

# materiNumpy

September 9, 2022

Nama : Muhammad Reesa Roysid

Program : Python for Data Science

## 1 NumPy

### 1.1 Definisi NumPy

NumPy (Numerical Python) library Python yang fokus pada scientific computing. NumPy memiliki kemampuan untuk membentuk objek N-dimensional array, yang mirip dengan list pada Python. Awalnya sebuah pustaka dengan nama Numeric muncul pada tahun 1995 yang digunakan pada bidang ilmiah dan engineering untuk mengolah angka. Namun library Numeric dinilai memiliki kekurangan pada efisiensi waktu proses. Pada tahun 1997, pengembang Python membuat terobosan dengan merilis Numarray sebagai pengganti sepenuhnya Numeric, karena dapat memproses data dengan waktu yang lebih efisien. Namun Numarray masih terdapat kekurangan yaitu menjadi tidak efisien apabila digunakan untuk mengolah data yang jumlahnya sedikit. Lalu pada tahun 2006, Travis Oliphant membuat NumPy sebagai penyempurna dua library sebelumnya dan digunakan hingga sekarang.

### 1.2 Install Numpy

Ada beberapa cara menginstall NumPy diantaranya: 1. Apabila anda menggunakan Anaconda maka NumPy sudah sepaket dalam penginstallan. 2. Apabila anda menggunakan miniconda anda harus menginstall secara manual melalui cmd dengan mengetikkan conda “install numpy”. 3. Apabila anda menggunakan Python biasa anda juga harus menginstall melalui cmd dengan mengetikkan “pip install numpy”.

### 1.3 Inisiasi Numpy

```
[14]: import numpy as np
```

### 1.4 Organisasi Array

#### 1.4.1 Struktur data Array

**Perbedaan List dan Array** List: \* Jumlah data yang dinamis (mutable) \* Dapat menampung berbagai tipe data

Array: \* Jumlah data statis (immutable) \* Hanya dapat menampung satu tipe data

### 1.4.2 Demo List dan array

#### Python List

```
[10]: L = ["Hallo", True, 34, 3.7]
      L
```

```
[10]: ['Hallo', True, 34, 3.7]
```

```
[11]: L.append("Reesa")
      L
```

```
[11]: ['Hallo', True, 34, 3.7, 'Reesa']
```

```
[12]: L.insert(2, "0")
      L
```

```
[12]: ['Hallo', True, '0', 34, 3.7, 'Reesa']
```

```
[13]: L.pop(0)
      L
```

```
[13]: [True, '0', 34, 3.7, 'Reesa']
```

#### Python List

```
[15]: gpasAsList = [4.0, 3.2, 3.5, 4.0]
      gpasAsArr = np.array(gpasAsList)
```

```
[16]: gpasAsArr.dtype
```

```
[16]: dtype('float64')
```

```
[17]: gpasAsArr.size
```

```
[17]: 4
```

#### Study Log Part 1

```
[20]: study_minute = np.zeros(100, np.uint16)
      study_minute
```

```
[20]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
          0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
          0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
          0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
          0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=uint16)
```

```
[21]: %whos
```

Variable	Type	Data/Info
-----		

```

L                list                n=5
autopep8         module              <module 'autopep8' from
'<...>e-packages\\autopep8.py'>
gpasAsArr        ndarray             4: 4 elems, type `float64`, 32 bytes
gpasAsList       list                n=4
json             module              <module 'json' from
'C:\\<...>\\lib\\json\\__init__.py'>
np               module              <module 'numpy' from
'C:\\<...>ges\\numpy\\__init__.py'>
study_minute     ndarray             100: 100 elems, type `uint16`, 200 bytes

```

### 1.4.3 Array multidimensi

```

[22]: student_gpas = np.array([
        [4.0, 3.3, 3.5, 4.0],
        [3.96, 3.92, 4.0, 4.0],
        [3.2, 3.8, 4.0, 4.0]
    ], np.float16)
student_gpas

```

```

[22]: array([[4.  , 3.3 , 3.5 , 4.  ],
            [3.96, 3.92, 4.  , 4.  ],
            [3.2 , 3.8 , 4.  , 4.  ]], dtype=float16)

```

```

[23]: #cek dimensi
student_gpas.ndim

```

```

[23]: 2

```

```

[24]: #cek bentuk matrik
student_gpas.shape

```

```

[24]: (3, 4)

```

```

[25]: student_gpas[2]

```

```

[25]: array([3.2, 3.8, 4. , 4. ], dtype=float16)

```

```

[27]: student_gpas[2,3]

```

```

[27]: 4.0

```

### Study Log MultiDimensi

```

[30]: study_minute = np.array([
        study_minute,
        np.zeros(100, np.uint16)
    ])
study_minute.shape

```

[30] : (2,)

[illegible]

```
[35]: array([132, 122, 128,  44, 136, 129, 101,  95,  50, 132, 151,  64, 104,
          175, 117, 146, 139, 129, 133, 176,  98, 160, 179,  99,  82, 142,
           31, 106, 117,  56,  98,  67, 121, 159,  81, 170,  31,  50,  49,
           87, 179,  51, 116, 177, 118,  78, 171, 117,  88, 123, 102,  44,
           79,  31, 108,  80,  59, 137,  84,  93, 155, 160,  67,  80, 166,
          164,  70,  50, 102, 113,  47, 131, 161, 118,  82,  89,  81,  43,
```

```
81, 38, 119, 52, 82, 31, 159, 57, 113, 71, 121, 140, 91,
70, 37, 106, 64, 127, 110, 58, 93, 79], dtype=uint16)
```

## 1.5 Pemrograman Berbasis Array

### 1.5.1 Array Indexing

```
[52]: [fake_log[1], fake_log[21]]
```

[52]: [122, 160]

```
[38]: fake_log[[1,21]]
```

```
[38]: array([122, 160], dtype=uint16)
```

```
[42]: #membuat array baru menggunakan data array yang lama
index = np.array([
    [1,21],
    [1,2]
])
fake_log[index]
```

```
[42]: array([[122, 160],
            [122, 128]], dtype=uint16)
```

```
[43]: study_minute = np.append(study_minute, [fake_log])
      study_minute
```

[illegible]

```
0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
        dtype=uint16)
,
    132, 122, 128, 44, 136, 129, 101, 95, 50, 132, 151, 64, 104, 175,
    117, 146, 139, 129, 133, 176, 98, 160, 179, 99, 82, 142, 31, 106,
    117, 56, 98, 67, 121, 159, 81, 170, 31, 50, 49, 87, 179, 51, 116,
    177, 118, 78, 171, 117, 88, 123, 102, 44, 79, 31, 108, 80, 59, 137,
    84, 93, 155, 160, 67, 80, 166, 164, 70, 50, 102, 113, 47, 131, 161,
    118, 82, 89, 81, 43, 81, 38, 119, 52, 82, 31, 159, 57, 113, 71,
    121, 140, 91, 70, 37, 106, 64, 127, 110, 58, 93, 79], dtype=object)
```

### 1.5.2 Filter Array dengan kondisi

```
[44]: fake_log
```

```
[44]: array([132, 122, 128,  44, 136, 129, 101,  95,  50, 132, 151,  64, 104,
          175, 117, 146, 139, 129, 133, 176,  98, 160, 179,  99,  82, 142,
           31, 106, 117,  56,  98,  67, 121, 159,  81, 170,  31,  50,  49,
           87, 179,  51, 116, 177, 118,  78, 171, 117,  88, 123, 102,  44,
           79,  31, 108,  80,  59, 137,  84,  93, 155, 160,  67,  80, 166,
          164,  70,  50, 102, 113,  47, 131, 161, 118,  82,  89,  81,  43,
           81,  38, 119,  52,  82,  31, 159,  57, 113,  71, 121, 140,  91,
           70,  37, 106,  64, 127, 110,  58,  93,  79], dtype=uint16)
```

```
[45]: fake_log<60
```

```
[45]: array([False, False, False,  True, False, False, False, False,  True,
          False, False, False, False, False, False, False, False, False,
          False, False, False, False, False, False, False, False, False,
          False, False,  True, False, False, False, False, False, False,
           True,  True,  True, False, False,  True, False, False, False,
          False, False, False, False, False, False,  True, False,  True,
          False, False,  True, False, False, False, False, False, False,
          False, False, False, False,  True, False, False,  True, False,
          False, False, False, False, False,  True, False,  True, False,
           True, False,  True, False,  True, False, False, False, False,
          False, False,  True, False, False, False, False,  True, False,
          False])
```

```
[46]: fake_log[fake_log<60]
```

```
[46]: array([44, 50, 31, 56, 31, 50, 49, 51, 44, 31, 59, 50, 47, 43, 38, 52, 31,
          57, 37, 58], dtype=uint16)
```

```
[47]: #Menggunakan for
result = []
```

```

for value in fake_log:
    if value < 60:
        result.append(value)
np.array(result)

```

```

[47]: array([44, 50, 31, 56, 31, 50, 49, 51, 44, 31, 59, 50, 47, 43, 38, 52, 31,
          57, 37, 58], dtype=uint16)

```

### Study Log dengan kondisi

```

[48]: study_minute = np.zeros(100, np.uint16)
      study_minute = np.append(study_minute, [fake_log])
      study_minute

```

```

[48]: array([ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
            0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
            0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
            0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
            0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
            0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
            0,  0,  0,  0,  0,  0,  0,  0,  0,  0, 132, 122, 128, 44,
          136, 129, 101,  95,  50, 132, 151,  64, 104, 175, 117, 146, 139,
          129, 133, 176,  98, 160, 179,  99,  82, 142,  31, 106, 117,  56,
           98,  67, 121, 159,  81, 170,  31,  50,  49,  87, 179,  51, 116,
          177, 118,  78, 171, 117,  88, 123, 102,  44,  79,  31, 108,  80,
           59, 137,  84,  93, 155, 160,  67,  80, 166, 164,  70,  50, 102,
          113,  47, 131, 161, 118,  82,  89,  81,  43,  81,  38, 119,  52,
           82,  31, 159,  57, 113,  71, 121, 140,  91,  70,  37, 106,  64,
          127, 110,  58,  93,  79], dtype=uint16)

```

```

[51]: study_minute[(study_minute<60) & (study_minute>0)]

```

```

[51]: array([44, 50, 31, 56, 31, 50, 49, 51, 44, 31, 59, 50, 47, 43, 38, 52, 31,
          57, 37, 58], dtype=uint16)

```

### 1.5.3 Slicing Array

```

[60]: practice = np.arange(42)
      practice.shape = (7,6)
      practice

```

```

[60]: array([[ 0,  1,  2,  3,  4,  5],
            [ 6,  7,  8,  9, 10, 11],
            [12, 13, 14, 15, 16, 17],
            [18, 19, 20, 21, 22, 23],
            [24, 25, 26, 27, 28, 29],
            [30, 31, 32, 33, 34, 35],

```

```
[36, 37, 38, 39, 40, 41]])
```

```
[56]: practice[2,1]
```

```
[56]: 13
```

```
[57]: practice[2:5]
```

```
[57]: array([[12, 13, 14, 15, 16, 17],  
          [18, 19, 20, 21, 22, 23],  
          [24, 25, 26, 27, 28, 29]])
```

```
[58]: practice[2:5, 3]
```

```
[58]: array([15, 21, 27])
```

```
[59]: practice[2:5, 3:]
```

```
[59]: array([[15, 16, 17],  
          [21, 22, 23],  
          [27, 28, 29]])
```

#### 1.5.4 Copy and View

##### List

```
[63]: practice_list = [1,2,3]  
      practice_list_copy = practice_list[:]  
      practice_list, practice_list_copy  
      practice_list_copy[0] = 12345  
      practice_list, practice_list_copy  
      # Yang berubah list paling baru
```

```
[63]: ([1, 2, 3], [12345, 2, 3])
```

##### Array

```
[64]: practice
```

```
[64]: array([[ 0,  1,  2,  3,  4,  5],  
          [ 6,  7,  8,  9, 10, 11],  
          [12, 13, 14, 15, 16, 17],  
          [18, 19, 20, 21, 22, 23],  
          [24, 25, 26, 27, 28, 29],  
          [30, 31, 32, 33, 34, 35],  
          [36, 37, 38, 39, 40, 41]])
```

```
[66]: not_copied = practice[:]  
      not_copied[0,0] = 12345  
      practice, not_copied
```



```
# Yang berubah data originalnya
```

```
[66]: (array([[12345,    1,    2,    3,    4,    5],
              [    6,    7,    8,    9,   10,   11],
              [   12,   13,   14,   15,   16,   17],
              [   18,   19,   20,   21,   22,   23],
              [   24,   25,   26,   27,   28,   29],
              [   30,   31,   32,   33,   34,   35],
              [   36,   37,   38,   39,   40,   41]]),
      array([[12345,    1,    2,    3,    4,    5],
              [    6,    7,    8,    9,   10,   11],
              [   12,   13,   14,   15,   16,   17],
              [   18,   19,   20,   21,   22,   23],
              [   24,   25,   26,   27,   28,   29],
              [   30,   31,   32,   33,   34,   35],
              [   36,   37,   38,   39,   40,   41]]))
```

## 1.6 Visualisasi Data

### 1.6.1 Aljabar Linear

```
[68]: order = np.array([
        [2, 0, 0, 0],
        [1, 0, 2, 4],
        [1, 2, 1, 0],
        [4, 6, 0, 2]
    ])
total = np.array([20000, 32000, 20000, 61000])
prices = np.linalg.solve(order, total)
prices
```

```
[68]: array([10000.,  2500.,  5000.,  3000.])
```

### 1.6.2 Fungsi Universal

```
[73]: a,b = np.split(np.arange(1,11), 2)
      a,b
```

```
[73]: (array([1, 2, 3, 4, 5]), array([ 6,  7,  8,  9, 10]))
```

```
[74]: a+b
```

```
[74]: array([ 7,  9, 11, 13, 15])
```

```
[75]: a*b
```

```
[75]: array([ 6, 14, 24, 36, 50])
```

```
[76]: a-b
```

```
[76]: array([-5, -5, -5, -5, -5])
```

```
[77]: b-a
```

```
[77]: array([5, 5, 5, 5, 5])
```

```
[78]: a+3
```

```
[78]: array([4, 5, 6, 7, 8])
```

```
[80]: a+np.repeat(3,5)
```

```
[80]: array([4, 5, 6, 7, 8])
```

### 1.6.3 Operasi Matrix

```
[84]: study_less = study_minute[(study_minute<60) & (study_minute>0)]  
print(study_less)  
np.add.reduce(study_less)
```

```
[44 50 31 56 31 50 49 51 44 31 59 50 47 43 38 52 31 57 37 58]
```

```
[84]: 909
```

```
[85]: np.sum(study_less)
```

```
[85]: 909
```

```
[86]: np.sum(study_minute)
```

```
[86]: 10199
```

### 1.6.4 Visualisasi data menggunakan matplotlib

#### Inisiasi matplotlib

```
[87]: import matplotlib.pyplot as plt
```

```
[88]: study_minute
```

```
[88]: array([[ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  
         0,  0,  0,  0,  0,  0,  0,  0,  0, 132, 122, 128, 44,
```

```
136, 129, 101, 95, 50, 132, 151, 64, 104, 175, 117, 146, 139,
129, 133, 176, 98, 160, 179, 99, 82, 142, 31, 106, 117, 56,
98, 67, 121, 159, 81, 170, 31, 50, 49, 87, 179, 51, 116,
177, 118, 78, 171, 117, 88, 123, 102, 44, 79, 31, 108, 80,
59, 137, 84, 93, 155, 160, 67, 80, 166, 164, 70, 50, 102,
113, 47, 131, 161, 118, 82, 89, 81, 43, 81, 38, 119, 52,
82, 31, 159, 57, 113, 71, 121, 140, 91, 70, 37, 106, 64,
127, 110, 58, 93, 79], dtype=uint16)
```

```
[90]: study_minute[(study_minute>0)]
```

```
[90]: array([132, 122, 128, 44, 136, 129, 101, 95, 50, 132, 151, 64, 104,
175, 117, 146, 139, 129, 133, 176, 98, 160, 179, 99, 82, 142,
31, 106, 117, 56, 98, 67, 121, 159, 81, 170, 31, 50, 49,
87, 179, 51, 116, 177, 118, 78, 171, 117, 88, 123, 102, 44,
79, 31, 108, 80, 59, 137, 84, 93, 155, 160, 67, 80, 166,
164, 70, 50, 102, 113, 47, 131, 161, 118, 82, 89, 81, 43,
81, 38, 119, 52, 82, 31, 159, 57, 113, 71, 121, 140, 91,
70, 37, 106, 64, 127, 110, 58, 93, 79], dtype=uint16)
```

```
[91]: plt.hist(study_minute[(study_minute>0)])
plt.plot
```

```
[91]: <function matplotlib.pyplot.plot(*args, scalex=True, scaley=True, data=None,
**kwargs)>
```

