

**LAPORAN KECERDASAN BUATAN  
UJIAN TENGAH SEMESTER**



Disusun oleh :

Dwi Ramadhantiasari

21091397057

**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA  
FAKULTAS VOKASI  
UNIVERSITAS NEGERI SURABAYA  
2022**

## UTS 1

### A. Single Neuron

- i. Input layer feature 10
- ii. Neuron 1
- iii. Coding :

```
Nomor 1 numpy.py • Nomor 2 numpy.py Nomor 3 numpy.py
C: > Users > Hewlett Packard > Downloads > Programs > UTS AI > Nomor 1 numpy.py > ...
1  #Nama : Dwi RamadhaniSari
2  #NIM : 21091397057
3  #Kelas : Manajemen Informatika 2021 A
4  #single neuron (Pakai numpy)
5
6  #Contoh produk titik menggunakan numpy
7  #Import modul Numpy.
8  import numpy as np
9
10 #masukan perception dengan jumlah input 10
11 inputs = [12, 11, 1, 9, 3, 2, 7, 8, 6, 11]
12
13 #Bobot diteruskan ke perception
14 #panjang bobot harus sesuai dengan panjang input yaitu 10, lalu jumlah bobot sesuai dengan jumlah neuron yaitu 1
15 weights = [0.6, 0.8, 0.1, 0.3, 0.9, 0.7, 0.4, 0.2, 0.5, -0.11]
16
17 #bias untuk perception tertentu
18 #dengan jumlah bias berdasarkan jumlah neuron
19 bias = 9
20
21 #Ambil hasil kali titik antara bobot dan masukan
22 #dan tambahkan bias ke nilai penjumlahan
23 output = np.dot(weights, inputs) + bias
24
25 #mencetak output
26 print(output)
```

### Output

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Hewlett Packard\Downloads\Programs\UTS AI> & 'C:\Users\Hewlett Packard\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\Hewlett Packard\.vscode\extensions\ms-python.python-2022.16.0\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '63038' '--' 'c:\Users\Hewlett Packard\Downloads\Programs\UTS AI\Nomor 1 numpy.py'
38.09
PS C:\Users\Hewlett Packard\Downloads\Programs\UTS AI>
```

### Analisis :

Pada single neuron diatas terdapat 10 input, satu neuron, dan satu set bobot. Untuk melakukan the dot product/operasi dot, gunakan fungsi np.dot untuk menghitung dot product. Tambahkan input dan bobot (weights). Kemudian matriks / vector melakukan dot product/ operasi dot. Dengan memasukan bobot(weights), input, dan kemudian menambahkan bias, maka outputnya seperti gambar diatas.

### Perhitungan

$$12.0 \cdot 0.6 + 11.0 \cdot 0.8 + 1.0 \cdot 0.1 + 9.0 \cdot 0.3 + 3.0 \cdot 0.9 + 2.0 \cdot 0.7 + 7.0 \cdot 0.4 + 8.0 \cdot 0.2 + 6.0 \cdot 0.5 + 11.0 \cdot -0.11 + 9.0 = 38.09$$

## B. Multi Neuron

- i. Input layer feature 10
- ii. Neuron 5
- iii. coding

```
Nomor 1 numpy.py • Nomor 2 numpy.py • Nomor 3 numpy.py
C: > Users > Hewlett Packard > Downloads > Programs > UTS AI > Nomor 2 numpy.py > ...
1  #Nama : Dwi RamadhaniSari
2  #NIM : 21091397057
3  #Kelas : Manajemen Informatika 2021 A
4  #Multi neuron (Pakai numpy)
5
6  #membuat array dengan numpy
7  #Impor modul Numpy.
8  import numpy as np
9
10 # masukan variabel angka dengan jumlah input 10
11 inputs = [3.0, 8.0, 2.0, 9.0, 4.0, 1.0, 7.0, 5.0, 6.0, 10.0]
12
13 #Bobot diteruskan ke perception
14 #panjang bobot harus sesuai dengan panjang input ialah 10, lalu jumlah bobot sesuai dengan jumlah neuron yaitu 5
15 weights = [[0.6, 0.8, 0.4, 0.2, 0.3, 0.5, 0.7, 0.1, 0.0, -0.8],
16            [1.0, 11.00, 0.22, 3.2, 1.2, 8.3, 5.4, 1.49, 5.79, 3.29],
17            [0.64, 7.6, 3.7, 0.26, 8.24, -0.30, -0.38, 0.56, 0.34, -3.1],
18            [1.89, 9.29, 0.24, 3.3, 6.2, 4.50, 9.23, 7.7, 0.35, 0.32],
19            [9.0, 5.3, 0.11, 2.2, -0.63, -0.84, 7.7, -2.1, 0.41, -0.26]]
20
21 #bias untuk perception tertentu
22 #dengan jumlah bias berdasarkan jumlah neuron yaitu 5
23 biases = [9.1, 4.2, 1.0, 1.1, 0.6]
24
25 #Panggil fungsi np.dot
26 #Ambil hasil kali titik antara bobot dan masukan
27 #dan tambahkan bias ke nilai penjumlahan
28 layer_outputs = np.dot(weights, inputs) + biases
29
30 #cetak output
31 print(layer_outputs)
```

## Output


```
[ 19.   250.43  77.3  248.98 129.92]
```

## Analisis :

Untuk Multi Neuron disini fungsi yang digunakan sama seperti single neuron, yaitu fungsi `np.dot` untuk menghitung dot product/ operasi dot. Yang mana input disini sebagai sinyal/ atau umpan yang masuk ke dalam neuron lalu melakukan operasi dot dengan bobot dan menambahkan bias, untuk hasil akhirnya seperti gambar diatas.

### C. Multi Neuron Batch Input

- i. Input layer Feature 10
- ii. Per batch nya 6 input
- iii. Neuron 5
- iv. Coding

```
C:\Users\Hewlett Packard\Downloads\Programs\UTS AI>  Nomor 3 numpy.py > ...
3 #Kelas : Manajemen Informatika 2021 A
4 #Multi Neuron Batch Input
5
6 #Import modul Numpy.
7 import numpy as np
8
9 #memasukan variabel dengan matriks 6x10 (input 10 dan batch 6)
10 inputs = [[0.0, 2.4, 0.2, 3.0, 0.5, 1.0, 0.7, 9.0, 0.4, 2.0],
11           [2.2, 1.1, 3.4, 0.9, 6.0, 5.0, 6.0, 9.0, 2.7, 2.9],
12           [1.5, 3.2, 4.0, 1.0, 3.4, 5.9, 9.3, 7.7, 1.5, 4.8],
13           [7.3, 6.7, 7.8, 1.9, 8.8, 2.1, 7.0, 4.4, 8.8, 3.8],
14           [7.2, 9.9, 3.9, 1.6, 1.5, 1.4, 8.7, 6.6, 2.6, 1.3],
15           [0.3, 4.3, 0.7, 6.6, 8.1, 1.7, 8.0, 3.0, 5.0, 4.0]]
16
17 #panjang weights sinkron dengan panjang input yaitu 10, lalu jumlah weights sama dengan jumlah neuron ialah 5
18 weights = [[1.1, 3.1, 2.3, 2.2, 4.2, 5.9, 4.3, 8.7, 2.1, 9.7],
19           [4.0, 8.1, 1.3, 7.0, 4.3, 2.1, 3.0, 2.0, 4.7, 4.9],
20           [9.2, 6.5, 8.4, 7.5, 1.8, 3.7, 4.2, 2.5, 2.9, 1.8],
21           [2.9, 1.1, 0.2, 1.8, 2.4, 3.0, 1.0, 4.0, 8.8, 1.6],
22           [6.6, 1.7, 2.7, 3.7, 8.3, 4.9, 7.7, 5.2, 9.8, 3.0]]
23
24 #total bias sinkron dengan total neuron ialah 5
25 biases = [1.0, 3.1, 5.5, 2.8, 7.9]
26
27 #Panggil fungsi np.dot
28 #masukan input
29 #panggil fungsi array
30 #kalikan bobot dengan Transpos matriks
31 #tambahkan bias
32 layer_outputs = np.dot(inputs, np.array(weights).T) + biases
33
34 #cetak outputs
```

### Output

```
PS C:\Users\Hewlett Packard\Downloads\Programs\UTS AI> & 'C:\Users\Hewlett Packard\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\Hewlett Packard\.vscode\extensions\ms-python.python-2022.16.0\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '62371' '--' 'c:\Users\Hewlett Packard\Downloads\Programs\UTS AI\Nomor 3 numpy.py'
[[125.05 79.83 80.08 58.5 94.78]
 [209.23 130.73 158.25 112.49 239.26]
 [229.75 148.1 180.45 100.11 235.62]
 [224.99 242.04 292.35 171.86 353.31]
 [179.56 195.64 251.65 106.09 238.7 ]
 [184.64 197.74 175.26 115.36 257.26]]
```

### Analisis

Pada Multi Neuron batch input fungsi yang digunakan sama seperti yang sebelumnya yaitu fungsi np.dot untuk menghitung dot product/ operasi dot. Yang mana input disini terdiri dari matriks 6x10 sebagai sinyal/ atau umpan yang masuk ke dalam neuron lalu melakukan operasi dot dengan beberapa bobot(weight) menggunakan array selanjutnya tambahkan biases, untuk hasil akhirnya seperti gambar diatas.

## UTS 2

### Multi Neuron Batch Input

- Input Layer fitur 10
- Per batch nya 6 input
- Hidden layer 1, 5 neuron
- Hidden layer 2, 3 neuron

### Coding

```
Untitled-1  nomor 4 numpy.py
C: > Users > Hewlett Packard > Downloads > Programs > UTS AI > UTS AI ke 2 > nomor 4 numpy.py > ...

8  #import numpy
9  import numpy as np
10
11 #memasukan variabel dengan matriks 6x10 (input 10 dan batch 6)
12 inputs = [[1.2, 1.3, 2.2, 3.4, 3.1, 5.4, 4.8, 1.8, 2.6, 8.1],
13           [0.1, 0.2, 2.1, 2.2, 0.6, 1.2, 3.1, 1.4, 2.4, 0.4],
14           [1.5, 3.2, 4.0, 1.0, 3.4, 5.9, 9.3, 7.7, 1.5, 4.8],
15           [0.0, 0.1, 0.4, 1.2, 0.6, 0.4, 0.4, 0.7, 0.3, 0.2],
16           [8.1, 4.7, 0.1, 1.6, 0.1, 1.9, 0.9, 0.1, 0.6, 4.2],
17           [0.3, 4.3, 0.7, 6.6, 8.1, 1.7, 8.0, 3.0, 5.0, 4.0]]
18
19 #panjang bobot pada layer 1 sama dengan panjang input ialah 10, lalu jumlah bobot sama dengan jumlah neuron ialah 5
20 weight1 = [[5.1, 2.9, 1.3, 8.8, 0.2, 1.1, 2.6, 6.2, 1.7, 0.5],
21            [4.0, 8.1, 1.3, 7.0, 4.3, 2.1, 3.0, 2.0, 4.7, 4.9],
22            [0.2, 0.4, 8.1, 2.6, 0.6, 6.2, 7.1, 6.4, 8.4, 5.4],
23            [5.4, 2.8, 9.3, 8.3, 0.3, 2.1, 1.6, 1.2, 2.7, 1.5],
24            [8.2, 4.3, 3.1, 1.8, 1.2, 1.9, 3.9, 2.1, 3.6, 2.2]]
25
26 #total bias pada layer1 sesuai dengan total neuron ialah 5
27 biases1 = [1.0, 1.4, 5.1, 2.8, 6.1]
28 #Panjang bobot pada layer 2 sama dengan input ialah 3 neuron
29 weight2 = [[0.2, 1.4, 5.1, 1.9, 4.6]
30            [2.1, 5.3, 2.4, 1.1, 4.9]
31            [2.3, 1.5, 4.1, 7.1, 6.6]]
32
33 #total bias pada layer sesuai dengan total neuron ialah 3
34 biases2 = [1.2, 1.4, 2.2]
35
36 #Perhitungan di layer 1
37 layer1_outputs = np.dot(inputs, np.array(weight1).T) + biases1
38 #Perhitungan di layer 2
39 layer2_outputs = np.dot(layer1_outputs, np.array(weight2).T) + biases2

#cetak outputs
print(layer2_outputs)
```

### Output

```
PS C:\Users\Hewlett Packard> & "C:/Users/Hewlett Packard/AppData/Local/Microsoft/WindowsApps/python3.10.exe" "c:/Users/Hewlett Packard/Downloads/Programs/UTS AI/UTS AI ke 2/nomor 3 numpy.py"
[[ 82.34 137.97 179.04 102.88  98.13]
 [ 46.64  51.99  89.02  58.11  45.8 ]
 [115.97 146.4  234.13 114.14 130.05]
 [ 18.92  19.54  25.26  20.37  15.96]
 [ 78.34 113.92  60.16  87.41 114.77]
 [127.38 196.04 184.71 119.65 118.35]]
```

## Analisis

Pada Multi Neuron batch input tahap pertama ialah import atau inisialisasi numpy as np, lanjut memasukan input matriks 6x10, lalu masukan weight layer 1 dengan panjang input 1 berjumlah 5 baris sesuai dengan neuron, masukan bias layer 1 yang sama dengan neuron yaitu 5, lanjut masuk ke layer dua masukan weights 2 sama dengan layer 1 yaitu 5 dengan jumlah baris sesuai dengan neuron layer 2 yaitu 3 dan memasukan bias layer 2 sama dengan neuron yaitu 3. Selanjutnya untuk proses perhitungan dengan menghitung layer 1 terlebih dahulu hasil dari perhitungan layer 1 kita masukan dalam perhitungan ke dua sesuai dengan codingan. Untuk hasil akhirnya seperti gambar diatas.