World Academy of Science, Engineering and Technology
International Journal of Electrical and Computer Engineering
Vol:1, No:4, 2007

Application of Artificial Neural Networks for Temperature Forecasting

Mohsen Hayati, and Zahra Mohebi

***Abstract*—**In this paper, the application of neural networks to study the design of short-term temperature forecasting (STTF) Systems for Kermanshah city, west of Iran was explored. One important architecture of neural networks named Multi-Layer Perceptron (MLP) to model STTF systems is used. Our study based on MLP was trained and tested using ten years (1996-2006) meteorological data. The results show that MLP network has the minimum forecasting error and can be considered as a good method to model the STTF systems.

***Keywords*—**Artificial neural networks, Forecasting, Weather, Multi-layer perceptron.

I. Introduction

WEATHER forecasting is the application of science and technology to predict the state of the atmosphere for a future time and a given location. Human kind has attempted to predict the weather since ancient times. Today, weather forecasts are made by collecting quantitative data about the current state of the atmosphere and using scientific understanding of atmospheric processes to project how the atmosphere will evolve. The chaotic nature of the atmosphere, the massive computational power required to solve the equations that describe the atmosphere and incomplete understanding of atmospheric processes mean that forecasts become less accurate as the range of the forecast increases. The need for accurate weather prediction is apparent when considering the benefits that it has. These predictions may not stop a tornado, hurricane or flood, but they can help us prepare for one. Great progress was made in the science of meteorology till now. Meteorologist use different methods for weather prediction. The possibility of numerical weather prediction was proposed by Lewis Fry Richardson in 1922. Practical use of numerical weather prediction began in 1955, spurred by the development of programmable electronic computers. Observation of atmospheric pressure, temperature, wind speed, wind direction, humidity, and precipitation are made near the earth's surface by trained observers, automatic weather stations.

Manuscript received March 11, 2007. This work was supported in part by Electrical Engineering Department, Faculty of Engineering, Razi University, Kermanshah, Iran.

M. Hayati is with Electrical Engineering Department, Faculty of Engineering, Razi University, Kermanshah, Iran (e-mail: hayati@razi.ac.ir) as well as with Electrical Engineering Department, Islamic Azad University, Kermanshah Branch, Kermanshah, Iran.

Z. Mohebi is M.E student in Electrical Engineering Department, Faculty of Engineering, Razi University, Kermanshah, Iran.

The World Meteorological Organization acts to standardize the instrumentation, observing practices and timing of these observations world wide. Artificial neural networks (ANN) are parallel computational models, comprising closely interconnected adaptive processing units. The important characteristic of neural networks is their adaptive nature, where 'learning by example replaces programming'. This feature makes the ANN techniques very appealing in application domains for solving highly nonlinear phenomena. During last four decades various complex problems like weather prediction [1],[2], stock market prediction etc has been proved to be areas with ample scope of application of this sophisticated mathematical tool. A multilayer neural network can approximate any smooth, measurable function between input and output vectors by selecting a suitable set of connecting weight and transfer functions.

Despite so much of emphasis given to the application of ANN in prediction of different weather events all over the world, Iranian meteorological forecasters did not put much precedence on application of this potent mathematical tool in atmospheric prediction. Our study was based on Multi Layer Perceptron(MLP) which trained and tested using past ten years (1996-2006) meteorological data.

The objective of this study is to develop ANN-based model by using meteorological data of Kermanshah city located in west of Iran for one day ahead forecasting of temperature of this area.

II. Data Collection

Weather data of ten years were collected from the meteorological department of Kermanshah, Iran, which has shown in Table I and Table II. Table I shows the part of data which has measured every 3 hours, and Table II is daily value of variables. Another variable which we got every 6 hours was soil temperature at 5, 10,20,30,50 and 100 cm of soil depth. The general structure of input/outputs i.e., ANN model for temperature forecasting is shown in Fig. 1.

The chosen weather data were divided into two randomly selected groups, the training group, corresponding to 67% of the patterns, and the test group, corresponding to 33% of patterns; so that the generalization capacity of network could be checked after training phase. Also four random days were selected as unseen data. We used the Mean Absolute Error (MAE) as a measure of error made by the neural network.

$$MAE = \frac{1}{M_{total}} \sum_{i=1}^{M_{total}} \left| P_i - P_i^* \right|$$

Where P_i , P_i^* and M_{total} are exact values, predicted values and total number of the test data respectively. The variation of Kermanshah temperature over last ten years (1996-2006) is shown in Fig. 2.

TABLE I
METEOROLOGICAL VARIABLES

No.	Meteorological variables in the every 3 hours time frame	unit
1	Wind speed	Knot
2	Wind direction	Deg
3	Dry Bulb temperature	Deg.C
4	Wet Bulb temperature	Deg.C
5	Relative humidity	%Rh
6	Dew point	Deg.C
7	Pressure	mb
8	Visibility	Km
9	Amount of cloud	octa

TABLE II
METEOROLOGICAL VARIABLES

No.	Daily meteorological variables	unit
1	Gust wind	Knot
2	Mean temperature	Deg.C
3	Maximum temperature	Deg.C
4	Minimum temperature	Deg.C
5	Precipitation	mm
6	Mean humidity	%H
7	Mean pressure	mb
8	Sunshine	H
9	Radiation	Cal/m2
10	Evaporation	Mm

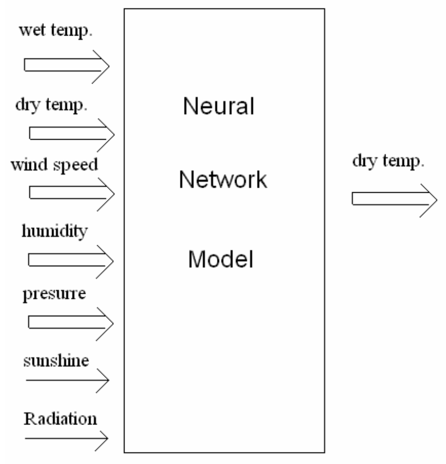

Fig. 1 General structure

III. NEURAL NETWORKS

Neural networks have seen an explosion of interest over the last few years, and are being successfully applied across an extraordinary range of problem domains, in areas as diverse as finance medicine, engineering, geology and physics. Indeed, anywhere that there are problems of prediction, classification or control, neural networks are being introduced. Neural networks could be define as an interconnected of simple processing element whose functionality is based on the biological neuron. Biological neuron is a unique piece of equipment that carries information or a bit of knowledge and transfers to other neuron in the chain of networks. Artificial neuron imitates these functions and their unique process of learning [3]. An artificial neuron is shown in Fig. 3 [4].

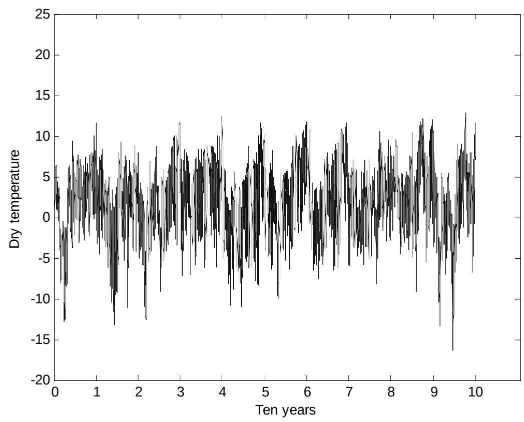

Fig. 2 (a) Dry temperature variations over last ten years (1996-2006)

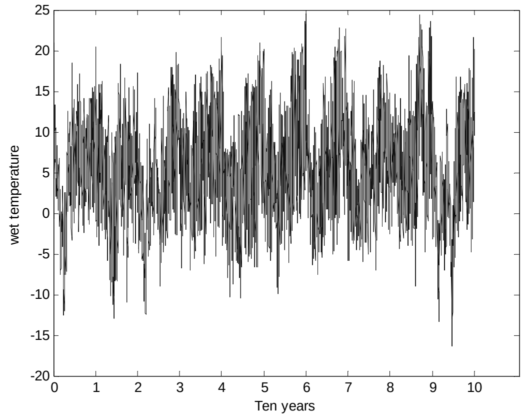

Fig. 2 (b) Wet temperature variations over last ten years (1996-2006)

The developed ANN model is based on one of the neural network architecture named multi-layer perceptron [5],[6].

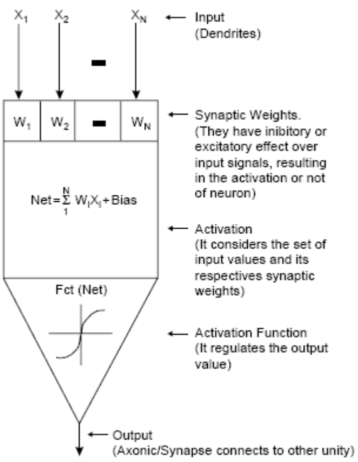

Fig. 3 Artificial neuron

A. *Multi-Layer Perceptron (MLP)*

This is perhaps the most popular network architecture in use today. Its units each perform a biased weighted sum of their inputs and pass this activation level through a transfer function to produce their input, and the units are arranged in a layered feed forward topology. The network thus has a simple interpretation as a form of input-output model, with the weights and thresholds (biases) the free parameters of the model.

Such networks can model functions of almost arbitrary complexity with the number of layers, and the number of units in each layer, determining the function complexity. Important issues in Multilayer Perceptron design include specification of the number of hidden layers and the number of units in these layers [5],[7]. Once the number of layers, and number of units in each layer has been selected, the network's weights and thresholds must be set so as to minimize the prediction error made by the network. This is the role of the training algorithms. The best known example of a neural network training algorithm is back propagation [6],[8],[9]. Modern second-order algorithm such as conjugate gradient descent and Levenberg-Marquardt [7] are substantially faster for many problems, but Back propagation still has advantages in some circumstances, and is the easiest algorithm to understand. MLP structure [6] is shown in Fig. 4.

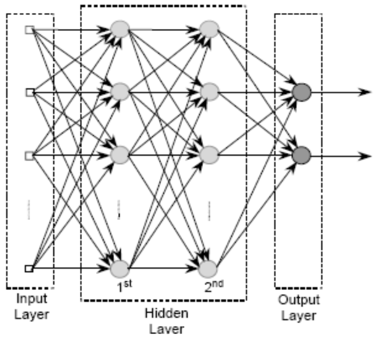

Fig. 4 Example of two layer MLP

With this background we designed and trained this network as below:

The three-layer network with sigmoid transfer function for hidden layer and linear transfer function for output layer can represent any functional relationship between inputs and outputs, if the sigmoid layer has enough neurons [5], so we selected this three layer structure. Back propagation training algorithms are often too slow for practical problems, so we can use several high performance algorithms that can converge from ten to one hundred times faster than back propagation algorithms. These faster algorithms fall into two main categories: heuristic technique (variable learning rate back propagation, resilient back propagation) and numerical optimization techniques (conjugate gradient, quasi-Newton, Levenberg-Marquardt). We tried several of these algorithms to get the best result. Levenberg-Marquardt is the fastest algorithm but as the number of weights and biases in the network increase, the advantage of this algorithm decrease, so we tried another algorithm which perform well on function approximation and converge rather fast. From these algorithms, scaled conjugate gradient was suitable for our purpose. From an optimization point of view learning in a neural network is equivalent to minimizing a global error function, which is a multivariate function that depends on the weights in the network. Many of the training algorithms are based on the gradient descent algorithm. Minimization is a local iterative process in which an approximation to the function, in a neighborhood of the current point in the weight space, is minimized. Most of the optimization methods used to minimize functions are based on the same strategy. The Scaled Conjugate Gradient (SCG) algorithm [10] denotes the quadratic approximation to the error E in a neighborhood of a point w by:

$$E_{qw}(y) = E(w) + E'(w)^T y + \frac{1}{2} y^T E''(w) y$$

In order to determine the minimum for $E_{qw}(y)$ the critical points for $E_{qw}(y)$ must be found. The critical points are the solution to the linear system defined by Moller [10].

$$E'_{qw}(y) = E''(w)y + E'(w) = 0$$

SCG belongs to the class of Conjugate Gradient Methods, which show super linear convergence on most problems. By using a step size scaling mechanism SCG avoids a time consuming line-search per learning iteration, which makes the algorithm faster than other second order algorithms. And also we got better results than with other training methods and neural networks tested, as standard back-propagation. Neural networks generally provide improved performance with the normalized data. The use of original data as input to neural network may cause a convergence problem. All the weather data sets were therefore, transformed into values between -1 and 1 through dividing the difference of actual and minimum values by the difference of maximum and minimum values subtract by 1. At the end of each algorithm, outputs were denormalized into the original data format for achieving the desired result. We know that from one initial condition the algorithm converged to global minimum point, while from the other initial condition the algorithm converged to a local minimum so it is best to try several different initial conditions in order to ensure that optimum solution has been obtained [5]. For a network to be able to generalize, it should have fewer parameters than there are data points in the training set [5],[11]. Training goal for the networks was set to 10^{-4}. Finding appropriate architecture needs trial and error method. Networks were trained for a fixed number of epochs. Performance of the network was evaluated by increasing the number of hidden neurons. After finding hidden neurons, epochs increase till we find the suitable epochs [12].

IV. Result And Discussion

The optimal structures for developed MLP neural network for obtaining minimum prediction error are shown in Table III.

TABLE III
MLP Structure

Number of hidden layers	1
Number of hidden neuron	6
Number of epochs	2000
Activation function used in hidden layer	tan-sig
Activation function used in output layer	pure linear

The over all mean absolute error for test days is shown in Fig. 5.

Fig. 5 Mean absolute error

Table V shows the minimum and maximum error between exact values and predicted values of unseen days so that the generalization capacity of network can be checked after training and testing phase.

TABLE IV
Exact and Predicted Values for Unseen Days

Unseen Days	Minimum Error	Maximum Error
02-Jan-1997	0.0079	0.6905
27-Aug-2000	0.1257	0.8005
09-Jun-2004	0.0809	1.0006
29-Nov-2006	0.0336	1.2916

The exact and predicted values for each unseen day by MLP network is shown in Fig. 6. From the result shown in Fig. 5, Fig. 6 and Table IV, it is observed that the predicted values are in good agreement with exact values and the predicted error is very less. Therefore the proposed ANN model with the developed structure shown in Table III can perform good prediction with least error.

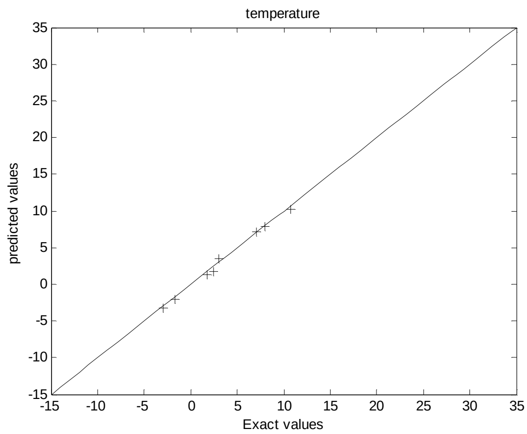

Fig. 6 (a) Comparison between exact and predicted values for unseen day i.e., 2-Jan-1997

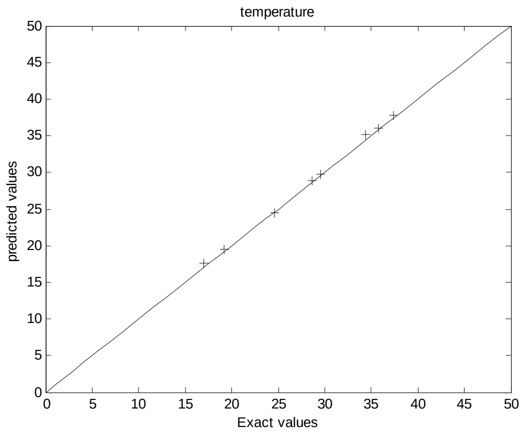

Fig. 6 (b) Comparison between exact and predicted values for unseen day i.e., 27-Aug-2000

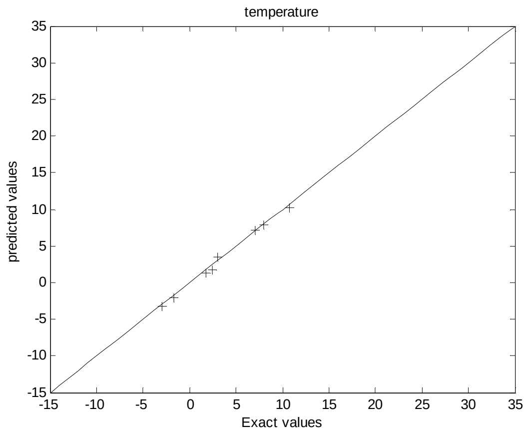

Fig. 6 (c) Comparison between exact and predicted values for unseen day i.e., 09-Jun-2004

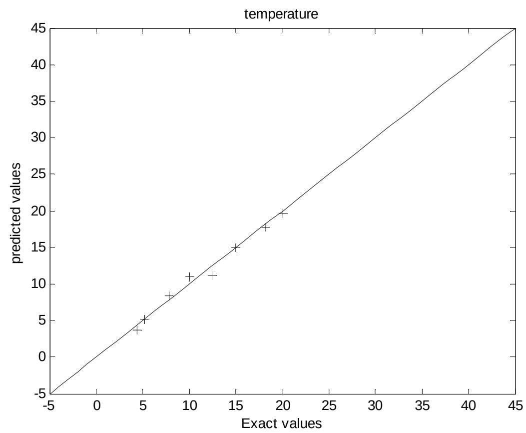

Fig. 6 (d) Comparison between exact and predicted values for unseen day i.e., 29-Nov-2006

V. Conclusion

The result of MLP network model used for one day ahead temperature forecast in the Kermanshah, Iran, shows that MLP network has a good performance and reasonable prediction accuracy was achieved for this model. It's forecasting reliabilities were evaluated by computing the mean absolute error between the exact and predicted values. The results suggest that this neural network could be an important tool for temperature forecasting.

Acknowledgment

We would like to express our gratitude to Mr. M. Ghasemi, manager of west of Iran meteorological research center for providing us the required data.

References

[1] I. Maqsood, I., Khan, M.R., Abraham, A., Intelligent weather monitoring systems using connectionist models. *International Journal of Neural, Parallel and Scientific Computations*, No. 10, 2000, pp.157–178.

[2] I. Maqsood, M.R. Khan, A. Abraham, Neuro-computing based Canadian weather analysis, *The 2nd International Workshop on Intelligent Systems Design and Applications, Computational Intelligence and Applications*. Dynamic Publishers, Atlanta, Georgia, 2002, pp. 39–44.

[3] L. Fausett, *Fundamental of Neural Networks*, New York , Prentice Hall. A well-written book, with very detailed worked examples to explain how the algorithm function, 1994.

[4] J.M. Zurada, *Introduction to Artificial Neural Systems*, West Publishing Company, Saint Paul, Minnesota, 1992.

[5] M.T. Hagan. H.B. Demuth, M.H. Beale. *Neural Network Design*. PWS Publishing Company, Boston, Massachusetts, 1996.

[6] S. Haykin. *Neural Networks*, A Comprehensive Foundation, New York, Macmillan Publishing. A comprehensive book, with an engineering perspective. Requires a good mathematical background , and contains a great deal of background theory, 1994.

[7] C. Bishop, *Neural Networks for Pattern Recognition*, University press. Extremely well-Written, up-to-date. Require a good mathematical background , but rewards careful reading, putting neural networks firmly into a statistical context, 1995.

[8] D. Patterson. *Artificial neural networks*. Singapore, Prentice Hall. Good wide-ranging coverage of topics, although less detailed than some other books, 1996.

[9] S. Haykin, *Neural networks*—a comprehensive foundation. Prentice-Hall, New Jersey, 1999.

[10] A. Moller. Scaled Conjugate Gradient Algorithm for Fast Supervised Learning, *Neural Networks*, 6 (4), 1993, pp.525-533.

[11] I. Maqsood, M.R. Khan, A. Abraham, Canadian weather analysis using connectionist learning paradigms. *The Seventh Online World Conference on Soft Computing in Industrial Application*, On the Internet. Springer, Germany, 2000.

[12] Y. Linde, A. Buzo, R. Gray, An algorithm for vector quantizer design. *IEEE Transactions on Communications*, No. 28, 1980, pp. 84–95.

[13] P.P. Vander Smagt, Minimization methods for training feedforward neural networks. *Neural Networks*, 1, 1994, pp.1–11.