Water Resour Manage (2013) 27:871–883
DOI 10.1007/s11269-012-0220-0

Prediction of Ground Water Levels in the Uplands of a Tropical Coastal Riparian Wetland using Artificial Neural Networks

L. Karthikeyan · D. Nagesh Kumar · Didier Graillot · Shishir Gaur

Received: 3 June 2011 / Accepted: 19 November 2012 /
Published online: 14 December 2012

Abstract Artificial Neural Networks (ANNs) have been found to be a robust tool to model many non-linear hydrological processes. The present study aims at evaluating the performance of ANN in simulating and predicting ground water levels in the uplands of a tropical coastal riparian wetland. The study involves comparison of two network architectures, Feed Forward Neural Network (FFNN) and Recurrent Neural Network (RNN) trained under five algorithms namely Levenberg Marquardt algorithm, Resilient Back propagation algorithm, BFGS Quasi Newton algorithm, Scaled Conjugate Gradient algorithm, and Fletcher Reeves Conjugate Gradient algorithm by simulating the water levels in a well in the study area. The study is analyzed in two cases-one with four inputs to the networks and two with eight inputs to the networks. The two networks-five algorithms in both the cases are compared to determine the best performing combination that could simulate and predict the process satisfactorily. Ad Hoc (Trial and Error) method is followed in optimizing network structure in all cases. On the whole, it is noticed from the results that the Artificial Neural Networks have simulated and predicted the water levels in the well with fair accuracy. This is evident from low values of Normalized Root Mean Square Error and Relative Root Mean Square Error and high values of Nash-Sutcliffe Efficiency Index and Correlation Coefficient (which are taken as the performance measures to calibrate the networks) calculated after the analysis. On comparison of ground water levels predicted with those at the observation well, FFNN trained with Fletcher Reeves Conjugate Gradient algorithm taken four inputs has outperformed all other combinations.

Keywords Ground water levels · Rainfall · Stream flow · Artificial Neural Networks · Prediction, Algorithms

L. Karthikeyan · D. N. Kumar (✉)
Department of Civil Engineering, Indian Institute of Science, Bangalore 560012, India
e-mail: nagesh@civil.iisc.ernet.in

L. Karthikeyan
Formerly with Department of Civil Engineering, SASTRA University, Thanjavur, Tamilnadu 613401, India

D. Graillot · S. Gaur
UMR 5600, Department of Geosciences and Environment, Ecole Nationale Supérieure des Mines,
Saint Etienne 42023, France

1 Introduction

In nature, several hydrological processes occur to sustain the water cycle. Some of them include rainfall-runoff process, groundwater process, river systems etc. These processes need to be modeled making use of events that occurred in the past to predict what the nature has in the future and when to expect any calamities. Research has been undertaken to assess these complex hydro-meteorological processes using Artificial Neural Networks (ANNs).

ANN models are 'black box' abstract models with particular properties (structure, processing nature etc.) which are greatly suited to dynamic nonlinear system modeling. Complex problems such as nonlinear modeling, classification etc. are solved by ANN through its ability of identifying relationship from the patterns given to it (ASCE 2000). Since the past decade, ANNs have been successfully used to model dynamic systems in diverse fields of engineering including water resources. ANNs have been used for modeling rainfall runoff process (Mantoglou and Kourakos 2002) and for forecasting and generation of stream flow (Raman and Sunilkumar 1995; Nagesh Kumar et al. 2004; Edossa and Babel 2011). Cancelliere et al. (2002) derived operating policy for an irrigation supply reservoir using mixture of dynamic programming and ANN which yielded optimized policy that could properly simulate the real system at all time periods. ANN applications have also been observed in groundwater hydrology problems (Nayak et al. 2006; Ghose et al. 2010; Sreekanth and Datta 2011; Trichakis and Nikolos 2011; Gaur et al. 2012). Nikolos et al. (2008) used ANNs combined with Differential Evolution algorithm to optimize pumping strategy to meet the demand considering environmental constraints in Greece. Fu and Kapelan (2010) used ANN combined with Genetic Algorithms for multi objective optimal design of water distribution systems to improve computational efficiency. Sudheer and Jain (2004) worked on understanding the underlying methodology (number of neurons in hidden layer, distribution of their weights etc.) of neural networks by examining whether or not the physical processes in a watershed are inherent in a trained ANN rainfall-runoff model.

Although ANNs have been applied extensively in hydrology, very few applications have been observed in the field of wetland management with specific interest towards uplands of the wetlands which form the major source of water to the wetlands and hence any changes in their environment will have a direct effect on them. The present work aims to apply two ANN architectures with five search algorithms in modeling ground water levels in the uplands of coastal riparian wetland using hydro-meteorological data as inputs (with one case having one time step lagged inputs and the other having lagged inputs up to two time steps). A rigorous analysis had been done to obtain the optimum structure of network considering the performances in all the cases (described in later sections) studied. Data considered is initially analyzed for redundancy and modeled using ANNs. Optimal results are obtained using Ad Hoc procedure and best network is identified.

2 Study Area

Figure 1 shows the study area comprising of the uplands of the humid tropical 'Padre Wetland' (13° 00 ′00″ N to 13° 02′ 00″ N and 74° 47′ 30″ E to 74° 48′ 30″E), near the National Institute of Technology, Karnataka (NITK), Surathkal, Karnataka State, India and the location of observation well (O1) in the region. Part of Pavanje river flows through the study region. Ground water aquifer is relatively homogenous and ground water level fluctuations are observed to be uniform in different wells in the study region.

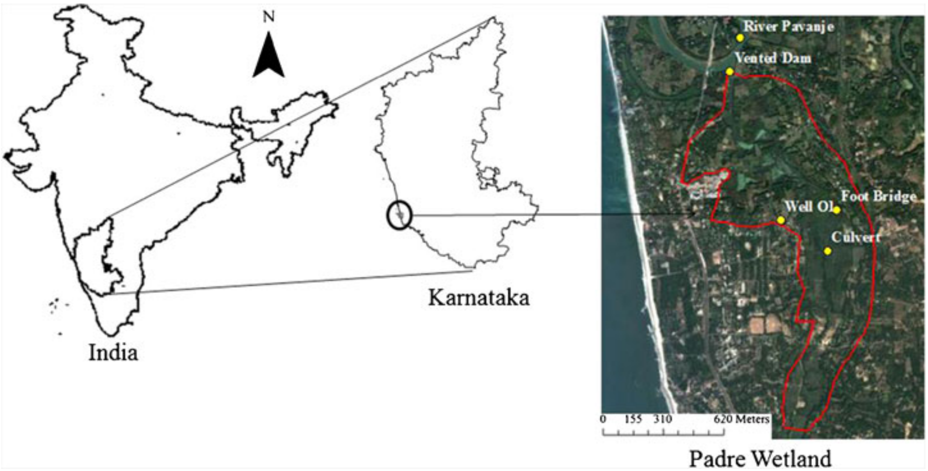

Fig. 1 Location of Padre Wetland and observation well (O1) in the region

3 Data Used and Data Analysis

3.1 Data Availability

The input hydrological parameters considered for the present study are weekly water level observations of well O1 and stream levels at two locations, one at the foot bridge and one at the culvert and the input meteorological parameters considered are the rainfall, average temperature and evaporation, measured at the nearest meteorological station located in Surathkal, India (Nyamathi 2008; Bharath 2009). These observations are converted from daily to weekly (averaging) level to maintain consistency between all the inputs on time scale.

3.2 Dependency Analysis

The study area has in total eight observation wells of which one of the wells is analyzed. Correlation analysis is done across well data to check how the water levels are varying with respect to well O1 (Table 1). It is observed that significant correlations exist mutually between the wells from which it can be said that conclusions obtained from one well might be applied to others and therefore can be generalized for the whole region.

Table 1 Correlation between the wells of study region

Corr	O1	O2	O3	O4	O5	O6	O7	O8
O1	1.000	0.976	0.867	0.865	0.652	0.972	0.932	0.859
O2	0.976	1.000	0.891	0.902	0.734	0.974	0.950	0.888
O3	0.867	0.891	1.000	0.894	0.737	0.866	0.923	0.873
O4	0.865	0.902	0.894	1.000	0.839	0.835	0.904	0.929
O5	0.652	0.734	0.737	0.839	1.000	0.641	0.752	0.844
O6	0.972	0.974	0.866	0.835	0.641	1.000	0.924	0.812
O7	0.932	0.950	0.923	0.904	0.752	0.924	1.000	0.912
O8	0.859	0.888	0.873	0.929	0.844	0.812	0.912	1.000

Four parameters rainfall, stream flow, evaporation and water level in the well were considered at weekly time scale for the study. Figure 2a–d shows the weekly data of observed rainfall, water levels at well O1, stream flow and evaporation rate.

In order to find out whether the inputs selected from correlation test have any association between themselves and with the output, Kendall tau rank correlation coefficient test is performed on the data. This test is used to measure the strength of relationship between two parameters. Specifically, it is a measure of rank correlation i.e., the similarity of the orderings of the data when ranked by each of the quantities.

3.3 Kendall Tau Test

Let (x_1, y_1), (x_2, y_2)… (x_n, y_n) be a set of joint observations from two random parameters X and Y respectively, such that all the values of (x_i) and (y_i) are unique. Any pair of observations (x_i, y_i) and (x_j, y_j) are said to be concordant if the ranks for both elements agree i.e., if both $x_i > x_j$ and $y_i > y_j$ or if both $x_i < x_j$ and $y_i < y_j$, and they are said to be discordant, if $x_i > x_j$ and $y_i < y_j$ or if $x_i < x_j$ and $y_i > y_j$. If $x_i = x_j$ or $y_i = y_j$, the pair is neither concordant, nor discordant.

The Kendall τ coefficient is defined as:

$$\tau = \frac{(n_c) - (n_d)}{\frac{1}{2}n(n-1)} \tag{1}$$

where, n_c is the number of concordant pairs, n_d is the number of discordant pairs and n is the total number of variables in a series. The denominator denotes the total number of possible pairs and hence the τ value always ranges between −1 and 1. If X and Y values agree in increasing order, the coefficient tends to 1 and if the agreement is in reverse order, coefficient tends to −1. The coefficient would be nearly 0 in case X and Y are independent. Tau coefficients calculated for the data are presented in the Table 2.

Proper understanding of the physical characteristics of the study area helps in the selection of the hydro-meteorological parameters and their lag time to be considered as

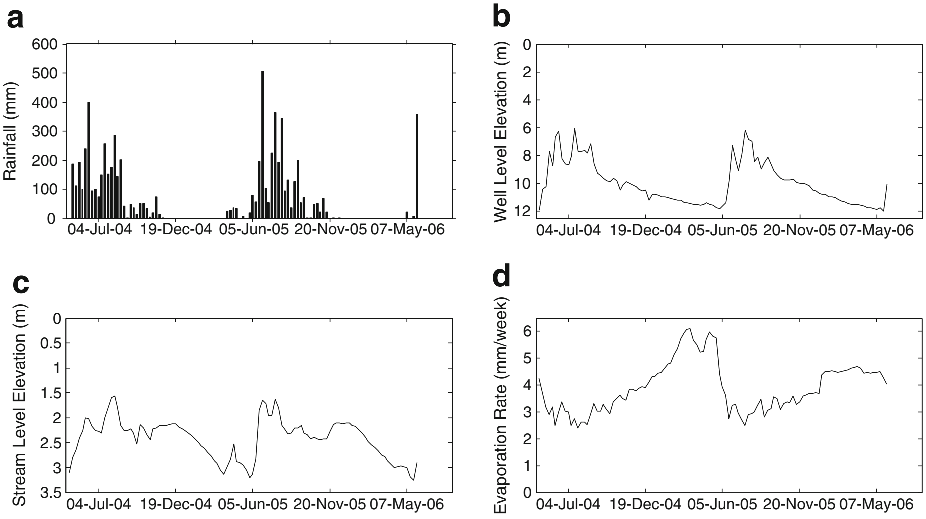

Fig. 2 Plots of (**a**) Rainfall (**b**) Ground water levels at O1 (**c**) Stream levels (**d**) Evaporation rate

Table 2 Kendall τ values between input and output parameters at well O1

τ	P_{t-1}	S_{t-1}	W_{t-1}	E_{t-1}	W_t
P_{t-1}	1.000	−0.334	−0.609	−0.611	−0.624
S_{t-1}	−0.334	1.000	0.613	0.494	0.545
W_{t-1}	−0.609	0.612	1.000	0.691	0.817
E_{t-1}	−0.610	0.493	0.691	1.000	0.755
W_t	−0.624	0.545	0.817	0.755	1.000

τ Kendall Coefficient; P_{t-1} Precipitation at time $t-1$; S_{t-1} Stream flow at time $t-1$; W_{t-1} Water level in well at time $t-1$; E_{t-1} Evaporation rate at time $t-1$; W_t Water level in well at time t

the inputs. Initially the inputs to the network involved only up to two time steps i.e., all the input parameters provided to network are lagged by $t-1$ and output parameter (well data) at t. This is because the study area is small and the soil being predominantly laterite, the recharge rate is high and hence the influence of hydrological parameters beyond the considered time step was initially assumed to be negligible. But, it is later found that high dependency exists between hydrological parameters lagged at $t-2$ and well data at t which made the initial assumption doubtful. So, in order to identify the extent up to which the lagged input parameters are affecting the predictions, the analysis is dealt in two cases-one with only two time steps at time t, $t-2$ and other with time steps up to a lag of two weeks (t, $t-1$, $t-2$). The input parameters for well O1 data applied to the network for the two cases are shown in Table 3.

The water levels at well O1 during time t, W (t) were considered as the targeted output parameters both during training and testing for afore mentioned cases.

3.4 Training and Testing Datasets

The dataset available is divided into two subsets-one for training and the other for testing. Due to non-availability of patterns, no separate validation could be set apart.

In the present study, two cases are analyzed-one with four inputs, and other with eight inputs. In both the cases, a total of 109 (previous) weekly observations were available from 2nd May 2004 to 28th May 2006. Data is provided in 70–30 ratio for training and testing sets respectively. The training set consisted of two wet seasons and one dry season and the testing set was predominantly a dry season data. The distribution of the data is as given below:

- Training Set: 2nd May 2004 to 16th October 2005–77 values.
- Testing Set: 23rd October 2005 to 28th May 2006–32 values.

Table 3 Input parameters to the network for the two cases

Well O1	Parameters (First Case)	Parameters (Second Case)
Rainfall	P($t-1$)	P($t-1$), P($t-2$)
Stream levels at Foot Bridge	S($t-1$)	S ($t-1$), S($t-2$)
Evaporation	E($t-1$)	E ($t-1$), E($t-2$)
Antecedent well Observations	W($t-1$)	W($t-1$), W($t-2$)

4 Methodology

Two network architectures namely Feed Forward Neural Network and Recurrent Neural Network have been used for comparative analysis.

In a feed forward neural network the network is organized in the form of layers. The information is passed on from the input layer to the output layer without any feedback mechanism. When a hidden layer having hidden neurons the function of which is to intervene between input and output is introduced, the network is called a multi layered FFNN.

A recurrent neural network (RNN) is a modification to FFNN in which a delay layer is added to the structure, which retains information between observations. At each time step, new inputs are fed into the RNN. The previous contents of the hidden layer are passed into the delay layer. These then feed back into the hidden layer in the next time step. Typical layout of FFNN and RNN can be seen in Nagesh Kumar et al. (2004).

4.1 Network Calibration

Ad hoc (trial and error) procedure is adopted to obtain the optimum structure of the network in which a rigorous analysis is carried out with one neuron initially and the number of neurons has been increased up to 40 with a step size of one in the hidden layer. As it is observed that the network's parameters were changed when it is run for multiple iterations keeping the number of neurons in hidden layer constant (which could be due to variable initial search point), the network is run for 50 iterations at each number of neurons in hidden layer (1–40). Initial weights are randomly assigned at the start of iterations.

In order for an ANN to generate an output vector that is as close as possible to the target vector, a training procedure is employed, the objective of which is to minimize mean square. Training is a process by which the connection weights of an ANN are adapted through a continuous process of stimulation by the environment in which the network is embedded (ASCE 2000). The training process involves active adjustment of the synaptic weights and the bias terms such that the primary function of minimizing the error function is achieved.

The training algorithms are optimization techniques that help in fulfilling the objective function. There are several algorithms for training neural network of which five have been considered for the present study. They are:

a) The Levenberg Marquardt Training Algorithm (LM)
b) The Resilient Back propagation Algorithm (RP)
c) The Scaled Conjugate Gradient Algorithm (SCG)
d) The BFGS quasi Newton Algorithm (BFGS)
e) The Fletcher -Reeves Conjugate Gradient Algorithm (CGF)

4.2 Performance Measures

In order to determine which network structure is optimal, the performance of a calibrated model is evaluated. ANN model performance is usually assessed using a quantitative error metric. The following performance measures were used to evaluate the efficiency of the network in all the cases.

a) Normalised Root Mean Square Error (*NRMSE*)
b) Relative Root Mean Square Error (*RRMSE*)

c) Nash-Sutcliffe Efficiency Index (E_f)
d) Correlation Coefficient (R)

a) Normalised Root Mean Square Error (*NRMSE*)

NRMSE is the Root Mean Square Error (*RMSE*) divided by the range of observed values:

$$NRMSE = \frac{RMSE}{x_{\max} - x_{\min}} = \frac{\sqrt{\frac{\sum_{i=1}^{N}(y_i - x_i)^2}{N}}}{x_{\max} - x_{\min}} \tag{2}$$

where, $x_{\max}$, $x_{\min}$ are maximum, minimum of observed values; y_i, x_i are actual and obtained values of output; N is the number of values.

In *NRMSE*, the *RMSE* obtained is normalized by the range of actual values to ensure that *RMSE* doesn't go beyond the range. If *NRMSE* value surpasses 1, it can be interpreted that the average root deviation (from respective observed) is greater than the range of values which though might occur in scenarios can be used to recommend the poor performance of the model. The definition of *NRMSE* need not pertain to only normalizing with range of observed values. It can be changed according to requirement.

b) Relative Root Mean Square Error (*RRMSE*)

Relative Root Mean Square Error is *RMSE* expressed in percentage.

$$RRMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - x_i}{y_i}\right)^2} * 100 \tag{3}$$

where, y_i and x_i are actual and obtained values of output, and N is the number of values. *RRMSE* would be lying between 0 and 100 which is the ratio between the deviation and actual values ideally close to zero. Incase *RRMSE* value goes beyond 100, it can be said that the simulated values have consistently under predicted the actual values indicating the poor performance of the model.

c) Nash-Sutcliffe Efficiency Index (E_f)

Traditionally, the correlation coefficient and standard error of estimate have been used to measure the goodness of fit of the model calibration. While the correlation coefficient is a useful goodness-of-fit index, it is theoretically applicable only to linear models that include an intercept.

The Nash-Sutcliffe Efficiency Index or coefficient of Efficiency provides an indication of how good a model is at predicting the values away from the mean. This criterion can provide some indication of how good the model will perform in either high or low magnitudes of the observed phenomenon. Closer the E_f value to 1, better the network fit. It is given by

$$E_f = 1 - \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4}$$

where, y_i and x_i are actual and obtained values of output, $\bar{y}$ is the mean of actual output values. Nash-Sutcliffe efficiencies can range from $-\infty$ to 1. E_f=1 corresponds to a perfect match of modeled discharge to the observed data. E_f=0 indicates that the model predictions are as accurate as the mean of the observed data, whereas E_f<0 occurs when the residual variance (described by the numerator of second term in Eq. 4), is larger than the data variance (described by the denominator of second term in Eq. 4). Essentially, the closer the model efficiency is to 1, the more accurate the model is.

d) Correlation Coefficient (R)

Correlation is widely used as a measure of the strength of linear dependence between two variables. The correlation coefficient (R) between the outputs and the targets are a measure of how well the variations in the target are explained by the outputs which is given by

$$R = \frac{\sum_{i=1}^{N}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \overline{y})^2}} \qquad (5)$$

where, y_i and x_i are actual and obtained values of output, $\overline{y}$ is the mean of actual output values. The correlation coefficient ranges from −1 to 1. A value of 1 implies that a linear equation describes the relationship in which Y increases as X increases and −1 implies that all data points lie on a line for which Y decreases as X increases. A value of 0 implies that there is no linear correlation between the variables.

5 Results

The methodology is applied to two kinds of inputs over two network architectures and five algorithms. The best network obtained in each case is screened based on optimum values of four performance measures simultaneously. The predictions of all the algorithms in four cases are presented in Fig. 3.

5.1 Comparison of Networks

From a comparison of the levels of accuracy of the best performing network architectures in modeling the water table fluctuations, it can be seen from Table 4 that feed forward neural networks with four inputs, five neurons in hidden layer and 23 iterations, using Fletcher-

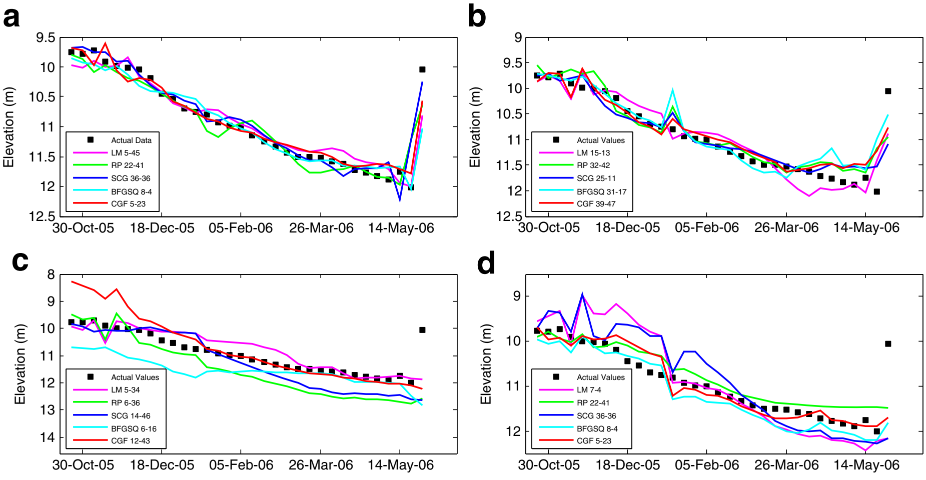

Fig. 3 Network performance for five algorithms (at optimized networks) in (**a**) FNN-4 Inputs (**b**) FNN-8 Inputs (**c**) RNN-4 inputs d) RNN-8 inputs

Table 4 Best networks and corresponding performance measures

Architecture	Inp.	Algo.	N	ITE	*NRMSE*	*RRMSE*	*Ef*	*R*
FNN	4	CGF	5	23	0.2335	1.4920	0.9538	0.9798
	8	CGF	39	47	0.4126	2.3398	0.8742	0.9452
RNN	4	CGF	12	43	0.3928	3.6418	0.7499	0.9260
	8	CGF	8	42	0.4885	2.8120	0.8251	0.9225

Inp Inputs; *Algo* Algorithm; *CGF* Fletcher-Reeves Conjugate Gradient algorithm; *N* Neurons in hidden layer; *ITE* Iterations

Reeves Conjugate Gradient algorithm has performed well over others. The performance of FFNN is observed to be better in both the cases over RNN which could be due to simpler architecture of the former network in which there is no scope of redundancy of input data that might have occurred with the latter case. Also, the training time for RNN in case of eight inputs is highest due to buildup of heavy architecture (in case of SCG algorithm, which optimally had 36 neurons in the hidden layer, RNN had to process 44 inputs (36+8) in total from second epoch on its search process) which finally produced poorer predictions compared to other scenarios. Figure 4 compares the simulated values obtained from the optimized situations under all the four afore mentioned cases with the actual well readings at well O1. Out of the four curves, the first curve i.e., FNN with four inputs is better in forecasting without any problem of over fitting. In case of RNN plots, it is observed that the predictions are under performed for at least 10 weeks of time period that signifies the poor modeling nature of RNN for the study region.

In order to check if preparation of training and testing datasets (on 70–30 basis respectively) with former set having one dry period and two wet periods and later having dominant dry period affected the performance of network, the entire data is shared between training and testing datasets with each of them having equal number of dry and wet periods on

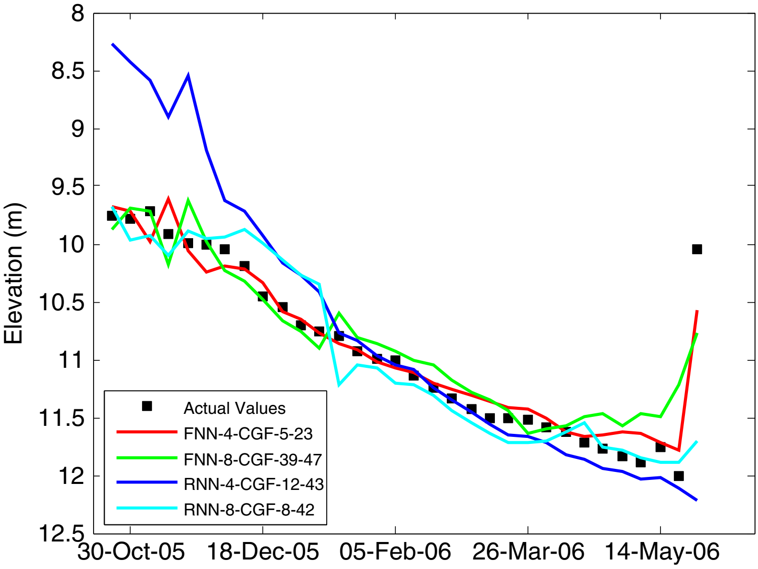

Fig. 4 Network performance of best networks obtained from all cases

hydrologic year basis (50–50 ratio). These datasets are applied to the best architecture obtained from the previously obtained results (FFNN- CGF- 4 Inputs- 5 Neuron Hidden Layer- 23 Iterations) and corresponding simulation and predictions are obtained. Figure 5 has box plots of observed ground water levels for total data and the levels obtained during training and testing applied over two sets of data with optimum architecture obtained from previous analysis. Supporting these plots, Fig. 6 shows box plots of errors obtained under the same scenarios. The high variance created by the presence of both dry and wet periods in training data (70–30 ratio) is almost reproduced by the network (Fig. 5). When prediction errors are considered, median is concentrated near zero and extremes are varied up to a range of 2 m which can be attributed to learning characteristics of the network. Similar results have been obtained during training of same network with data of 50–50 ratio which might be due to the presence of dry and wet periods in both the datasets. In the case of testing, 70–30 ratio data has predominant dry period limiting the variance which is reproduced sparing the exact prediction of water table deficit during this period which is important in wetland scenario. Most of the prediction errors are concentrated around zero (Fig. 6) which supports the efficiency of the optimum architecture obtained. 50–50 data during testing also has high variance (due to presence of dry and wet periods). Box plot is prepared to check if the network is able to predict these variations and corresponding prediction errors (Figs. 5 and 6). It is observed that the network could not successfully predict the values in both ways which may be attributed to insufficient number training datasets.

6 Conclusions

The study carried out proves the efficiency of Artificial Neural Networks in modeling weekly water table fluctuations. The selection of parameters to be considered as inputs for

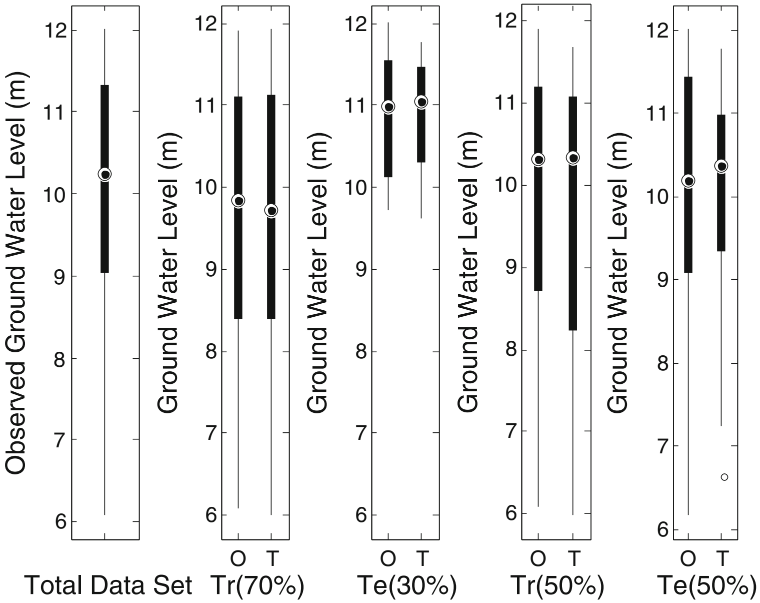

Fig. 5 Box plots of ground water levels for (**a**) Total data (**b**) Training (70 %) (**c**) Testing (30 %) (**d**) Training (50 %) (**e**) Testing (50 %) (*O* Observed; *T* Tested; *Tr* Training; *Te* Testing)

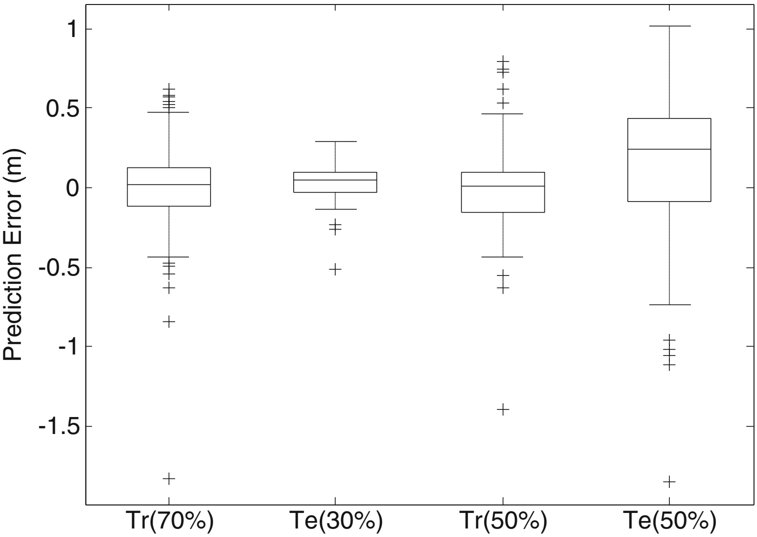

Fig. 6 Box Plots of Prediction Error during (**a**) Training (70 %) (**b**) Testing (30 %) (**c**) Training (50 %) (**d**) Testing (50 %) (*Tr* Training; *Te* Testing)

the network is to be done properly. The network inputs should be relevant to the physical independent influence of each parameter over the output. This can be measured in different ways but for the present work only correlation analysis is carried out.

The following conclusions can be drawn in two perspectives.

1. Neural Network architectures are problem specific and data dependent.
2. (a) If a proper physical relationship doesn't exist or an empirical model could not properly represent a particular hydrologic process, neural networks can give a good solution in simulating the patterns and forecasting the results.
   (b) If any reliable relationship is available, it can be compared with ANNs and attempts can be made to discuss the processes running inside the neural network to extend the usage of neural networks for similar predictions.

Following conclusions can be made from the results obtained for the study area.

Results of current study contradict the literature (Anmala et al. 2000; Nagesh Kumar et al. 2004; Dogan et al. 2007) that suggests the better performance of recurrent neural networks when compared with standard feed forward neural network. This reinforces the fact that neural networks are completely problem specific and data dependent and the results of which cannot be generalized for all the cases.

FNN with four inputs is the only case which nearly predicted the sudden depletion in water level that has occurred during last time step of the data (May, dry season) which indicates that this kind of architecture can represent better the fluctuations present in data (rainfall etc.).

Input parameters considered for network are selected based on correlation analysis. The reason for exclusion of temperature series and not evapotranspiration series corresponds to the range of both the series. In case of evapotranspiration, the range of values are more or less similar to the other parameters which gave an idea that the relationship established by network with this parameter will be smoother when compared to relationship made through considering temperature as parameter that has totally different range which might make the network unstable.

Input parameters are fed to network aiming to check the extent to which the hydro meteorological parameters are affecting the well data predictions. In this process, it is concluded that these parameters have effect only up to lag of 1 week. But, due to the manner in which inputs are prepared, it is not possible to estimate which of the parameters carry most weightage in monitoring the inputs. Better predictions can be obtained if input parameters are prepared taking only minimum number of parameters (may be four) which are selected based on highest dependency with well observations.

The better performance of FFNN over RNN indicates that proper balance has to be maintained between input data and the architecture. In case of relationship that involves too many input parameters, usage of simple networks like FFNN might be sufficient to achieve reasonable predictions. RNN can be used in case of non-availability of datasets where network needs more data to establish stronger regression between parameters.

Number of neurons in hidden layer influences computation time, learning rate, accuracy etc. The study concluded usage of 5 neurons in hidden layer in FFNN architecture through Ad Hoc procedure which is tested for the range of 1 to 40. It is concluded from the study that in case of FFNN, optimum number of neurons' range can be fixed in and around the number of inputs being fed to network that each neuron dominantly influences a particular input in the network. This can be supported by the kind of results obtained by Sudheer and Jain (2004).

Usage of different error measures is found out to be useful as it induced rigor in selection of models. Also, in order to check if there is any kind of dependency, failure of correlation analysis must be succeeded by tests that check non-linear dependency between those parameters. Integrating results from both tests will show the existence of any kind of dependency.

From the box plots presented, it can be concluded that during training, network tries to give higher weightage to extreme values and it tends to learn from them with greater accuracy producing almost same variance as that of observed values in both the cases (70–30 & 50–50). As a trade-off for this, the prediction errors are obtained with a range of 2 m although median is lying around zero. In case of testing, the network in both cases produced similar range of values although it failed to predict the extremes of data. But, error-wise, the network could predict 30 % of the data accurately which can be supported by the presence of monotonic dry period values. Whereas, testing with 50 % of data in strict sense can be said as unsatisfactory in terms of prediction error from which it can be concluded that lesser the number of training data sets poorer will be the final predictions.

It can be concluded that FFNN is the optimum network both in terms of performance and computational expense for prediction of water levels in uplands of wetlands considering proper inputs for the network. In both the cases analyzed (4, 8 inputs), FFNN and RNN gave respective best predictions with CGF algorithm which can be concluded as optimum search algorithm for this kind of region.

References

Anmala J, Zhang B, Govindaraju S (2000) Comparison of ANNs and empirical approaches for predicting watershed runoff. J Water Res Plann Manag ASCE 126(3):156–166

ASCE Task Committee (2000) Artificial neural networks in hydrology, part I. J Hydrolog Eng 5(2):115–137

Bharath R (2009) Simulation of Water Levels in open Wells at Uplands of a Wetland Using Artificial Neural Networks, M.E thesis, Bangalore Univ, India

Cancelliere A, Giuliano G, Ancarani A, Rossi G (2002) A neural networks approach for deriving irrigation reservoir operating rules. Water Res Manag 16:71–88

Dogan D, Isik S, Toluk T, Sandalci M (2007) Daily streamflow forecasting using artificial neural networks, Int Cong on River Basin Mgmt, River Basin Flood Mgmt, Ch IV, 448–459

Edossa DC, Babel MS (2011) Application of ANN-based streamflow forecasting model for agricultural water management in the Awash river basin, Ethiopia. Water Res Manag 25:1759–1773

Fu G, Kapelan Z (2010) Embedding Neural Networks in Multi Objective Genetic Algorithms for Water Distribution System Design, Wat Dist Syst Ana, ASCE, 888–898

Gaur S, Sudheer Ch, Graillot D, Chahar B R and Nagesh Kumar D (2012) Application of Artificial Neural Networks and Particle Swarm Optimization for the Management of Groundwater Resources, Water Res Mgmt, in print

Ghose DK, Panda SS, Swain PC (2010) Prediction of water table depth in western region, Orissa using BPNN and RBFN neural networks. J Hydrol 394:296–304

Mantoglou A, Kourakos G (2002) Modeling the rainfall-runoff process using neural networks with emphasis on optimization of network configuration using genetic algorithms, Proc. of Inter. Conf., W. Res. Mgmt. in the ERA of Transition, Euro Wat Res Ass, Ath, 229–242

Nagesh Kumar D, Srinivasa Raju K, Sathish T (2004) River flow forecasting using recurrent neural networks. Water Resour Manag 18:143–161

Nayak PC, Rao YRS, Sudheer KP (2006) Groundwater level forecasting in a shallow aquifer using artificial neural network approach. Water Res Manag 20:77–90

Nikolos IK, Stergiadi M, Papadopoulou MP, Karatzas GP (2008) Artificial neural networks an alternative approach to groundwater numerical modeling and environmental design. Hydrol Proc 22(17):3337–3348

Nyamathi SJ (2008) Characterizing Hydrological Responses of Coastal Humid Tropical Wetland, Ph.D thesis, NITK Surathkal, India

Raman H, Sunilkumar N (1995) Multivariate modelling of water resources time series using artificial neural networks. J Hyd Sci 40(2):145–163

Sreekanth J, Datta B (2011) Comparative evaluation of genetic programming and neural network as potential surrogate models for coastal aquifer management. Water Res Manag 25:3201–3218

Sudheer KP, Jain A (2004) Explaining the internal behaviour of artificial neural network river flow models. Hydrol Proc 18:833–844

Trichakis IC, Nikolos IK (2011) Artificial Neural Network (ANN) based modeling for Karstic groundwater level simulation. Water Res Manag 25:1143–1152