

**UTS KECERDASAN BUATAN**

**PART-2**



**DISUSUN OLEH:**

**ZESYCA DWI ANJARSARI**

**21091397010**

**D4 MANAJEMEN INFORMATIKA**

**UNIVERSITAS NEGERI SURABAYA**

## 1. Single neuron

Input layer feature 10, batch 6 input, hidden layer 1 (5 neuron), hidden layer 2 (3 neuron)

Input coding

```
1 # Zesyca Dwi Anjarsari
2 # 21091397010
3 # Multiple perceptron / Neuron batch and multiple layer 2
4
5 # menginisialisasi numpy
6 import numpy as np
7
8 # menginisialisasi variabel
9 # menginput nilai variabel layer feature 10 dan batch 6
10 inputs = [
11     [4.7, 4.5, 5.0, 3.0, 2.1, 4.3, 4.7, 4.5, 2.0, 2.5],
12     [2.0, 3.0, 2.1, 4.3, 4.5, 2.3, 3.5, 7.5, 3.4, 6.5],
13     [3.2, 1.2, 4.3, 0.6, 1.5, 6.2, 2.4, 3.2, 1.5, 3.4],
14     [4.5, 2.3, 3.5, 0.8, 4.6, 1.2, 9.2, 9.4, 4.7, 4.5],
15     [3.6, 3.8, 4.5, 3.6, 2.0, 3.5, 4.7, 4.5, 5.0, 5.5],
16     [1.5, 2.3, 8.5, 9.3, 4.2, 1.2, 4.3, 0.6, 1.5, 6.2],
17 ]
18 # memberikan nilai bobot pada variabel sesuai dengan jumlah input
19 # menginput jumlah weight sesuai dengan jumlah neuron yaitu sejumlah 5
20 weights1 = [
21     [1.0, 1.5, 2.0, 2.5, 3.7, 3.5, 4.7, 4.5, 5.0, 5.5],
22     [1.5, 1.4, 2.2, 2.4, 3.2, 3.4, 4.2, 4.4, 5.2, 5.4],
23     [2.7, 1.8, 2.6, 2.8, 3.6, 3.8, 4.6, 4.8, 5.6, 5.8],
24     [2.5, 6.4, 7.2, 7.4, 8.2, 8.4, 9.2, 9.4, 10.2, 10.4],
25     [3.5, 18.5, 18.0, 20.5, 30.0, 30.5, 40.0, 40.5, 50.0, 50.5],
26 ]
```

```

26 ]
27 # menginisialisasi biases pada layer1 sesuai dengan neuron yang ditentukan yaitu layer 1 = 5 neuron
28 biases1 = [1.5, 2.3, 3.1, 4.7, 5.8]
29
30 # menginisialisasi jumlah weight 2, weight layer 2 = neuron layer 1 yaitu 5
31 # menginput jumlah weight sesuai dengan neuron layer 2 yaitu 3 neuron
32 weights2 = [
33     [6.3, 6.4, 2.2, 1.2, 5.2],
34     [2.0, 5.0, 3.2, 6.4, 2.3],
35     [7.1, 3.2, 8.5, 4.3, 2.4]]
36
37 # menginisialisasi biases pada layer2 dengan neuron yang ditentukan yaitu 3
38 biases2 = [2.4, 4.3, 5.9]
39 transpose = np.dot(inputs, np.array(weights1).T)
40 print(transpose)
41 # output
42 # membuat perhitungan layer1 dengan (inputs*weight1) dan biases1
43 layer1_outputs = np.dot(inputs, np.array(weights1).T) + biases1
44
45 # membuat perhitungan layer2 dengan hasil perhitungan pada layer1
46 layer2_outputs = np.dot(layer1_outputs, np.array(weights2).T) + biases2
47
48 # print output layer2
49 print(layer2_outputs)

```

Penjelasan:

- Line 6 pada input code diatas menginisialisasikan numpy ke np yang memudahkan untuk mengoperasikan komputasi angka
- Line 10 pada input code diatas terdapat variable dengan jumlah 10 baris angka sesuai dengan soal yaitu feature layer 10 dan 6 kolom angka dengan batch 6
- Line 20 terdapat inputan nilai weight dengan 5 kolom yang sesuai dengan neuron yaitu 5 dan 10 baris angka.
- Line 28 terdapat input bias sesuai dengan jumlah neuron yaitu 5
- Line 32 terdapat inputan weight2 yang memiliki jumlah sama dengan neuron di layer1 yaitu 5 dengan 3 kolom yang sesuai dengan neuron layer2 yaitu 3
- Line 38 terdapat inputan biases2 dengan jumlah neuron sama dengan layer 2 yaitu 3
- Pada line 43 terdapat inputan yang berfungsi melakukan perhitungan di layer 1 dengan weight dan transpose dulu
- Pada line 46 terdapat inputan yang berfungsi melakukan perhitungan dari hasil layer 1 di kalikan weight transpose ditambah biases2
- Pada line 49 terdapat inputan yang berfungsi untuk menampilkan hasil output dari kode tersebut

## Output

```
[[ 117.86  116.33  135.01  284.03 1041.85]
 [ 149.1   144.84  162.08  332.34 1335.6 ]
 [  94.23   93.58  107.14  218.28  829.5 ]
 [ 171.96  167.13  188.61  372.63 1523.25]
 [ 144.54  142.8   161.96  331.9   1302.45]
 [ 129.45  125.99  143.03  317.99 1138.45]]
```

```
[[ 7611.698  5535.589  6162.806]
 [ 9636.     6812.052  7618.67 ]
 [ 6072.795  4376.19   4892.973]
 [11005.168  7743.569  8701.416]
 [ 9421.044  6713.287  7497.294]
 [ 8307.255  6072.257  6722.045]]
```

Process finished with exit code 0

## Perhitungan output

input 10 <sup>5</sup>		weight 10 <sup>6</sup>		
[4.7, 4.5, 5.0, 3.0, 2.1, 4.3, 4.7, 4.5, 2.0, 2.5], [2.0, 3.0, 2.1, 4.3, 4.5, 2.3, 3.5, 7.5, 3.4, 6.5], [3.2, 1.2, 4.3, 0.6, 1.5, 6.2, 2.4, 3.2, 1.5, 3.4], [4.5, 2.3, 3.5, 0.8, 4.6, 1.2, 9.2, 9.4, 4.7, 4.5], [3.6, 3.8, 4.5, 3.6, 2.0, 3.5, 4.7, 4.5, 5.0, 5.5], [1.5, 2.3, 8.5, 9.3, 4.2, 1.2, 4.3, 0.6, 1.5, 6.2],	X	[1.0, 1.5, 2.0, 2.5, 3.7, 3.5, 4.7, 4.5, 5.0, 5.5], [1.5, 1.4, 2.2, 2.4, 3.2, 3.4, 4.2, 4.4, 5.2, 5.4], [2.7, 1.8, 2.6, 2.8, 3.6, 3.8, 4.6, 4.8, 5.6, 5.8], [2.5, 6.4, 7.2, 7.4, 8.2, 8.4, 9.2, 9.4, 10.2, 10.4], [3.5, 18.5, 18.0, 20.5, 30.0, 30.5, 40.0, 40.5, 50.0, 50.5],	=	[ 117.86 116.33 135.01 284.03 1041.85] [ 149.1 144.84 162.08 332.34 1335.6 ] [ 94.23 93.58 107.14 218.28 829.5 ] [ 171.96 167.13 188.61 372.63 1523.25] [ 144.54 142.8 161.96 331.9 1302.45] [ 129.45 125.99 143.03 317.99 1138.45]
		biases1		
[ 117.86 116.33 135.01 284.03 1041.85] [ 149.1 144.84 162.08 332.34 1335.6 ] [ 94.23 93.58 107.14 218.28 829.5 ] [ 171.96 167.13 188.61 372.63 1523.25] [ 144.54 142.8 161.96 331.9 1302.45] [ 129.45 125.99 143.03 317.99 1138.45]	+	[1.5, 2.3, 3.1, 4.7, 5.8]	=	[ 119.36 118.63 138.11 288.73 1047.65] [ 150.6 147.14 165.18 337.04 1341.4 ] [ 95.73 95.88 110.24 222.98 835.3 ] [ 173.46 169.43 191.71 377.33 1529.05] [ 146.04 145.1 165.06 336.6 1308.25] [ 130.95 128.29 146.13 322.69 1144.25]
		weight2		
[ 119.36 118.63 138.11 288.73 1047.65] [ 150.6 147.14 165.18 337.04 1341.4 ] [ 95.73 95.88 110.24 222.98 835.3 ] [ 173.46 169.43 191.71 377.33 1529.05] [ 146.04 145.1 165.06 336.6 1308.25] [ 130.95 128.29 146.13 322.69 1144.25]	X	[6.3, 6.4, 2.2, 1.2, 5.2], [2.0, 5.0, 3.2, 6.4, 2.3], [7.1, 3.2, 8.5, 4.3, 2.4]	=	7609.298 5531.289 6156.906 9360.6 6807.752 7612.77 6070.392 4371.89 4887.073 11002.768 7739.269 8695.516 9418.644 6708.987 7491.394 8304.855 6067.957 6716.145
		biases2		
7609.298 5531.289 6156.906 9360.6 6807.752 7612.77 6070.392 4371.89 4887.073 11002.768 7739.269 8695.516 9418.644 6708.987 7491.394 8304.855 6067.957 6716.145	+	[2.4, 4.3, 5.9]	=	[ 7611.698 5535.589 6162.806] [ 9636. 6812.052 7618.67 ] [ 6072.795 4376.19 4892.973] [11005.168 7743.569 8701.416] [ 9421.044 6713.287 7497.294] [ 8307.255 6072.257 6722.045]