Pogosian S A_KVBO-07-23_WorkBook 2

September 22, 2024

[1]:
```
import numpy as np

a = np.zeros(64).reshape(8, 8)
a[0::2, 1::2] = 1
a[1::2, 0::2] = 1

print("                        : ", a, sep="\n\n")
```

                        :

```
[[0. 1. 0. 1. 0. 1. 0. 1.]
 [1. 0. 1. 0. 1. 0. 1. 0.]
 [0. 1. 0. 1. 0. 1. 0. 1.]
 [1. 0. 1. 0. 1. 0. 1. 0.]
 [0. 1. 0. 1. 0. 1. 0. 1.]
 [1. 0. 1. 0. 1. 0. 1. 0.]
 [0. 1. 0. 1. 0. 1. 0. 1.]
 [1. 0. 1. 0. 1. 0. 1. 0.]]
```

[2]:
```
matrix = np.tile(np.arange(5), (5, 1))


print("                    0   4: ", matrix, sep="\n\n")
```

                    0   4:

```
[[0 1 2 3 4]
 [0 1 2 3 4]
 [0 1 2 3 4]
 [0 1 2 3 4]
 [0 1 2 3 4]]
```

[3]:
```
matrix = np.random.random((3, 3, 3))
print("                        : ", matrix, sep="\n\n")
```

                        :

```
[[[0.6455588  0.66187321 0.88458828]
  [0.99168515 0.24353967 0.82002836]
```

```
 [0.37264454 0.21198344 0.27402907]]

[[0.01490704 0.12330038 0.88189278]
 [0.60117292 0.13800556 0.17414293]
 [0.84692389 0.3439131  0.35293331]]

[[0.65015167 0.99336299 0.66147603]
 [0.98026167 0.00960463 0.09462423]
 [0.71823769 0.01455659 0.50636635]]]
```

[6]:
```
n = int(input("                : "))

matrix = np.zeros(n ** 2).reshape(n, n)

matrix[0] = 1
matrix[0::1, 0::n-1] = 1
matrix[n - 1] = 1

print(matrix)
```

```
              :  10
[[1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]]
```

[7]:
```
array = np.random.randint(1, 100, size=10)


sorted_array = np.sort(array)[::-1]

print("         :", array)
print("                     :", sorted_array)
```

```
        : [36 95 93 37 44 56 50 43 45 64]
                  : [95 93 64 56 50 45 44 43 37 36]
```

[8]:
```
matrix = np.random.randint(1, 100, size=(3, 4))


print("    :\n", matrix)
```

```
print("       :", matrix.shape)


print("        :", matrix.size)


print("          :", matrix.ndim)
```

```
   :
[[59 18 45 64]
 [32 42 82 51]
 [27 48 72  1]]
       : (3, 4)
        : 12
           : 2
```

[2]:
```
import pandas as pd

def euclidean_distance(a, b):
    #     ,       Series
    if len(a) != len(b):
        raise ValueError("Series a   b                 .")

    #
    return ((a - b) ** 2).sum() ** 0.5

#            :
a = pd.Series([1, 2, 3])
b = pd.Series([4, 6, 8])

distance = euclidean_distance(a, b)
print(f"           : {distance}")
```

```
          : 7.0710678118654755
```

[18]:
```
dataframe = pd.read_csv("adult.csv")

dataframe.head(5)
```

[18]:
```
   age         workclass  education_num      marital_status  \
0   39         State-gov             13       Never-married
1   50  Self-emp-not-inc             13  Married-civ-spouse
2   38           Private              9            Divorced
3   53           Private              7  Married-civ-spouse
4   28           Private             13  Married-civ-spouse

          occupation   relationship   race  gender  hours_per_week  \
```

```
0       Adm-clerical  Not-in-family  White    Male             40
1    Exec-managerial        Husband  White    Male             13
2  Handlers-cleaners  Not-in-family  White    Male             40
3  Handlers-cleaners        Husband  Other    Male             40
4     Prof-specialty           Wife  Other  Female             40

  native_country  capital income_status
0  United-States     2174         <=50K
1  United-States        0         <=50K
2  United-States        0         <=50K
3  United-States        0         <=50K
4          Other        0         <=50K
```

[19]:
```
dataframe.tail(5)
```

[19]:
```
       age     workclass  education_num      marital_status       occupation  \
45217   33       Private             13       Never-married   Prof-specialty
45218   39       Private             13            Divorced   Prof-specialty
45219   38       Private             13  Married-civ-spouse   Prof-specialty
45220   44       Private             13            Divorced     Adm-clerical
45221   35  Self-emp-inc             13  Married-civ-spouse  Exec-managerial

        relationship   race  gender  hours_per_week native_country  capital  \
45217      Own-child  White    Male              40  United-States        0
45218  Not-in-family  White  Female              36  United-States        0
45219        Husband  White    Male              50  United-States        0
45220      Own-child  Other    Male              40  United-States     5455
45221        Husband  White    Male              60  United-States        0

      income_status
45217         <=50K
45218         <=50K
45219         <=50K
45220         <=50K
45221          >50K
```

[20]:
```
dataframe.describe()
```

[20]:
```
                age  education_num  hours_per_week       capital
count  45222.000000   45222.000000    45222.000000  45222.000000
mean      38.547941      10.118460       40.938017   1012.834925
std       13.217870       2.552881       12.007508   7530.315380
min       17.000000       1.000000        1.000000  -4356.000000
25%       28.000000       9.000000       40.000000      0.000000
50%       37.000000      10.000000       40.000000      0.000000
75%       47.000000      13.000000       45.000000      0.000000
max       90.000000      16.000000       99.000000  99999.000000
```

[21]:
```
dataframe.shape
```

[21]:
```
(45222, 12)
```

[24]:
```
dataframe.iloc[4: 7]
```

[24]:
```
   age workclass  education_num         marital_status       occupation  \
4   28   Private             13     Married-civ-spouse   Prof-specialty
5   37   Private             14     Married-civ-spouse  Exec-managerial
6   49   Private              5  Married-spouse-absent    Other-service

    relationship   race  gender  hours_per_week native_country  capital  \
4           Wife  Other  Female              40          Other        0
5           Wife  White  Female              40  United-States        0
6  Not-in-family  Other  Female              16          Other        0

  income_status
4         <=50K
5         <=50K
6         <=50K
```

[31]:
```
dataframe[dataframe["age"] >= 50].head(5)
```

[31]:
```
    age         workclass  education_num      marital_status  \
1    50  Self-emp-not-inc             13  Married-civ-spouse
3    53           Private              7  Married-civ-spouse
7    52  Self-emp-not-inc              9  Married-civ-spouse
20   54           Private              9           Separated
23   59           Private              9            Divorced

           occupation relationship   race  gender  hours_per_week  \
1     Exec-managerial      Husband  White    Male              13
3   Handlers-cleaners      Husband  Other    Male              40
7     Exec-managerial      Husband  White    Male              45
20      Other-service    Unmarried  Other  Female              20
23       Tech-support    Unmarried  White  Female              40

   native_country  capital income_status
1   United-States        0         <=50K
3   United-States        0         <=50K
7   United-States        0          >50K
20  United-States        0         <=50K
23  United-States        0         <=50K
```

[7]:
```
from sklearn.preprocessing import MinMaxScaler, StandardScaler
import pandas as pd

df = pd.read_csv("iris.csv")
```

```
min_max_scaler = MinMaxScaler()
standard_scaler = StandardScaler()

df['sepal.length_normalized'] = min_max_scaler.fit_transform(
    df[['sepal.length']])

df['sepal.width_standardized'] = standard_scaler.fit_transform(
    df[['sepal.width']])

df[['sepal.length', 'sepal.length_normalized', 'sepal.width',
    'sepal.width_standardized']].head()
```

[7]:

	sepal.length	sepal.length_normalized	sepal.width	sepal.width_standardized
0	5.1	0.222222	3.5	1.019004
1	4.9	0.166667	3.0	-0.131979
2	4.7	0.111111	3.2	0.328414
3	4.6	0.083333	3.1	0.098217
4	5.0	0.194444	3.6	1.249201

[ ]: