



ISSN (E): 2277- 7695
ISSN (P): 2349-8242
NAAS Rating: 5.23
TPI 2021; 10(4): 386-391
© 2021 TPI
www.thepharmajournal.com
Received: 04-02-2021
Accepted: 09-03-2021

PB Hittanagi
M. Tech. Scholar, SWCE,
CAET, Dr. BSKKV, Dapoli,
Maharashtra, India

HN Bhangre
Assistant Professor, Department
of Computer Science, Dr.
BSKKV, Dapoli, Maharashtra,
India

BL Ayare
Professor and Head, SWCE, Dr.
BSKKV, Dapoli, Maharashtra,
India

MS Mane
Head, Inter faculty, MPKV,
Rahuri, Maharashtra, India

Forecasting of water table fluctuation for Priyadarshini watershed using Ann with different algorithms

PB Hittanagi, HN Bhangre, BL Ayare and MS Mane

DOI: <https://doi.org/10.22271/tpi.2021.v10.i4f.5962>

Abstract

Groundwater is an important natural resource essential for sustenance of life. Over 98% of the freshwater on the Earth lies below its surface. The ANN technique is applied as a new approach and an attractive tool to study and predict groundwater levels without applying physically based hydrologic parameters. It is observed that the maximum LM value of R for training and validation are 0.908 and 0.903 shown in well 2 (2-9-1) and testing is 0.949 shown in well 8 (2-9-1), whereas minimum value for training is 0.684 shown in well 1 (4-4-1) for validation and testing are 0.159 and 0.773 well 9 (3-5-1) The maximum observed R value for CG training, validation and testing are 0.76 shown in well 3 (4-5-1), 0.85 shown in well 9(3-5-1) and 0.891 shown in well 7 (2-5-1) whereas the minimum value for training and testing are 0.671 and 0.458 shown in well 1 (4-4-1) for validation the minimum R value is 0.638 well 2 (4-7-1). Considering training, validation and testing period and all the statistics it is difficult to say which algorithm is better among the two selected for study. Because there was a lot of variation in all the statistics among the two selected algorithms for training, validation and testing period. But considering the testing period of all the nine wells it was found that LM algorithm was better than CG for wells i.e., well 1 (2-9-1), well 2 (2-9-1), well 3 (1-8-1), well 4 (1-6-1), well 5 (2-9-1), well 6 (1-9-1), well 8 (2-9-1) while CG algorithm was better than LM for wells i.e., well 7 (2-5-1) and well 9 (3-5-1) So these algorithms for particular well were selected for sensitivity analysis.

Keywords: ANN, forecasting water level Priyadarshini watershed

Introduction

Groundwater is an important natural resource essential for sustenance of life. Over 98% of the freshwater on the Earth lies below its surface. It is located below the soil surface and largely contained in interstices of bedrocks, sands, gravels, and other interspaces through which precipitation infiltrates and percolates into the underground aquifers due to gravity. (Wagh *et al.*, 2014) [8].

In general, in major part of the Ratnagiri district, rise in water level in the range of 0.05 m (at Sakarpa, Taluka-Sangmeshwar) to 7.22 m (at Jaigarh, Taluka-Ratnagiri) is recorded between pre-monsoon and post-monsoon season of the year-2011, (Anonymous, 2017) [2].

The basic concept of an artificial neural network (ANN) is derived from an analogy with the biological nervous system of the human brain and how the latter processes information through its millions of neurons interconnected to each other by synapses. Borrowing this analogy, an ANN is a massively parallel system composed of many processing elements (neurons), where the synapses are actually variable weights, specifying the connections between individual neurons and which are adjusted.

The ANN technique is applied as a new approach and an attractive tool to study and predict groundwater levels without applying physically based hydrologic parameters. The approach may improve the understanding of complex groundwater system and is able to show the effects of hydrologic, meteorological and anthropic impacts on the groundwater conditions. (Sirhan and Koch, 2013) [6].

Groundwater is one of the major sources of supply for domestic, industrial and agricultural purposes. To gain insight in the processes including the groundwater system, one needs knowledge about the essential variables and how they fluctuate over time. Forecasting the ground water level fluctuations is an important requirement for planning conjunctive use in any basin.

Corresponding Author:
PB Hittanagi
M. Tech. Scholar, SWCE,
CAET, Dr. BSKKV, Dapoli,
Maharashtra, India

Materials and Methods

The research work has been carried out at the Priyadarshini watershed, College of Agricultural Engineering and Technology, Dr. Balasaheb Sawant Konkan Krishi Vidyapeeth, Dapoli, Dist.- Ratnagiri (M.S.). The Priyadarshini Watershed is located at 17.1° N latitude, 73.26° E longitudes and 250 m above mean sea level. The region comes under heavy rainfall with average annual rainfall of 3500 mm. Priyadarshini watershed has 38.72 ha area. The ambient temperature of the region varies from 7.5°C to 38.5°C and relative humidity varies from 55 percent to 99 percent in different seasons. The climate of the region is hot and humid. The region has hilly topography with lateritic soils.

Artificial neural network is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, the network is composed of a large number of highly interconnected processing elements called as neuron. They typically consist of hundreds of simple processing units which are wired together in a complex communication network. Each unit or node is a simplified model of real neuron which sends off a new signal or fires if it receives a sufficiently strong Input signal from the other nodes to which it is connected. Learning in this system involves the adjustment between neurons through synaptic connection. (Maind and Wankar, 2014) [3]. In this study feed-forward neural networks architecture has been used in predicting weekly water table depths. The weekly data of 9

years (2005-2014) related to Rainfall, Temperature, Solar, Well depth and Permeability data of Priyadarshini Watershed was used. Neuro Solutions Software was used for analysis.

ANN Architecture: In this study, four input parameters have been used as input, rainfall data, permeability data, solar data, temperature and one output parameter as well depth. The ANN architecture consists of three layers namely Input, Hidden and output layer, shown in Fig 1.

Input Nodes – neurons interfaces to the real world to receive its inputs as “Input Layer. The layer of input neurons receives the data either from input files or directly from electronic sensors in real-time applications they just pass on the information to hidden nodes. (Maind and Wankar, 2014) [3].

Hidden Nodes -hidden layer receives the signals from all of the neurons in a layer above it, typically an input layer. After a neuron performs its function it passes its output to all of the neurons in the layer below it (Maind and Wankar, 2014) [3]. To calculate number of hidden layers to be used we use $(2n+1)$. Where n = no. of nodes.

Output Nodes – neurons provide the real world with the network's outputs. Output nodes are collectively referred to as “Output Layer” and are responsible for computations and transferring information from the network to outside world. In this study, the groundwater level will be estimated. (Maind and Wankar, 2014) [3]

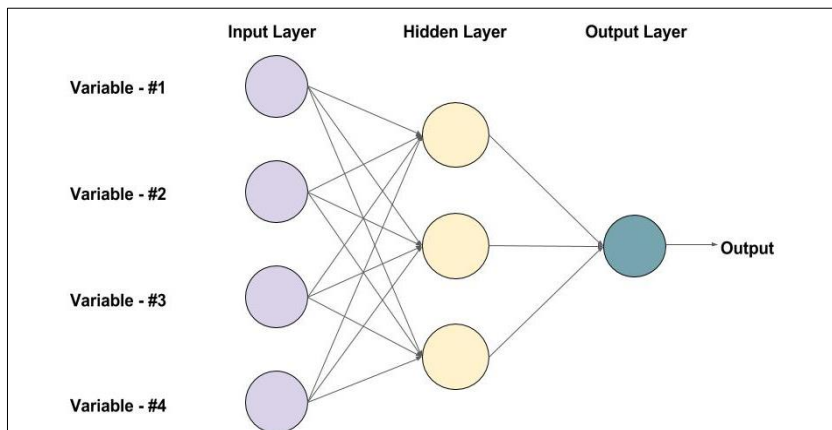


Fig 1: Artificial Neural Network

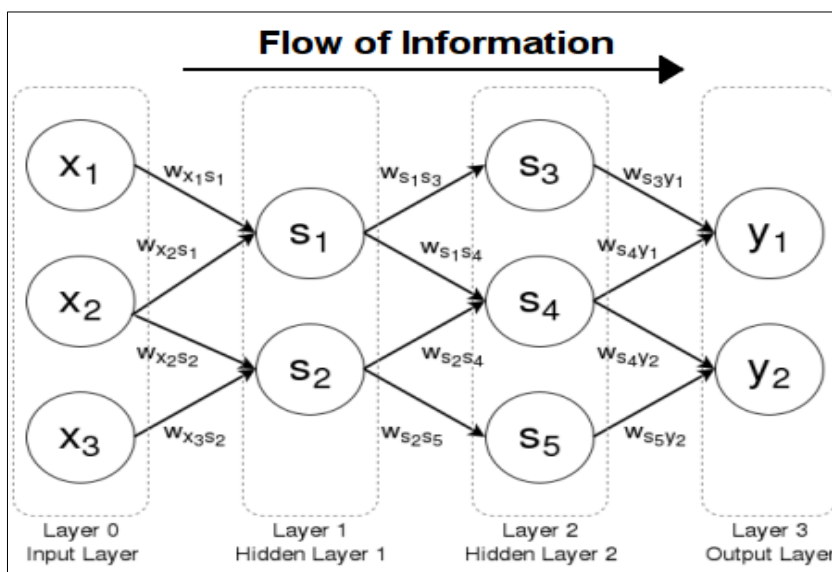


Fig 2: Flow of data in a Feed Forward Network

Feed-forward neural network (FNN): Feed-forward neural networks have been applied successfully in many different problems since advent of error back propagation learning algorithm. This network architecture and the corresponding learning algorithm can be viewed as a generalization of popular least-mean-square (LMS) algorithm. In feed-forward networks, data flow through network in one direction from input layer to output layer through hidden layer(s). Each output value is based solely on current set of inputs. In most networks, nodes of one layer are fully connected to the nodes in the next layer; however, this is not a requirement of feed-forward networks. A multilayer perceptron network consists of an input layer, one or more hidden layers of computation nodes, and an output layer. Fig. 2 shows a typical feed-forward network with two hidden layer, three input neurons and two output. Input signal propagates through the network in a forward direction, layer by layer. Key disadvantages are that it trains slowly, and require lots of training data.

Building of neural networks: For developing ANN model generally data sets are required for the training, validation and testing of the ANN networks. In this study, observed rainfall data, infiltration data, Water level, Permeability data, Temperature data and Solar data have been used to train and validate an artificial neural-network. Levenberg–Marquardt

(LM), Conjugate Gradient Algorithm (CG) used as the learning algorithm. The Neural Network will be optimized using Neuro Solutions. In the training stage, to define the output accurately, the number of nodes will be increased step-by-step in the hidden layer. The software normalizes the given data. Neurons in the input layer have no transfer function. Logistic sigmoid (logsig) transfer function will be used in hidden and output layer. After the successful training of the network, the network will be tested with the test data. Using the results produced by the network, statistical methods will be used to make comparisons.

Transfer function: The output activation function for binary classification problems (i.e. outputs values that range (0,1) Is the logistic sigmoid. The logistic sigmoid has the following form:

$$f(x) = \frac{1}{1+e^{-x}} \quad \dots 1$$

and outputs values that range (0,1). The logistic sigmoid is motivated somewhat by biological neurons and can be interpreted as the probability of an artificial neuron “firing” given its inputs (Fig. 3).

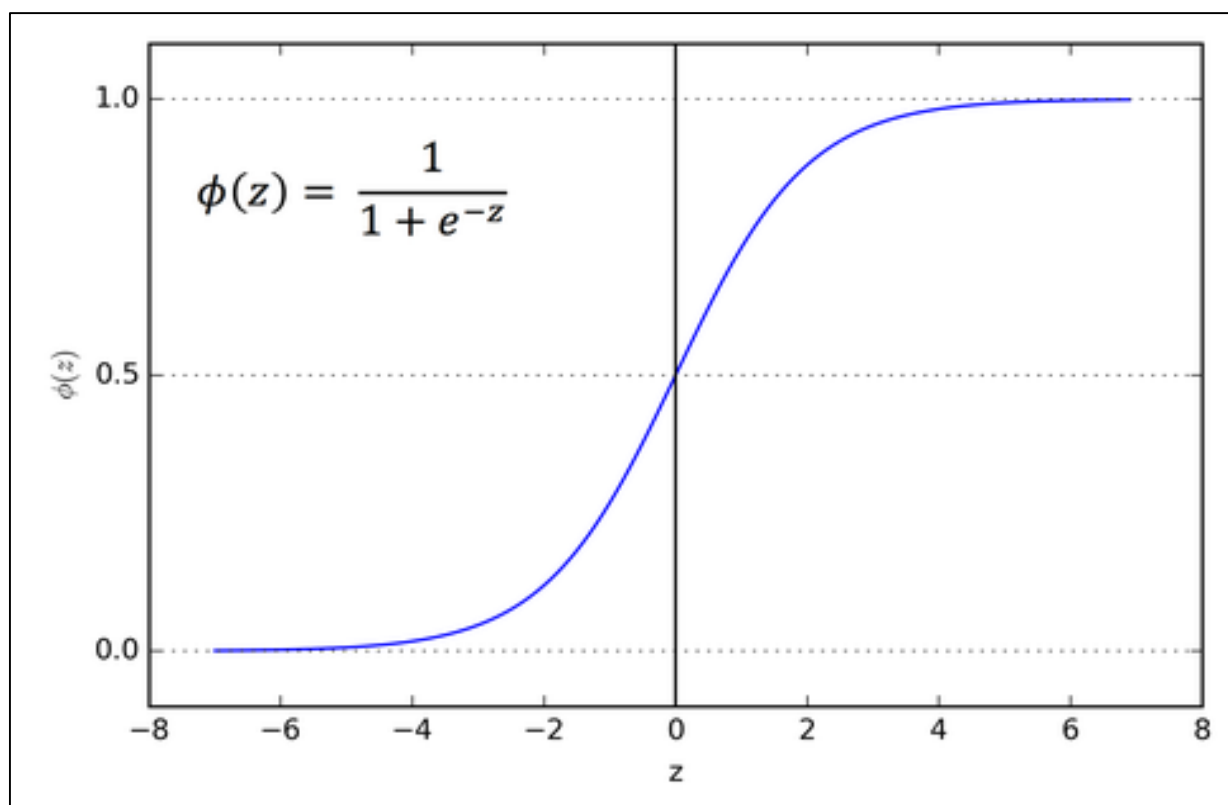


Fig 3: The sigmoid curve is in “S” shape

Learning Algorithm Supervised Learning

In supervised training, both the inputs and the outputs are provided. The network then processes the inputs and compares its resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights which control the network. This process occurs over and over as the weights are continually tweaked. The set of data which enables the training is called the "training set." During the training of a

network the same set of data is processed many times as the connection weights are ever refined. The current commercial network development packages provide tools to monitor how well an artificial neural network is converging on the ability to predict the right answer. These tools allow the training process to go on for days, stopping only when the system reaches some statistically desired point, or accuracy. When finally, the system has been correctly trained, and no further learning is needed, the weights can, if desired, be "frozen." (Maind and Wankar, 2014) [3].

Training with different algorithms

Determining the best values of all the weights is called training the ANN. In a supervised learning mode, actual output of a neural network is compared to predicted output. Weights, which are usually randomly set to begin with, are then adjusted so that next result will produce less variation between predicted and actual output. Training consists of presenting input and output data to network and allowing to run for certain epochs. These data are training data. For each input provided to the network, the corresponding predicted output set is given as well as processed through 5000 epochs. It is considered complete when the artificial neural network reaches a desired performance level. At this level the network has achieved the desired statistical accuracy as it produces required outputs for a given sequence of inputs. When further learning is found to be unnecessary, resulting weights are typically fixed for the application. Once a supervised network performs well on the training data, it is important to see what it can do with a new set of data. If a system does not give desired output for this test set, then training period should continue. Testing is important to ensure that network has learned the basic patterns involved in application and has not memorized all the data. Two different algorithms are being used in this study in order to identify the one which trains a given network more efficiently.

Conjugate gradient algorithm (CG)

This is the direction in which the performance function is decreasing most rapidly. It turns out that, although the function decreases most rapidly along the negative of the gradient, this does not necessarily produce the fastest convergence

$$\beta_k = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \quad 2$$

Levenberg-Marquardt (LM)

Levenberg-Marquardt algorithm was designed to approach second-order training speed without having to compute the Hessian matrix. When performance function has form of a sum of squares (as is typical in training feed forward networks), then the Hessian matrix can be approximated as

$$H = J^T J \quad 3$$

and gradient can be computed as

$$g = J^T e \quad 4$$

where, J is Jacobian matrix that contains first derivatives of network errors with respect to weights and biases, and e is a vector of network errors. Jacobian matrix can be computed through a standard back propagation technique that is much less complex than computing the Hessian matrix.

Levenberg-Marquardt algorithm uses this approximation to the Hessian matrix in the following Newton-like update:

$$x_{k+1} = x_k - [J^T J + \mu I]^{-1} J^T e \quad 5$$

When scalar μ is zero, this is just Newton's method, using the approximate Hessian matrix. When μ is large, this becomes gradient descent with a small step size. Newton's method is faster and more accurate near an error minimum, so aim is to

shift towards Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step would increase performance function. In this way, performance function will always be reduced at each iteration of the algorithm.

Results and Discussion

Comparison of algorithms

Nash-Sutcliffe coefficient (E) (Nash and Sutcliffe, 1970) [4], root mean square error (RMSE), mean absolute error (MAE), and Pearson coefficient (R) were used to assess the models response to that of observed value for different algorithms for developed ANN models during training, validation and testing period and presented in table 1. It is observed that the maximum LM value of R for training and validation are 0.908 and 0.903 shown in well 2 (2-9-1) and testing is 0.949 shown in well 8 (2-9-1), whereas minimum value for training is 0.684 shown in well 1 (4-4-1) for validation and testing are 0.159 and 0.773 well 9 (3-5-1) as presented in the table 1.

The maximum observed R value for CG training, validation and testing are 0.76 shown in well 3(4-5-1), 0.85 shown in well 9 (3-5-1) and 0.891 shown in well 7 (2-5-1) whereas the minimum value for training and testing are 0.671 and 0.458 shown in well 1 (4-4-1) for validation the minimum R value is 0.638 well 2 (4-7-1).

It is observed that the Pearson coefficient (R) indicates the strength and direction of linear relationship between two variable the correlation is +1 in case of perfect increasing linear relationship and -1 in case of decreasing linear relationship a correlation coefficient of 0 means there is no linear relationship between the variables minimum value is (0.159) during validation period of well 9(2-9-1) for the LM algorithm and was maximum value is (0.949) during testing period of well 8 (2-9-1) for the LM algorithms.

The variation of root mean square error (RMSE) statistics, a measure of residual variance which illustrates the results between the computed and observed water table depths, was minimum (0.050) during training period of well 2 (2-9-1) for LM algorithm and was maximum (0.303) during validation period of well 1(2-9-1) for the LM algorithm.

The mean absolute error (MAE) was found to be minimum (0.005) during validation period of well 8(2-8-1) for CG algorithm and was maximum (0.216) during validation period of well 6 (4-4-1) for LM algorithm.

The coefficient of efficiency (E) was found to be varying from -506.05 (during validation period of well 1 (2-9-1) for LM algorithm) to 0.896 (during testing period of well 1 (2-9-1) for LM algorithms). Fig.4 shows observed and predicted weekly water table depths of all the nine wells for different algorithms during training, validation and testing period. It was observed that the predicted water table depths followed the observed water table pattern.

Considering training and testing period and all the statistics it is difficult to say which algorithm is better among the two selected for study. Because there was a lot of variation in all the statistics among the two selected algorithms for training, validation and testing period. But considering the testing period of all the nine wells it was found that LM algorithm was better than CG for wells i.e., well 1 (2-9-1), well 2 (2-9-1), well 3 (1-8-1), well 4 (1-6-1), well 5 (2-9-1), well 6 (1-9-1), well 8 (2-9-1) while CG algorithm was better than LM for wells i.e., well 7 (2-5-1) and well 9 (3-5-1) So these algorithms for particular well were selected for sensitivity analysis

Table 1: Statistics of LM and CG algorithms for developed ANN models

Well no	Model steps	R		RMSE		E		MAE	
		LM	CG	LM	CG	LM	CG	LM	CG
1	Training	0.684	0.671	0.146	0.084	-66.98	-6.450	0.011	0.007
	Validation	0.369	0.664	0.303	0.084	-506.05	-12.450	0.060	0.006
	Testing	0.905	0.458	0.058	0.188	0.896	-201.44	0.011	0.014
2	Training	0.908	0.713	0.050	0.081	-5.660	-13.610	0.004	0.006
	Validation	0.903	0.638	0.123	0.081	-45.007	-2.722	0.024	0.006
	Testing	0.943	0.695	0.067	0.084	-24.252	-1.827	0.012	0.006
3	Training	0.855	0.766	0.071	0.101	-7.542	-17.912	0.006	0.009
	Validation	0.901	0.773	0.117	0.108	-1.598	-11.151	0.023	0.008
	Testing	0.901	0.757	0.101	0.066	-4.390	-1.55	0.019	0.005
4	Training	0.764	0.680	0.0991	0.113	-0.332	-3.752	0.009	0.008
	Validation	0.868	0.749	0.263	0.152	-37.447	-4.230	0.052	0.011
	Testing	0.893	0.800	0.133	0.122	0.361	-1.615	0.025	0.009
5	Training	0.829	0.704	0.099	0.104	-14.548	-7.452	0.009	0.009
	Validation	0.805	0.708	0.177	0.083	-7.663	-67.926	0.0355	0.006
	Testing	0.865	0.776	0.118	0.098	-571.59	-0.154	0.022	0.007
6	Training	0.854	0.681	0.080	0.133	-60.563	-29.970	0.007	0.012
	Validation	0.889	0.722	0.108	0.167	0.194	-1.990	0.216	0.012
	Testing	0.944	0.857	0.107	0.183	-3.157	-1.043	0.020	0.014
7	Training	0.715	0.693	0.149	0.169	-51.661	-61.645	0.013	0.015
	Validation	0.638	0.826	0.202	0.123	-45.523	-0.154	0.040	0.009
	Testing	0.880	0.891	0.220	0.084	-8.186	0.655	0.042	0.006
8	Training	0.884	0.756	0.067	0.096	-166.52	-7.742	0.006	0.008
	Validation	0.870	0.775	0.132	0.066	-21.706	0.169	0.026	0.005
	Testing	0.949	0.535	0.087	0.079	-0.368	0.354	0.016	0.006
9	Training	0.738	0.763	0.086	0.089	-61.015	-14.5	0.007	0.008
	Validation	0.159	0.855	0.141	0.132	-1.176	0.821	0.028	0.026
	Testing	0.773	0.864	0.090	0.049	-3.211	0.830	0.017	0.003

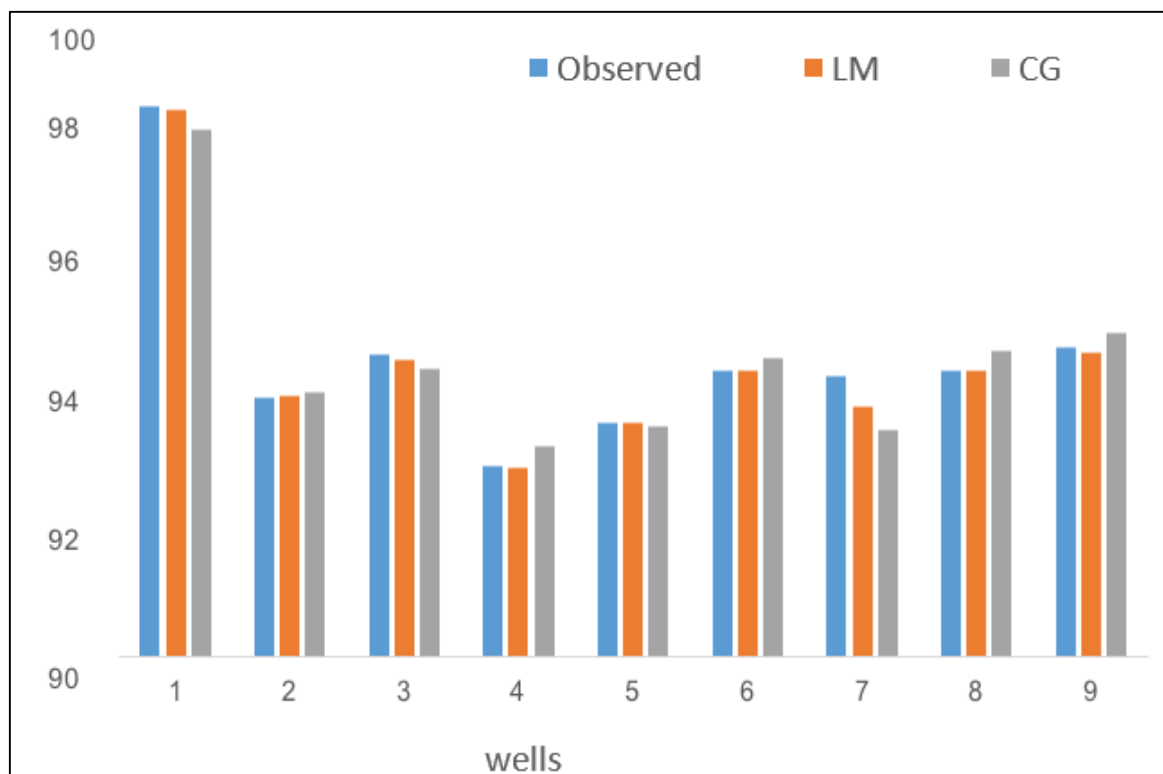


Fig 4: Observed and predicted weekly water table depth of average value of LM and CG algorithm

Similar study was carried out by Sujatha and Kumar (2005) [7] suggested that feed forward network trained with training algorithm Levenberg-Marquardt showed best performance in predicting the ground water levels with data of relatively shorter period. Similarly, Al-Aboodi *et al.* (2016) [1] reported that LM is best ANN structure for predicting ground water flow.

As the results found were based on trial and error methods Levenberg- Marquardt (LM) algorithm provides better results than Conjugate Gradient algorithm as shown in the Fig.4. Levenberg- Marquardt (LM) best results for ANN network architecture of best model for well 1(2-9-1), well 2(2-9-1), well 3(1-8-1), well 4 (1-6-1), well 5(2-9-1), well 6(1-9-1), well 7(3-5-1), well 8 (2-9-1), well 9(2-9-1).

Conclusions

Based on building of ANN models for predicting groundwater levels for 9 wells in the Priyadarshini watershed, it was found that;

1. Levenberg- Marquardt (LM) best results for ANN network architecture of best model for well
2. The best structure of ANN model for predicting groundwater flow in the study area is of three layers feed-forward network type. This ANN model is developed with Logistic sigmoid transfer function.

References

1. Al-Aboodi AH, Kifah M, Khudhair Ali S, Al-Aidani. Prediction of Groundwater Level in Safwan-Zubair Area Using Artificial Neural Networks. Basrah. Journal for Engineering Sciences 2016;16(1):42-50.
2. Anonymous. Government of India Ministry of Water Resource River Development And Ganga Rejuvenation CGWB, Ground Water Year Book of Maharashtra and Union Territory of Dadar And Nagar Haveli 2017.
3. Maind SB, Wankar P. Research Paper on Basic of Artificial Neural Network. International Journal on Recent and Innovation Trends in Computing and Communication. 2014;2(1):96-100.
4. Nash JE, Sutcliffe JV. River Flow Forecasting Through Conceptual Models Part I- A Discussion Of Principles. Journal Of Hydrology 1970;10:282-290.
5. Sahoo S, Russo TA, Elliott J, Foster I. Water Resources Machine learning algorithms for modeling groundwater level changes in agricultural regions of the U.S. Journal of water resource research 2017. DOI: 10.1002/2016wr019933.
6. Sirhan H, Koch M. Numerical Modeling of three-dimensional coupled Flow and Salinity Transport in the Gaza Coastal Aquifer, South Palestine, In: Proceedings of the "6th International Conference on Water Resources and Environment Research, ICWRER 2013", Koblenz, Germany 2013.
7. Sujatha P, GN, Kumar P. Prediction of Groundwater Levels Using Different Artificial Neural Network Architectures and Algorithms, Proceedings on the international conference on artificial intelligence (ICAI), Unpublished Doctor of Philosophy thesis submitted to Dept of civil engineering, Sri Venkateshwara University, Tirupati 2005.
8. Wagh VM, Pawar RS, Panaskar DB, Aamalawar ML. Status of Groundwater Quality. Asian Journal of Biochemical and Pharmaceutical Research 2014;4(3):59-65.