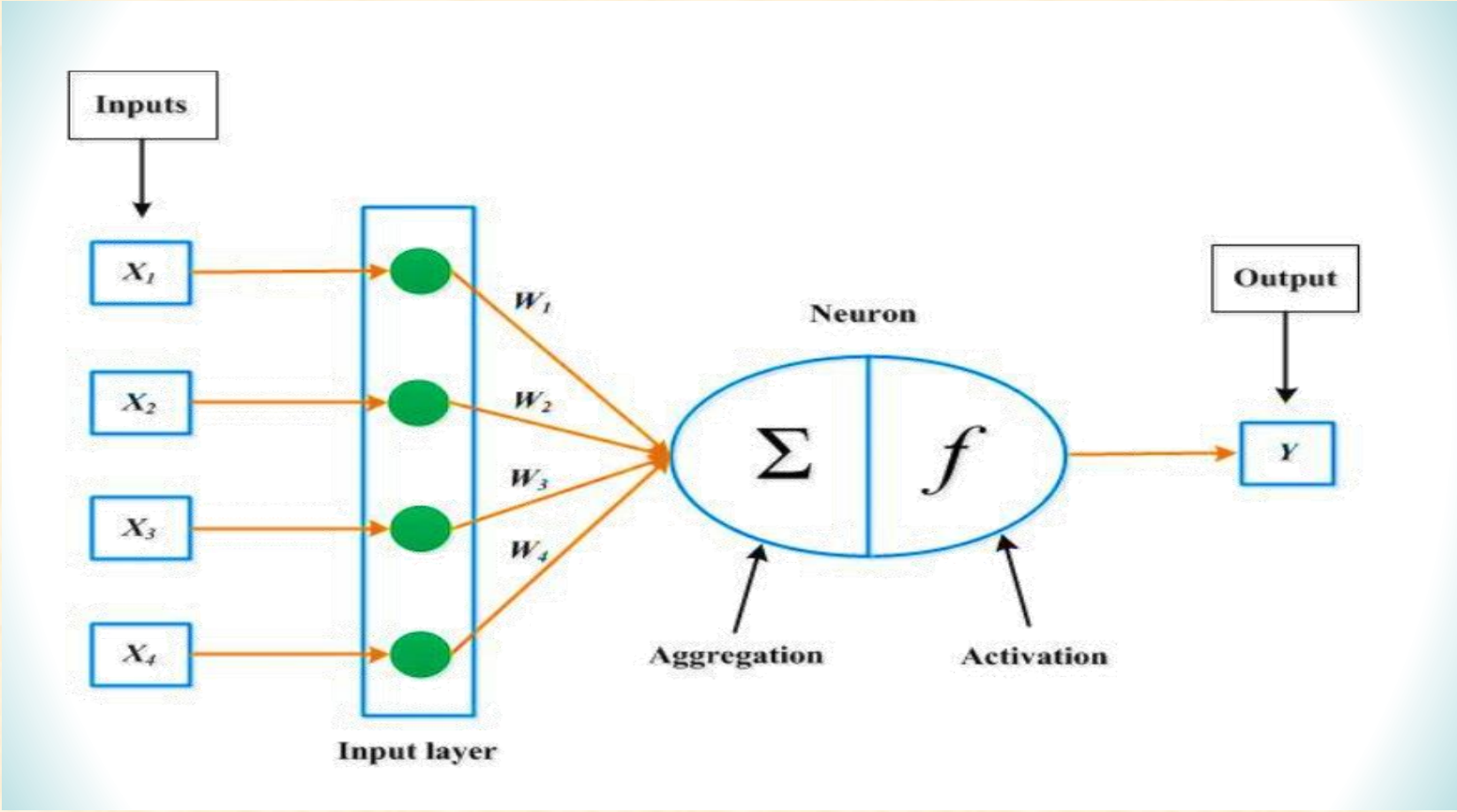
Single Layer Perceptron (SLP)



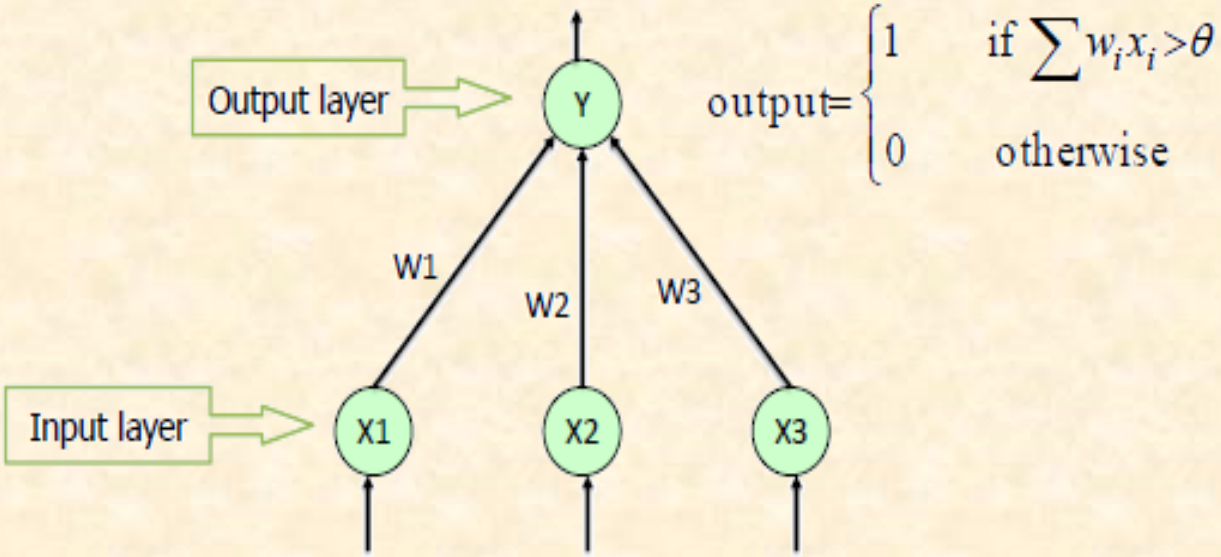
Preview

- ***Perceptron***
- ***Learning Algorithm: Training Perceptron***
- ***Perceptron Learning Algorithm***

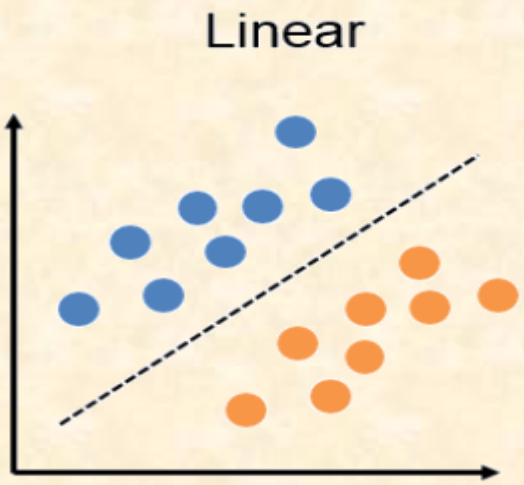
Perceptron

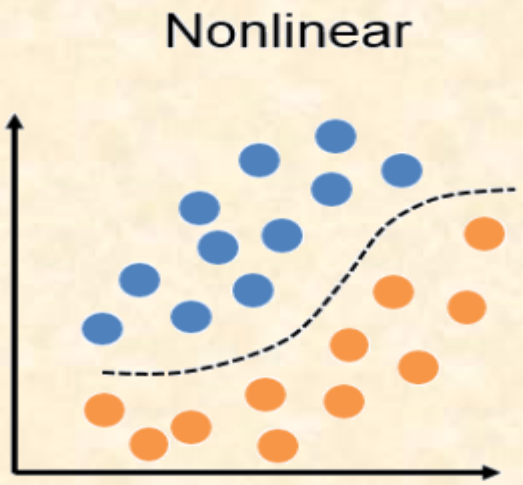
- ***A single layer perceptron (SLP) is a feed-forward network based on a threshold transfer function. SLP is the simplest type of artificial neural networks and can only classify linearly separable cases with a binary target (1, 0).***

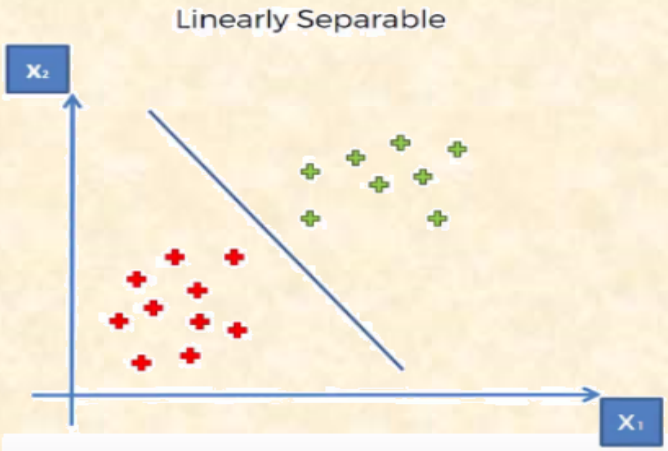
Single Layer Perceptron

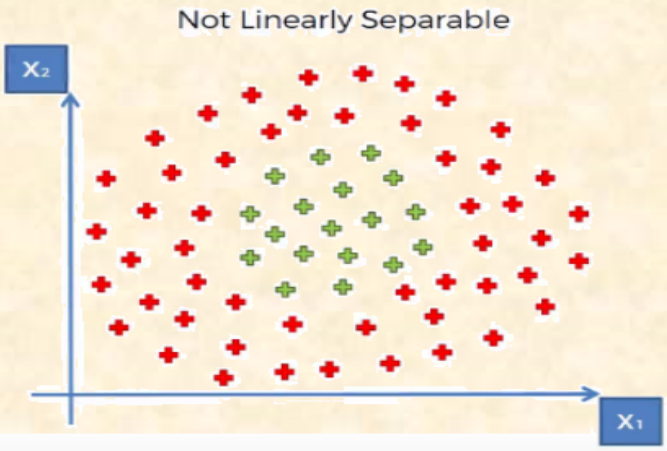


Continue…

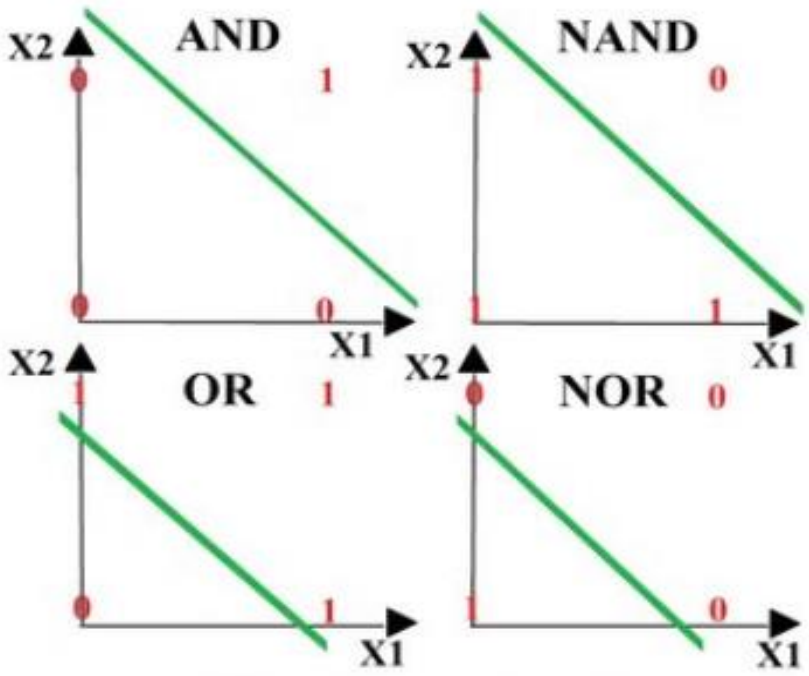

Linear


Nonlinear

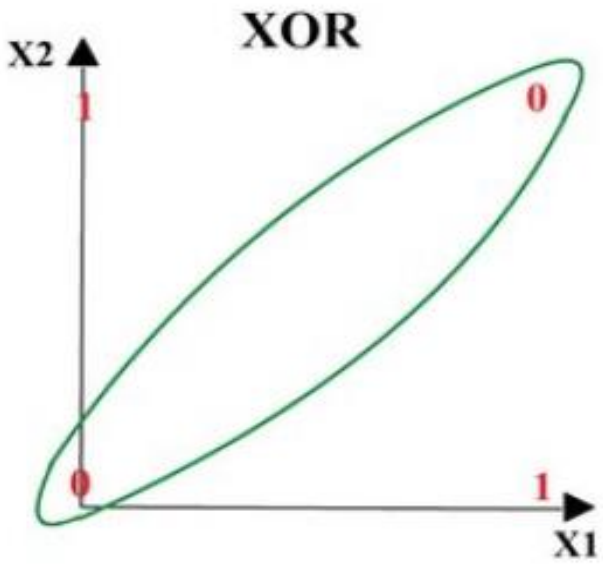

Linearly Separable
X2
X1


Not Linearly Separable
X2
X1

Continue…



(a) Linearly separable problems: AND, NAND, OR, NOR operators



(b) Non-linearly separable problems: XOR operator

Continue…

- ***The single layer perceptron does not have a prior knowledge, so the initial weights are assigned randomly. SLP sums all the weighted inputs and if the sum is above the threshold (some predetermined value), SLP is said to be activated (output=1).***

Output

$$w_1 x_1 + w_2 x_2 + \ldots + w_n x_n > \theta \Rightarrow 1$$

$$w_1 x_1 + w_2 x_2 + \ldots + w_n x_n \leq \theta \Rightarrow 0$$

Learning Algorithm: Training Perceptron

The training of Perceptron is a supervised learning algorithm where weights are adjusted to minimize error when ever the output does not match the desired output.

- If the output is correct then no adjustment of weights is done.

  i.e. $w_{ij}^{K+1} = w_{ij}^{K}$

- If the output is **1** but should have been **0** then the weights are decreased on the active input link

  i.e. $w_{ij}^{K+1} = w_{ij}^{K} - \alpha E x_i$

- If the output is **0** but should have been **1** then the weights are increased on the active input link

  i.e. $w_{ij}^{K+1} = w_{ij}^{K} + \alpha E x_i$

Continue…

Where

w_{ij}^{K+1} is the new adjusted weight, w_{ij}^{K} is the old weight

x_i is the input and α is the learning rate parameter.

α small leads to slow and α large leads to fast learning.

Perceptron Learning Algorithm

The algorithm is illustrated step-by-step.

- **Step 1 :**

  Create a peceptron with $(n+1)$ input neurons $x_0, x_1, \ldots\ldots, x_n$,

  where $x_0 = 1$ is the bias input.

  Let o be the output neuron.

- **Step 2 :**

  Initialize weight $W = (w_0, w_1, \ldots\ldots, w_n)$ to random weights.

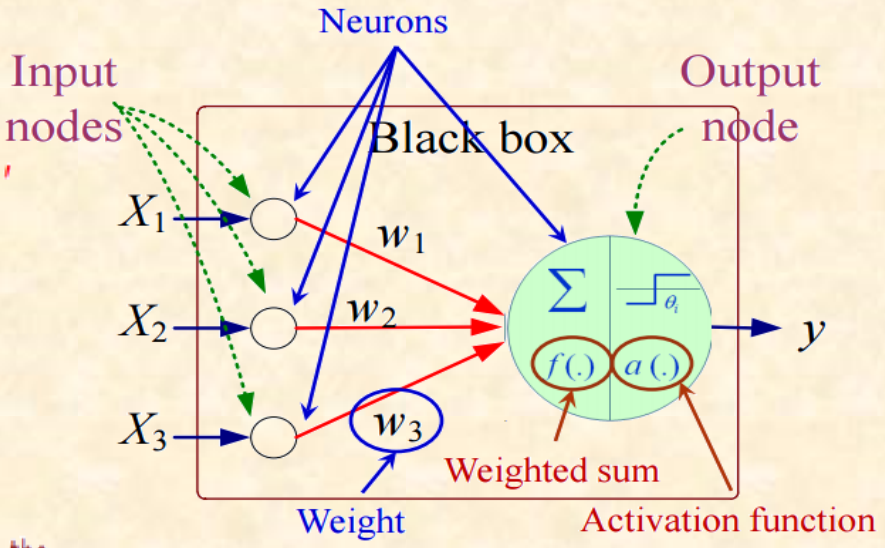
- **Step 3 :**

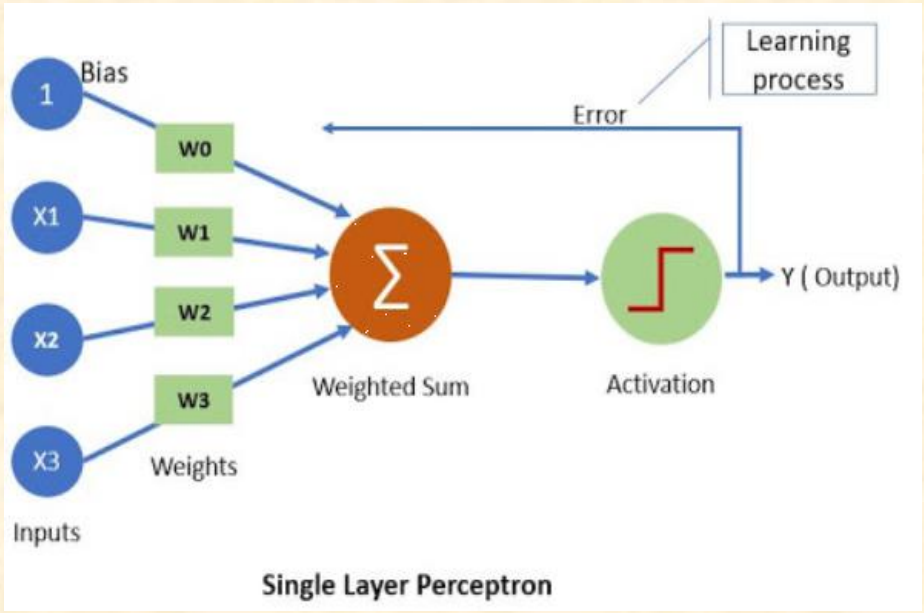
  Iterate through the input patterns X_j of the training set using the weight set; ie compute the weighted sum of inputs $net_j = \sum_{i=1}^{n} x_i w_i$ for each input pattern j.

- **Step 4 :**

  Compute the output y_j

$$y_j = f(net_j) = \begin{cases} 1 & \text{if } net_j > \theta \\ 0 & \text{if } net_j \le \theta \end{cases} \quad \text{where} \quad net_j = \sum_{i=1}^{n} x_i w_{ij}$$





Continue…

- **Step 5 :**

  Compare the computed output y_j with the target output y_j for each input pattern j .

  Error signal= $\triangle i = E = (O_{desired} - O_{actual})$

  $E = (d - y)$

  If all the input patterns have been classified correctly, then output (read) the weights and exit.

- **Step 6 :**

  Otherwise, update the weights as given below :

  If the computed outputs y_j is **1** but should have been **0**,

  Then $w_i = w_i - \alpha x_i$, $i = 0, 1, 2, \ldots, n$

  If the computed outputs y_j is **0** but should have been **1**,

  Then $w_i = w_i + \alpha x_i$, $i = 0, 1, 2, \ldots, n$

  where α is the learning parameter and is constant.

- **Step 7 :**

  goto step 3

- **END**

Question

Q1:

Explain in detail the steps of Perceptron Learning Algorithm.

Q2 :

Choose the an Architectural type suitable for the perceptron network ?

a- Single Layer Feed-forward Network
b- Multi Layer Feed-forward Network
c- Recurrent Network

Question for discussion in the next lecture

Q: In phase training of Perceptron, it is a supervised learning algorithm, where weights are adjusted to minimize error when ever the output does not match the desired output**.**

Prove the following phrase *"If the output is 0 but should have been 1 then the weights are increased on the active input link".*

Thank You

Any Question?

Dear students,
Please, contact via Google Classroom