

LAPORAN KECERDASAN BUATAN



ALWAN ROFIQI

21091397020

2021 B

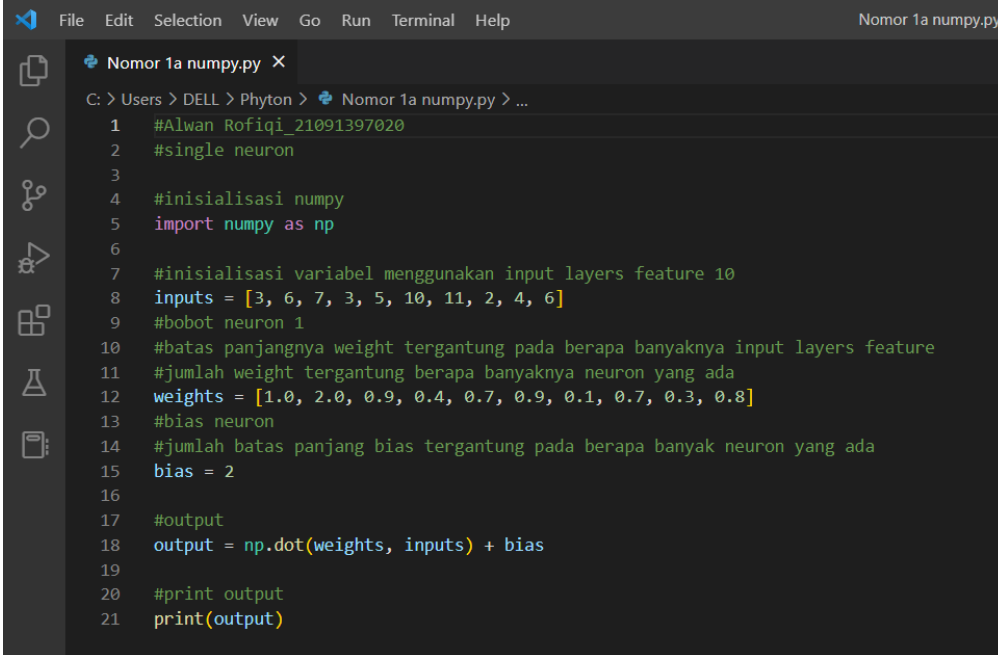
UNIVERSITAS NEGERI SURABAYA

D4 MANAJEMEN INFORMATIKA

REVISI UTS 1

1. Single neuron

Input :



```
File Edit Selection View Go Run Terminal Help
Nomor 1a numpy.py

Nomor 1a numpy.py X
C: > Users > DELL > Python > Nomor 1a numpy.py > ...
1 #Alwan Rofiqi_21091397020
2 #single neuron
3
4 #inisialisasi numpy
5 import numpy as np
6
7 #inisialisasi variabel menggunakan input layers feature 10
8 inputs = [3, 6, 7, 3, 5, 10, 11, 2, 4, 6]
9 #bobot neuron 1
10 #batas panjangnya weight tergantung pada berapa banyaknya input layers feature
11 #jumlah weight tergantung berapa banyaknya neuron yang ada
12 weights = [1.0, 2.0, 0.9, 0.4, 0.7, 0.9, 0.1, 0.7, 0.3, 0.8]
13 #bias neuron
14 #jumlah batas panjang bias tergantung pada berapa banyak neuron yang ada
15 bias = 2
16
17 #output
18 output = np.dot(weights, inputs) + bias
19
20 #print output
21 print(output)
```

Pada **line 8** yaitu inputs dimana pada line ini kita memasukkan angka awal yang mana batasannya sebanyak 10 angka.

Pada **line 12** yaitu weights disini kita memasukkan angka juga yang dimana banyaknya angka sesuai dengan Batasan tadi yaitu 10 angka

Pada **line 15** yaitu bias yang mana bias ini nantinya akan jadi bahan tambah, dimana banyaknya angka yang ada pada bias ini tergantung berapa neuron yang disediakan, terhubung disini neuron yang disediakan hanya 1 maka angka yang di masukkan yaitu hanya satu angka

Pada **line 18** yaitu output yang mana disini terjadinya operasi antara weights, inputs dan bias, disini weights dan inputs terjadi pengoperasian dot yaitu perkalian setelah weights dan inputs itu di kalikan maka Langkah akhir yaitu di tambahkan dengan bias

Pada **line 21** yaitu print (output) ini berfungsi untuk menampilkan hasil pengoperasian tadi

Ouput :

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
[Running] python -u "c:\Users\DELL\Phyton\Nomor 1a numpy.py"
45.5

[Done] exited with code=0 in 0.461 seconds
```

2. Multi Neuron

Input :

```
File Edit Selection View Go Run Terminal Help
Nomor 1a numpy.py Nomor 1b numpy.py X
C: > Users > DELL > Phyton > Nomor 1b numpy.py > ...
1 #Alwan Rofiqi_21091397020
2 #Multi neuron
3
4 #inisialisasi numpy
5 import numpy as np
6
7 #inisialisasi variabel menggunakan input layers feature 10
8 input = [3.0, 8.0, 2.0, 9.0, 4.0, 1.11, 7.0, 5.0, 6.0, 0.11]
9 #bobot neuron 5
10 #batas panjangnya weight tergantung pada berapa banyaknya input layers
11 #jumlah weight tergantung berapa banyaknya neuron yang ada
12 weight = [[0.11, 0.9, 0.13, 0.1, 0.7, 0.31, 0.3, 1.11, 0.7, 3.0],
13           [0.2, 0.45, 0.1, 0.53, 0.8, 0.9, 0.3, 1.23, 0.99, 0.21],
14           [0.5, 0.65, 0.2, 0.8, 0.21, 0.10, 0.4, 0.21, 0.11, 1.0],
15           [0.1, 0.3, 0.41, 0.7, 0.11, 0.6, 0.9, 0.1, 0.3, 0.43],
16           [1.32, 0.1, 0.21, 0.45, 0.6, 0.7, 0.22, 0.8, 0.11, 0.9]]
17 #bias neuron
18 #jumlah batas panjang bias tergantung pada berapa banyak neuron yang ada
19 bias = [7.0, 9.0, 6.0, 3.0, 9.0]
20 #output
21 layer_outputs = np.dot(weight, input) + bias
22
23 #print output
24 print(layer_outputs)
```

Pada **line 8** yaitu inputs dimana pada line ini kita memasukkan angka awal yang mana batasannya sebanyak 10 angka.

Pada **line 12** yaitu weights disini kita memasukkan angka juga yang dimana banyaknya angka sesuai dengan Batasan tadi yaitu 10 angka, dan untuk jumlah nya itu tergantung berapa banyak neuron terhubung disini kita ada 5 neuron maka banyaknya baris yaitu 5

Pada **line 19** yaitu bias yang mana bias ini nantinya akan jadi bahan tambah, dimana banyaknya angka yang ada pada bias ini tergantung berapa neuron yang disediakan, terhubung disini neuron yang disediakan hanya 5 maka angka yang di masukkan yaitu 5 angka

Pada **line 18** yaitu output yang mana disini terjadinya operasi antara weights, inputs dan bias, disini weights dan inputs terjadi pengoperasian dot yaitu perkalian setelah weights dan inputs itu di kalikan maka Langkah akhir yaitu di tambahkan dengan bias

Pada **line 24** yaitu print (output) ini berfungsi untuk menampilkan hasil pengoperasian tadi

Output :

```
[Running] python -u "c:\Users\DELL\Phyton\Nomor 1b numpy.py"
[31.0141 36.5821 25.871 22.5733 27.706 ]

[Done] exited with code=0 in 0.349 seconds
```

3. Multi Neuron Batch input

Input :

```
Go Run Terminal Help
Nomor 1c numpy.py - Phyton - Visual Studio Code

Nomor 1c numpy.py x
Nomor 1c numpy.py > ...
1 #Nama : Alwan Rofiqi_21091397020
2 #Multi Neuron Batch Input
3
4 #inisialisasi numpy
5 import numpy as np
6
7 #inisialisasi variabel dengan matriks 10 x 6 yang mana input layers feature 10 dan input perbatchnya 6
8 inputs = [[1.0, 0.8, 0.9, 1.0, 0.5, 0.6, 0.1, 0.3, 0.4, 1.0],
9           [0.9, 0.2, 1.0, 0.3, 0.7, 0.10, 0.9, 0.2, 0.3, 2.0],
10          [2.0, 0.5, 0.2, 0.8, 0.1, 1.0, 0.3, 0.12, 0.5, 0.2],
11          [0.1, 0.3, 0.8, 0.6, 0.3, 0.9, 0.2, 0.7, 0.8, 0.3],
12          [0.11, 0.4, 1.0, 0.21, 0.6, 0.6, 0.6, 0.4, 0.1, 0.7],
13          [0.4, 0.1, 0.14, 0.4, 0.1, 0.2, 0.2, 0.6, 0.3, 0.5]]
14
15 #bobot per neuron 5
16 #panjang weights tergantung pada berapa banyaknya input layers yang telah ditentukan
17 #jumlah weights itu tergantung pada berapa banyak neuron yang ada
18 weights = [[0.1, 2.0, 0.3, 1.0, 2.0, 3.0, 1.0, 0.3, 3.0, 0.4],
19            [2.0, 0.3, 0.7, 0.5, 1.0, 0.11, 0.5, 1.0, 0.7, 1.0],
20            [1.0, 0.2, 0.5, 0.1, 2.0, 0.2, 2.0, 0.8, 1.0, 0.4],
21            [-0.20, 0.4, 3.0, 1.0, 3.4, 1.9, 6.0, 0.7, 0.3, 0.1],
22            [0.11, 0.4, 1.0, 0.15, 0.3, 0.2, 2.0, 0.6, 1.0, 0.6]]
23
24 #bias neuron
25 #jumlah batas panjang bias tergantung pada berapa banyak neuron yang ada
26 biases = [4.0, 1.0, 0.20, 6.0, 3.0]
27 #output, output ini menggunakan metode numpy
28 layers_outputs= np.dot(inputs, np.array(weights).T) + biases
29 #print output
30 print(layers_outputs)
```

Pada **line 8** yaitu inputs dimana pada line ini kita memasukkan variable matrix yaitu matriks 10 x 6 yang mana banyaknya kolom yaitu 6 dan banyaknya baris itu 10

Pada **line 22** yaitu weights disini kita memasukkan angka juga yang dimana banyaknya angka atau kolom sesuai dengan input perbatchnya yang telah ditentukan yaitu 6 angka, dan untuk jumlah barisnya itu tergantung berapa banyak neuron terhubung disini kita ada 5 neuron maka banyaknya baris yaitu 5

Pada **line 29** yaitu bias yang mana bias ini nantinya akan jadi bahan tambah, dimana banyaknya angka yang ada pada bias ini tergantung berapa neuron yang disediakan, terhubung disini neuron yang disediakan hanya 5 maka angka yang di masukkan yaitu 5 angka

Pada **line 31** yaitu output yang mana disini terjadinya operasi antara weights, inputs dan bias, disini weights dan inputs terjadi pengoperasian dot yaitu perkalian yang mana pada weights diatas itu ada np.array, np array adalah salah satu library yang disediakan oleh Python dalam memudahkan operasi komputasi tipe data numerik setelah weights dan inputs itu di kalikan maka Langkah selanjutnya yaitu dikalikan dengan transpose atau disimbolkan dengan huruf T, Langkah akhir yaitu di tambahkan dengan bias

Pada **line 33** yaitu print (output) ini berfungsi untuk menampilkan hasil pengoperasian tadi

Output :

```
[Running] python -u "c:\Users\DELL\Phyton\Nomor 1c numpy.py"
[[[11.56    6.566    4.27    13.69    6.13 ]
 [ 9.45    7.281    6.15    17.6     7.874 ]
 [11.176    6.72    4.156    11.494    5.262 ]
 [11.68    4.209    3.48    13.79    6.091 ]
 [ 9.621    4.281    4.131    16.508    6.4636]
 [ 6.962    3.66     2.35     9.26     4.714 ]]]

[Done] exited with code=0 in 0.317 seconds
```

UTS 2

INPUT

```
UTS 2 >  uts2.py > [e] layers1_outputs
1  #Nama : Alwan Rofiqi_21091397020
2  #Multi Neuron Batch Input
3
4  #inisialisasi numpy
5  import numpy as np
6
7  #inisialisasi variabel dengan matriks 10 x 6 yang mana input layers feature 10 dan input perbatchnya 6
8  inputs = [[1, 8, 2, 1, 5, 6, 2, 3, 4, 1],
9            [2, 1, 1, 3, 7, 10, 9, 2, 3, 1],
10           [7, 1, 3, 4, 7, 1, 4, 2, 3, 10],
11           [1, 3, 8, 2, 3, 9, 2, 7, 2, 4],
12           [2, 4, 1, 2, 6, 3, 2, 4, 1, 2],
13           [3, 2, 3, 1, 3, 2, 1, 3, 10, 3]],
14
15  #panjang weights tergantung pada berapa banyaknya input layers yang telah ditentukan
16  #jumlah weights itu tergantung pada berapa banyak neuron yang ada
17  weights1 = [[1, 2, 6, 1, 2, 3, 1, 3, 3, 4],
18             [2, 3, 7, 5, 1, 11, 5, 1, 2, 1],
19             [1, 2, 6, 8, 9, 10, 11, 3, 1, 3],
20             [-2, 4, -10, 4, 1, 2, 2, 3, 1, 3],
21             [-3, 5, 1, 4, 1, 6, 8, 10, 6, 11]]
22  #bias neuron
23  #jumlah batas panjang bias tergantung pada berapa banyak neuron yang ada
24  biases1 = [4, 1, 2, 6, 3]
25  #jumlah weights itu tergantung pada berapa banyak neuron pada layer1
26  weights2 = [[3, 1, 3, 3, 4],
27             [7, 8, 1, 6, 1],
28             [2, 5, 2, 2, 4]]
29
30  #jumlah batas panjang bias tergantung pada berapa banyak neuron yang ada
31  biases2 = [4, 2, 6]
32  #output
33  layers1_outputs= np.dot(inputs, np.array(weights1).T) + biases1
34  #output
35  layers2_outputs= np.dot(layers1_outputs, np.array(weights2).T) + biases2
36  #print output
37  print(layers2_outputs)
```

Pada **line 8** yaitu inputs dimana pada line ini kita memasukkan variable matrix yaitu matriks 10 x 6 yang mana banyaknya kolom yaitu 6 dan banyaknya baris itu 10

Pada **line 17** yaitu weights 1 disini kita memasukkan angka juga yang dimana banyaknya angka atau kolom sesuai dengan input perbatchnya yang telah ditentukan yaitu 6 angka, dan untuk jumlah barisnya itu tergantung berapa banyak neuron terhubung disini kita ada 5 neuron maka banyaknya baris yaitu 5

Pada **line 24** yaitu bias 1 yang mana bias ini nantinya akan jadi bahan tambah, dimana banyaknya angka yang ada pada bias ini tergantung berapa neuron yang disediakan, terhubung disini neuron yang disediakan hanya 5 maka angka yang di masukkan yaitu 5 angka

Pada **line 26** yaitu weights 2 disini kita memasukkan angka juga yang dimana banyaknya angka atau kolom sesuai dengan neuron pada layer1 yaitu 5 angka, dan untuk jumlah barisnya itu

tergantung berapa banyak neuron pada layer 2 terhubung disini kita ada 3 neuron maka banyaknya baris yaitu 3

Pada **line 31** yaitu bias 2 yang mana bias ini nantinya akan jadi bahan tambah, dimana banyaknya angka yang ada pada bias ini tergantung berapa neuron yang disediakan, terhubung disini neuron yang disediakan hanya 3 maka angka yang di masukkan yaitu 3 angka

Pada **line 33** yaitu output yang mana disini terjadinya operasi antara weights1, inputs dan bias1, disini weights1 dan inputs terjadi pengoperasian dot yaitu perkalian yang mana pada weights diatas itu ada np.array, np array adalah salah satu library yang disediakan oleh Python dalam memudahkan operasi komputasi tipe data numerik setelah weights dan inputs itu di kalikan maka Langkah selanjutnya yaitu dikalikan dengan transpose atau disimbolkan dengan huruf T, Langkah akhir yaitu di tambahkan dengan bias1

Pada **line 35** yaitu output yang mana disini terjadinya operasi antara layers1_output, weights 2 dan bias2, disini weights1 dan inputs terjadi pengoperasian dot yaitu perkalian yang mana pada weights diatas itu ada np.array, np array adalah salah satu library yang disediakan oleh Python dalam memudahkan operasi komputasi tipe data numerik setelah weights dan inputs itu di kalikan maka Langkah selanjutnya yaitu dikalikan dengan transpose atau disimbolkan dengan huruf T, Langkah akhir yaitu di tambahkan dengan bias2.

Pada **line 37** yaitu print (output) ini berfungsi untuk menampilkan hasil pengoperasian tadi

OUTPUT

```
[Running] python -u "c:\n
[[[1799 2429 2029]
  [2409 3136 2751]
  [2013 2343 2109]
  [2299 3116 2731]
  [1465 1805 1553]
  [1429 1812 1577]]]
```