

LAPORAN UJIAN TENGAH SEMESTER KECERDASAN BUATAN



Oleh:

Fisma Meividianugraha Subani

21091397017

D4 MANAJEMEN INFORMATIKA

FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

TAHUN AJARAN 2022/2023

1. Buat Kodingan

a) Source Code Single Neuron

```
C:\Users\LENOVO\Downloads\UTS\017_fisma_no1a_singelneuron.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

017_fisma_no1a_singelneuron.py x 017_fisma_no1b_multineuron.py x 017_fisma_no1c_multineuronbatchinput.py x
1 # Nama : Fisma Meividianugraha Subani
2 # NIM : 21091397017
3 # Kelas : 2021 A
4
5 # No 1a. Single Neuron
6 # i. Input layer fature 10
7 # ii. Neuron 1
8
9 import numpy as np
10 # inisialisasi variabel inputs dengan panjang input layer = 10
11 inputs = [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]
12
13 # inisialisasi variabel weights berdasarkan jumlah neuron = 1 dengan panjang sesuai input layer = 10
14 weights = [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1]
15
16 # inisialisasi variabel bias berdasarkan panjang neuron = 1
17 bias = 4.0
18
19 # Rumus dot product vector [inputs[0-9]*weights[0-9]] + [bias]
20 outputs = np.dot(weights, inputs) + bias
21
22 # Menampilkan hasil output
23 print(outputs)
24
```

Hasil Output:

```
C:\Users\LENOVO\Downloads\UTS\017_fisma_no1a_singelneuron.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

017_fisma_no1a_singelneuron.py x 017_fisma_no1b_multineuron.py x 017_fisma_no1c_multineuronbatchinput.py x
1 # Nama : Fisma Meividianugraha Subani
2 # NIM : 21091397017
3 # Kelas : 2021 A
4
5 # No 1a. Single Neuron
6 # i. Input layer fature 10
7 # ii. Neuron 1
8
9 import numpy as np
10 # inisialisasi variabel inputs dengan panjang input layer = 10
11 inputs = [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]
12
13 # inisialisasi variabel weights berdasarkan jumlah neuron = 1 dengan panjang sesuai input layer = 10
14 weights = [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1]
15
16 # inisialisasi variabel bias berdasarkan panjang neuron = 1
17 bias = 4.0
18
19 # Rumus dot product vector [inputs[0-9]*weights[0-9]] + [bias]
20 outputs = np.dot(weights, inputs) + bias
21
22 # Menampilkan hasil output
23 print(outputs)
24
38.18
[Finished in 9.2s]
```

b) Source Code Multi Neuron

```
C:\Users\LENOVO\Downloads\UTS\017_fisma_no1b_multineuron.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

017_fisma_no1a_singelneuron.py x 017_fisma_no1b_multineuron.py x 017_fisma_no1c_multineuronbatchinput.py x
1 # Nama : Fisma Meividianugraha Subani
2 # NIM : 21091397017
3 # Kelas : 2021 A
4
5 # No 1b. Multi Neuron
6 # i. Input layer fature 10
7 # ii. Neuron 5
8
9 import numpy as np
10 # inisialisasi variabel inputs dengan panjang input layer = 10
11 inputs = [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]
12
13 # inisialisasi variabel weights berdasarkan jumlah neuron = 5 dengan panjang sesuai input layer = 10
14 weights = [[1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1],
15            [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5],
16            [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1],
17            [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5],
18            [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1]]
19
20 # inisialisasi variabel biases berdasarkan panjang neuron = 5
21 biases = [1.0, 5.0, 1.1, 5.0, 1.1]
22
23 # Rumus dot product vector [np.dot(weight[0], input), np.dot(weight[1], input), np.dot(weight[2], input),
24 # np.dot(weight[3], input), np.dot(weight[4], input)] + biases
25 layer_outputs = np.dot(weights, inputs) + biases
26
27 # Menampilkan hasil output
28 print(layer_outputs)
29
```

Hasil Output:

```
C:\Users\LENOVO\Downloads\UTS\017_fisma_no1b_multineuron.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
017_fisma_no1a_singleneuron.py 017_fisma_no1b_multineuron.py 017_fisma_no1c_multineuronbatchinput.py
1 # Nama : Fisma Meividianugraha Subani
2 # NIM : 21091397017
3 # Kelas : 2021 A
4
5 # No 1b. Multi Neuron
6 # i. Input layer fature 10
7 # ii. Neuron 5
[35.18 94.08 51.12 40.48 50.5 ]
[Finished in 1.1s]
```

c) Source Code Multi Neuron Batch Input

```
C:\Users\LENOVO\Downloads\UTS\017_fisma_no1c_multineuronbatchinput.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
017_fisma_no1a_singleneuron.py 017_fisma_no1b_multineuron.py 017_fisma_no1c_multineuronbatchinput.py
1 # Nama : Fisma Meividianugraha Subani
2 # NIM : 21091397017
3 # Kelas : 2021 A
4
5 # No 1c. Multi Neuron Batch Input
6 # i. Input layer fature 10
7 # ii. Per batchnya 6 input
8 # iii. Neuron 5
9
10 import numpy as np
11 # inisialisasi variabel inputs berdasarkan jumlah batch = 6 dengan panjang sesuai input layer = 10 [matriks 6x10]
12 inputs = [[1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5],
13           [3.0, 0.9, 2.0, 0.3, 0.3, 9.0, 0.2, 3.0, -3.0, -0.9],
14           [8.0, 0.3, 2.0, 0.3, 0.8, 3.0, 0.2, 3.0, -8.0, -0.3],
15           [1.6, 1.2, 2.0, 2.0, 6.1, 2.1, 0.2, 0.2, -1.6, -1.2],
16           [6.0, 7.0, 2.0, 0.3, 0.6, 0.7, 0.2, 3.0, -6.0, -7.0],
17           [1.9, 1.0, 2.0, 2.2, 9.1, 0.1, 0.2, 2.2, -1.9, -1.0]]
18
19 # inisialisasi variabel weights berdasarkan jumlah neuron = 5 dengan panjang sesuai input layer = 10 [matriks 5x10]
20 weights = [[1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1],
21            [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5],
22            [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1],
23            [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5],
24            [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1]]
25
26 # inisialisasi variabel biases berdasarkan panjang neuron = 5
27 biases = [5.0, 9.0, 3.0, 2.1, 0.7]
28
29 # Rumus dot product vector [inputs batch * hasil transpose weights] + [biases]
30 # [(matriks 6x10) * (matriks 10x5)] + [biases1, biases2, biases3, biases4, biases5]
31 layer_outputs = np.dot(inputs, np.array(weights).T) + biases
32
33 # Menampilkan hasil output
34 print(layer_outputs)
35
Line 35, Column 1 Spaces: 4 Python
```

Hasil Output:

```
C:\Users\LENOVO\Downloads\UTS\017_fisma_no1c_multineuronbatchinput.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
017_fisma_no1a_singleneuron.py 017_fisma_no1b_multineuron.py 017_fisma_no1c_multineuronbatchinput.py
1 # Nama : Fisma Meividianugraha Subani
2 # NIM : 21091397017
3 # Kelas : 2021 A
4
5 # No 1c. Multi Neuron Batch Input
6 # i. Input layer fature 10
7 # ii. Per batchnya 6 input
[[ 39.18  98.08  53.02  37.58  50.1 ]
 [ 46.38  99.46  54.04  61.1  41.94]
 [ 50.31  92.41  60.67  40.55  66.12]
 [ 36.95 102.83  51.56  22.05  69.47]
 [ 45.33  78.19  67.06  31.73  65.49]
 [ 49.51 137.61  75.54  14.77  97.15]]
[Finished in 2.0s]
```

2. Buat dokumentasi dengan bahasa kalian sendiri, jelaskan :

- Masukkan Source Code kodingan dan jelaskan dengan bahasa kalian sendiri step by step pengerjaannya. (Dijelaskan dibagian komen)

b) Dijelaskan cara kerja setiap penghitungan output setiap matrix nya.

i. Single perceptron → dot product dan penambahan bias

Pada single perceptron atau neuron terdapat inputs, weights, dan bias. Panjang input dapat mempengaruhi panjang weights, misalnya inputs [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5] dan weights [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1]. Di dalam inputs memiliki panjang inputs sebanyak 10, begitu juga dengan weights memiliki panjang yang sama seperti inputs, yaitu sebanyak 10. Jumlah neuron dapat mempengaruhi jumlah weights dan bias, misalnya jumlah neuron pada contoh diatas adalah 1 karena contoh tersebut merupakan single neuron, maka weights [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1] dan bias [4.0]. Dapat diketahui banyaknya jumlah weights dan bias yaitu 1 (bukan weights1, weights2, bias1, bias2).

$$\vec{a} \cdot \vec{b} = \sum_{i=1}^n a_i b_i = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

Gambar 2.b. 1 Rumus Dot Product Vector

Cara menyelesaikan perhitungan pada single neuron dapat menggunakan rumus dot product vector kemudian dijumlahkan dengan bias. Terdapat beberapa langkah mudah dalam menyelesaikan single neuron pada source code di atas:

1. Inputs dimisalkan sebagai vector a dan weights dimisalkan sebagai vector b, jawab:

vector a [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]

vector b [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1]

2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:

$$(1.3*1.0) + (0.5*0.1) + (2.0*1.9) + (0.3*9.5) + (3.1*0.1) + (5.0*1.0) + (0.2*9.1) + (3.0*5.9) + (-1.3*-1.0) + (-0.5*-0.1) = 34.18$$

3. Hasil dari dot product vector tersebut kemudian dijumlahkan dengan bias, jawab:

$$34.18 + 4.0 = 38.18$$

4. Output yang dihasilkan, jawab:

$$34.18$$

ii. Multi perceptron → dot product dan penambahan bias.

Pada multi perceptron atau neuron terdapat inputs, weights, dan bias. Panjang input dapat mempengaruhi panjang weights, misalnya inputs [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5], weights1 [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1], weights2 [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5], weights3 [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1], weights4 [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5], weights5 [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1]. Di dalam inputs memiliki panjang inputs sebanyak 10, begitu juga dengan weights1, weights2, weights3, weights4, weights5 memiliki panjang yang sama seperti inputs, yaitu sebanyak 10. Jumlah neuron pada multi neuron dapat berjumlah 2 atau lebih dan jumlah tersebut dapat mempengaruhi jumlah weights dan bias, misalnya jumlah neuron pada contoh diatas adalah 5 karena contoh tersebut merupakan multi neuron, maka weights1 [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1], weights2 [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5], weights3 [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1], weights4 [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5], weights5 [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1] dan bias1 [1.0], bias2 [5.0], bias3 [1.1], bias4 [5.0], bias5 [1.1]. Dapat diketahui banyaknya jumlah weights dan bias yaitu 5.

Cara menyelesaikan perhitungan pada multi neuron dapat menggunakan rumus dot product vector kemudian dijumlahkan dengan bias untuk menemukan hasil neuron1, neuron2, neuron3, neuron4, dan neuron5 pada contoh tersebut. Terdapat beberapa langkah mudah dalam menyelesaikan multi neuron pada source code di atas:

Neuron1

1. Inputs dimisalkan sebagai vector a dan weights1 dimisalkan sebagai vector b, jawab:

vector a [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]

vector b [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1]

2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:

$$(1.3*1.0) + (0.5*0.1) + (2.0*1.9) + (0.3*9.5) + (3.1*0.1) + (5.0*1.0) + (0.2*9.1) + (3.0*5.9) + (-1.3*-1.0) + (-0.5*-0.1) = 34.18$$

3. Hasil dari dot product vector pada neuron1 kemudian dijumlahkan dengan bias1, jawab:

$$34.18 + 1.0 = 35.18 \rightarrow \text{hasil neuron 1}$$

Neuron2

1. Inputs dimisalkan sebagai vector a dan weights2 dimisalkan sebagai vector b, jawab:

vector a [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]

vector b [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5]

2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:

$$(1.3*1.9) + (0.5*0.5) + (2.0*1.9) + (0.3*6.7) + (3.1*9.1) + (5.0*5.0) + (0.2*9.1) + (3.0*7.6) + (-1.3*-1.9) + (-0.5*-0.5) = 89.08$$

3. Hasil dari dot product vector pada neuron2 kemudian dijumlahkan dengan bias2, jawab:

$$89.08 + 5.0 = 94.08 \rightarrow \text{hasil neuron 2}$$

Neuron3

1. Inputs dimisalkan sebagai vector a dan weights3 dimisalkan sebagai vector b, jawab:

vector a [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]

vector b [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1]

2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:

$$(1.3*1.3) + (0.5*1.1) + (2.0*1.9) + (0.3*6.7) + (3.1*3.1) + (5.0*1.1) + (0.2*9.1) + (3.0*7.6) + (-1.3*-1.3) + (-0.5*-1.1) = 50.02$$

3. Hasil dari dot product vector pada neuron3 kemudian dijumlahkan dengan bias3, jawab:

$$50.02 + 1.1 = 51.12 \rightarrow \text{hasil neuron 3}$$

Neuron4

1. Inputs dimisalkan sebagai vector a dan weights4 dimisalkan sebagai vector b, jawab:

vector a [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]

vector b [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5]

2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:

$$(1.3*1.0) + (0.5*0.5) + (2.0*2.0) + (0.3*0.1) + (3.1*0.1) + (5.0*5.0) + (0.2*0.2) + (3.0*1.0) + (-1.3*-1.0) + (-0.5*-0.5) = 35.48$$

3. Hasil dari dot product vector pada neuron4 kemudian dijumlahkan dengan bias4, jawab:

$$35.48 + 5.0 = 40.48 \rightarrow \text{hasil neuron 4}$$

Neuron5

1. Inputs dimisalkan sebagai vector a dan weights5 dimisalkan sebagai vector b, jawab:

vector a [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5]

vector b [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1]

2. Lakukan perkalian vector a dan vector b sesuai dengan rumus dot product vector, jawab:

$$(1.3*2.8) + (0.5*1.1) + (2.0*2.0) + (0.3*0.2) + (3.1*8.2) + (5.0*1.1) + (0.2*0.2) + (3.0*2.0) + (-1.3*-2.8) + (-0.5*-1.1) = 49.4$$

3. Hasil dari dot product vector pada neuron5 kemudian dijumlahkan dengan bias5, jawab:

$$49.4 + 1.1 = 50.5 \rightarrow \text{hasil neuron 5}$$

Output

1. Output yang dihasilkan adalah hasil dari [neuron1, neuron2, neuron3, neuron4, neuron5], jawab:

[35.18 94.08 51.12 40.48 50.5]

iii. Multi perceptron batch input → dot product, transpose dan penambahan bias

Pada multi perceptron atau neuron batch input terdapat inputs, weights, dan bias. Jumlah batch input pada multi neuron dapat mempengaruhi jumlah inputs, misalnya jumlah per batch nya 6 input, maka:

inputs1 [1.3, 0.5, 2.0, 0.3, 3.1, 5.0, 0.2, 3.0, -1.3, -0.5],

inputs2 [3.0, 0.9, 2.0, 0.3, 0.3, 9.0, 0.2, 3.0, -3.0, -0.9],

inputs3 [8.0, 0.3, 2.0, 0.3, 0.8, 3.0, 0.2, 3.0, -8.0, -0.3],

inputs4 [1.6, 1.2, 2.0, 2.0, 6.1, 2.1, 0.2, 0.2, -1.6, -1.2],

inputs5 [6.0, 7.0, 2.0, 0.3, 0.6, 0.7, 0.2, 3.0, -6.0, -7.0],

inputs6 [1.9, 1.0, 2.0, 2.2, 9.1, 0.1, 0.2, 2.2, -1.9, -1.0]

Dapat diketahui banyaknya jumlah inputs yaitu 6 sehingga bisa juga diartikan inputs-inputs tersebut sebagai matriks (6x10). Panjang input dapat mempengaruhi panjang weights, misalnya panjang inputs di atas sebanyak 10, maka:

weights1 [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1],

weights2 [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5],

weights3 [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1],

weights4 [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5],

weights5 [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1]

Hal tersebut mempengaruhi pada weights1, weights2, weights3, weights4, weights5 memiliki panjang yang sama seperti inputs, yaitu sebanyak 10 sehingga dapat diartikan weights-weights tersebut sebagai matriks (5x10). Jumlah neuron pada multi neuron batch input dapat berjumlah 2 atau lebih dan jumlah tersebut dapat mempengaruhi jumlah weights dan bias, misalnya jumlah neuron pada contoh diatas adalah 5 karena contoh tersebut merupakan multi neuron, maka weights1 [1.0, 0.1, 1.9, 9.5, 0.1, 1.0, 9.1, 5.9, -1.0, -0.1], weights2 [1.9, 0.5, 1.9, 6.7, 9.1, 5.0, 9.1, 7.6, -1.9, -0.5], weights3 [1.3, 1.1, 1.9, 6.7, 3.1, 1.1, 9.1, 7.6, -1.3, -1.1], weights4 [1.0, 0.5, 2.0, 0.1, 0.1, 5.0, 0.2, 1.0, -1.0, -0.5], weights5 [2.8, 1.1, 2.0, 0.2, 8.2, 1.1, 0.2, 2.0, -2.8, -1.1] dan bias1 [5.0], bias2 [9.0], bias3 [3.0], bias4 [2.1], bias5 [0.7]. Dapat diketahui banyaknya jumlah weights dan bias yaitu 5.

Kalikan sesuai urutannya

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix} \times \begin{pmatrix} p & q \\ r & s \\ t & u \end{pmatrix} = \begin{pmatrix} ap+br+ct & aq+bs+cu \\ dp+er+ft & dq+es+fu \\ gp+hr+it & gq+hs+iu \end{pmatrix}$$

Gambar 2.b. 3 Rumus Perkalian Matriks

Cara menyelesaikan perhitungan pada multi neuron batch input dapat menggunakan rumus perkalian matriks dengan salah satu matriksnya ditransposekan terlebih dahulu agar kedua matriks tersebut dapat dikalikan kemudian dijumlahkan dengan bias untuk menemukan matriks yang dihasilkan dari semua neuron pada contoh tersebut. Terdapat beberapa langkah mudah dalam menyelesaikan multi neuron batch input pada source code di atas:

1. Inputs1, inputs2, inputs3, inputs4, inputs5, inputs6 dimisalkan sebagai matriks (6x10), jawab:

$$\begin{bmatrix} 1.3 & 0.5 & 2.0 & 0.3 & 3.1 & 5.0 & 0.2 & 3.0 & -1.3 & -0.5 \\ 3.0 & 0.9 & 2.0 & 0.3 & 0.3 & 9.0 & 0.2 & 3.0 & -3.0 & -0.9 \\ 8.0 & 0.3 & 2.0 & 0.3 & 0.8 & 3.0 & 0.2 & 3.0 & -8.0 & -0.3 \\ 1.6 & 1.2 & 2.0 & 2.0 & 6.1 & 2.1 & 0.2 & 0.2 & -1.6 & -1.2 \\ 6.0 & 7.0 & 2.0 & 0.3 & 0.6 & 0.7 & 0.2 & 3.0 & -6.0 & -7.0 \\ 1.9 & 1.0 & 2.0 & 2.2 & 9.1 & 0.1 & 0.2 & 2.2 & -1.9 & -1.0 \end{bmatrix}$$

2. Weights1, weights2, weights3, weights4, weights5 dimisalkan sebagai matriks (5x10), jawab:

$$\begin{bmatrix} 1.0 & 0.1 & 1.9 & 9.5 & 0.1 & 1.0 & 9.1 & 5.9 & -1.0 & -0.1 \\ 1.9 & 0.5 & 1.9 & 6.7 & 9.1 & 5.0 & 9.1 & 7.6 & -1.9 & -0.5 \\ 1.3 & 1.1 & 1.9 & 6.7 & 3.1 & 1.1 & 9.1 & 7.6 & -1.3 & -1.1 \\ 1.0 & 0.5 & 2.0 & 0.1 & 0.1 & 5.0 & 0.2 & 1.0 & -1.0 & -0.5 \\ 2.8 & 1.1 & 2.0 & 0.2 & 8.2 & 1.1 & 0.2 & 2.0 & -2.8 & -1.1 \end{bmatrix}$$

3. Dikarenakan matrik inputs (6x10) dan matriks weights (5x10) tidak dapat dikalikan karena jumlah kolom (10) matriks inputs tidak sama dengan jumlah baris (5) pada matriks weights, maka salah satu matriks yaitu matriks weights harus ditransposekan terlebih dahulu, jawab:

$$\begin{bmatrix} 1.0 & 1.9 & 1.3 & 1.0 & 2.8 \\ 0.1 & 0.5 & 1.1 & 0.5 & 1.1 \\ 1.9 & 1.9 & 1.9 & 2.0 & 2.0 \\ 9.5 & 6.7 & 6.7 & 0.1 & 0.2 \\ 0.1 & 9.1 & 3.1 & 0.1 & 8.2 \\ 1.0 & 5.0 & 1.1 & 5.0 & 1.1 \\ 9.1 & 9.1 & 9.1 & 0.2 & 0.2 \\ 5.9 & 7.6 & 7.6 & 1.0 & 2.0 \\ -1.0 & -1.9 & -1.3 & -1.0 & -2.8 \\ -0.1 & -0.5 & -1.1 & -0.5 & -1.1 \end{bmatrix}$$

4. Kemudian setelah jumlah kolom (10) matriks inputs sama dengan jumlah baris (10) pada matriks weights, maka matriks inputs (6x10) dan matriks weights (10x5) dapat dikalikan menggunakan rumus perkalian matriks dengan cara baris dikali kolom sehingga matriks yang dihasilkan menjadi matriks (6x5), jawab:

$$\begin{bmatrix} 39.18 & 98.08 & 53.02 & 37.58 & 50.1 \\ 46.38 & 99.46 & 54.04 & 61.1 & 41.94 \\ 50.31 & 92.41 & 60.67 & 40.55 & 66.12 \\ 36.95 & 102.83 & 51.56 & 22.05 & 69.47 \\ 45.33 & 78.19 & 67.06 & 31.73 & 65.49 \\ 49.51 & 137.61 & 75.54 & 14.77 & 97.15 \end{bmatrix}$$