$_o$ (Smith et al. 1998). Although, it requires several climatic data some of which have no actual measurements but are estimated from measured weather parameters.

The final form of the FAO Penman-Monteith equation is as given below:

$$ET_0 = \frac{0.408\Delta(R_n - G) + \gamma\left[\frac{900}{T+273}\right]u_2(e_s - e_a)}{\Delta + \gamma(1 + 0.34u_2)} \quad \text{... (1)}$$

where, ET$_o$ - ETo (mm day^{-1}),
Δ - slope of saturation vapour pressure curve (kPa^{0}C^{-1}),
R$_n$ - net radiation (MJ m^{-2} day^{-1}),
G - soil heat flux density (MJ m^{-2} day $^{-1}$),
γ - psychometric constant (kPa^{0}C^{-1}),
T - mean daily air temperature (^{0}C)
e$_s$ - Saturation vapour pressure (kPa),
e$_a$ - actual vapour pressure (kPa) and
u$_2$ is average daily wind speed at 2 m height (m sec^{-1}).

Calculation of ET$_o$ using equation 1 on daily basis requires meteorological data consisting of maximum and minimum daily air temperatures (T$_{max}$ and T$_{min}$), mean daily actual vapour pressure (e$_a$) derived from dew point temperature or relative humidity (Rh$_1$ and Rh$_2$), daily average of 24 hours of wind speed measured at 2m height (u$_2$), net radiation (R$_n$) measured or computed from solar and long wave radiation or the actual duration of sunshine hours (n). The extra terrestrial radiation (R$_n$) and day light hours (n) for a specific day of the month are also computed. As the magnitude of soil heat flux (G) beneath the reference grass surface is relatively small, it is ignored for daily time stake. This Peneman-Monteith method has been used in this study for finding out ET$_o$. The values were used for training, testing and validation of ANN model.

2.3 Multivariate Adaptive Regression Splines

The MARS 2.0 software (Friedman, 1991) was used to estimate the relative significance of the available data in predicting the ET$_o$. The concepts used in MARS tool permits the user to analyze the data set and generate the input parameters, which are of more significance in generation of the desired output and display the empirical model of best fit. Finally, the validation input data set is entered into the model and the output is compared with the observed values to decide on the future model applicability.

In this study, the data of maximum, minimum and average temperature (T_{max}, T_{min} and T), average relative humidity (Rh), , daily average of 24 hours of wind speed measured at 2m height (u_2), rainfall (P), net radiation (R_n) measured or computed from solar and long wave radiation, actual duration of sunshine hours (n) were used as input. ET_o were used as output parameter. After running the MARS tool with the data sets of Jan 2010 to June 2012 the input parameters: mean temperature (T), mean relative humidity (Rh) and daily average of 24 hours of wind speed measured at 2m height (u_2) topped the list. The rest of the parameters were towards the bottom in the list of relative importance. Therefore, the input data of T, Rh and u_2 were used as the input parameters for ANN models against the output parameter ET_o. In general, the ANN model is operated by using the available input and output responses without considering the inherent system parameters. Therefore, in an attempt to elevate the complete ‘‘black box’’ approximation of the ANN model, in this study, the MARS tool was used to consider most of the system response parameters as detailed in Penman-Monteith method and the relatively important parameters obtained were used as input nodes (PEs) to selected ANN model. So the completely ‘‘black box’’ nature of prediction through ANN model were minimized by inclusion of the MARS derived system response parameters for predicting ET_o. Therefore, the inclusion of MARS generated parameters as PEs of the input layer was an effort towards elevating the ANNs from ‘‘black box’’ modelling approach to a ‘‘gray box’’ approximation.

2.4 ANN model architecture

ANN, which emulates the parallel distributed processing of the human nervous system, has proven to be successful in dealing with complicated problems such as function approximation and pattern recognition. The stored information-processing elements are interconnected and organized in layers. The connection strengths, also called network weights, can be adapted such that the output of the network matches a desired response. In hydrology, prediction of evapotranspiration, runoff and sediment loss from watershed systems has been a difficult subject due to complexity of the physical processes involved and variability of rainfall in space and time. The most commonly used ANN for hydrological modelling is a feed-forward network with the BP training algorithm (Zhang and Govindaraju, 2003), which is also capable of nonlinear pattern recognition and memory association. Standard multi-layer feed-forward network is capable of approximating any measurable function to any desirable degree of accuracy. In that sense, the multi-layer feed-forward architecture gives neural networks the potential of being universal approximations rather than the specific choice of an activation function. In general, application of ANN in modelling, design or problem-solving is preferred in situations where the system response parameters of a real-world phenomenon are either poorly defined or misunderstood, and where observations of the process may be difficult or impossible to perform, and also when it is difficult to recognize the complex relationships between aspects of the process under investigation (Haykin, 1999).

2.4.1 BPNN

Among different ANN algorithms, feed-forward network with the BP training is widely used and is capable of recognizing the nonlinear pattern and memory association (Zhang and Govindaraju, 2003). BPNN is a multilayer perceptron network in which each neuron is connected with a number of input arcs (U_1 to U_n). The network is associated with each neuron (i) having weight W_{ij}, which represents a multiplication factor to a value passing to the neuron. Finally, a neuron sums the values of all inputs and represented as:

$$S_j = \sum_{i=1}^{n} W_{ij} U_i \qquad \text{... (2)}$$

In Fig. 1, Wu corresponds to the summation term used in Eq. (1). The term b is called bias. Finally, an activation function is applied to S_j for obtaining the final output from the neuron. When a BPNN training algorithm is used, the sigmoid activation function is most often preferred (Sivakumar et al., 2002). The sigmoid function (φ) is given by

$$\varphi(S_j) = \frac{1}{1+e^{S_j}} \qquad \text{... (3)}$$

It is obvious from the sigmoid activation function that the range of nodal output in entire network must be within 0 to 1. That means the input variables are to be kept smaller in order to avoid saturation effect, caused by the sigmoid function during the analysis. Thus, input and output variables must be normalized before the initiation of the training of the neural network. Most popular scheme to normalize input and output variables is as follows:

The normalization is carried out with respect to the maximum value of variables under consideration. That is, the value of X and Y at time t can to normalized as,

X_1 [normalised input] = X_1 / X_{max} ... (4)
Y_1 [normalised input] = Y_1 / Y_{max} ... (5)

where the subscript max defines the maximum value of respective variable and t is the time. After the normalization, the input and output variables vary in between 0 to 1 (0 < = normalized value < =1).

In the present study, the BPNN had one input layer, hidden layer and output layer each. Each neuron in the input layer is connected to each neuron in the hidden layer by weight W_{ij}. After an input neuron receives a signal U_i, it transmits it to the hidden neuron. Each hidden neuron then computes the sum $U_i W_{ij}$ entering from each input neuron, and transforms this value to an output signal using sigmoid function.

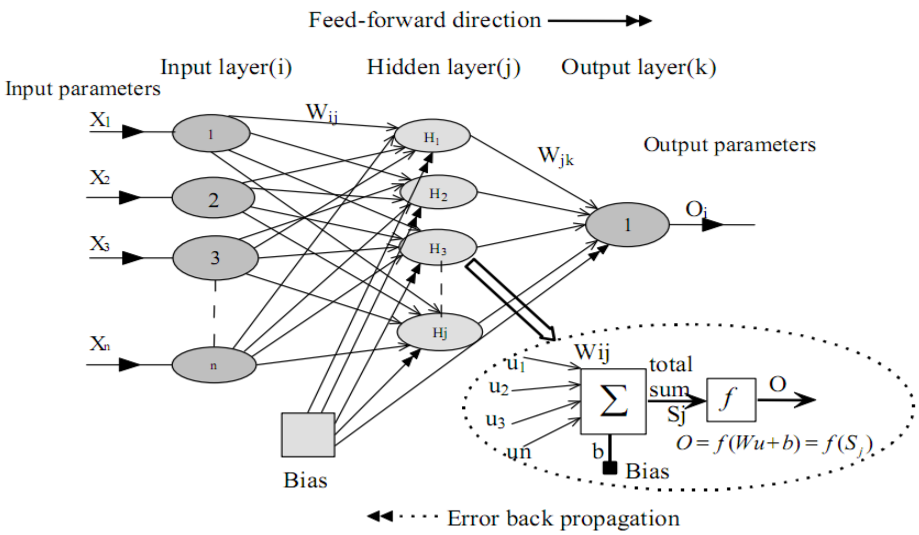


Figure: 2 ANN model Architecture

The ANN input layer in this study consisted of three processing elements (T, Rh, u_2) and the processing elements (PEs) of the hidden layer were finalized by comparing the root mean square error (RMSE) of the network training using different numbers of PEs. The output layer consisted of one processing element representing the ET_o. The entire data was divided to training, testing and validation sets and the training and testing data were fed to the BPNN to select the optimal architecture based on the RMSE values. During the process of training, the actual output value was compared with the desired output and the error was calculated. The error values were then propagated back into the network to update connection weights between the different layers. These processes were repeated until the network has been trained to the lowest RMSE.

2.4.2 Radial basis function neural network (RBFNN)

RBFNN has a feed forward architecture. It consists of three layers viz. one input layer, one hidden layer and one output layer with a number of PEs or nodes in each layer. There is one neuron for each predictor variable. In case of categorical variables N-1 are used where N is number of categories. Input neurons feed normalized data to each of the neuron in the hidden layer. However, the structure of an Radial Basis Function Neural Network is unique which allows for adaptive determination of the hidden neurons during training of the network (Sharma et al.,2003). Each input neuron is completely connected to all hidden neurons, and hidden neurons and output neurons are also interconnected to each other by a set of weights. Information fed into the network through input neurons is transmitted to hidden neurons. Each hidden neuron then transforms the input signal using a transfer function f. For the present model, the Gaussian function was used. It is a positive radial symmetric function (kernel) with a center m and a spread s. The spread is the radial distance from the center of the kernel, within which the value of the function is significantly different from zero. This is called the receptive field $(\mu \pm \sigma)$ of a hidden neuron. An input pattern falling within the receptive field will cause a significant response. For each input pattern, the hidden neurons compute the distance between the input signal and the center of the receiving field. For Gaussian function, the response is unity if this distance is zero, and decays to zero when the distance is greater than the spread.

The basic difference between Back Propagation Neural Network and Radial Basis Function Neural Network is that the latter model represents the inputs presented to the network during training phase in local spaces with each local space being represented by a hidden neuron. Therefore, any input to the model in the testing phase that lies near a local space is closely predicted. On the other hand, the Back Propagation Neural Network model maps the relationships between the inputs and outputs in global space for the training scenarios. Therefore, the model fails to predict the localized variation in data in the testing phase and this is also reflected in model validation phases. In relation to the processing time, the Radial Basis Function Neural Network is faster than the Back Propagation Neural Network for larger data sets (Kim et al., 2003). In training a Radial Basis Function Neural Network, the hidden neurons are self-organized with the training process (Zhang and Kushwaha, 1999). For this purpose, the orthogonal least squares (OLS) algorithm proposed by Chen et al. (1991) was employed. According to this algorithm, the number of hidden neurons at the beginning of training is zero. The hidden neurons are added one by one with training until the output of the network is within a target precision. For each iteration, the RMSE from the network is computed. If the error is lower than a predefined tolerance (selected from the lowest RMSE in Back Propagation Neural Network training), the training is stopped and the number of neurons added to the hidden layer represent the number of hidden neurons required.

2.4.3 General regression neural network

The General Regression Neural Network algorithm has little resemblance to the more widely used Back Propagation Neural Network but it is one of the variants of Radial Basis Function Neural Network (Huang and Williamson, 1994). It has, at its origin, one of the most frequently used statistical techniques, i.e.

regression analysis and no iteration are involved in computing with the General Regression Neural Network algorithm. Another striking feature is that, unlike Back Propagation Neural Network, a General Regression Neural Network does not touch to local minima, can handle incomplete patterns and approaches the problem on the basis of the probability density function (pdf) of the training data (Huang and Williamson, 1994). General Regression Neural Network uses one-pass learning algorithm which can be used for the estimation of continuous variables, and converge to the underlying regression surface. Mathematically, General Regression Neural Network uses a standard statistical algorithm for calculating the conditional mean Y of a scalar random variable y given a measurement X of a vector random variable x. The vector x corresponds to the input to the network and the random variable y corresponds to the output of the network. If there is more than one output node, the same algorithm is used on each output node. General Regression Neural Network is dominated by the estimation of probability density function of x and y and is also used as a static regression technique. General Regression Neural Network can be used in situations where the statistics of the data are changing over time. This is achieved by specifying a time constant and a threshold, which are used to reset a pattern node if it has not been used recently. The principal advantages of General Regression Neural Network are its quick learning and fast convergence to optimal regression surface with large numbers of data sets when compared with Back Propagation Neural Network and Radial Basis Function Neural Network (Kim et al., 2003).

2.5 Neural Network Simulation

The available data were divided into training (50% of data), testing (30% of data) and validation (20% of data), with the training and testing files comprising three inputs and one output, and the validation file comprising only the input parameters that were not used for the training and testing processes. The data were partitioned as per the indicated percentages to prepare separate data sets for training, testing and validation processes of the ANN model. However, the percentage of data used for partitioning is based on the concept that major share of it should be used for training processes followed by testing and validation processes. The data were further shuffled within the spreadsheet and prepared for analysis to nullify the presence of any existing trend and inherent properties within the data (Sarangi and Bhattacharya, 2005; Zhang and Govindaraju, 2003; Patel et al., 2002)

2.6 Sensitivity Analysis and optimum network configuration

Sensitivity analysis was done to determine the optimum network configuration for BPNN and GRNN by varying two network parameters, learning rate and number of hidden neurons that minimized the error of estimation. Learning rate indicates the rate of change of connection weights during training. A high learning rate causes oscillation of the connection weights resulting in large generalization error, while a low learning rate results in a significant increase in training time. It was observed that, with use of more neurons (>25) in hidden layer of BPNN, the network becomes over fitted, in which case it is capable of fitting the training data very well but incapable of generalizing for unknown inputs. Also, a large number of hidden neurons significantly increase the network training time. A small number of hidden units results in under fitting due to the lack of enough processing units to map the input/output relationship.

Sensitivity analysis was also done to determine the optimum value of tolerance and receptive field for the RBFNN. Also, for each iteration, the sum of squared error from the network was computed. When the error became lower than a predefined tolerance, the training was stopped. At this stage, the numbers of neurons added to the hidden layer represented the number of hidden neurons required. If the sum of squared error was above the tolerance then the input pattern with largest error was identified and added to the hidden layer. This process was continued till the network error was minimum and within the tolerance limit. It was also observed that with the increase in the number of hidden neurons the computational time increased. Keeping this in view, the trial and error approach was employed in assigning the number of neurons in RBFNN. It was revealed that, with the change of receptive field from 5 to 40 with an increment of 5, keeping the tolerance constant at 5, the number of neurons in hidden layer increased and the statistical parameters (coefficient of determination (R^2) and model efficiency (E)) improved, indicating better network performance. But, with further increase in receptive field (>25), there was no improvement in statistical parameters (R^2 , E). In this study, receptive field of 20 with tolerance constant of 10 leading to 30 numbers of neurons in the hidden layer was found optimal (RMSE = 0.012) for the operation of RBFNN.

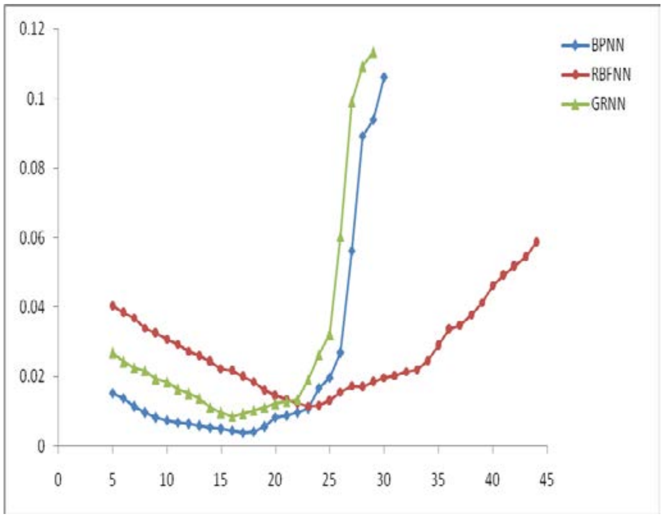


Figure 3: The RMSE of BPNN, RBFNN, GRNN for prediction of ET_o

The number of neurons of the hidden layer in BPNN model and the optimal nodes were selected based on the RMSE of the learning results (Fig. 3). It is seen from Fig. 3 that the RMSE was lowest for the BPNN model (0.0045) for 17 hidden layered PEs than GRNN model () and RBFNN () with 24 layered for modelling ET_o. Finally, the BPNN model with the optimal architecture (1 input layer with 3 PEs, 1 hidden layer with 17 hidden neurons and 1 output layer with 1 PEs) resulted in the statistical parameters (R^2 , E) comparatively higher than RBFNN and GRNN algorithms. In this study, the learning rate was varied from 0.02 to 0.08, hidden neurons varied from 5 to 40. The low ranges of learning rate were chosen because high fluctuations in error were observed at higher learning rates.

2.7 Evaluation of ANN model

The BPNN model was run with the selected network architecture (Fig. 3) using the independent training and testing data sets. Then the trained model was validated with the unexposed data followed by estimation of R^2 and E, using the relation:

$$E = 1 - \frac{\sum_{i=1}^{r} (m_i - p_i)^2}{\sum_{i=1}^{r} (p_i - p)^2} \quad \ldots (6)$$

where r, total number of observations; p_i, i^{th} value from Penman-Monteith method; p, mean of values from Penman-Monteith method and m_i, i^{th} model predicted value. The best model was selected based on the R^2 and E value approaching 1.0 (James and Burgess, 1982)

III. Results and Discussion

The results of BPNN are validated with Penman-Monteith method results. Monthly mean of Penman-Monteith method was compared with BPNN predicted monthly mean and monthly variation of evapotranspiration was observed.

2.8 Performance of ANN model

Based on the statistical parameters (R^2, E) as discussed, the BPNN neural network with 17 neurons in the hidden layer and a learning rate of 0.02 was found optimum for prediction of ET_o. It was also observed that the change in the epoch numbers for normalized cumulative delta learning rule in BPNN did not affect the prediction accuracy significantly. The epoch is the number of sets of training data sets presented to the learning cycles during weight updates. The variation of epoch from 5 to 25 tried for training the BPNN model did not yield any significant variation of the RMSE value. Therefore, the developed BPNN model was validated using the model efficiency factor E and R^2 of observed and model predicted values (Fig. 4). It was observed that BPNN performed well for prediction of ET_o with R^2 and E values approaching 1.

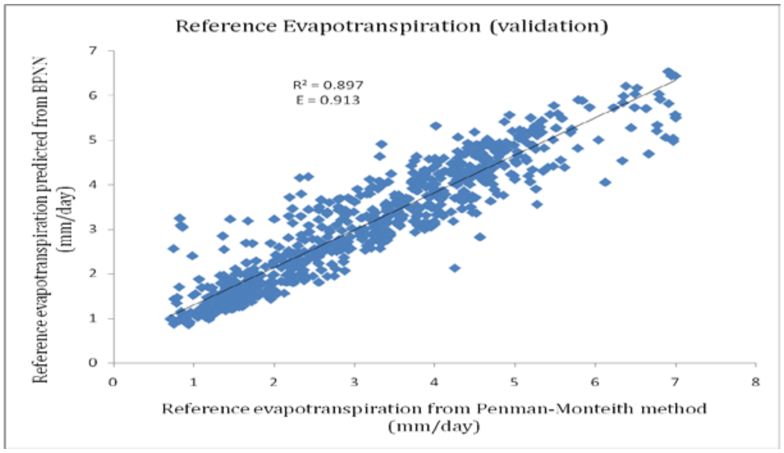


Figure 4 : The ET_o from Penman- Monteith and predicted using BPNN for year 2012.

2.9 Comparison of monthly evapotranspiration variation

The monthly mean of Penman-Monteith method and predicted value from BPNN of ET_o for the year 2012 is plotted in figure. It observed that BPNN has predicted the monthly ET value significantly well. In the month of April, May and June BPNN has slightly overestimated the ET value where as in other months close agreement has been found between Penman-Monteith method and BPNN predicted evapotranspiration.

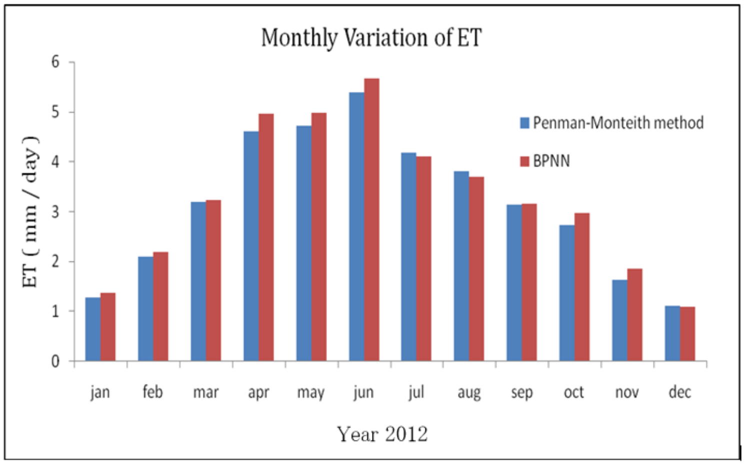


Figure 5: Comparison of monthly mean of ET_o predicted using BPNN and estimated by Penman-Monteith method.

IV. Conclusion

This study was done to investigate the applicability of ANN approach in modelling the ET_o from Ameleke watershed in the hilly Gadio zone, Southern Ethiopia. The input parameters and the ANN model architecture was decided based on use of MARS tool and trial and error approach leading to optimal error statistics. The results of the predictability of ANN model (BPNN model) were then compared with those of Penman-Monteith method to validate the model performance. It was concluded that, BPNN model performed better for prediction of ET_o. Therefore, BPNN modelling approach detailed in this study can be used for prediction of ET_o with acceptable accuracy using minimal input data. The ANN approaches are simpler, relatively faster in model development and simulation and can operate on minimal data structure in comparison to Penman-Monteith method. Also, consideration of the system based parameters in preparation of the PEs of ANN input layers resulted in elevating the complete ‘‘black-box’’ approximation of ANN towards ‘‘gray box’’ model representations.

REFERENCES

[1] Abraham, A., Steinberg, D., 2001. Is neural network a reliable forecaster on earth? A MARS query. In: Jose, M., Alberto, P.(Eds.), Bio-Inspired Applications of Connectionism, Lecture Notes in Computer Science, vol. 2085. Springer-Verlag,Germany, Spain, pp. 679–686.

[2] Allen, R.G. 1983. New approaches to estimating crop evapotranspiration. Acta Hort, 335; 287-294.

[3] Allen, R.G. Smith, M., Pereira, A. and Pereira, L.S. 1994 a. An update for the definition of ETo. ICID Bulletin, 43 (2): 1-34.

[4] Allen, R.G. Smith, M, Pereira, A. and Pereira, L.S. 1994 b. An update for the calculation of ETo. ICID Bulletin. 43 (2) 35-92.

[5] Allen, R. G. Pereira, L.S., Raes, D. and Smith, M. 1998. Crop: Guidelines for computing crop water requirements. Irrigation and Drainage paper No. 56. FAO, Rome Italy.

[6] ASCE, 2000. Artificial neural networks in hydrology. II: Hydrologic applications. J. Hydrologic Engineering, 5(2): 124-137.

[7] Bhattacharya, A.K., 1999. Drainage of agricultural lands. In:Singh, G.B., Sharma, B.R. (Eds.), 50 Years of Natural Resource Management Research. Division of Natural Resource Management, ICAR, Krishi Bhavan, New Delhi, India, pp. 347–362.

[8] Chen, S., Cowan, C.F., Grant, P.M., 1991. Orthogonal least squares learning algorithm for radial basis function networks. IEEE Trans. Neural Networks 2, 302–309.

[9] Friedman, J.H., 1991. Multivariate adaptive regression splines. Ann. Stat. 19, 1–141.

[10] H. P., Brakensiek, D.L. (Eds.), Hydrological Modeling of Small Watersheds. American Society of Agricultural Engineers, St. Joseph, MI, pp. 215–257.

[11] Hill R. W., Johns, E. L. and Frevert. D. K. 1983. Comparison of equations used for estimating agricultural crop evapotranspiration with field research. Bureau of Reclamation, U. S. Dep. Of the Interior Eng. and Res. Ctr., Denver, Colo.

[12] Huang, Z., Williamson, M.A., 1994. Geological pattern recognition and modeling with a general regression neural network. Can. J. Explor. Geophys. 30 (1), 60–68.

[13] Hus, K., Gupta, H. V. and Sorooshian, S. 1995. Artificial neural network modeling of the rainfall-runoff process. Water Resour. Res., 31(10): 2517-2530.

[14] James, I.D., Burgess, S.J., 1982. Selection, calibration and testing of hydrologic models. In: Haan, C.T., Johnson .

[15] Jensen, M.e., Burman, R.D., and Allen R.G. 1990 Crop and irrigation water requirements. Manual and Reports on Engineering Practice No. 70, ASCE, New York.

[16] Jyothi Prakash, V., Ramchandran, M.R. and Shanmuganathan, P. 2002. Artificial neural network model for estimation of REFET. J. The Institution of Engineers (India). 83: 17-24.

[17] Kim, B., Kim, S., Kim, K., 2003. Modelling of plasma etching using a generalized regression neural network. Vaccum 71, 497–503.

[18] Kuligowski. R. J. and Barros, A.P. 1998. Experiments in short tyerm precipitation forecasting using artificail neural networks. Monthly Weather Rev., 126(2): 470-482.

[19] Kumar, M. Raghuwanshi, N.S., Singh, R. Wallender, W.W. and Pruitt, O 2002 Estimating evapotranspiration using ArtificialNeural Network. J Irrig. and Drain. Engg., ASCE, 128 (4): 224-233.

[20] Markus, M Salas J.D. and Shin, H.K. 1995. Predicting streamflows based on neural networks. Proc. Ist. Int. Conf. On Water Resour. Engg. 1641-1646.

[21] Monteith, J. L. 1981. Evaporation and surface temperature. Quart. J. Roy. Meteor. Soc., 107:1-27.

[22] Patel, R.M., Prasher, S.O., Goel, P.K., Bassi, R., 2002. Soil salinity prediction using artificial neural networks. J. Am. Water Resour. Assoc. 38 (1), 91–100

[23] Patil. V.V., Singh Ramesh, Sharma. H.C. and Mehta, Rama. 2006. Groundwater modelling in selected area of north -westUttar Prades using fuzzy logic. Pantnagar Journal of Research 4(2) : 127-132.

[24] Salehi, F., Prasher, S.O., Amin, S., Madani, A., Jebelli, S.J., Ramaswamy, H.S., Tan, C., Drury, C.F., 2000. Prediction of annual nitrate-N losses in drain outflows with artificial neural networks. Trans. ASAE 43 (5), 1137–1143.

[25] Sarangi, A., Bhattacharya, A.K., 2005. Comparison of artificial neural network and regression models for sediment loss prediction from Banha watershed in India. Agric. Water Manage. 78, 195–208.

[26] Sharma, V., Negi, S.C., Rudra, R.P., Yang, S., 2003. Neural networks for predicting nitrate-nitrogen in drainage water. Agric. Water Manage. 63, 169–183.

[27] Shukla, M.B., Kok, R., Prasher, S.O., Clark, G., Lacroix, R., 1996. Use of artificial neural networks in transient drainage design. Trans. ASAE 39, 119–124.

[28] Singh, M., Bhattacharya, A.K., Nair, T.V.R., Singh, A.K.,2002. Nitrogen losses through subsurface drainage effluent in coastal rice fields from India. Agric. Water Manage. 52, 249–260.

[29] Singh, M., Bhattacharya, Singh, A.K., Singh, A., 2002b. Application of SALTMOD in coastal clay soils in India.Irrigation Drainage Syst. 16, 213–231.

[30] Sivakumar, B., Jayawardena, A.W., Fernando, T.M.K.G., 2002. River flow forecasting: use of phase-space reconstruction and artificial neural networks approaches. J. Hydrol. 265, 225–245.

[31] Smith, J. and Eli, R. N. 1995. Neural network modelloing of rainfall-runoff process. J Water Research Planning and Management 126: pp 499-508.

[32] Smith, M., Allen, R. G., Pereira, L.S. and Raes, D. 1998. Crop evapotranspiration. FAO Irrigation and Drainage Paper 56, FAO Rome, Italy.

[33] Sudheer, K.P., Gosain, A.K., Ramasastri, K.S., 2002. A data-driven algorithm for constructing artificial neural network rainfall-runoff models. Hydrol.Process. 16, 1325–1330.

[34] Tsoukalas, L.H., Uherg, R.E., 1996. Fuzzy and Neural Approaches in Engineering. John Wiley & Sons Inc., New York, pp. 196–267.

[35] Tyagi, N.K., Tyagi, K.C., Pillai, N.N., Willardson, L.S.,1993. Decision support for irrigation system improvement in saline environment. Agric. Water Manage. 23 (4), 285–301.

[36] Yang, C.C., Prasher, S.O., Lacroix, R., 1996. Applications of artificial neural networks to land drainage engineering. Trans. ASAE 39, 525–533.

[37] Yitian, L., Gu, R.R., 2003. Modeling flow and sediment transport in a river system using an artificial neural network. Environ. Manage. 31 (1), 122–134.

[38] Yu, C., Northcott, W.J., McIsaac, G.F., 2004. Development of an artificial neural network model for hydrologic and water quality modeling of agricultural watersheds. Trans. ASAE 47 (1), 285–290.

[39] Zhang, B., Govindaraju, R., 2003. Geomorphology-based artificial neural networks (GANNs) for estimation of direct runoff over watersheds. J. Hydrol. 273, 18–34.

[40] Zhang, G.P., 2000. Neural networks for classification: a survey. IEEE Trans. Syst. Man Cybernetics C: Appl. Rev. 30, 451–462.

[41] Zhang, Z.X., Kushwaha, R.L., 1999. Applications of neural networks to simulate soil–tool interaction and soil behavior. Can. J. Agric. Eng. 41, 119–125

AUTHORS

First Author – Pakhale G.K, Dept. of Hydraulics and Water Resources Engineering, College of Engineering and Technology, Dilla University, Ethiopia
Second Author – Nale J.P, SWCE, Dr. A.S. College of Agricultural Engineering, Rahuri, Maharashtra, India
Third Author – Temesgen W.B, Dept. of Hydraulics and Water Resources Engineering, College of Engineering and Technology, Dilla University, Ethiopia
Fourth Author – Muluneh W.D, Dept. of Civil Engineering, College of Engineering and Technology, Dilla University, Ethiopia