SOMAIYA VIDYAVIHAR UNIVERSITY

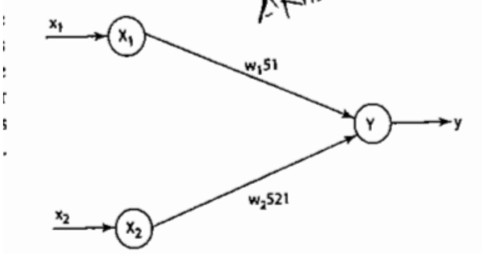
<table>
<tr><td colspan="4">Semester: July 2023 –October 2023
Maximum Marks: 100   Examination: ESE Examination   Duration:3 Hrs.</td></tr>
<tr><td colspan="2">Programme code: 01
Programme: B.Tech</td><td>Class: TY</td><td>Semester: V (SVU 2020)</td></tr>
<tr><td colspan="2">Name of the Constituent College:
K. J. Somaiya College of Engineering</td><td colspan="2">Name of the department: Computer Engg.</td></tr>
<tr><td>Course Code: 116U01E514</td><td colspan="3">Name of the Course: Soft Computing</td></tr>
<tr><td colspan="4">Instructions: 1)Draw neat diagrams 2) All questions are compulsory
3) Assume suitable data wherever necessary</td></tr>
</table>

Que. No.	Question	M M
Q1	Solve any **Four**	**20**
i)	Explain Biological Neuron and its working with the help of diagram. Diagram: 2M Description of parts: 3M	5
ii)	Implement ANDNOT McCulloch-Pitts Neuron with proper architecture and explanation.	5

x_1	x_2	y
0	0	0
0	1	0
1	0	1
1	1	0

The given function gives an output only when $x_1 = 1$ and $x_2 = 0$. The weights have to be decided only after the analysis. The net can be represented as shown in Figure 5.

ANDNOT i0010



Case 1: Assume that both weights w_1 and w_2 are excitatory, i.e.,

$$w_1 = w_2 = 1$$

Then for the four inputs calculate the net input using

$$y_{in} = x_1 w_1 + x_2 w_2$$

For inputs

$$(1,1),\quad y_{in} = 1\times 1 + 1\times 1 = 2$$
$$(1,0),\quad y_{in} = 1\times 1 + 0\times 1 = 1$$
$$(0,1),\quad y_{in} = 0\times 1 + 1\times 1 = 1$$
$$(0,0),\quad y_{in} = 0\times 1 + 0\times 1 = 0$$

From the calculated net inputs, it is not possible to fire the neuron for input (1, 0) only. Hence, these weights are not suitable. after Analyzing

Assume one weight as excitatory and the other as inhibitory, i.e.,

$$w_1 = 1,\ w_2 = -1$$

Now calculate the net input. For the inputs

$$(1,1),\quad y_{in} = 1\times 1 + 1\times -1 = 0$$
$$(1,0),\quad y_{in} = 1\times 1 + 0\times -1 = 1$$
$$(0,1),\quad y_{in} = 0\times 1 + 1\times -1 = -1$$
$$(0,0),\quad y_{in} = 0\times 1 + 0\times -1 = 0$$

From the calculated net inputs, now it is possible to fire the neuron for input (1, 0) only by fixing a threshold of 1, i.e., $\theta \geq 1$ for Y unit. Thus,

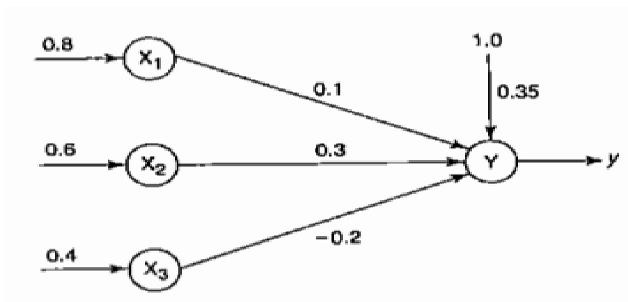
$$w_1 = 1;\ w_2 = -1;\ \theta \geq 1$$

Note: The value of θ is calculated using the following:

$$\theta \geq nw - p$$
$$\theta \geq 2\times 1 - 1 \quad \text{[for "}p\text{" inhibitory only magnitude considered]}$$
$$\theta \geq 1$$

Thus, the output of neuron Y can be written as

$$y = f(y_{in}) = \begin{cases} 1 & \text{if } y_{in} \geq 1 \\ 0 & \text{if } y_{in} < 1 \end{cases}$$

iii)	Explain ANN and give its advantages. ANN Definition : 2M Diagram : 1M Advantages : 2M	5
iv)	Explain Fuzzification methods. Atleast 3 Methods	5
v)	Obtain the output of the neuron Y using activation functions as: (i) Binary sigmoidal (ii) Bipolar sigmoidal  Net = 0.08+0.18-0.08+0.35 = 0.53 Sigmoidal = 0.629 Bipolar = 0.258	5
vi)	Consider a fuzzy set A defined on the interval x=[0, 10] of integers by the membership function. μA(x) = x / x+ 2 Find α-cut corresponding to α = 0.5. Ans = {2,3,4,5,6,7,8,9,10}	5

Que. No.	Question	Max. Marks
Q2 A	Solve the following	10
i)	The height of the building term set, h, comprises tall and low buildings. A building that is below 12.7 metres is definitely a low building and it is definitely not low if the height is greater than 31.88 meters. Contrariwise, a tall building should measure more than 12.44 meters and a building with a height greater than or equal to 34.49 meters is definitely a tall building. Derive the membership functions for the linguistic variables tall and low from the description given above. Tall – Ramp – 2.5 M (0,12.7,31.88) Low - Ramp – 2.5 M (12.44,34.49,infinity)	5
ii)	Describe the process of Simulated Annealing in Neural Network and its application. Description : 2M Diagram : 2M Application:1M	5
	OR	
Q2 A	Design a Hebb net to implement logical AND function (use bipolar inputs and targets).	10

Inputs				Weight changes			Weights		
x_1	x_2	b	y	Δw_1	Δw_2	Δb	w_1	w_2	b
							(0	0	0)
1	1	1	1	1	1	1	1	1	1
1	−1	1	−1	−1	1	−1	0	2	0
−1	1	1	−1	1	−1	−1	1	1	−1
−1	−1	1	−1	1	1	−1	2	2	−2

The separating line equation is given by

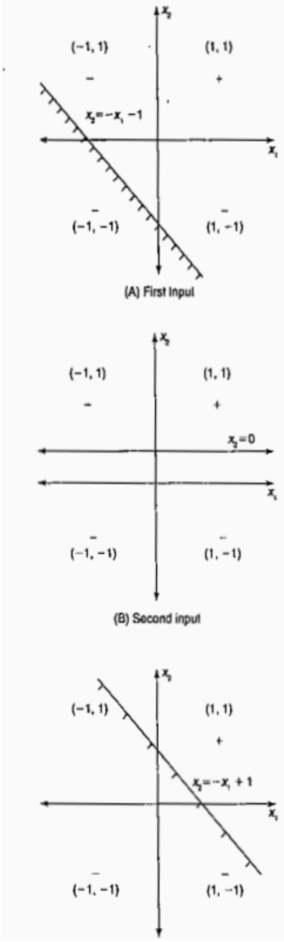
$$x_2 = \frac{-w_1}{w_2}x_1 - \frac{b}{w_2}$$

For all inputs, use the final weights obtained for each input to obtain the separating line. For the first input [1 1 1], the separating line is given by

$$x_2 = \frac{-1}{1}x_1 - \frac{1}{1} \Rightarrow x_2 = -x_1 - 1$$

Similarly, for the second input [1 −1 1], the separating line is

$$x_2 = \frac{-0}{2}x_1 - \frac{0}{2} \Rightarrow x_2 = 0$$



For the third input [−1 1 1], it is

$$x_2 = \frac{-1}{1}x_1 + \frac{1}{1} \Rightarrow x_2 = -x_1 + 1$$

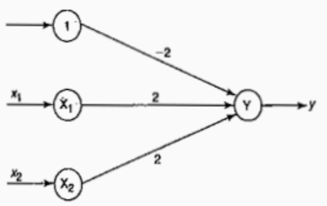
Finally, for the fourth input [−1 −1 1], the separating line is

$$x_2 = \frac{-2}{2}x_1 + \frac{2}{2} \Rightarrow x_2 = -x_1 + 1$$

The graphs for each of these separating lines obtained are shown in Figure 11. In this figure "+" mark is used for output "1" and "−" mark is used for output "−1." From Figure 11, it can be noticed that for the first input, the decision boundary differentiates only the first and fourth inputs, and not all negative responses are separated from positive responses. When the second input pattern is presented, the decision boundary separates (1, 1) from (1, −1) and (−1, −1) and not (−1, 1). But the boundary line is same for the both third and fourth training pairs. And, the decision boundary line obtained from these input training pairs separates the positive response region from the negative response region. Hence, the weights obtained from this are the final weights and are given as

$$w_1 = 2; \quad w_2 = 2; \quad b = -2$$

The network can be represented as shown in Figure 12.



Q 2 B	Solve any One	10
i)	Explain Error Back Propagation Training Algorithm using Bipolar Sigmoidal Activation Function. Give the Neural Network Architecture and weight updation formulas. Neural Network Architecture and vectors :2M Forward Pass : 4M Backward Pass : 4M	10
ii)	Using Hebb rule, find the weights required to perform following classifications. The vectors (1 -1 1 -1) and (1 1 1 -1) belong to class (target value +1) and vectors (-1 -1 1 1) and (1 1 -1 -1) do not belong to class (target value -1). Also using each of training vectors as input, test the response of network. Weight Calculation: 7M Testing : 3M	10

Que. No.	Question	Max. Marks
Q3	Solve any Two	20
i)	Construct an ART 1 network for clustering two input vectors with low vigilance parameter = 0.4 into two clusters. The two input vectors are [0 0 0 1] and [0 1 0 1].	10

Solution: The values assumed in this case are $\rho = 0.4$, $\alpha = 2$. Also it can be noted that $n = 4$ and $m = 3$. Hence,
Bottom-up-weights, $b_{ij}(0) = 1/(1+n) = 1/1+4 = 0.2$.
Top-down-weights $t_{ji}(0) = 1$.
For $i = 1$ to 4 and $j = 1$ to 2,

$$b_{ij} = \begin{bmatrix} 0.2 & 0.2 \\ 0.2 & 0.2 \\ 0.2 & 0.2 \\ 0.2 & 0.2 \end{bmatrix} \qquad t_{ji} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step 0: Initialize the parameters:

$$\rho = 0.4; \quad \alpha = 2$$

Initialize weights:

$$b_{ij}(0) = 0.2; \quad t_{ji}(0) = 1$$

Step 1: Start computation.

Step 2: For the first input vector [0 0 0 1], perform Steps 3–12.

Step 3: Set activations of all F_2 units to zero. Set activations of F_1(a) units to input vector $s = [0\ 0\ 0\ 1]$.

Step 4: Compute norm of s:

$$\|s\| = 0 + 0 + 0 + 1 = 1$$

Step 5: Compute activations for each node in the F_1 layer:

$$x = [0\ 0\ 0\ 1]$$

Step 6: Compute net input to each node in the F_2 layer:

$$y_j = \sum_{i=1}^{4} b_{ij} x_i$$

For $j = 1$ to 3,

$$y_1 = 0.2(0) + 0.2(0) + 0.2(0) + 0.2(1) = 0.2$$
$$y_2 = 0.2(0) + 0.2(0) + 0.2(0) + 0.2(1) = 0.2$$
$$y_3 = 0.2(0) + 0.2(0) + 0.2(0) + 0.2(1) = 0.2$$

Step 7: When reset is true, perform Steps 8–11.

Step 8: Since all the inputs pose same net input, there exists a tie and the unit with the smallest index is the winner, i.e., $J=1$.

Step 9: Recompute the F_1 activations (for $J=1$):

$$x_i = s_i t_{Ji}$$
$$x_1 = s_1 t_{J1} = [0\ 0\ 0\ 1][1\ 1\ 1\ 1]$$
$$x_1 = [0\ 0\ 0\ 1]$$

Step 10: Calculate norm of x:

$$\|x\| = 1$$

Step 11: Test for reset condition:

$$\frac{\|x\|}{\|s\|} = \frac{1}{1} = 1.0 \geq 0.4\ (\rho)$$

Hence reset is false. Proceed to Step 12.

Step 12: • Update bottom-up weights for $\alpha = 2$:

$$b_{ij}(\text{new}) = \frac{\alpha x_i}{\alpha - 1 + \|x\|} = \frac{2x_i}{2 - 1 + \|x\|} = \frac{2x_i}{1 + \|x\|}$$

$$b_{11} = \frac{2 \times 0}{1+1} = 0;$$
$$b_{21} = \frac{2 \times 0}{1+1} = 0;$$
$$b_{31} = \frac{2 \times 0}{1+1} = 0;$$
$$b_{41} = \frac{2 \times 1}{1+1} = \frac{2}{2} = 1$$

Therefore, the bottom-up weight matrix b_{ij} becomes

$$b_{ij} = \begin{bmatrix} 0 & 0.2 \\ 0 & 0.2 \\ 0 & 0.2 \\ 1 & 0.2 \end{bmatrix} \qquad t_{ji} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

For second Input [0101]

Step 2: For the second input vector [0 1 0 1], perform Steps 3–12.

Step 3: Set activations of all F_2 units to zero. Set activations of F_1(a) units to input vector $s = [0\ 1\ 0\ 1]$.

Step 4: Compute norm of s:

$$\|s\| = 0 + 1 + 0 + 1 = 2$$

Step 5: Compute activations of each node in the F_1 layer:

$$x = [0\ 1\ 0\ 1]$$

Step 6: Compute net input to each node in the F_2 layer:

$$y_j = \sum_{i=1}^{4} b_{ij} x_i$$

For $j = 1$ to 3,

$$y_1 = 0(0) + 0(1) + 0(0) + 1(1) = 1$$
$$y_2 = 0.2(0) + 0.2(1) + 0.2(0) + 0.2(1) = 0.4$$
$$y_3 = 0.2(0) + 0.2(1) + 0.2(0) + 0.2(1) = 0.4$$

Step 7: When reset is true, perform Steps 8–11.

Step 8: The unit with largest net input is the winner, i.e., $J = 1$.

Step 9: Recompute F_1 activations (for $J = 1$):

$$x_i = s_i t_{Ji} = [0\ 1\ 0\ 1][0\ 0\ 0\ 1] = [0\ 0\ 0\ 1]$$

Step 10: Calculate norm of x:

$$\|x\| = 0 + 0 + 0 + 1 = 1$$

Step 11: Test for reset condition:

$$\frac{\|x\|}{\|s\|} = \frac{1}{2} = 0.5 \geq 0.4\ (\rho)$$

Hence reset is false. Proceed to Step 12.

Step 12: • Update bottom-up weights for $\alpha = 2$:

$$b_{ij}(\text{new}) = \frac{2x_i}{1 + \|x\|}$$

$$b_{11} = \frac{2 \times 0}{1+1} = 0;$$
$$b_{21} = \frac{2 \times 0}{1+1} = 0;$$
$$b_{31} = \frac{2 \times 0}{1+1} = 0;$$
$$b_{41} = \frac{2 \times 1}{1+1} = \frac{2}{2} = 1$$

Therefore, the bottom-up weight matrix b_{ij} becomes

$$b_{ij} = \begin{bmatrix} 0 & 0.2 \\ 0 & 0.2 \\ 0 & 0.2 \\ 1 & 0.2 \end{bmatrix}$$

• Update the top-down weights:

$$t_{Ji}(\text{new}) = x_i$$

$$t_{ji} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

First Input : 5M
Second Input: 5M

<table>
<tr><td>ii)</td><td>Explain Kohonen Self Organizing Feature Map in detail with suitable diagrams. Also give its training algorithm.

Network Architecture : 2M
Explanation: 4M
Training algorithm: 4M</td><td>10</td></tr>
<tr><td>iii)</td><td>With the help of a block diagram explain the working of a Fuzzy Logic Air Conditioner Controller. (Assume suitable input and output)

Input: 1M
Output: 1M
Fuzzification: 3M
Fuzzy Rules: 2M
Defuzzification: 3M</td><td>10</td></tr>
</table>

<table>
<tr><th>Que. No.</th><th>Question</th><th>Max. Marks</th></tr>
<tr><td>Q4</td><td>Solve any Two</td><td>20</td></tr>
<tr><td>i)</td><td>Train a Heteroassociative Memory Network to store the input vectors s = (s1, s2, s3, s4) to the output vectors t = (t1, t2) as given in below Table. Also test the performance of the network using its training input as testing input.

<table>
<tr><th>Input and targets</th><th>s_1</th><th>s_2</th><th>s_3</th><th>s_4</th><th>t_1</th><th>t_2</th></tr>
<tr><td>1st</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>2nd</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>3rd</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>4th</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
</table></td><td>10</td></tr>
</table>

For $p = 1$ to 4,

$$W = \sum_{p=1}^{4} s^{T}(p)\, t(p)$$

$$= s^{T}(1)t(1) + s^{T}(2)t(2) + s^{T}(3)t(3) + s^{T}(4)t(4)$$

$$= \begin{bmatrix}1\\0\\0\\0\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix} + \begin{bmatrix}1\\1\\0\\0\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix} + \begin{bmatrix}0\\0\\0\\1\end{bmatrix}\begin{bmatrix}1 & 0\end{bmatrix} + \begin{bmatrix}0\\0\\1\\1\end{bmatrix}\begin{bmatrix}1 & 0\end{bmatrix}$$

$$= \begin{bmatrix}0 & 1\\0 & 0\\0 & 0\\0 & 0\end{bmatrix} + \begin{bmatrix}0 & 1\\0 & 1\\0 & 0\\0 & 0\end{bmatrix} + \begin{bmatrix}0 & 0\\0 & 0\\0 & 0\\1 & 0\end{bmatrix} + \begin{bmatrix}0 & 0\\0 & 0\\1 & 0\\1 & 0\end{bmatrix}$$

$$W = \begin{bmatrix}0 & 2\\0 & 1\\1 & 0\\2 & 0\end{bmatrix}$$

This is the final weight of the matrix.

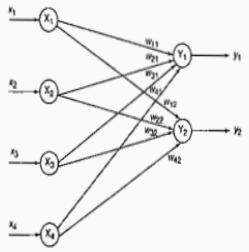


Figure 2 Network architecture.

Testing the Network

Method I

The testing algorithm for a heteroassociative memory network is used to test the performance of the net. The weight obtained from training algorithm is the initial weight in testing algorithm.

For 1st testing input

Step 0: Initialize the weights:

$$W = \begin{bmatrix}w_{11} & w_{12}\\w_{21} & w_{22}\\w_{31} & w_{32}\\w_{41} & w_{42}\end{bmatrix} = \begin{bmatrix}0 & 2\\0 & 1\\1 & 0\\2 & 0\end{bmatrix}$$

Step 1: Performs Steps 2–4 for each testing input–output vector.

Step 2: Set the activations, $x = [1\ 0\ 0\ 0]$.

Step 3: Compute the net input, $n = 4$, $m = 2$. For $i = 1$ to 4 and $j = 1$ to 2:

$$y_{inj} = \sum_{i=1}^{n} x_i w_{ij}$$

$$y_{in1} = \sum_{i=1}^{n} x_i w_{i1}$$

$$= x_1 w_{11} + x_2 w_{21} + x_3 w_{31} + x_4 w_{41}$$

$$= 1 \times 0 + 0 \times 0 + 0 \times 1 + 0 \times 2 = 0$$

$$y_{in2} = \sum_{i=1}^{n} x_i w_{i2}$$

$$= x_1 w_{12} + x_2 w_{22} + x_3 w_{32} + x_4 w_{42}$$

$$= 1 \times 2 + 0 \times 1 + 0 \times 0 + 0 \times 0 = 2$$

Step 4: Applying activation over the net input to calculate the output.

$$y_1 = f(y_{in1}) = f(0) = 0$$
$$y_2 = f(y_{in2}) = f(2) = 1$$

The output is [0, 1] which is correct response for first input pattern.

For 2nd testing input

Set the activation $x = [1\ 1\ 0\ 0]$. Computing the net input, we obtain

$$y_{in1} = x_1 w_{11} + x_2 w_{21} + x_3 w_{31} + x_4 w_{41}$$
$$= 0 + 0 + 0 + 0 = 0$$
$$y_{in2} = x_1 w_{12} + x_2 w_{22} + x_3 w_{32} + x_4 w_{42}$$
$$= 2 + 1 + 0 + 0 = 3$$

Compute the output by applying activations over net input,

$$y_1 = f(y_{in1}) = f(0) = 0$$
$$y_2 = f(y_{in2}) = f(3) = 1$$

The output is [0, 1] which is correct response for second input pattern.

For 3rd testing input

Set the activation $x = [0\ 0\ 0\ 1]$. Computing net input, we obtain

$$y_{in1} = x_1 w_{11} + x_2 w_{21} + x_3 w_{31} + x_4 w_{41}$$
$$= 0 + 0 + 0 + 2 = 2$$
$$y_{in2} = x_1 w_{12} + x_2 w_{22} + x_3 w_{32} + x_4 w_{42}$$
$$= 0 + 0 + 0 + 0 = 0$$

Calculate output of the network,

$$y_1 = f(y_{in1}) = f(2) = 1$$
$$y_2 = f(y_{in2}) = f(0) = 0$$

The output is [1 0] which is correct response for third testing input pattern.

For 4th testing input

Set the activation $x = [0\ 0\ 1\ 1]$. Calculating the net input, we obtain

$$y_{in1} = x_1 w_{11} + x_2 w_{21} + x_3 w_{31} + x_4 w_{41}$$
$$= 0 + 0 + 1 + 2 = 3$$
$$y_{in2} = x_1 w_{12} + x_2 w_{22} + x_3 w_{32} + x_4 w_{42}$$
$$= 0 + 0 + 0 + 0 = 0$$

Calculate the output of the network,

$$y_1 = f(y_{in1}) = f(3) = 1$$
$$y_2 = f(y_{in2}) = f(0) = 0$$

The output is [1 0] which is correct response for fourth testing input pattern.

Training : 5M
Testing : 5M

ii) Train an Autoassociative Network for input vector [-I 1 1 - l]. Test the Autoassociative Network with one missing, one mistake, two missing and two mistake entries in test vector. — 10

Training:2M
one missing:2M
one mistake:2M
two missing:2M
two mistake: 2M

iii) Consider a rule base for a 2-Inputs-1-Output Mamdani controller: — 10

The input variables, λ1 and λ2 and η is the output variable. The variables belong to the universes of discourse X1, X2 and Y respectively, such that λ1 ∈ X1 = [-1,1], λ2 ∈ X2 = [-1,1] and η ∈ Y = [-1,1]. The rule base for the controller is given as:

λ2 \ λ1	***Neg***	***Zero***	***Pos***
Neg	Neg		Zero
Zero		Zero	
Pos	Zero		Pos

The fuzzy partitions of the sets X1, X2 and Y are given as :

	$$\chi_{Neg}(x) = \begin{cases} 0 & if\ x \geq 0, \\ 1 & if\ x = -1\ , \\ \alpha_1 + \beta_1\, x & otherwise. \end{cases}$$ $$\chi_{Pos}(x) = \begin{cases} 0 & if\ x \leq 0, \\ 1 & if\ x = 1 \\ \alpha_2 + \beta_2\, x & otherwise. \end{cases}$$ $$\chi_{zero}(x) = \max\left(\min(-(x+1), 1-x), 0\right)$$ (α and β are constatnts.) Compute the control values η for (λ1, λ2) = (-1, 0) Show all the four stages of the computation: fuzzification, inference, composition and defuzzification. Input: {-1,0}, Fuzzification: λ1 is Neg and λ2 is Zero, inference Rules Fired: If λ1 is Neg and λ2 is Zero 5 composition: η is Zero defuzzification η: η is 0 2 M each	

Que. No.	Question	Max. Marks
Q5	Write Short notes on any **four**	20
i)	ANFIS Explanation: 3M Diagram: 2M	5
ii)	Hopfield Network Explanation: 3M Diagram: 2M	5
iii)	Discrete BAM Explanation: 3M Diagram: 2M	5
iv)	Neocognitron Explanation: 3M Diagram: 2M	5
v)	Defuzzification Explanation: 3M Diagram: 2M	5
vi)	Full Counter Propagation Network Explanation: 3M Diagram: 2M	5