

**Laboratory Manual**  
For  
**Data Analysis and Information Extraction**  
**(IT 704)**

B.Tech (IT)  
SEM VII



June 2010

Faculty of Technology  
Dharmsinh Desai University  
Nadiad.  
[www.ddu.ac.in](http://www.ddu.ac.in)

## **Table of Contents**

### **EXPERIMENT-1**

To perform various commands given in PL/SQL in Oracle 8.0(For brushing up.) .....7

### **EXPERIMENT-2**

To perform multi-dimensional data model using SQL queries. E.g. Star, snowflake and Fact constellation schemas .....8

### **EXPERIMENT-3**

To perform various OLAP operations such slice, dice, roll up, drill up, pivot etc. ....9

### **EXPERIMENT-4**

To perform the text mining on the given data warehouse..... 10

### **EXPERIMENT-5**

To perform the correlation ship analysis between for the given data set. .... 11

### **EXPERIMENT-6**

To perform the attribute relevance analysis on the given data..... 12

### **EXPERIMENT-7**

To perform the information gain for a particular attribute in the given data. .... 13

### **EXPERIMENT-8**

To perform the experiment to predict the class using the Bayesian classification ..... 14

### **EXPERIMENT-9**

Write a program to find out a weight or bias updating using the back propagation in Neural network..... 15

### **EXPERIMENT-10**

To perform various data mining algorithms on the give data base using Clementine.....16

## **LABWORK BEYOND CURRICULA**

### **EXPERIMENT-11**

To perform data mining using Weka mining tool. .... 17

### **EXPERIMENT-12**

To perform data mining using Excel mining tool..... 18

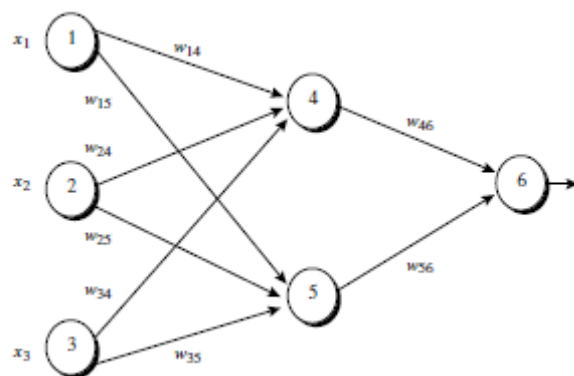
### Sample experiment

**1 AIM:** To perform Simple Neural Network Simulation by using Back Propagation

**2 TOOLS/APPARATUS:** Visual studio 2005 or C++, MS Access.

### 3 STANDARD PROCEDURES:

**3.1 Analyzing the Problem:** For example calculate Neural Network for the following example using back propagation.



Initial input, weight, and bias values.

$x_1$	$x_2$	$x_3$	$w_{14}$	$w_{15}$	$w_{24}$	$w_{25}$	$w_{34}$	$w_{35}$	$w_{46}$	$w_{56}$	$\theta_4$	$\theta_5$	$\theta_6$
1	0	1	0.2	-0.3	0.4	0.1	-0.5	0.2	-0.3	-0.2	-0.4	0.2	0.1

Back propagation is a neural network learning algorithm, the field of neural network was originally kindled by psychologists and neuron biologist who sought to developed and test computational analogues of neurons. The back propagation algorithm performs learning on a multi layer feed forward neural network. The inputs corresponds to the attributes measured for each training sample ,the inputs are feed simultaneously into a layer of units making of the input layer , the weighted outputs of this units are intern feed simultaneously towards second layer of Neurons like units know as hidden layer. The weighted output of the last hidden layer is input to units making of the output layer, which emits the networks prediction for given samples.

### 3.2 Designing the Solution:

Back propagation learns by iteratively processing a data set of training tuples, comparing the network's prediction for each tuple with the actual known *target* value. The target value may be the known class label of the training tuple (for classification problems) or a continuous value (for prediction). For each training tuple, the weights are modified so as to minimize the mean squared error between the network's prediction and the actual target value. These modifications are made in the "backwards" direction, that is, from the output layer, through each hidden layer down to

the first hidden layer (hence the name *back propagation*). Although it is not guaranteed, in general the weights will eventually converge, and the learning process stops. The steps involved are expressed in terms of inputs, outputs, and errors, and may seem awkward if this is your first look at neural network learning. However, once you become familiar with the process, you will see that each step is inherently simple. The steps are described below.

**Algorithm: Backpropagation.** Neural network learning for classification or prediction, using the backpropagation algorithm.

**Input:**

- $D$ , a data set consisting of the training tuples and their associated target values;
- $l$ , the learning rate;
- *network*, a multilayer feed-forward network.

**Output:** A trained neural network.

**Method:**

- (1) Initialize all weights and biases in *network*;
- (2) while terminating condition is not satisfied {
  - (3) for each training tuple  $X$  in  $D$  {
    - (4) // Propagate the inputs forward:
    - (5) for each input layer unit  $j$  {
      - (6)  $O_j = I_j$ ; // output of an input unit is its actual input value
    - (7) for each hidden or output layer unit  $j$  {
      - (8)  $I_j = \sum_i w_{ij} O_i + \theta_j$ ; // compute the net input of unit  $j$  with respect to the previous layer,  $i$
      - (9)  $O_j = \frac{1}{1 + e^{-I_j}}$ ; } // compute the output of each unit  $j$
    - (10) // Backpropagate the errors:
    - (11) for each unit  $j$  in the output layer
      - (12)  $Err_j = O_j(1 - O_j)(T_j - O_j)$ ; // compute the error
    - (13) for each unit  $j$  in the hidden layers, from the last to the first hidden layer
      - (14)  $Err_j = O_j(1 - O_j) \sum_k Err_k w_{jk}$ ; // compute the error with respect to the next higher layer,  $k$
    - (15) for each weight  $w_{ij}$  in *network* {
      - (16)  $\Delta w_{ij} = (l) Err_j O_i$ ; // weight increment
      - (17)  $w_{ij} = w_{ij} + \Delta w_{ij}$ ; } // weight update
      - (18) for each bias  $\theta_j$  in *network* {
        - (19)  $\Delta \theta_j = (l) Err_j$ ; // bias increment
        - (20)  $\theta_j = \theta_j + \Delta \theta_j$ ; } // bias update
      - (21) } }

Class-name	Function-Definition	Purpose of function
NeuralNetwork	NeuralNetwork(double x1, double x2, double x3, double w14, double w15, double w24, double w25, double w34, double w35, double w46, double w56, double t1, double t2, double t3)	Constructor of Neural Network class use to initialize the values of all the parameters

	Void Calculate_Input()	<i>This function is used to calculate the input using equation</i>
	Void calculate_Output()	<i>This function is used to calculate the output using equation</i>
	Void calculate_error()	<i>This function is used to calculate the error prorogate to next layer</i>
	void Calculate_weight()	<i>This function calculates the new weight of every existing weight input.</i>
	void Calculate_Bias()	<i>This function will calculate the All Bias value for neural network</i>
Attribute_relevance	protected void Button1_Click(object sender, EventArgs e)	<i>This function used to execute the whole neural network.</i>

### 3.3 Implementing the Solution

#### 3.3.1 Writing Source Code

```
namespace NewralNetworkclass
{
    public class NeuralNetwork
    {
        const double Tj = 1;
        double Onode4, Onode5, Onode6;
        double Inode4, Inode5, Inode6;
        double Error4, Error5, Error6;
        double x1, x2, x3;
        double W14, W15, W24, W25, W34, W35, W46, W56;
        double O4, O5, O6;
```

```

double learnning_rate = 0.9;

public NeuralNetwork(double x1, double x2, double x3, double w14,
    double w15, double w24, double w25, double w34, double w35,
    double w46, double w56, double t1, double t2, double t3)
{
//Consturctor for the NeuralNetwork Class
    this.x1 = x1;
    this.x2 = x2;
    this.x3 = x3;
    this.W14 = w14;
    this.W15 = w15;
    this.W24 = w24;
    this.W25 = w25;
    this.W34 = w34;
    this.W35 = w35;
    this.W46 = w46;
    this.W56 = w56;
    this.O4 = t1;
    this.O5 = t2;
    this.O6 = t3;
}
public void Calculate_Input()
{
//calculation of input to the nural network
    Inode4 = x1 * W14 + x2 * W24 + x3 * W34 + O4;
    Inode5 = x1 * W15 + x2 * W25 + x3 * W35 + O5;
    Inode6 = W46 * Onode4 + W56 * Onode5 + O6;
}
public void calculate_Output()
{
//calculation of Output to the nural network

    Onode4 = (1 / (1 + Math.Exp(-Inode4)));
    Onode5 = (1 / (1 + Math.Exp(-Inode5)));
    Onode6 = (1 / (1 + Math.Exp(-Inode6)));
}
public void calculate_error()
{
//calculation of error propogation to the next layer of the nural network
    Error6 = Onode6 * (1 - Onode6) * (Tj - Onode6);
    Error5 = Onode5 * (1 - Onode5) * (Error6 * W56);
    Error4 = Onode4 * (1 - Onode4) * (Error6 * W46);
}
public void Calculate_weight()
{

```

```
//calculation of Weight of new nural network
W46 = W46 + (learnning_rate * Error6) * (Onode4);
W56 = W56 + (learnning_rate * Error6) * (Onode5);
W14 = W14 + (learnning_rate * Error4) * (x1);
W15 = W15 + (learnning_rate * Error5) * (x1);
W24 = W24 + (learnning_rate * Error4) * (x2);
W25 = W25 + (learnning_rate * Error5) * (x2);
W34 = W34 + (learnning_rate * Error4) * (x3);
W35 = W35 + (learnning_rate * Error5) * (x3);
}
public void Calculate_Bias()
{
//calculation of Bias of new nural network
O6 = O6 + learnning_rate * Error6;
O5 = O5 + learnning_rate * Error5;
O4 = O4 + learnning_rate * Error4;
}
}
```

### 3.3.2 Compilation /Running and Debugging the Solution

- The Compiling, Running, and Debugging can be done in the .NET IDE.

1. Open Visual studio 2005 editor
2. Open the program
3. Compile program using shift+F6
4. Run program using F5
5. Enter Threshold value of information gain
6. Program shows the output of resulting attributes with their gains.

### 3.4 Testing the Solution

The test cases have set of data as the input and New Weight, Bias are calculated and will display as an Output.

Untitled Page - Microsoft Internet Explorer

Address: <http://localhost:1513/AttributeRelevance/Default.aspx>

sofonic Search Games Play now! MTV Free TV Login

x1  x2  x3

w14  w15   
w24  w25   
w34  w35   
w46  w56   
THETA 4  THETA 5  THETA 6

Button

**output is:**

W46= -0.3  
W56= -0.2  
W14= 0.2  
W15= -0.3  
W24= 0.4  
W25= 0.1  
W34= -0.5  
W35= 0.2  
O6= 0.1  
O5= 0.2  
O4= -0.4

Done Local intranet 5:42 PM

### 4 Conclusions:

Neural Network using back propagation is gives us the trained neural network.



### **EXPERIMENT-1**

**Aim:** To perform various commands given in PL/SQL in Oracle 8.0(For brushing up.)

**Tools/ Apparatus:** PL/SQL programming tool.

**Procedure:**

- 1) Open SQL 8.0 tool and it will display the login window.
- 2) Enter the username = 'SCOTT' and password = 'TIGER'
- 3) After the successfully login try to write down the SQL queries in correct syntax and run them successfully.

## **EXPERIMENT-2**

**Aim:** To perform multi-dimensional data model using SQL queries. E.g. Star, snowflake and Fact constellation schemas

**Tools/ Apparatus:** Oracle 8i.

**Procedure:**

- 1) Open Oracle 8i tool and it will display the login window where one has to enter the login details.
- 2) Create the fact and the required dimensions tables as per the given business problem.  
There are three basic types of the multidimensional data model. They are Star, snowflake and Fact constellation schemas
- 3) As per the guidelines given in the theory draw all the three dimensional models.  
Use the following queries :

Define cube sales\_star[time, item, branch, location];

Dollars\_sold = sum(sales\_in\_dollars), units\_sold = count(\*)

Define dimension time as(time\_key, day, day\_of\_week, month, quarter, year).

Write the same queries for all other dimensions resp. Then run the following query:

Select s.time\_key, s.item\_key, s.branch\_key, s.location\_key,

Sum(s.number\_of\_units\_sold\*s.price), sum(s.number\_of\_units\_sold)

From time t, item I, branch b, location l, sales s,

Where s.time\_key = t.time\_key and s.item\_key = i.item\_key and s.branch\_key = b.branch\_key and s.location\_key = l.location\_key

Group by s.time\_key, s.item\_key, s.branch\_key, s.location\_key.

Run the queries to create the various multi-dimensional models.

### **EXPERIMENT-3**

**Aim:** To perform various OLAP operations such slice, dice, roll up, drill up, pivot etc.

**Tools/ Apparatus:** Oracle 8i, PL/SQL.

**Procedure:**

- 1) Open SQL 8.0 tool and login successfully.
- 2) Write down the queries to perform slice. In which one should keep one of the dimensions as constant and other dimensions should range from min to max.
- 3) Write down the queries to perform the dice. In which one has to keep two of the dimensions constant.
- 4) Write down the queries to perform roll-up by keeping one dimension constant and others should range from min to max. It is more like a generalization.
- 5) Write down the queries to perform roll-up by keeping one dimension constant and others should range from min to max. It is more like a specialization.

### **EXPERIMENT-4**

**Aim:** To perform the text mining on the given data warehouse.

**Tools/ Apparatus:** .NET Framework 2.0.

**Procedure:**

- 1) Given the database for mining the given text as part of the rule in DM.
- 2) Write a program in .NET Framework to find the text amongst the given transactions.
- 3) Try to find out the support and confidence level threshold for the given text by finding the no. of occurrences for the same.

### **EXPERIMENT-5**

**Aim:** To perform the correlation ship analysis between two attributes for the given data set.

**Tools/ Apparatus:** .NET Framework 2.0.

**Procedure:**

- 1) Given the database for mining the given dimensions as part of the rule in DM.
- 2) Write a program in .NET Framework to find the support s and confidence c for the given rule.
- 3) Find out the lift value for the given dimensions. Find the given value :  $\text{Lift}(A,B) = \frac{P(A \text{ Union } B)}{P(A)P(B)}$ .
- 4) If the value of life is  $< 1$  then the dimensions are negatively correlated, if it is  $> 1$  then they are positively co-correlated. If it is  $=1$  then there is no relationship between the two.

## **EXPERIMENT-6**

**Aim:** To perform the attribute relevance analysis on the given data.

**Tools/ Apparatus:** .NET Framework 2.0.

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rule in DM.
- 2) Write a program in .NET Framework to find the attribute which has most occurrences in the given databases.
- 3) Use the decision tree induction method to find the attribute which carries most amount of information.

### **EXPERIMENT-7**

**Aim:** To perform the information gain for a particular attribute in the given data.

**Tools/ Apparatus:** .NET Framework 2.0.

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rule in DM.
- 2) Define the classes for the given data set so that the data can be classified.
- 3) Write a program in .NET Framework to find the gain by using the formula for the given set of rules.
- 4) Use the attribute with the maximum gain to do the further classification or prediction.

### **EXPERIMENT-8**

**Aim:** To perform the experiment to predict the class label using the Bayesian classification.

**Tools/ Apparatus:** .NET Framework 2.0.

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rule in DM.
- 2) Define the classes for the given data set so that the data can be classified.
- 3) Write a program in .NET Framework to find the probability by using the formula for the set of classes for the given set of rules.
- 4) Find the class which has got the higher probability to belong to a given set of predicates and that will be the answer.



### **EXPERIMENT-9**

**Aim:** Write a program to find out a weight or bias updating using the back propagation in Neural network

**Tools/ Apparatus:** .NET Framework 2.0.

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rule in DM.
- 2) Draw the appropriate network for the given set of values/data to show it graphically.
- 3) Apply the formulas to calculate the input  $I_j$ , output  $O_j$ , error  $E_j$ , weight  $W_j$  and Bias  $\theta_j$  for the given layers and their nodes.
- 4) Find out the adjusted values for the error  $E_j$ , weight  $W_j$  and Bias  $\theta_j$  to make it more accurate.

Use the following formulas :

$I_j = \sum W_{ij} O_i + \theta_j$ ,  $O_j = 1 / (1 + \text{power}(e, -I_j))$ ,  $\text{Err}_j = O_j(1 - O_j)(E_j - O_j)$  (For output layer),  
 $\text{Err}_j = O_j(1 - O_j) \sum \text{Err}_k w_{jk}$ ,  $\Delta w_{ij} = (l) \text{Err}_j O_i$ ,  $\Delta \theta_j = (l) \text{Err}_j$ .

### **EXPERIMENT-10**

**Aim:** To perform various data mining algorithms on the give data base using Clementine.

**Tools/ Apparatus:** SPSS Clemenentine.

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rule in DM.
- 2) Use the IDE options to load the files in the editor.
- 3) Use the source options to draw the appropriate links with the other algorithmic operations using the editor options.
- 4) Represent the data in the appropriate format using the proper algorithm such as decision tree,neural network etc.

### **EXPERIMENT-11**

**Aim:** To perform data mining using Weca mining tool.

**Tools/ Apparatus:** Weca mining tool..

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rule in DM.
- 2) Use the IDE options to load the files in the editor.
- 3) Use the source options to draw the appropriate links with the other algorithmic operations using the editor options.
- 4) Represent the data in the appropriate format using the proper algorithm such as decision tree, neural network etc..

## **EXPERIMENT-12**

**Aim:** To perform data mining using Excel mining tool.

**Tools/ Apparatus:** Excel mining tool.

**Procedure:**

- 1) Given the database for mining the given business problem as part of the rules in DM.
- 2) Use the IDE options to load the files in the editor.
- 3) Use the source options to draw the appropriate links with the other algorithmic operations using the editor options.
- 4) Represent the data in the appropriate format using the proper algorithm such as decision tree, neural network etc..

## References

### Reference books:

- “Data warehousing in the real world”,
  - Sam Anahory & Dennis Murray
- “Information retrieval: Data structures and algorithms”,
  - W.B.Frakes and R. Baeza-Yates, Eds., Prentice-Hall, New Jersey,(1992)
- “Data mining techniques: For marketing, sales, customer support”,
  - Michael J A Berry, & Gordon Linoff.
- “Data mining”,
  - Pieter Adriaans & Dolf Zantinge, Pearson Education Asia(2001).
- “Statistics for Business and Economics”,
  - Jit S. Chandan, Vikas Publishing(1999).