

Laporan Kecerdasan Buatan

Ujian Tengah Semester

Part 2



Disusun Oleh:

Riska Octavia Arianto

21091397059

2021A

DIV MANAJEMEN INFORMATIKA PROGRAM VOKASI

UNIVERSITAS NEGERI SURABAYA

TAHUN AJARAN 2021/2022

Revisi UTS 1

Soal

1. Buat kodingan
 - a. Single Neuron,
 - i. Input layer feature 10
 - ii. Neuron 1
 - b. Multi Neuron
 - i. Input layer feature 10
 - ii. Neuron 5
 - c. Multi Neuron Batch Input
 - i. Input layer feature 10
 - ii. Per batch nya 6 input
 - iii. Neuron 5
2. Buat dokumentasi dengan bahasa kalian sendiri, jelaskan :
 - a. Masukkan Source Code kodingan dan jelaskan dengan bahasa kalian sendiri step by step pengerjaannya
 - b. Dijelaskan cara kerja setiap penghitungan output setiap matrix nya.
 - i. single perceptron → dot product dan penambahan bias
 - ii. multi perceptron → dot product dan penambahan bias
 - iii. multi perceptron batch input → dot product, transpose dan penambahan bias

1. Single neuron

Source code:

```
1A.py X
1A.py > ...
1 #Nama= Riska Octavia Arianto
2 #NIM= 21091397059
3 #Single Neuron
4
5 #inisialisasi numpy
6 import numpy as np
7
8 #inisialisasi variabel
9 #Input layer feature 10
10 inputs = [5, 3, 2, 6, 4, 8, 7, 1, 10, 9]
11 #Neuron 1
12 weights = [0.1, 0.3, -0.6, 0.8, 0.4, 0.2, 0.1, 0.9, 0.5, -0.3]
13 #banyak bias tergantung dari berapa banyak neuron yang ada
14 bias = 6
15
16 #output dari rumus numpy
17 output = np.dot(weights, inputs) + bias
18
19 #print out
20 print(output)
```

Pada line 6 kita menginisialisasikan numpy sebagai metode perhitungan.

Selanjutnya menginisialisasikan variable yang mana pada line 10 kita berikan input sebanyak 10 angka seperti yang diminta pada soal “input layer feature 10”.

Pada line 12 kita masukan weights, dimana banyak angka dalam weight sesuai dengan banyaknya angka layer feature.

Bias yang terdapat pada line 14 tersebut harus sesuai dengan banyaknya neuron yang diminta.

Pada line 17 adalah output dari operasi perkalian antara weights dan inputs dan ditambahkan dengan bias.

Untuk memunculkan output tersebut maka harus di print out seperti pada line 20.

Output:

```
PS C:\Users\Riska Octavia\Documents\Numpy> & "C:/Users/Riska Octavia/AppData/Local/Programs/Python/Python38-32/Python.exe" -i -c "import sys; sys.path.append('C:/Users/Riska Octavia/Documents/Numpy'); import 1A.py"
18.1
PS C:\Users\Riska Octavia\Documents\Numpy>
```

Gambaran matriksnya

Weight

$$\begin{bmatrix} 0.1 \\ 0.3 \\ -0.6 \\ 0.8 \\ 0.4 \\ 0.2 \\ 0.1 \\ 0.9 \\ 0.5 \\ -0.3 \end{bmatrix}$$

Input

$$\times \begin{bmatrix} 5 & 3 & 2 & 6 & 4 & 8 & 7 & 1 & 10 & 9 \end{bmatrix}$$

+ Bias

$$\begin{bmatrix} 6 \end{bmatrix}$$

2. Multi Neuron

Source code:

```
1B.py ×
1B.py > ...
1  #Nama= Riska Octavia Arianto
2  #NIM= 21091397059
3  #Multi Neuron
4
5  #inisialisasi numpy
6  import numpy as np
7
8  #inisialisasi variabel
9  #Input layer feature 10
10 inputs = [3.0, 0.5, 1.0, 0.32, -1.0, 0.27, 7.0, 5.0, 6.0, 10.0]
11
12 #Neuron 5
13 weights = [[1.0, 7.0, 5.0, 0.2, 0.8, 0.2, 0.1, 0.3, 0.5, -0.4],
14            [0.22, 0.24, 0.29, 0.2, 0.8, 0.25, -2.0, 0.33, 0.89, 0.49],
15            [0.34, 0.6, 0.7, 0.23, 0.24, -0.29, -0.46, 0.78, 0.99, -0.1],
16            [3.0, 0.29, 0.9, 6.0, 4.0, 4, 8, 0.75, 0.35, 0.22],
17            [1.0, 0.3, -2.0, 1.0, -0.25, -0.44, 7.0, -0.1, 0.34, -0.56]]
18
19 #banyak bias tergantung dari berapa banyak neuron yang ada
20 biases = [9.0, 4.0, 2.0, 8.0, 3.0]
21
22 #output dari rumus numpy
23 output = np.dot(weights, inputs) + biases
24
25 #print out
26 print(output)
```

Line 6, menginisialisasikan numpy.

Line 8, inisialisasikan variable.

Line 10, memasukan inputs sesuai soal “input layer feature 10”.

Line 13, memasukan weights dengan angka sebanyak angka layer dan baris sebanyak neuron yang diminta.

Line 20, bias yang dimasukkan sesuai dengan banyak neuron

Line 23, output dari operasi perkalian antara weight dan inputs yang selanjutnya ditambah oleh bias.

Line 26, print out dari output untuk memunculkan hasil output tersebut.

Output:

```
PS C:\Users\Riska Octavia\Documents\Numpy> & "C:/Users/Riska Octavia/Ap
mpy/1B.py"
[21.018  2.2915  9.3953 81.095  49.5412]
PS C:\Users\Riska Octavia\Documents\Numpy>
```

Bentuk matriksnya:

Weight

$$\begin{bmatrix} 1.0 & 7.0 & 5.0 & 0.2 & 0.8 & 0.2 & 0.1 & 0.3 & 0.5 & -0.4 \\ 0.22 & 0.24 & 0.29 & 0.2 & 0.8 & 0.25 & -2.0 & 0.33 & 0.89 & 0.49 \\ 0.34 & 0.6 & 0.7 & 0.23 & 0.24 & -0.29 & -0.46 & 0.78 & 0.99 & -0.1 \\ 3.0 & 0.29 & 0.9 & 6.0 & 4.0 & 4 & 8 & 0.75 & 0.35 & 0.22 \\ 1.0 & 0.3 & -2.0 & 1.0 & -0.25 & -0.44 & 7.0 & -0.1 & 0.34 & -0.56 \end{bmatrix}$$

insert

\times

$$\begin{bmatrix} 3.0 & 0.5 & 1.0 & 0.32 & -1.0 & 0.27 & 7.0 & 5.0 & 6.0 & 10.0 \end{bmatrix}$$

+ Biases

$$\begin{bmatrix} 9.0, 4.0, 2.0, 8.0, 3.0 \end{bmatrix}$$

3. Multi Neuron Batch Input

Source code:

```
1C.py > ...
1  #Nama= Riska Octavia Arianto
2  #NIM= 21091397059
3  #Multi Neuron Batch Input
4
5  #inisialisasi numpy
6  import numpy as np
7
8  #inisialisasi variabel
9  #Input layer feature 10
10 #per batch nya 6 input
11 inputs = [[3.0, 2.0, 1.0, 7.0, 5.0, 10.0, 0.33, 9.0, 0.8, 0.1],
12           [4.0, 7.0, 1.5, 2.0, 0.5, 9.0, 1.25, 0.7, 3.0, 1.0],
13           [3.0, 0.5, 1.0, 0.32, -1.0, 0.27, 0.3, 0.9, 6.0, 4.0],
14           [-2.0, 6.0, 1.0, 7.0, 5.0, 0.2, 1.0, 2.0, 3.0, -1.0],
15           [0.7, 2.0, 1.0, 0.3, -5.0, 0.22, 1.0, -0.25, 0.73, 6.0],
16           [0.25, -2.0, 0.33, 9.0, 0.8, 0.1, 0.5, -2.0, 1.0, -0.25]]
17
18 #Neuron 5
19 weights = [[ 0.4, 0.6, 0.2, 2.3, 0.5, -0.4, 9.0, 0.8, 0.1, 0.5],
20            [-0.4, 2.0, 0.89, 3.0, -0.5, 1.4, 2.0, 1.0, 0.3, -5.0],
21            [6.0, -0.6, 8.0, 0.3, 0.25, -0.7, 1.0, 0.32, -1.0, 2.0],
22            [8.0, 0.45, -4.0, 7.0, 5.0, 0.4, 7.0, 5.0, 10.0, 0.33],
23            [9.0, 4.0, -0.2, 0.89, -2.0, -0.4, 0.5, -2.0, 1.0, -0.25]]
24
25 #banyak bias tergantung dari berapa banyak neuron yang ada
26 biases = [8.0, 3.0, 4.0, 9.0, 1.0]
27
28 #output dari rumus numpy
29 output = np.dot(inputs, np.array(weights).T) + biases
30
31 #print out
32 print(output)
33
```

Line 6, inisialisasi metode perhitungan numpy

Line 8, inisialisasi variable.

Line 11, memasukkan input dengan layer feature nya 10 dan batch sebanyak 6.

Line 23, memasukkan weights sesuai dengan neuron yang diminta

Line 30, memasukkan bias yang banyak angkanya sesuai dengan banyaknya neuron yang ada.

Line 33, output operasi dari perkalian antara inputs dan weight yang kemudian dikalikan dengan transpose yang disimbolkan dengan "T" dan kemudian ditambah dengan bias.

Line 36, print out dari output tersebut.

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER

[ 15.35   35.085   30.225   58.25   61.88 ]
[  9.828   5.528   29.357   26.573   31.9768]
[ 29.52   35.47   -0.39   65.78    2.95 ]
[  7.782   11.318   13.686  -11.312   25.279 ]
[ 28.026   25.9337  12.17   75.82    1.554 ]
[ 10.64    5.3925   13.3    22.52   -8.427 ]
[ 10.93   11.767  -13.23   22.15  -34.859 ]
[  4.59    6.975   12.1075   12.4   -6.7825]
[ 19.5    17.79   28.45   60.3    47.98 ]]
PS C:\Users\Riska Octavia\Documents\Numpy>
```

Gambaran matriksnya

$$\begin{array}{c} \text{Input} \\ \left[\begin{array}{cccccccccccc} 3.0 & 2.0 & 1.0 & 7.0 & 5.0 & 10.0 & 0.33 & 9.0 & 0.8 & 0.1 \\ 4.0 & 7.0 & 1.5 & 2.0 & 0.5 & 9.0 & 1.25 & 0.7 & 3.0 & 1.0 \\ 3.0 & 0.5 & 1.0 & 0.32 & -1.0 & 0.27 & 0.3 & 0.9 & 6.0 & 4.0 \\ -2.0 & 6.0 & 1.0 & 7.0 & 5.0 & 0.2 & 1.0 & 2.0 & 3.0 & -1.0 \\ 0.7 & 2.0 & 1.0 & 0.3 & -5.0 & 0.22 & 1.0 & -0.25 & 0.73 & 6.0 \\ 0.25 & -2.0 & 0.33 & 9.0 & 0.8 & 0.1 & 0.5 & -2.0 & 1.0 & -0.25 \end{array} \right] \end{array} \times \begin{array}{c} \text{weight} \\ \left[\begin{array}{cccccccccccc} 0.4 & 0.6 & 0.2 & 2.3 & 0.5 & -0.4 & 9.0 & 0.8 & 0.1 & 0.5 \\ -0.4 & 2.0 & 0.89 & 3.0 & -0.5 & 1.4 & 2.0 & 1.0 & 0.3 & -5.0 \\ 6.0 & -0.6 & 8.0 & 0.3 & 0.25 & -0.7 & 1.0 & 0.32 & -1.0 & 2.0 \\ 8.0 & 0.45 & -4.0 & 7.0 & 5.0 & 0.4 & 7.0 & 5.0 & 10.0 & 0.33 \\ 9.0 & 4.0 & -0.2 & 0.89 & -2.0 & -0.4 & 0.5 & -2.0 & 1.0 & -0.25 \end{array} \right] \end{array}$$
$$\begin{array}{c} + \text{Biases} \\ \left[\begin{array}{cccccc} 8.0 & 3.0 & 4.0 & 9.0 & 1.0 \end{array} \right] \end{array}$$

UTS Part 2

Source code:

```
UTS2.py > ...
1  #Nama= Riska Octavia Arianto
2  #NIM= 21091397059
3  #UTS part 2
4
5  #inisialisasi numpy
6  import numpy as num
7
8  #inisialisasi variabel
9  #Input layer feature 10
10 #per batch nya 6 input
11 inputs = [[3.0, 2.0, 1.0, 7.0, 5.0, 10.0, 0.33, 9.0, 0.8, 0.1],
12 [4.0, 7.0, 1.5, 2.0, 0.5, 9.0, 1.25, 0.7, 3.0, 1.0],
13 [3.0, 0.5, 1.0, 0.32, -1.0, 0.27, 0.3, 0.9, 6.0, 4.0],
14 [-2.0, 6.0, 1.0, 7.0, 5.0, 0.2, 1.0, 2.0, 3.0, -1.0],
15 [0.7, 2.0, 1.0, 0.3, -5.0, 0.22, 1.0, -0.25, 0.73, 6.0],
16 [0.25, -2.0, 0.33, 9.0, 0.8, 0.1, 0.5, -2.0, 1.0, -0.25]]
17
18 #Hidden layer 1, Neuron 5
19 weights1 = [[ 0.4, 0.6, 0.2, 2.3, 0.5, -0.4, 9.0, 0.8, 0.1, 0.5],
20 [-0.4, 2.0, 0.89, 3.0, -0.5, 1.4, 2.0, 1.0, 0.3, -5.0],
21 [6.0, -0.6, 8.0, 0.3, 0.25, -0.7, 1.0, 0.32, -1.0, 2.0],
22 [8.0, 0.45, -4.0, 7.0, 5.0, 0.4, 7.0, 5.0, 10.0, 0.33],
23 [9.0, 4.0, -0.2, 0.89, -2.0, -0.4, 0.5, -2.0, 1.0, -0.25]]
24
25 #banyak bias tergantung dari berapa banyak neuron layer1
26 biases1 = [8.0, 3.0, 4.0, 9.0, 1.0]
27
28 #Hidden layer2, neuron 3
29 weights2 = [[ -0.5, 1.4, 2.0, 2.3, 0.5],
30 [-0.4, 2.0, 0.89, 2.0, -5.0],
31 [0.45, -4.0, 7.0, -1.0, 2.0]]
32
33 #banyak bias tergantung dari berapa banyak neuron layer2
34 biases2 = [8.0, 3.0, 4.0]
35
36 #output layer1
37 layer1_output = num.dot(inputs, num.array(weights1).T) + biases1
38 #output layer2
39 layer2_output = num.dot(layer1_output, num.array(weights2).T) + biases2
40
41 #print out
42 print(layer2_output)
```

Line 6, inisialisasi numpy

Line 11, memasukkan input dengan input layer feature 10 dan per batch nya sebanyak 6 input

Line 19, weight layer 1 sebanyak 5 neuron

Line 26, memasukkan bias sebanyak neuron layer 1

Line 29, weight kedua memiliki neuron sebanyak hasil dari output layer 1 yaitu 3 dan layer feature sebanyak 5

Line 34, bias sebanyak neuron di weight2

Line 37, memanggil output layer 1

Line 39, memanggil output layer 2

Line 42, memprintout hasil layer kedua

Output:

```
[Running] python -u "c:\Users\Riska Octavia\Documents\Numpy\UTS2.py"
[[ 27.5    45.59    23.76   154.243    9.97 ]
 [ 19.96    31.185   26.699    91.83    62.855 ]
 [  7.848  -14.172   27.945    85.493   34.3268]
 [ 31.92    42.37    -7.75   103.45    1.7 ]
 [ 11.655  -19.713   21.876    -5.282   24.509 ]
 [ 22.901   23.4837    6.53    70.2375    5.8665]]
```

Gambaran matriksnya

Layer 1

$$\begin{array}{c} \text{inputs} \\ \left[\begin{array}{ccccccccc} 3.0 & 2.0 & 1.0 & 7.0 & 5.0 & 10.0 & 0.33 & 9.0 & 0.8 & 0.1 \\ 4.0 & 7.0 & 1.5 & 2.0 & 0.5 & 9.0 & 1.25 & 0.7 & 3.0 & 1.0 \\ 3.0 & 0.5 & 1.0 & 0.32 & -1.0 & 0.27 & 0.3 & 0.9 & 6.0 & 4.0 \\ -2.0 & 6.0 & 1.0 & 7.0 & 5.0 & 0.2 & 1.0 & 2.0 & 3.0 & -1.0 \\ 0.7 & 2.0 & 1.0 & 0.3 & -5.0 & 0.22 & 1.0 & -0.25 & 0.73 & 6.0 \\ 0.25 & -2.0 & 0.33 & 9.0 & 0.8 & 0.1 & 0.5 & -2.0 & 1.0 & -0.25 \end{array} \right] \end{array} \quad \begin{array}{c} \text{weight1} \\ \left[\begin{array}{cccccccccc} 0.4 & 0.6 & 0.2 & 2.3 & 0.5 & -0.4 & 9.0 & 0.8 & 0.1 & 0.5 \\ -0.4 & 2.0 & 0.89 & 3.0 & -0.5 & 1.4 & 2.0 & 1.0 & 0.3 & -5.0 \\ 6.0 & -0.6 & 8.0 & 0.3 & 0.25 & -0.7 & 1.0 & 0.32 & -1.0 & 2.0 \\ 8.0 & 0.45 & -4.0 & 7.0 & 5.0 & 0.4 & 7.0 & 5.0 & 10.0 & 0.33 \\ 9.0 & 4.0 & -0.2 & 0.89 & -2.0 & -0.4 & 0.5 & -2.0 & 1.0 & -0.25 \end{array} \right] \end{array}$$
$$\begin{array}{c} \text{num.dot} \\ \left[\begin{array}{ccccc} 27.5 & 45.59 & 23.76 & 154.243 & 9.97 \\ 19.96 & 31.185 & 26.699 & 91.83 & 62.855 \\ 7.848 & -14.172 & 27.945 & 85.493 & 34.3268 \\ 31.92 & 42.37 & -7.75 & 103.45 & 1.7 \\ 11.655 & -19.713 & 21.876 & -5.282 & 24.509 \\ 22.901 & 23.4837 & 6.53 & 70.2375 & 5.8665 \end{array} \right] \end{array} + \begin{array}{c} \text{biases1} \\ \left[\begin{array}{c} 8.0, 3.0, 4.0, 9.0, 1.0 \end{array} \right] \end{array}$$

Hasil Layer 1

```
[ [ 35.5      48.59      27.76      163.243      10.97    ]
  [ 27.96      34.185      30.699      100.83      63.855    ]
  [ 15.848     -11.172      31.945      94.493      35.3268   ]
  [ 39.92      45.37       -3.75      112.45       2.7      ]
  [ 19.655     -16.713      25.876       3.718      25.509    ]
  [ 30.901      26.4837     10.53      79.2375      6.8665   ] ]
```

Layer 2

$$\begin{array}{c} \text{Layer1} \\ \left[\begin{array}{ccccc} 35.5 & 48.59 & 27.76 & 163.243 & 10.97 \\ 27.96 & 34.185 & 30.699 & 100.83 & 63.855 \\ 15.848 & -11.172 & 31.945 & 94.493 & 35.3268 \\ 39.92 & 45.37 & -3.75 & 112.45 & 2.7 \\ 19.655 & -16.713 & 25.876 & 3.718 & 25.509 \\ 30.901 & 26.4837 & 10.53 & 79.2375 & 6.8665 \end{array} \right] \end{array} \times \begin{array}{c} \text{Weight2} \\ \left[\begin{array}{ccccc} -0.5 & 1.4 & 2.0 & 2.3 & 0.5 \\ -0.4 & 2.0 & 0.89 & 2.0 & -5.0 \\ 0.45 & -4.0 & 7.0 & -1.0 & 2.0 \end{array} \right] \end{array}$$

$$+ \begin{array}{c} \text{Biases2} \\ \left[\begin{array}{ccc} 8.0 & 3.0 & 4.0 \end{array} \right] \end{array}$$

Hasil layer 2

```
[ [ 27.5      45.59      23.76      154.243      9.97    ]
  [ 19.96      31.185      26.699      91.83      62.855    ]
  [ 7.848     -14.172      27.945      85.493      34.3268   ]
  [ 31.92      42.37       -7.75      103.45       1.7      ]
  [ 11.655     -19.713      21.876      -5.282      24.509    ]
  [ 22.901      23.4837      6.53      70.2375      5.8665   ] ]
```