

Mata Kuliah : Kecerdasan Buatan
Kelas : TI-2H
NIM : 2041720074
Nama : Mochammad Hairullah

Praktikum

1. Lakukan percobaan dengan code program diatas

```
def predict(row, weights):
    activation = weights[0]
    for i in range(len(row)-1):
        activation += weights[i + 1]*row[i]
    return 1.0 if activation >= 0.0 else 0.0

import numpy as np
arr_error=[]
def train_weights(train, l_rate, n_epoch):
    weights = [0.0 for i in range(len(train[0]))]
    for epoch in range(n_epoch):
        sum_error = 0.0
        for row in train:
            prediction = predict(row, weights)
            error = row[-1] - prediction
            sum_error += error**2
            weights[0] = weights[0] + l_rate * error
            for i in range(len(row)-1):
                weights[i + 1] = weights[i + 1] + l_rate * error * row[i]
        print('>epoch=%d, lrate=%.3f, error=%.3f' % (epoch, l_rate,
sum_error))
        arr_error.append(sum_error)
    return weights, arr_error

#Contoh Kasus OR
dataset = [[1,1,1],
           [1,0,1],
           [0,1,1],
           [0,0,0]]

weights = [0, 0, 0] #bias,w1,w2
l_rate = 1
n_epoch = 7
weights, arr_error = train_weights(dataset, l_rate, n_epoch)
print(weights)

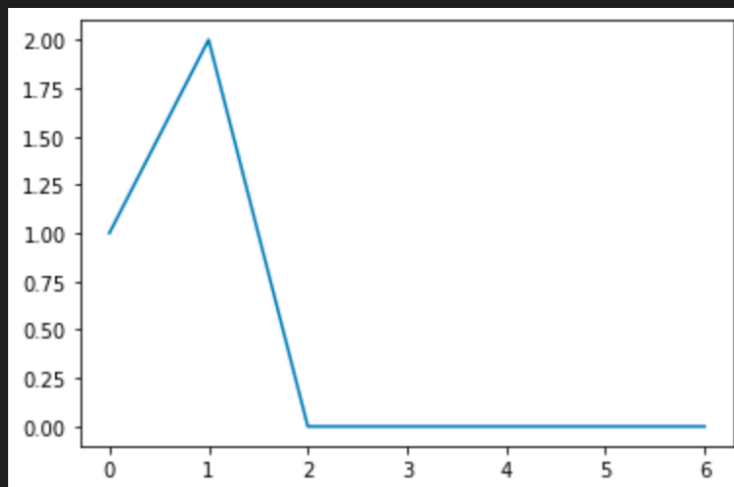
#visualisasi
import matplotlib.pyplot as plt
plt.plot(arr_error)
```

Output

```
... >epoch=0, lrate=1.000, error=1.000
>epoch=1, lrate=1.000, error=2.000
>epoch=2, lrate=1.000, error=0.000
>epoch=3, lrate=1.000, error=0.000
>epoch=4, lrate=1.000, error=0.000
>epoch=5, lrate=1.000, error=0.000
>epoch=6, lrate=1.000, error=0.000
[-1.0, 1.0, 1.0]

[<matplotlib.lines.Line2D at 0x11a318f40>]
```

</>



```
▷ ✓ Nweights = [-1,1,1]
for row in dataset:
    prediction = predict(row, weights)
    print("Expected=%d, Predicted=%d" % (row[-1], prediction))

[2] ✓ 0.2s
```

```
... Expected=1, Predicted=1
Expected=1, Predicted=1
Expected=1, Predicted=1
Expected=0, Predicted=0
```

+ Code

+ Markdown

```
Nweights = [-1,1,1]
dataset = [[1,0,1]]
for row in dataset:
    prediction = predict(row, weights)
    print("Expected=%d, Predicted=%d" % (row[-1], prediction))

[3] ✓ 0.2s

... Expected=1, Predicted=1
```

2. Lakukan perhitungan manual dengan data logika AND

Menentukan bobot awal secara acak

$w_1 = 0, w_2 = 0$, learning rate = 1, biasa = 0, maksimal epoch = 6

- **Epoh 1**

- **Data ke-1**

$$x = \{0,0\}, w = \{0,0\}, b = 0, target = 0$$

$$y_{in} = (0*0) + (0*0) + 0 = 0$$

$$y = sign(0) = 1$$

karena $y \neq target$

update bobot

$$error = target - y = 0 - 1 = -1$$

$$w1_baru = w1_lama + (learning_rate * error * x1)$$

$$= 0 + (1 * (-1) * 0) = 0$$

$$w2_baru = w2_lama + (learning_rate * error * x2)$$

$$= 0 + (1 * (-1) * 0) = 0$$

$$bias_baru = bias_lama + (learning_rate * error)$$

$$= 0 + (1 * (-1)) = -1$$

maka didapat bobot baru $w = \{0,0\}$, $b = -1$ dan lanjutkan untuk data yang ke dua

- **Data ke-2**

$$x = \{0,1\}, w = \{0,0\}, b = -1, target = 0$$

$$y_{in} = (x1*w1) + (x2*w2) + b = (0*0) + (1*0) + (-1) = -1$$

$$y = sign(-1) = 0$$

karena $y == target$ maka tidak perlu dilakukan perbaikan bobot dan lanjutkan untuk data yang ke tiga

- **Data Ke-3**

$$x = \{1,0\}, w = \{0,0\}, b = -1 target = 0$$

$$y_{in} = (x1*w1) + (x2*w2) + b = (1*0) + (0*0) + (-1) = -1$$

$$y = sign(-1) = 0$$

karena $y == target$ maka tidak perlu dilakukan perbaikan bobot dan lanjutkan untuk data yang ke empat

- **Data ke-4**

$$x = \{1,1\} w = \{0,0\}, b = -1 target = 1 y_{in} = (x1*w1) + (x2*w2) + b =$$

$$(1*0) + (1*0) + (-1) = -1 y = sign(-1) = 0$$

karena $y \neq target$ maka hitung error dan update bobot $error = target - y = 1 - 0 = 1$

$$w1_baru = w1_lama + (learning_rate * error * x1) = 0 + (1 * (1) * 1) = 1$$

$$w2_baru = w2_lama + (learning_rate * error * x2) = 0 + (1 * (1) * 1) = 1$$

$$bias_baru = bias_lama + (learning_rate * error) = -1 + (1 * 1) = 0$$

maka didapat bobot baru $w = \{1,1\}$, $b = 0$ dan lanjutkan untuk data selanjutnya karena di epoh pertama masih ada eror, maka perhitungan di lanjutkan ke epoh berikutnya

- **Epoh ke-2**

- **Data ke-1**

$$x = \{0,0\}, w = \{1,1\}, b = 0, target = 0$$

$$y_{in} = (x1*w1) + (x2*w2) + b = (0*1) + (0*1) + 0 = 0$$

$$y = sign(0) = 1$$

karena $y \neq target$ maka hitung error dan update bobot $error = target - y = 0 - 1 = -1$

$$w1_baru = w1_lama + (learning_rate * error * x1)$$

$$= 1 + (1 * (-1) * 0) = 1$$

$$w2_baru = w2_lama + (learning_rate * error * x2)$$

$$= 1 + (1 * (-1) * 0) = 1$$

$$\text{bias_baru} = \text{bias_lama} + (\text{learning_rate} * \text{error})$$

$$= 0 + (1 * (-1)) = -1$$

maka didapat bobot baru $w = \{1, 1\}$, $b = -1$ dan lanjutkan untuk data yang ke dua

- **Data ke-2**

$$x = \{0, 1\}, w = \{1, 1\}, b = -1, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (0 * 1) + (1 * 1) + (-1) = 0$$

$$y = \text{sign}(0) = 1$$

karena $y \neq \text{target}$ maka hitung error dan update bobot $\text{error} = \text{target} - y = 0 - 1 = -1$

$$w1_baru = w1_lama + (\text{learning_rate} * \text{error} * x1)$$

$$= 1 + (1 * (-1) * 0) = 1$$

$$w2_baru = w2_lama + (\text{learning_rate} * \text{error} * x2)$$

$$= 1 + (1 * (-1) * 1) = 0$$

$$\text{bias_baru} = \text{bias_lama} + (\text{learning_rate} * \text{error})$$

$$= -1 + (1 * (-1)) = -2$$

maka didapat bobot baru $w = \{1, 0\}$, $b = -2$ dan lanjutkan untuk data yang ke tiga

- **Data ke-3**

$$x = \{1, 0\}, w = \{1, 0\}, b = -2, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (1 * 1) + (0 * 0) + (-2) = -1$$

$$y = \text{sign}(-1) = 0$$

karena $y \neq \text{target}$ maka tidak perlu dilakukan perbaikan bobot dan lanjutkan untuk data yang ke empat

- **Data ke-4**

$$x = \{1, 1\}, w = \{1, 0\}, b = -2, \text{target} = 1$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (1 * 1) + (1 * 0) + (-2) = -1$$

$$y = \text{sign}(-1) = 0$$

karena $y \neq \text{target}$ maka hitung error dan update bobot

$$\text{error} = \text{target} - y = 1 - 0 = 1$$

$$w1_baru = w1_lama + (\text{learning_rate} * \text{error} * x1)$$

$$= 1 + (1 * 1 * 1) = 2$$

$$w2_baru = w2_lama + (\text{learning_rate} * \text{error} * x2)$$

$$= 0 + (1 * 1 * 1) = 1$$

$$\text{bias_baru} = \text{bias_lama} + (\text{learning_rate} * \text{error})$$

$$= -2 + (1 * 1) = -1$$

maka didapat bobot baru $w = \{2, 1\}$, $b = -1$. Karena di epoch ke dua masih ada error, maka di lanjut ke epoch selanjutnya.

• **Epoch ke-3**

- **Data ke-1**

$$x = \{0, 0\}, w = \{2, 1\}, b = -1, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (0 * 2) + (0 * 1) + (-1) = -1$$

$$y = \text{sign}(-1) = 0$$

karena $y \neq \text{target}$ maka tidak perlu dilakukan perbaikan bobot dan lanjutkan untuk data yang ke dua

- **Data ke-2**

$$x = \{0, 1\}, w = \{2, 1\}, b = -1, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (0 * 2) + (1 * 1) + (-1) = 0$$

$$y = \text{sign}(0) = 1$$

karena $y \neq \text{target}$ maka hitung error dan update bobot

$$\text{error} = \text{target} - y = 0 - 1 = -1$$

$$w1_baru = w1_lama + (\text{learning_rate} * \text{error} * x1) \\ = 2 + (1 * (-1) * 0) = 2$$

$$w2_baru = w2_lama + (\text{learning_rate} * \text{error} * x2) \\ = 1 + (1 * (-1) * 1) = 0$$

$$\text{bias_baru} = \text{bias_lama} + (\text{learning_rate} * \text{error}) \\ = -1 + (1 * (-1)) = -2$$

maka didapat bobot baru $w = \{2, 0\}$, $b = -2$ dan lanjutkan untuk data yang ke tiga

- **Data ke-3**

$$x = \{1, 0\}, w = \{2, 0\}, b = -2, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (1 * 2) + (0 * 0) + (-2) = 0$$

$$y = \text{sign}(0) = 1$$

karena $y \neq \text{target}$ maka hitung error dan update bobot

$$\text{error} = \text{target} - y = 0 - 1 = -1$$

$$w1_baru = w1_lama + (\text{learning_rate} * \text{error} * x1) \\ = 2 + (1 * (-1) * 1) = 1$$

$$w2_baru = w2_lama + (\text{learning_rate} * \text{error} * x2) \\ = 0 + (1 * (-1) * 0) = 0$$

$$\text{bias_baru} = \text{bias_lama} + (\text{learning_rate} * \text{error}) \\ = -2 + (1 * (-1)) = -3$$

maka didapat bobot baru $w = \{1, 0\}$, $b = -3$ dan lanjutkan untuk data yang ke empat

- **Data ke-4**

$$x = \{1, 1\}, w = \{1, 0\}, b = -3, \text{target} = 1$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (1 * 1) + (1 * 0) + (-3) = -2 \quad y = \text{sign}(-2) = 0$$

karena $y \neq \text{target}$ maka hitung error dan update bobot $\text{error} = \text{target} - y = 1 - 0 = 1$

$$w1_baru = w1_lama + (\text{learning_rate} * \text{error} * x1) \\ = 1 + (1 * 1 * 1) = 2$$

$$w2_baru = w2_lama + (\text{learning_rate} * \text{error} * x2) \\ = 0 + (1 * 1 * 1) = 1$$

$$\text{bias_baru} = \text{bias_lama} + (\text{learning_rate} * \text{error}) \\ = -3 + (1 * 1) = -2$$

maka didapat bobot baru $w = \{2, 1\}$, $b = -2$. Karena di epoch ke dua masih ada error, maka dilanjut ke epoch selanjutnya.

• **Epoch ke-4**

- **Data ke-1**

$$x = \{0, 0\}, w = \{2, 1\}, b = -2, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (0 * 2) + (0 * 1) + (-2) = -2$$

$$y = \text{sign}(-2) = 0$$

karena $y == \text{target}$ maka tidak perlu dilakukan perbaikan bobot dan lanjutkan untuk data yang ke dua

- **Data ke-2**

$$x = \{0, 1\}, w = \{2, 1\}, b = -2, \text{target} = 0$$

$$y_in = (x1 * w1) + (x2 * w2) + b = (0 * 2) + (1 * 1) + (-2) = -1$$

$$y = \text{sign}(-1) = 0$$

karena $y == \text{target}$ maka tidak perlu dilakukan perbaikan bobot dan lanjutkan untuk data yang ke tiga

- **Data ke-3**

$x = \{1, 0\}$, $w = \{2, 1\}$, $b = -2$, $target = 0$

$y_{in} = (x1 * w1) + (x2 * w2) + b = (1 * 2) + (0 * 1) + (-2) = 0$

$y = \text{sign}(0) = 1$

karena $y \neq target$ maka hitung error dan update bobot $error = target - y = 0 - 1 = -1$

$w1_baru = w1_lama + (learning_rate * error * x1)$

$= 2 + (1 * (-1) * 1) = 1$

$w2_baru = w2_lama + (learning_rate * error * x2)$

$= 1 + (1 * (-1) * 0) = 1$

$bias_baru = bias_lama + (learning_rate * error)$

$= -2 + (1 * (-1)) = -3$

maka didapat bobot baru $w = \{1, 1\}$, $b = -3$ dan lanjutkan untuk data yang ke empat

- **Data ke-4**

$x = \{1, 1\}$, $w = \{1, 1\}$, $b = -3$, $target = 1$

$y_{in} = (x1 * w1) + (x2 * w2) + b = (1 * 1) + (1 * 1) + (-3) = -1$

$y = \text{sign}(-1) = 0$

karena $y \neq target$ maka hitung error dan update bobot

$error = target - y = 1 - 0 = 1$

$w1_baru = w1_lama + (learning_rate * error * x1)$

$= 1 + (1 * 1 * 1) = 2$

$w2_baru = w2_lama + (learning_rate * error * x2)$

$= 1 + (1 * 1 * 1) = 2$

$bias_baru = bias_lama + (learning_rate * error)$

$= -3 + (1 * 1) = -2$

maka didapat bobot baru $w = \{2, 2\}$, $b = -2$. Karena di epoch ke empat masih ada error, maka di lanjut ke epoch selanjutnya

• **Epoch ke-5**

- **Data ke-1**

$x = 0, 0$, $w = 2, 2$, $b = -2$

$target = 0$

$y_{in} = (0 * 2) + (0 * 2) + (-2) = -2$ $y = \text{sign}(-2) = 0$

- **Data ke-2**

$x = 0, 1$, $w = 2, 2$, $b = -2$

$target = 0$

$y_{in} = (0 * 2) + (1 * 2) + (-2) = 0$ $y = \text{sign}(0) = 1$

karena $y \neq target$

update bobot = error = target - $y = 0 - 1 = -1$ $w1_baru = 2 + (1 * -1 * 0) = 2$ $w2$

$baru = 2 + (1 * -1 * 1) = 1$ $bias_baru = -2 + (1 * -1) = -3$

- **Data ke-3**

$x = 1, 0$, $w = 2, 1$, $b = -3$

$target = 0$

$y_{in} = (1 * 2) + (0 * 1) + (-3) = -1$ $y = \text{sign}(-1) = 0$

- **Data ke-4**

$x = 1, 1$, $w = 2, 1$, $b = -3$

$target = 1$

$y_{in} = (1 * 2) + (1 * 1) + (-3) = 0$ $y = \text{sign}(0) = 1$

• **Epoch ke-6**

- **Data ke-1**

- $x = 0,0, w = 2,1, b = -3$
 $target = 0$
 $y_in = (0*2)+(0*1)+(-3) = -3 \ y = sign(-3) = 0$
- **Data ke-2**

$x = 0,1, w = 2,1, b = -3$
 $target = 0$
 $y_in = (0*2)+(1*1)+(-3) = -2 \ y = sign(-2) = 0$
 - **Data ke-3**

$x = 1,0, w = 2,1, b = -3$
 $target = 0$
 $y_in = (1*2)+(0*1)-3 = -1 \ y = sign(-1) = 0$
 - **Data ke-4**

$x = 1,1, w = 2,1, b = -3$
 $target = 1$
 $y_in = (1*2)+(1*1)-3 = 0 \ y = sign(0) = 1$

3. Ubah kode program diatas dengan dataset berupa logika AND kemudian bandingkan hasilnya!

```
def predict(row, weights):
    activation = weights[0]
    for i in range(len(row)-1):
        activation += weights[i + 1] * row[i]
    return 1.0 if activation >= 0.0 else 0.0

import numpy as np
arr_error=[]
def train_weights(train, l_rate, n_epoch):
    weights = [0.0 for i in range(len(train[0]))]
    for epoch in range(n_epoch):
        sum_error = 0.0
        for row in train:
            prediction = predict(row, weights)
            error = row[-1] - prediction
            sum_error += error**2
            weights[0] = weights[0] + l_rate * error
            for i in range(len(row)-1):
                weights[i + 1] = weights[i + 1] + l_rate * error * row[i]
        print('>epoch=%d, lrate=%.3f, error=%.3f' % (epoch, l_rate,
sum_error))
        arr_error.append(sum_error)
    return weights, arr_error

dataset = [[0, 0, 0],
           [0, 1, 0],
           [1, 0, 0],
           [1, 1, 1]]

weights = [0, 0, 0] #bias,w1,w2
l_rate = 1
n_epoch = 7
```

```

weights, arr_error = train_weights(dataset, l_rate, n_epoch)
print(weights)

#visualisasi
import matplotlib.pyplot as plt
plt.plot(arr_error)

```

Output

```

... >epoch=0, lrate=1.000, error=2.000
    >epoch=1, lrate=1.000, error=3.000
    >epoch=2, lrate=1.000, error=3.000
    >epoch=3, lrate=1.000, error=2.000
    >epoch=4, lrate=1.000, error=1.000
    >epoch=5, lrate=1.000, error=0.000
    >epoch=6, lrate=1.000, error=0.000
    [-3.0, 2.0, 1.0]

[<matplotlib.lines.Line2D at 0x12ca6cf10>]

```

</>

