ISSN No: 2309-4893
International Journal of Advanced Engineering and Global Technology
Vol-03, Issue-05, May 2015

Predicting Weather Using Data Mining Techniques

Rahul Moriwal, Shiv Kumar Dubey

Department of Computer Science & Engineering
Acropolis Institute of Technology & Research,Indore,India
rmoriwal@gmail.com,shivmgcgv@gmail.com

Abstract-In this paper we present an analysis of the prediction of weather using Data mining techniques. We have investigated data mining techniques is neural network with Back propagation network for supervised learning using recorded data. To obtain the accurate result different testing and training rules used with Weka.
Keywords- Data mining, ANN, Temperature forecating, Back propagation ,Weka.

I. INTRODUCTION

Weather forecasting is one of the important applications of science and technology to predict the state of the atmosphere for a particular location. Weather forecasts are made by collecting data about the current state of the atmosphere and using scientific understanding of atmospheric processes to project how the atmosphere will evolve.

Now weather forecasting relies on computer-based models that take many types of atmospheric factors. Human input is still required to pick the best possible forecast model to base the forecast upon, which involves pattern recognition skills, teleconnections, knowledge of model performance, and knowledge of model biases. There are different types of end uses to weather forecasts. Weather warnings are one of the important forecasts because they help to protect life and property. For agriculture temperature and precipitation forecast is important. Utility companies also used temperature forecast to estimate demand over coming days. People use weather forecasts to determine what to wear on a given day.

Atmosphere is not ordered form, so massive computational power is required to solve the equations which describe the atmosphere, error involved in measuring the initial conditions, and an incomplete understanding of atmospheric processes. This means that forecasts become less accurate as the difference in current time and the time for which the forecast is being made (the range of the forecast) increases.

Several steps to predict the temperature are
1. Data collection
2. Data assimilation and analysis,
3. Numerical weather prediction,
4. Model output post processing.

II. OVERVIEW

An Artificial Neural Network (ANN) is an information processing system composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example is configured for a specific application, such as pattern recognition or data classification, through a learning process.

A neural network is a massively parallel distributed processor made up of simple processing units, which has a neutral property fro storing experiential knowledge and making it available for use. It resembles the brain in two aspects-

- Knowledge is acquired by the network from its environment through a learning process.
- Interneurons connection strengths, known as synaptic weights, are used to store the acquired
- Knowledge.

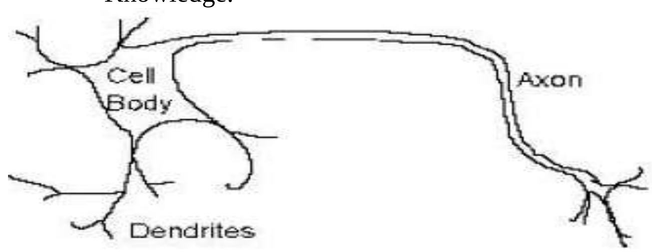

Figure:-1

III. WHY USE NEURAL NETWORKS?

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyses. Benefits of Neural Network:

1. Adaptive learning: An ability to learn how to do tasks based on the data given for training or initial experience.
2. Self-Organization: An ANN can create its own organization or representation of the information it receives during learning time.

1. Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.
2. Fault Tolerance via Redundant Information Coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

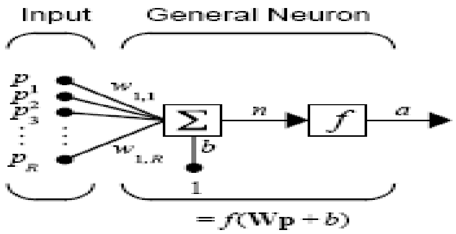

Fig 2

IV. MULTILAYER PERCEPTRONS

The network consists of a set of sensory units that constitute the input layer, one or more hidden layers of computation nodes, and an output layer of computation nodes. The input signal propagates through the networks are commonly referred to as multilayer perceptrons (MLPs) which represent a generalization of the single layer perceptron..

A multilayer perceptron (MLP) is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate output. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron with a nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training the network.MLP is a modification of the standard linear perceptron, which can distinguish data that is not linearly separable.

The commonest type of artificial neural network consists of three groups, or layers, of units: a layer of "input" units is connected to a layer of "hidden" units, which is connected to a layer of "output" units. (See Figure 3)

- The activity of the input units represents the raw information that is fed into the network.
- The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.

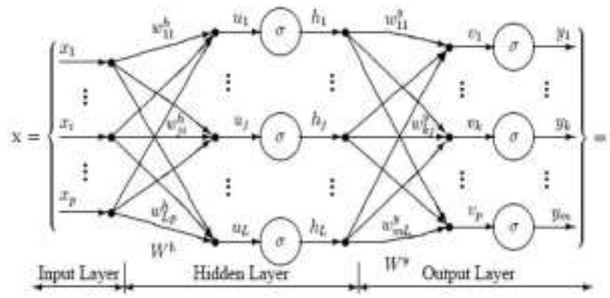

Figure:- 3

- The behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units.

V. THE BACK-PROPAGATION ALGORITHM

In order to train a neural network to perform some task, we must adjust the weights of each unit in such a way that the error between the desired output and the actual output is reduced. This process requires that the neural network compute the error derivative of the weights. In other words, it must calculate how the error changes as each weight is increased or decreased slightly. The back propagation algorithm is the most widely used method for determining the EW.

Back propagation is a common method of teaching artificial neural networks how to perform a given task. It was first described by Arthur E. Bryson and Yu-Chi Ho in 1969. It is a supervised learning method, and is a generalization of the delta rule.

The back-propagation algorithm is easiest to understand if all the units in the network are linear. The algorithm computes each EW by first computing the EA, the rate at which the error changes as the activity level of a unit is changed. For output units, the EA is simply the difference between the actual and the desired output. To compute the EA for a hidden unit in the layer just before the output layer, we first identify all the weights between that hidden unit and the output units to which it is connected. We then multiply those weights by the EAs of those output units and add the products. This sum equals the EA for the chosen hidden unit. After calculating all the EAs in the hidden layer just before the output layer, we can compute in like fashion the EAs for other layers, moving from layer to layer in a direction opposite to the way activities propagate through the network. This is what gives back propagation its name. Once the EA has been computed for a unit, it is straight forward to compute the EW for each incoming connection of the unit. The EW is the product of the EA and the activity through the incoming connection.

The back-propagation Algorithm

Units are connected to one another. There is a real number associated with each connection, which is called the weight of the connection. We denote by Wij the weight of the connection from unit ui to unit uj.Two types of connection are usually distinguished: excitatory and inhibitory. A positive weight represents an excitatory connection whereas a negative weight represents an inhibitory connection.

A unit in the output layer determines its activity by following a two step procedure.

- First, it computes the total weighted input xj, using the formula:

$$X_j = \sum_i y_i W_{ij}$$

where yi is the activity level of the jth unit in the previous layer and Wij is the weight of the connection between the ith and the jth unit.

Next, the unit calculates the activity y_j using some function of the total weighted input. Typically we use the sigmoid function:

$$y_j = \frac{1}{1+e^{-x_j}}$$

Once the activities of all output units have been determined, the network computes the error E, which is defined by the expression:

$$E = \frac{1}{2}\sum_i (y_i - d_i)^2$$

Where yj is the activity level of the jth unit in the top layer and dj is the desired output of the jth unit.

The back-propagation algorithm consists of four steps:

1. Compute how fast the error changes as the activity of an output unit is changed. This error derivative (EA) is the difference between the actual and the desired activity.

$$\mathrm{EA}_j = \frac{\partial E}{\partial y_j} = y_j - d_j$$

2. Compute how fast the error changes as the total input received by an output unit is changed. This quantity (EI) is the answer from step 1 multiplied by the rate at which the output of a unit changes as its total input is changed.

$$\mathrm{EI}_j = \frac{\partial E}{\partial x_j} = \frac{\partial E}{\partial y_j} \times \frac{dy_j}{dx_j} = \mathrm{EA}_j y_j (1 - y_j)$$

3. Compute how fast the error changes as a weight on the connection into an output unit is changed.

$$\mathrm{EW}_{ij} = \frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial x_j} \times \frac{\partial x_j}{\partial W_{ij}} = \mathrm{EI}_j y_i$$

4. Compute how fast the error changes as the activity of a unit in the previous layer is changed. When the activity of a unit in the previous layer changes, it affects the activates of all the output units to which it is connected. So to compute the overall effect on the error, we add together all these separate effects on output units.

$$\mathrm{EA}_i = \frac{\partial E}{\partial y_i} = \sum_j \frac{\partial E}{\partial x_j} \times \frac{\partial x_j}{\partial y_i} = \sum_j \mathrm{EI}_j W_{ij}$$

The following conditions are to be analyzed for input to BPN,
(i) Atmospheric Min & Max Temperature
(ii) Rainfall

VI. Experimentation and Result

In this paper, the accuracy of the data is analyzed. The goal is to have high accuracy, besides high precision. We have used the Weka toolkit to experiment with the data mining algorithms. The Weka is an ensemble of tools for data classification, regression, clustering, association rules, and visualization. The toolkit is developed in Java and is open source software issued under the GNU General Public License.

Table 1 shows the graph for Bhopal climate in different time interval. We tested our as training set and obtained the related temperature with Weka. The training time vs error graph is shown in fig.5.

From the conducted experiment with Weka we can examine that the temperature peeks during the month of May, its range is 40.7 degree centigrade. The temperature falls during the month of January. The minimum temperature is 10.2 C.

The Training-Error graph explains that the error is high when the iteration is less and vice versa. In the above graph it explains that when the iteration count is 500 the sum squared error is maximum (i.e. 3.38) and when the count reaches 4000 the error value is merely 1.0849.

This result shows that for more accurate results, the iteration count should be high.

VII. Conclusion

In this paper, back propagation neural network is used with weka for predicting the temperature based on the training sets. For good results Back propagation neural network approach for temperature forecasting is capable and it is considered as an alternative to traditional meteorological approaches. Make a prediction of temperature in future, this approach is able to determine the non-linear relationship

ISSN No: 2309-4893
International Journal of Advanced Engineering and Global Technology
Vol-03, Issue-05, May 2015

that exists between the historical data (temperature) supplied to the system during the training phase and on that basis.

Temperatures in India (by month)	Average Minimum Temperatures in Bhopal, India	Average Maximum Temperature in Bhopal, India	Average Rainfall/ Precipitation (mm)	Wet Days
January	10.2	25.3	12.9	1.3
February	12.4	28.6	7.8	0.7
March	17.1	33.6	7.2	0.7
April	21.8	38.3	4.5	0.4
May	25.5	40.7	8.0	0.8
June	25.3	37.0	114.0	6.7
July	23.1	30.6	355.8	14.5
August	22.4	28.8	388.4	14.9
September	21.4	30.5	195.8	8.2
October	18.4	32.0	26.2	1.7
November	14.1	29.0	13.7	0.8
December	10.9	25.9	12.4	0.7

Table: - 1

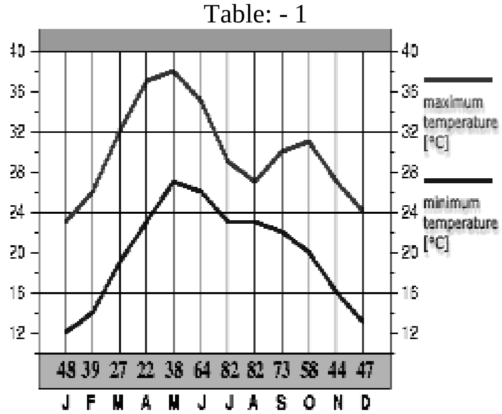

Figure: 1

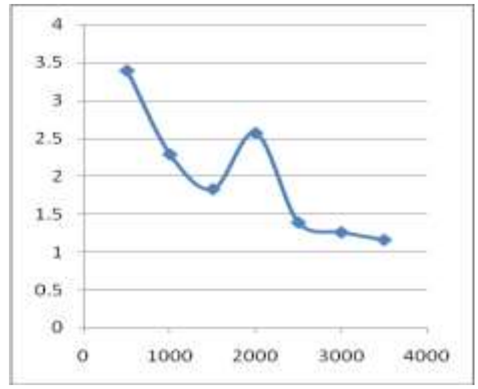

Figure: 2

Training Time-Root Relative squared Error Graph

REFERENCE

[1]Xinghuo Yu, M. Onder Efe, and Okyay Kaynak,” A General Back propagation Algorithm for Feedforward Neural Networks Learning,”

[2] R. Rojas: Neural Networks, Springer-Verlag, Berlin, 1996.

[3] Y.Radhika and M.Shashi,” Atmospheric Temperature Prediction using Support Vector Machines,” International Journal of Computer Theory and Engineering, Vol. 1, No. 1, April 2009 1793-8201.

[4] An Introduction to Back-Propagation Neural Networks by Pete McCollum.

[5] Mohsen Hayati, and Zahra Mohebi,” Application of Artificial Neural Networks for Temperature Forecasting,” World Academy of Science, Engineering and Technology 28 2007.

[6] Brian A. Smith, Ronald W. McClendon, and Gerrit Hoogenboom,” Improving Air Temperature Prediction with Artificial Neural Networks” International Journal of Computational Intelligence 3;3 2007.

[7] Arvind Sharma, Prof. Manish Manoria,” A Weather Forecasting System using concept of Soft Computing,”

[9] Surajit Chattopadhyay ,” Multilayered feed forward Artificial Neural Network model to predict the average summer-monsoon rainfall in India ,”

[10] Raúl Rojas,” The back propagation algorithm of Neural Networks - A Systematic Introduction, “chapter 7, ISBN 978-3540605058

[11] Mike O'Neill,” Neural Network for Recognition of Handwritten Digits,” Standard Reference Data Program National Institute of Standards and Technology.

[12] http://www.climatetemp.info/india/

[13] S. Huber, M. Rohde, M. Tamme, “Rule Extraction from Artificial Neural Networks”, PS Natural Computation, SS 2006.

[14] http://en.wikipedia.org/wiki/Weather_forecasting.