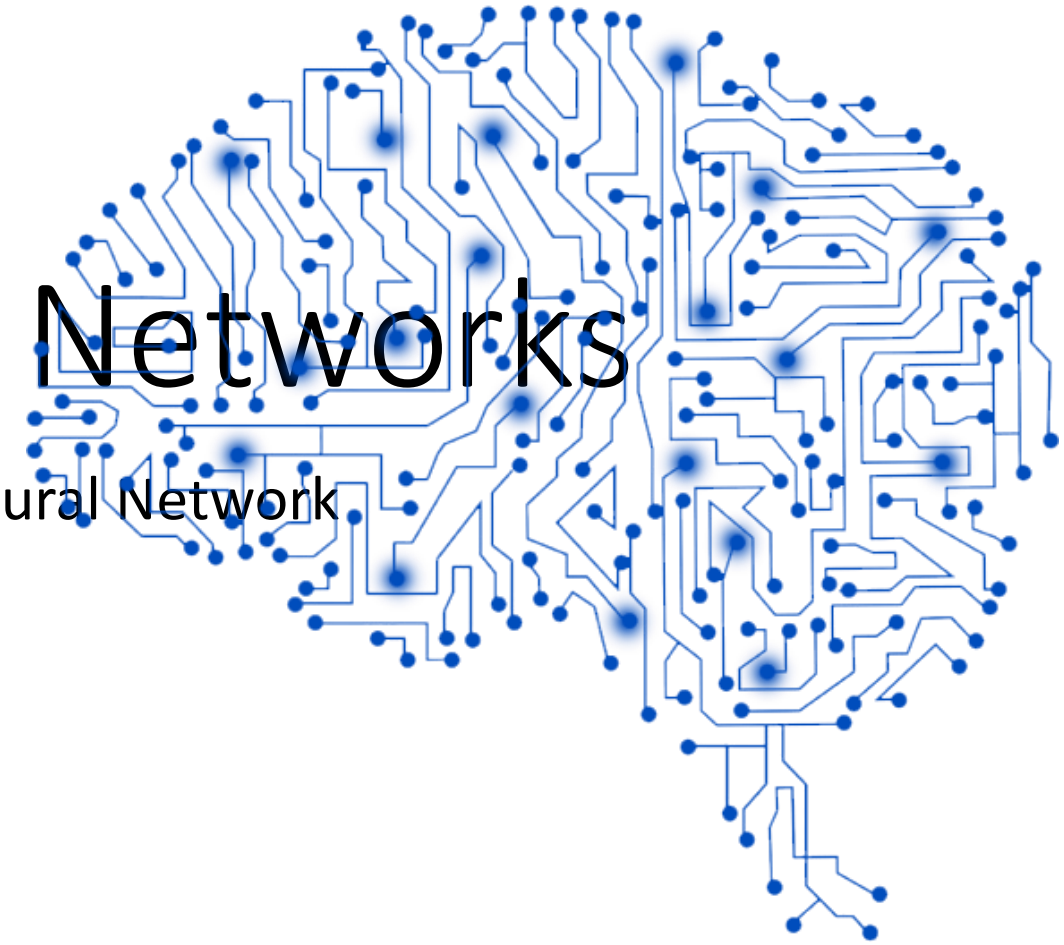
Artificial Neural Networks

Introduction to Artificial Neural Network



Arie Satia Dharma, S.T., M.Kom
Dosen Informatika

Topics

- ❑Perceptron Learning Rule
- ❑Perceptron Learning Rule Example

Perceptron Learning Rule

- perceptron is an early version of modern neural networks.
- Understanding the logic behind the classical single layer perceptron will help you to understand the idea behind deep learning as well.
- Because you can image deep neural networks as combination of nested perceptrons.
- You can also imagine single layer perceptron as legacy neural networks.

Perceptron Learning Rule Algorithm :

1.Initiation early weight, bias, learning rate and threshold.

2.Set input units with the input vector $X_i = S_i$ for i = 1 to n and target y = t.

3.Calculate $y_{in} = b + \sum_i x_i w_i$

4.Substitution y to activation unit :

$$y = \begin{cases} 1 \ , y_{in} > 0 \\ 0, -\theta \leq y_{in} \leq \theta \\ -1, y_{in} < \theta \end{cases}$$

5.Compare with target, if not same then update weight

6.Repeat steps 3-5

Perceptron Learning Example-1

- Contoh: Data fungsi logika “AND”

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	0

Perceptron Algorithm :

1. inisiasi lr dan threshold
2. set input dan target
3. hitung target prediksi
4. bandingkan target aktual
5. perbaiki bobot

Perceptron Learning Example-1

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1

- Epoch 1 Step 1 :

  Set weight randomly, $w_1 = 0.9$ and $w_2 = 0.9$. Bias (b) = 0

  Set learning rate value between 0 and 1, $\alpha = 0.5$ and threshold = 0.5.

- Step 2-5 :
  - 1st Instance:

    $Y_{1in} = 0 + (x1 * w1 + x2 * w2) = 0 + (0 * 0.9 + 0 * 0.9) = 0$

    Y_{1in} < threshold => $Y_{1in} = 0$ (sesuai dengan target aktual)
  - 2nd Instance :

    $Y_{2in} = 0 + (x1 * w1 + x2 * w2) = 0 + (0 * 0.9 + 1 * 0.9) = 0.9$

    Y_{2in} >= threshold => $Y_{2in} = 1$ (tidak sesuai dengan target aktual, update)

Perceptron Learning Example-1

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1

- Step 2-5 :
  - 2nd Instance :

    ε = actual – prediction = 0 – 1 = -1

    $w_1 = w_1 + \alpha * \varepsilon = 0.9 + 0.5 * (-1) = 0.9 - 0.5 = 0.4$

    $w_2 = w_2 + \alpha * \varepsilon = 0.9 + 0.5 * (-1) = 0.9 - 0.5 = 0.4$
  - 3rd Instance :

    $Y_{3in} = 0 + (x1 * w1 + x2 * w2) = 0 + (1 * 0.4 + 0 * 0.4) = 0.4$

    Y_{3in} < threshold => $Y_{3in} = 0$ (sesuai dengan target aktual)
  - 4th Instance :

    $Y_{4in} = 0 + (x1 * w1 + x2 * w2) = 0 + (1 * 0.4 + 1 * 0.4) = 0.8$

Perceptron Learning Example-1

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1

- Step 2-5 :
  - 4_{th} Instance :

    $Y_{4in} = 0 + (x1 * w1 + x2 * w2) = 0 + (1 * 0.4 + 1 * 0.4) = 0.8$

    Y_{4in} < threshold => $Y_{4in} = 0$ (sesuai dengan target aktual)

Epoch 2

- Step 1 :

  Set weight based previously, $w_1 = 0.4$ and $w_2 = 0.4$. Bias (b) = 0

  Set learning rate value between 0 and 1, $\alpha = 0.5$ and threshold = 0.5.

- Step 2-5 :
  - 1st Instance:

Perceptron Learning Example-1

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1

- Step 2-5 :
  - 1st Instance:

    $Y_{1in} = 0 + (x1 * w1 + x2 * w2) = 0 + (0 * 0.4 + 0 * 0.4) = 0$

    Y_{1in} < threshold => $Y_{1in} = 0$ (sesuai dengan target aktual)
  - 2nd Instance:

    $Y_{2in} = 0 + (x1 * w1 + x2 * w2) = 0 + (0 * 0.4 + 1 * 0.4) = 0.4$

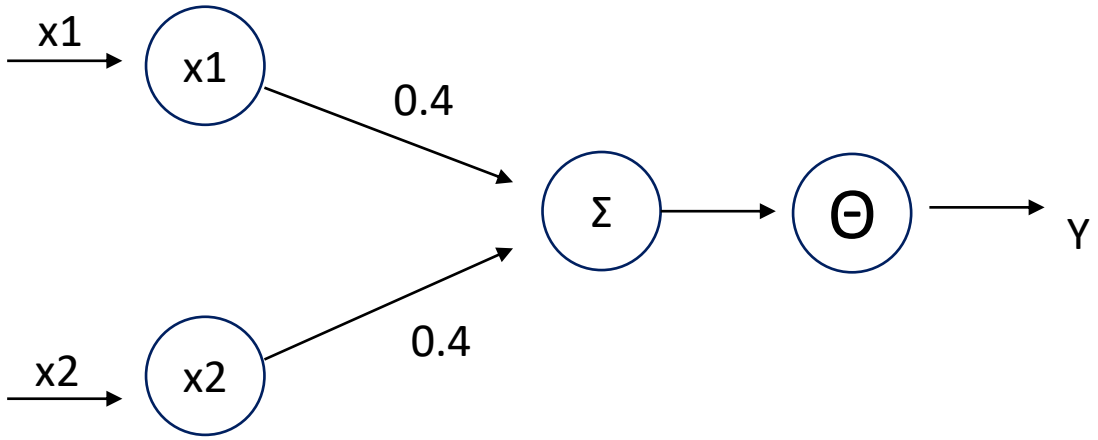
    Y_{2in} >= threshold => $Y_{2in} = 0$ (sesuai dengan target aktual)
  - 3rd Instance & 4th Instance : sudah dilakukan sebelumnya

Final weight matrix is [0.4 0.4]

Perceptron Learning Example-1

X1	X2	Y
0	0	0
0	1	0
1	0	0
1	1	1

- The network with the final weights :



$2x_1 + 2x_2 = Y$

Perceptron Learning Example-2

- Diketahui data fungsi logika “OR”:

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	1