Mod-3

The Perceptron Learning Algorithm

The Perceptron Algorithm

- **Initialisation**
  - set all of the weights w_{ij} to small (positive and negative) random numbers
- **Training**
  - for T iterations or until all the outputs are correct:
    - for each input vector:
      - compute the activation of each neuron j using activation function g:

$$y_j = g\left(\sum_{i=0}^{m} w_{ij}x_i\right) = \begin{cases} 1 & \text{if } \sum_{i=0}^{m} w_{ij}x_i > 0 \\ 0 & \text{if } \sum_{i=0}^{m} w_{ij}x_i \leq 0 \end{cases} \qquad (3.4)$$

      - update each of the weights individually using:

$$w_{ij} \leftarrow w_{ij} - \eta(y_j - t_j) \cdot x_i \qquad (3.5)$$

- **Recall**
  - compute the activation of each neuron j using:

$$y_j = g\left(\sum_{i=0}^{m} w_{ij}x_i\right) = \begin{cases} 1 & \text{if } w_{ij}x_i > 0 \\ 0 & \text{if } w_{ij}x_i \leq 0 \end{cases} \qquad (3.6)$$

- Consider the truth table for AND function
- The M–P neuron has no particular training algorithm
- In M-Pneuron, only analysis is being performed.
- Hence, assume the weights be w1 = 1 and w2 = 1.

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

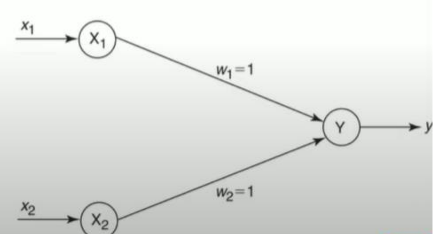
$(1, 1),\ y_{in} = x_1w_1 + x_2w_2 = 1 \times 1 + 1 \times 1 = 2$

$(1, 0),\ y_{in} = x_1w_1 + x_2w_2 = 1 \times 1 + 0 \times 1 = 1$

$(0, 1),\ y_{in} = x_1w_1 + x_2w_2 = 0 \times 1 + 1 \times 1 = 1$

$(0, 0),\ y_{in} = x_1w_1 + x_2w_2 = 0 \times 1 + 0 \times 1 = 0$

Threshold value is set equal to 2 (θ = 2).



- This can also be obtained by

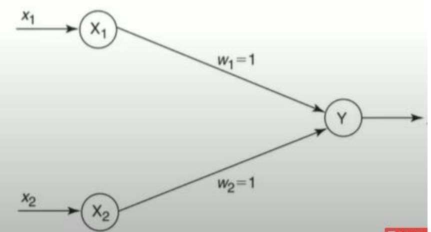
$$\theta \geq nw - p$$

- Here, n = 2, w = 1 (excitatory weights) and p = 0 (no inhibitory weights).
- Substituting these values in the above-mentioned equation we get $\theta \geq 2 \times 1 - 0 \Rightarrow \theta \geq 2$
- Thus, the output of neuron Y can be written

$$y = f(y_{in}) = \begin{cases} 1 & \text{if} \quad y_{in} \geq 2 \\ 0 & \text{if} \quad y_{in} < 2 \end{cases}$$

MORE VIDEOS

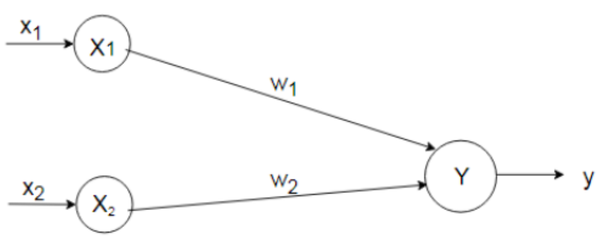
x_1	x_2	y
1	1	1
1	0	0
0	1	0
0	0	0



Problem-6

- Use McCulloch-Pitts Neuron to implement AND NOT function (take binary data representation).

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



Problem-6

Case 1: Assume both the weights as excitatory, i.e., $w_1 = w_2 = 1$,

$\theta \geq nw - p$
$\theta \geq 2 \times 1 - 0 \geq 2$

The net input,

(i) $(1,1) - y_{in} = x_1 w_1 + x_2 w_2 = 1 \times 1 + 1 \times 1 = 2$

(ii) $(1,0) - y_{in} = x_1 w_1 + x_2 w_2 = 1 \times 1 + 0 \times 1 = 1$

(iii) $(0,1) - y_{in} = x_1 w_1 + x_2 w_2 = 1 \times 0 + 1 \times 1 = 1$

Problem-6

(iv) $(0,0) - y_{in} = x_1w_1 + x_2w_2 = 0 \times 1 + 0 \times 1 = 0$

From the calculated net input, it is possible to fire the neuron with input $(1,0)$ only.

Case 2:

Assume one weight as excitatory and another one as inhibitory,

i.e., $w_1 = 1, w_2 = 1$

The net input,

(i) $(1,1) - y_{in} = x_1w_1 + x_2w_2 = 1 \times 1 + 1 \times (-1) = 0$

(ii) $(1,0) - y_{in} = x_1w_1 + x_2w_2 = 1 \times 1 + 0 \times (-1) = 1$

(iii) $(0,1) - y_{in} = x_1w_1 + x_2w_2 = 1 \times 0 + 1 \times (-1) = -1$

(iv) $(0,0) - y_{in} = x_1w_1 + x_2w_2 = 0 \times 1 + 0 \times (-1) = 0$

From the net inputs now it is possible to conclude that the neuron will only fire with input $(1,0)$ by fixing the threshold $\theta \geq 1$.

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

The value of θ is calculated as,

$$\theta \geq nw - p$$
$$\theta \geq 2 \times 1 - 1$$
$$\theta \geq 1$$

The output of the neuron Y can be written as,

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

Ms.Sujata Pathak, IT, KJSCE