Weather Prediction Using Artificial Neural Network

Abdul Manan Ahmad[1], Chia Su Chuan[2] and Fatimah Mohamad
[1] Department of Software Engineering,
Faculty of Computer Science & Information System
Universiti Teknologi Malaysia, 81310 Skudai, Johore, Malaysia
Tel : +607-550-3701, Fax no: +607-556-5044
email :manan@fsksm.utm.my
[2] Software Engineering Department,
Faculty of Computer Science & Information System
Universiti Teknologi Malaysia, 81310 Skudai, Johore, Malaysia

Abstract: The characteristic features of Malaysia's climate is has stable temperature, with high humidity and copious rainfall. Weather forecasting is an important task in Malaysia as it could affetcs man irrespective of mans job, lifestyle and activities especially in the agriculture. In Malaysia, numerical method is the common used method to forecast weather which involves a complex of mathematical computing. The models used in forecasting are supplied by other counties such as Europe and Japan. The goal of this project is to forecast weather using another techonology known as artificial neural network. This system is capable to learn the pattern of rainfall in order to produce a precise forecasting result. The supervised learning technique is used in the learning process.

1. Introduction

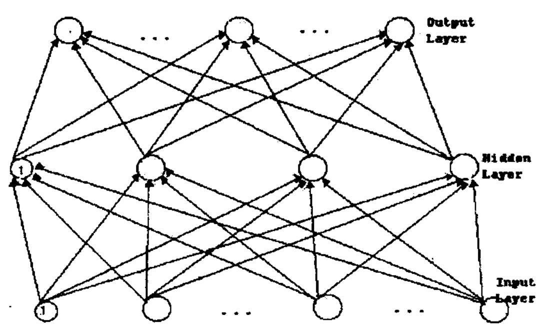
Rain forecasting is one of the most important operational practices carried out by Meteorological Services Malaysia. The most widespread technique used for rainfall forecasting is the numerical statistical method. Even though research within this field has been conducted for a long time, it is still failed from time to time.

Recently, artificial neural network (ANN) has been used for a wide range of different real-world applications. Many complex areas of science utilize neural networks for different kind of learning problem and their derivatives. This article will describe the multiplayer perceptron with error back-propagation algorithm to provide prediction of rainfall. The ANN technique is applied to the hourly local weather data for the area of Senai, Johor, Malaysia.

2. Approach And Methods

The brain consists of a large number is neurons, connected with each other by synapses. These networks of neurons are called neural networks, or natural neural networks. The ANN is a simplified mathematical model of a natural neural network. It is a directed graph where a vertex corresponds to a neuron and an edge to a synapse.

In general, climate and rainfall are highly non-linear phenomena. The multilayer perceptron neural network has been designed to function well in such non-linear phenomena. A feedforward multilayer neural network consists of an input layer and output layer with some number of input and output neurons respectively. There are also one or more hidden layers in between the input and the output layer with some number of neurons on each.



"Figure 1. Structure of a neural network with one hidden layer"

The values X_i of the ith neuron of the first hidden layer is given by

$$X_i = f(\Sigma_i w_{ji} O_j + \theta_i),$$

Where w_{ji} are the weights connecting the jth input neuron (whose value is O_j) to the ith hidden neuron whose activation threshold(bias) is θ_i and f is a smooth and bounded function called activation function. A similar rule applies to the neurons for the second hidden layer as well as output layer. The activation function being used in this work is sigmoid function:

$$f(x_i) = \frac{1}{1 + e^{-(x_i)}}$$

The learning algorithm under study, is the back propagation, by Rumelhart et. al. The basic idea of this method is the utilization of a non-linear perceptron-like system, capable of making decisions. The network learns the relationship between the input-output data in the training set via network training in which the weights are modified by presenting input-output patterns of the training set until a prescribed error criterion is fulfilled. For adjusting the weight (w_{ij}) from the ith input unit to the jth output, the derivative of the error function with respect to any weight in the network is computed as shown below:

$$\Delta w_{ij} = -k \frac{\partial E}{\partial w_{ij}}$$

$$w_{ij}(t+1) - w_{ij}(t) = \eta \delta_j x'_i + \alpha (w_{ij}(t) - w_{ij}(t-1))$$

Where $w_{ij}(t)$ is the weight from hidden node j or from an input node j at time t, x_i' is either the output of node i or is an input, η and α is the learning rate (gain) and momentum of the net respectively, and δ_j is the error term for node j.

3. The Data Set

The data input file contained 9 distinct features: time of the day, pressure, dry bulb temperature, wet bulb temperature, dew point, wind speed, wind direction, cloud cover and whether it rained in the past hour.

The followings is an samples of pre-normalized data input set:

"Table 1. Samples of Data Input"

Hour	Pressure	Temperature Wet Bulb	Temperature Dry Bulb
0.69565	0.42938	0.55894	0.43631
0	0.59618	0.24748	0.26429
0.65217	0.37948	0.43316	0.47088
0	0.59618	0.24748	0.26429
0.13043	0.35539	0.20342	0.25095
0.17391	0.41673	0.21743	0.25095
0.21739	0.48076	0.23211	0.25095
0.34783	0.7126	0324748	0.27113
0.3913	0.7126	0.26351	0.29231
0.43478	0.66833	0.2802	0.29231

Wind Direction	Wind Speed	Total Cloud Cover	Dew Point	Past Weather
0.83941	0.60735	0.86095	0.59235	0
0.76043	0.79806	0.69227	0.33819	1
0.34834	0.82852	0.83095	0.33819	0
0.76043	0.79806	0.69227	0.33819	1
0.76043	0.60735	0.69227	0.33819	0
0.76043	0.63881	0.86095	0.33819	0
0.79719	0.54195	0.86095	0.33819	1
0.30151	0.54195	0.86095	0.33819	1
0.83941	0.72556	0.86095	0.33819	1
0.81926	0.69809	0.86095	0.33819	1

The output layer consisted of two output neurons represented the raining and sunny weather respectively. Only one neuron will be set at 1 for each input pattern and 0 for the other neuron.

4. Experiment

Two different configuration of neural network architecture will be used in the training and testing purpose, which are 9-9-2 and 9-9-9-2. The table 2 shows that the configuration with 9-9-9-2 was performed better than the 9-9-2 network. Figure 2 and Figure 3 illustrate the Graph Mean Squared Error over a full epoch for the network 9-9-2 and 9-9-9-2 respectively for learning process. Further experiments with different configurations will be carried out in order to get a well-trained network.

"Table 2. Result of learning and testing for each architecture network"

Architecture	Parameter	Accuracy
9-9-9-2	Learning Rate: 0.25	77%
	Momentum : 0.15	
	Error Tolerance : 0.17	
	Epoch Needed : 2354	
9-9-2	Learning Rate: 0.25	78%
	Momentum : 0.25	
	Error Tolerance : 0.2	
	Epoch Needed: 1958	

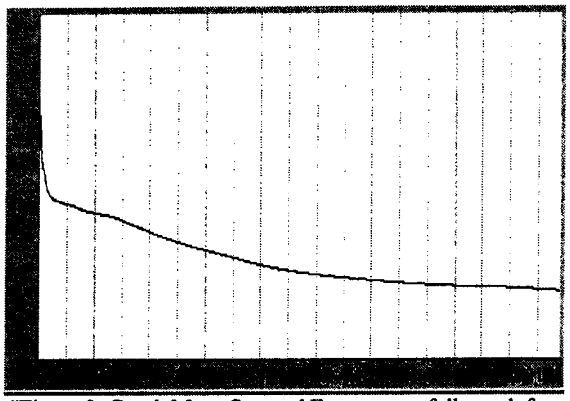
5. Sample Output

"Table 3. Result of testing process"

Hour	Pressure	Temperature Dry Bulb	Temperature Wet Bulb
0.04348	0.57099	0.18583	0.1068
0.08696	0.46783	0.1755	0.09902
0.04348	0.45495	0.25095	0.20342
0.65217	0.23763	0.35318	0.39231
0.3913	0.62088	0.54054	0.60001
0.95652	0.6909	0.24445	0.1901
0.13043	0.51962	0.25095	0.20342
0.17391	0.66833	0.19116	0.09902
0.65217	0.28771	0.65724	0.74649
0.69565	0.37948	0.54054	0.60001

Wind Direction	Wind Speed	Total Cloud Cover	Dew Point	Past Weather
0.79719	0.54195	0.86095	0.15233	1
0.76043	0.50855	0.86095	0.15233	1
0.76043	0.47507	0.69227	0.33819	1
0.34834	0.57498	0.69227	0.59235	1
0.83941	0.54195	0.69227	0.59235	0
0.34834	0.37712	0.69227	0.33819	
0.77317	0.40907	0.69227	0.33819	0
0.77317	0.82852	0.69227	0.15233	0
0.31673	0.90423	0.69227	0.80514	1
0.62569	0.3462	0.69227	0.59235	1

Target Sunny	Target Raining	Predicted Sunny (9-9-2)	Predicted Raining (9-9-2)	Predicted Sunny (9-9-9-2)	Predicted Raining (9-9-9-2)
0	1	0.002779	0.994230	0.005324	0.98806
1	0	0.000512	0.998833	0.006462	0.98586
1	0	0.010508	0.968163	0.004827	0.98902
0	1	0.000003	0.999988	0.000006	0.99997
1	0	0.579394	0.461290	0.986995	0.01210
1	0	0.999015	0.003424	0.724088	0.52250
1	0	0.998988	0.001472	0.982674	0.01554
0	1	0.000009	0.999956	0.022133	0.96150
0	1	0.689677	0.337886	0.117878	0.86162
1	0	0.859423	0.176072	0.99589	0.00649



"Figure 2. Graph Mean Squared Error over a full epoch for neural network 9-9-2 "

"Figure 3. Graph Mean Squared Error over a full epoch for neural network 9-9-2"

6. Conclusion

In this study, the back propagation algorithm was used. This method has a number of inherited limitations:

1. Heavy computational requirements.
2. non existence of ANN design methodologies for deciding the value of learning rate and momentum.

Many researches and experiments done in weather prediction with neural networks hint that a forecast can be done with at least as good accuracy as the models that are used today. But the reliability issue of a trained network is still in doubt.

The network performance could have been improved by providing more training data. It will be interesting to study further the robustness of neural network when compared to numerical-statistical method.

References

[1] Antognetti, P. and Milutinović, V. "Neural Networks Concepts, Applications, and Implementations," 5th. ed. Englewood Cliffs, N.J: Prentice-Hall, 1991.

[2] Bose, N.K. and Liang, P. "Neural Network Fundamentals with Graphs, Algotithms, and Applications," New York: McGraw-Hill, 1996.

[3] Collinge, V. and Kirby, A. "Weather Radar and Flood Forecasting," Chichester: JOHN WILEY & SONS, 1987.

[4] Fausett, L. "Fundamentals of Neural Networks Architectures, Algorithms, And Applications," Englewood Cliff, N.J.: Prentice Hall, 1994.

[5] Karayiannis, N.B. and Venetsanopoulos, A.N. "Artificial Neural Networks Learning Algorithms, Performance Evaluation, and Applications," Boston: Kluwer Academic Publisher, 1993.

[6] Mitchell, F. "Practical Weather Forecasting," Woodbury, N.Y. : BARRON'S, 1977.