

**LAPORAN KECERDASAN BUATAN
UJIAN TENGAH SEMESTER**



Disusun oleh :
Muvidha Fatmawati Putri (21091397057)
A2021 MI

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

2022

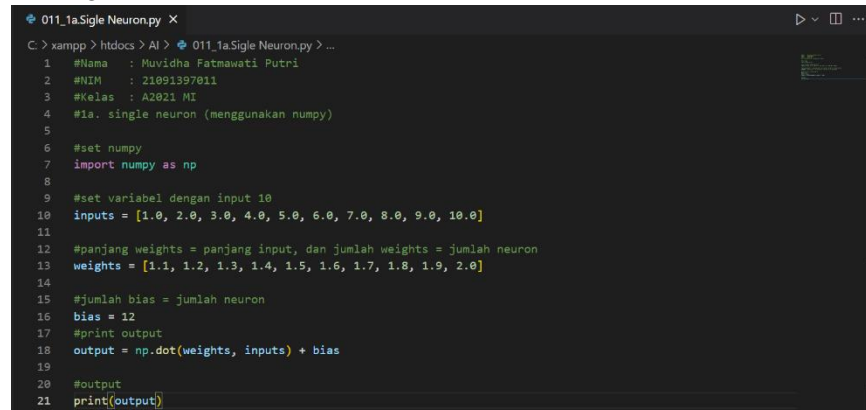
2. Buat dokumentasi dan jelaskan

a. Single Neuron

Input layer feature 10

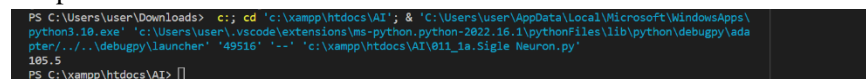
Neuron 1

- Kodingan :



```
011_1a.Single Neuron.py X
C:\> xampp > htdocs > AI > 011_1a.Single Neuron.py > ...
1  #Nama : Muvidha Fatmawati Putri
2  #NIM : 21091397011
3  #Kelas : A2021 MI
4  #1a. single neuron (menggunakan numpy)
5
6  #set numpy
7  import numpy as np
8
9  #set variabel dengan input 10
10 inputs = [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0]
11
12 #panjang weights = panjang input, dan jumlah weights = jumlah neuron
13 weights = [1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0]
14
15 #jumlah bias = jumlah neuron
16 bias = 12
17 #print output
18 output = np.dot(weights, inputs) + bias
19
20 #output
21 print(output)
```

- Output :



```
PS C:\Users\User\Downloads> cd 'c:\xampp\htdocs\AI'; & 'C:\Users\User\AppData\Local\Microsoft\WindowsApps\python3.10.exe' 'c:\Users\User\vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '49516' '--' 'c:\xampp\htdocs\AI\011_1a.Single Neuron.py'
105.5
PS C:\xampp\htdocs\AI> []
```

- Penjelasan :

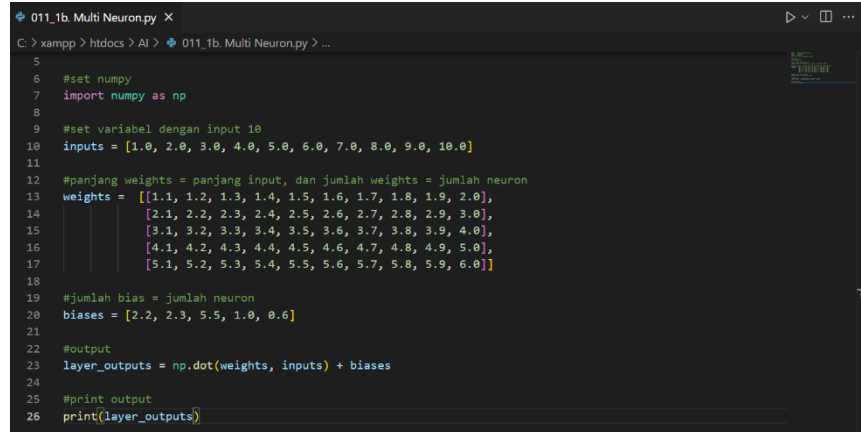
Program dimulai dengan menginisialisasi numpy dengan np pada baris 6. Selanjutnya membuat variabel input pada baris 10 (untuk menginput nilai neuron), bobot pada baris 13 (bobot neuron), dan bias pada baris 14 (untuk menyeimbangkan keluaran positif atau negatif). Serta mengisi setiap variable dengan nilai yang telah diinputkan, yaitu 10 value untuk input layers, dan Weight serta bias = 1, karena single neuron. Baris 17 mendefinisikan output dari perhitungan produk dalam dot.product (untuk mengalikan nilai input dengan bobot) dan ditambahkan ke nilai bias .Dan terus baris 20 terdapat fungsi print yang memanggil hasil perhitungan dengan output.

- Step perhitungan

1. Inputs dimisalkan sebagai vector a dan weights dimisalkan sebagai vector b, jawab:
vector a [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0]
vector b [1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0]
2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:
 $(1.0 \times 1.1) + (2.0 \times 1.2) + (3.0 \times 1.3) + (4.0 \times 1.4) + (5.0 \times 1.5) + (6.0 \times 1.6) + (7.0 \times 1.7) + (8.0 \times 1.8) + (9.0 \times 1.9) + (10.0 \times 2.0) = 93.5$
3. Hasil dari dot product vector tersebut kemudian dijumlahkan dengan bias, jawab: $93.5 + 12 = 105.5$
4. Output yang dihasilkan, jawab: 105.5

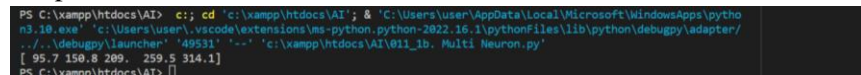
- b. Multi Neuron
Input layer feature 10
Neuron 5

- Kodingan :



```
5
6 #set numpy
7 import numpy as np
8
9 #set variabel dengan input 10
10 inputs = [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0]
11
12 #panjang weights = panjang input, dan jumlah weights = jumlah neuron
13 weights = [[1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0],
14            [2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0],
15            [3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4.0],
16            [4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5.0],
17            [5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 6.0]]
18
19 #jumlah bias = jumlah neuron
20 biases = [2.2, 2.3, 5.5, 1.0, 0.6]
21
22 #output
23 layer_outputs = np.dot(weights, inputs) + biases
24
25 #print output
26 print(layer_outputs)
```

- Output :



```
PS C:\xampp\htdocs\AI> cd "c:\xampp\htdocs\AI"; & "C:\Users\User\AppData\Local\Microsoft\WindowsApps\python3.10.exe" "c:\Users\User\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter/../../debugpy/launcher" "49531" "--" "c:\xampp\htdocs\AI\011_1b. Multi Neuron.py"
[ 95.7 150.8 209. 259.5 314.1]
PS C:\xampp\htdocs\AI>
```

- Penjelasan :

Seperti Single Neuron, Multi Neuron dimulai dengan inisialisasi numpy pada baris 6. Selanjutnya, inisialisasi variabel input (baris 10), bobot (baris 13), dan bias (baris 19). Untuk lapisan input = 10, bobot dan bias = 5 (neuron = 5). Kemudian, pada baris 22 , kami membuat produk titik perhitungan dot.product (mengkalikan bobot dengan input) pada lapisan output dan menambahkan bias . _ Selanjutnya pada baris ke - 25 , hasil perhitungan output layer dipanggil dengan perintah print . Untuk step perhitungan multi neuron hampir sama dengan single neuron tetapi dihitung per neuron satu per satu terlebih dahulu

c. Multi Neuron Batch Input

Input layer Feature 10

Per batch nya 6 input

Neuron 5

- Kodingan :

```
011_1c.Muti Neuron Batch Input.py X
C:\xampp\htdocs> AI > 011_1c.Muti Neuron Batch Input.py > ...
1 #Nama : Muvidha Fatmawati Putri
2 #NIM : 21091397011
3 #Kelas : A2021 MI
4 #1c. multi neuron batch input (menggunakan numpy)
5
6 #set numpy
7 import numpy as np
8
9 #set variabel dengan matriks 6x10 (input 10 dan batch 6)
10 inputs = [[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0],
11 [1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0],
12 [2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0],
13 [3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4.0],
14 [4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5.0],
15 [5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 6.0]]
16
17 #panjang weights = panjang input, dan jumlah weights = jumlah neuron
18 weights = [[1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0],
19 [2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0],
20 [3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4.0],
21 [4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5.0],
22 [5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 6.0]]
23 biases = [2.2, 2.3, 5.5, 1.0, 0.6]
24
25 #ouputs
26 layer_outputs = np.dot(inputs, np.array(weights).T) + biases
27
28 #print outputs
29 print(layer_outputs)
```

- Output :

```
PS C:\xampp\htdocs\AI> c:\cd "c:\xampp\htdocs\AI"; & "C:\Users\user\AppData\Local\Microsoft\WindowsApps\python3.10.exe" "c:\Users\user\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter/..\..\debugpy\launcher" "49539" "-c" "c:\xampp\htdocs\AI\011_1c.Muti Neuron Batch Input.py"
[[ 11.55  17.15  25.85  26.85  31.95]
 [ 27.85  42.65  61.35  72.95  87.45]
 [ 42.55  68.15  96.85 117.85 142.95]
 [ 58.85  93.65 132.35 163.35 198.45]
 [ 73.55 119.15 167.85 208.85 253.95]
 [ 89.05 144.65 203.35 254.35 309.45]]
PS C:\xampp\htdocs\AI>
```

- Penjelasan :

Pada baris 5, pertama - tama kita menginisialisasi numpy (yang nantinya akan digunakan dalam perhitungan dot product). Selanjutnya, inisialisasi variabel input (baris 10), bobot (baris 18), dan bias (baris 23). Untuk lapisan input = 10, input per batch = 6, bobot dan bias = 5 (neuron = 5). Selanjutnya, baris 26 melakukan perhitungan hasil kali dalam dot.product (mengkalikan setiap nilai input dengan bobot transposnya (T)) dan menambahkan setiap nilai bias . Ini memanggil fungsi " Lapisan PrintOutput " pada baris 29 dan menampilkan hasil perhitungan dalam bentuk Himpunan.