ELEC 576 / COMP 576 – Assignment 0

Pinfeng Huang

September 16, 2025

Task 1

```
         active environment : base
        active env location : /Users/pinfenghuang/Desktop/Anaconda/anaconda3
                shell level : 1
           user config file : /Users/pinfenghuang/.condarc
      populated config files : /Users/pinfenghuang/.condarc
              conda version : 24.3.0
        conda-build version : 24.1.2
              python version : 3.11.7.final.0
                     solver : libmamba (default)
           virtual packages : __archspec=1=m1
                              __conda=24.3.0=0
                              __osx=13.5=0
                              __unix=0=0
           base environment : /Users/pinfenghuang/Desktop/Anaconda/anaconda3
(writable)
           conda av data dir : /Users/pinfenghuang/Desktop/Anaconda/anaconda3/etc/
       conda av metadata url : None
                channel URLs : https://conda.anaconda.org/conda-forge/osx-arm64
                              https://conda.anaconda.org/conda-forge/noarch
                              https://repo.anaconda.com/pkgs/main/osx-arm64
                              https://repo.anaconda.com/pkgs/main/noarch
                              https://repo.anaconda.com/pkgs/r/osx-arm64
                              https://repo.anaconda.com/pkgs/r/noarch
              package cache : /Users/pinfenghuang/Desktop/Anaconda/anaconda3/pkgs
                              /Users/pinfenghuang/.conda/pkgs
            envs directories : /Users/pinfenghuang/Desktop/Anaconda/anaconda3/envs
                              /Users/pinfenghuang/.conda/envs
                   platform : osx-arm64
                 user-agent : conda/24.3.0 requests/2.31.0 CPython/3.11.7 Darwin/
                    UID:GID : 501:20
                 netrc file : None
                offline mode : False
```

Task 2

```python
import numpy as np
import scipy.linalg as linalg
from scipy.sparse.linalg import cg, eigs
```

```python
a = np.array([[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]])

b = np.array([[2,  -1,  1,  3,  5],
              [4,   6, -2,  1, -4],
              [2,   7,  8, -3,  2],
              [9,  -5,  3, 10,  6],
              [1,   2, -6,  4,  8]])

c = np.array([[3,   0,  1, -2,  4],
              [5,   7, -1,  3,  0],
              [2,  -3,  6,  1, -5],
              [8,   4, -2,  9,  3],
              [6,  -7,  5,  0, 11]])

d = np.array([[6,   2, -3,  1,  7],
              [0,   5,  4, -2, -6],
              [8,  -1,  3,  9,  2],
              [2,  10,  0, -5,  4],
              [7,  -4,  6,  1, 12]])

v = np.array([0.1, 0.2, 0.4, 0.6, 0.8])

x = np.array([[ 4,  -2,   7,   0,   3],
              [ 1,   6,  -5,   8,   2],
              [ 9,   0,   4,  -1,   5],
              [-3,   2,   6,   7,  -4],
              [ 8,  -6,   1,   3,   9]])
m = 3
```

```
n = 2
```

```python
np.ndim(a)
```

```
2
```

```python
np.size(a)
```

```
25
```

```python
np.shape(a)
```

```
(5, 5)
```

```python
a.shape[n-1]
```

```python
np.array([[1., 2., 3.], [4., 5., 6.]])
```

```
array([[1., 2., 3.],
       [4., 5., 6.]])
```

```python
np.block([[a, b], [c, d]])
```

```
array([[ 1,  2,  3,  4,  5,  2, -1,  0,  3,  5],
       [ 6,  7,  8,  9, 10,  4,  6, -2,  1, -4],
       [11, 12, 13, 14, 15,  0,  7,  8, -3,  2],
       [16, 17, 18, 19, 20,  9, -5,  3, 10,  6],
       [21, 22, 23, 24, 25,  1,  2, -6,  4,  8],
       [ 3,  0,  1, -2,  4,  6,  2, -3,  1,  7],
       [ 5,  7, -1,  3,  0,  0,  5,  4, -2, -6],
       [ 2, -3,  6,  1, -5,  8, -1,  3,  9,  2],
       [ 8,  4, -2,  9,  3,  2, 10,  0, -5,  4],
       [ 6, -7,  5,  0, 11,  7, -4,  6,  1, 12]])
```

```python
a[-1]
```

```
array([21, 22, 23, 24, 25])
```

```python
a[1, 4]
```

```
np.int64(10)
```

```python
a[1]
```

```
array([ 6,  7,  8,  9, 10])
```

```python
a[0:5]
```

```
array([[ 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]])
```

```python
a[-5:]
```

```
array([[ 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]])
```

```python
a[0:3, 4:9]
```

```
array([[ 5],
       [10],
       [15]])
```

```python
a[np.ix_([1, 3, 4], [0, 2])]
```

```
array([[ 6,  8],
       [16, 18],
       [21, 23]])
```

```python
a[2:21:2,:]
```

```
array([[11, 12, 13, 14, 15],
       [21, 22, 23, 24, 25]])
```

```python
a[::2, :]
```

```
array([[ 1,  2,  3,  4,  5],
       [11, 12, 13, 14, 15],
       [21, 22, 23, 24, 25]])
```

```python
a[::-1,:]
```

```
array([[21, 22, 23, 24, 25],
       [16, 17, 18, 19, 20],
       [11, 12, 13, 14, 15],
       [ 6,  7,  8,  9, 10],
       [ 1,  2,  3,  4,  5]])
```

```python
a[np.r_[:len(a),0]]
```

```
array([[ 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],
       [ 1,  2,  3,  4,  5]])
```

```python
a.transpose()
```

```
array([[ 1,  6, 11, 16, 21],
       [ 2,  7, 12, 17, 22],
       [ 3,  8, 13, 18, 23],
       [ 4,  9, 14, 19, 24],
       [ 5, 10, 15, 20, 25]])
```

```python
a.conj().transpose()
```

```
array([[ 1,  6, 11, 16, 21],
       [ 2,  7, 12, 17, 22],
       [ 3,  8, 13, 18, 23],
       [ 4,  9, 14, 19, 24],
       [ 5, 10, 15, 20, 25]])
```

```python
a @ b
```

```
array([[ 51,  22,   2,  56,  67],
       [131,  67,  17, 131, 152],
       [211, 112,  32, 206, 237],
       [291, 157,  47, 281, 322],
       [371, 202,  62, 356, 407]])
```

```python
a * b
```

```
array([[   2,   -2,    0,   12,   25],
       [  24,   42,  -16,    9,  -40],
       [   0,   84,  104,  -42,   30],
       [ 144,  -85,   54,  190,  120],
       [  21,   44, -138,   96,  200]])
```

```python
a/b
```

```
array([[ 0.5       , -2.        ,  3.        ,  1.33333333,  1.        ],
       [ 1.5       ,  1.16666667, -4.        ,  9.        , -2.5       ],
```

```
                    [ 5.5       ,  1.71428571,  1.625     , -4.66666667,  7.5       ],
                    [ 1.77777778, -3.4       ,  6.        ,  1.9       ,  3.33333333],
                    [21.        , 11.        , -3.83333333,  6.        ,  3.125     ]])
```

```python
a**3
```

```
    array([[    1,     8,     27,     64,    125],
           [  216,   343,    512,    729,   1000],
           [ 1331,  1728,   2197,   2744,   3375],
           [ 4096,  4913,   5832,   6859,   8000],
           [ 9261, 10648,  12167,  13824,  15625]])
```

```python
(a > 0.5)
```

```
    array([[ True,  True,  True,  True,  True],
           [ True,  True,  True,  True,  True],
           [ True,  True,  True,  True,  True],
           [ True,  True,  True,  True,  True],
           [ True,  True,  True,  True,  True]])
```

```python
np.nonzero(a > 0.5)
```

```
(array([0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4,
        4, 4, 4]),
 array([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]))
```

```python
a[:,np.nonzero(v > 0.5)[0]]
```

```
array([[ 4,  5],
       [ 9, 10],
       [14, 15],
       [19, 20],
       [24, 25]])
```

```python
a[:, v.T > 0.5]
```

```
array([[ 4,  5],
       [ 9, 10],
       [14, 15],
       [19, 20],
       [24, 25]])
```

```python
a[a < 0.5]=0
a
```

```
array([[ 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]])
```

```python
a * (a > 0.5)
```

```
array([[ 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]])
```

```python
a[:] = 3
a
```

```
array([[3, 3, 3, 3, 3],
       [3, 3, 3, 3, 3],
       [3, 3, 3, 3, 3],
       [3, 3, 3, 3, 3],
       [3, 3, 3, 3, 3]])
```

```python
y = x.copy()
y
```

```
array([[ 4, -2,  7,  0,  3],
       [ 1,  6, -5,  8,  2],
       [ 9,  0,  4, -1,  5],
       [-3,  2,  6,  7, -4],
       [ 8, -6,  1,  3,  9]])
```

```python
y = x[1, :].copy()
y
```

```
array([ 1,  6, -5,  8,  2])
```

```python
y = x.flatten()
y
```

```
array([ 4, -2,  7,  0,  3,  1,  6, -5,  8,  2,  9,  0,  4, -1,  5, -3,  2,
        6,  7, -4,  8, -6,  1,  3,  9])
```

```python
np.arange(1., 11.)
```

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

```python
np.arange(10.)
```

```
array([0., 1., 2., 3., 4., 5., 6., 7., 8., 9.])
```

```python
np.arange(1.,11.)[:, np.newaxis]
```

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

```python
np.zeros((3, 4))
```

```
array([[0., 0., 0., 0.],
       [0., 0., 0., 0.],
       [0., 0., 0., 0.]])
```

```python
np.zeros((3, 4, 5))
```

```
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.]]])
```

```python
np.ones((3, 4))
```

```
array([[1., 1., 1., 1.],
       [1., 1., 1., 1.],
       [1., 1., 1., 1.]])
```

```python
np.eye(3)
```

```
array([[1., 0., 0.],
       [0., 1., 0.],
       [0., 0., 1.]])
```

```python
np.diag(a)
```

```
array([3, 3, 3, 3, 3])
```

```python
np.diag(v, 0)
```

```
array([[0.1, 0. , 0. , 0. , 0. ],
       [0. , 0.2, 0. , 0. , 0. ],
       [0. , 0. , 0.4, 0. , 0. ],
       [0. , 0. , 0. , 0.6, 0. ],
```

```
                [0. , 0. , 0. , 0. , 0.8]])
```

```python
from numpy.random import default_rng
rng = default_rng(42)
rng.random((3, 4))
```

```
    array([[0.77395605, 0.43887844, 0.85859792, 0.69736803],
           [0.09417735, 0.97562235, 0.7611397 , 0.78606431],
           [0.12811363, 0.45038594, 0.37079802, 0.92676499]])
```

```python
np.linspace(1,3,4)
```

```
    array([1.        , 1.66666667, 2.33333333, 3.        ])
```

```python
np.mgrid[0:9.,0:6.]
```

```
    array([[[0., 0., 0., 0., 0., 0.],
            [1., 1., 1., 1., 1., 1.],
            [2., 2., 2., 2., 2., 2.],
            [3., 3., 3., 3., 3., 3.],
            [4., 4., 4., 4., 4., 4.],
```

```
       [5., 5., 5., 5., 5., 5.],
       [6., 6., 6., 6., 6., 6.],
       [7., 7., 7., 7., 7., 7.],
       [8., 8., 8., 8., 8., 8.]],

      [[0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.],
       [0., 1., 2., 3., 4., 5.]]])
```

```python
np.ix_(np.r_[0:9.],np.r_[0:6.])
```

```
    (array([[0.],
            [1.],
            [2.],
            [3.],
            [4.],
            [5.],
            [6.],
            [7.],
            [8.]]),
     array([[0., 1., 2., 3., 4., 5.]]))
```

```python
np.meshgrid([1,2,4],[2,4,5])
```

```
(array([[1, 2, 4],
        [1, 2, 4],
        [1, 2, 4]]),
 array([[2, 2, 2],
        [4, 4, 4],
        [5, 5, 5]]))
```

```python
np.tile(a, (m, n))
```

```
array([[ 1,  2,  3,  4,  5,  1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10,  6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15, 11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20, 16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25, 21, 22, 23, 24, 25],
       [ 1,  2,  3,  4,  5,  1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10,  6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15, 11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20, 16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25, 21, 22, 23, 24, 25],
       [ 1,  2,  3,  4,  5,  1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10,  6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15, 11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20, 16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25, 21, 22, 23, 24, 25]])
```

```python
np.concatenate((a,b),1)
```

```
array([[ 1,  2,  3,  4,  5,  2, -1,  1,  3,  5],
```

```
       [ 6,  7,  8,  9, 10,  4,  6, -2,  1, -4],
       [11, 12, 13, 14, 15,  2,  7,  8, -3,  2],
       [16, 17, 18, 19, 20,  9, -5,  3, 10,  6],
       [21, 22, 23, 24, 25,  1,  2, -6,  4,  8]])
```

```python
np.concatenate((a,b))
```

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

```python
a.max()
```

```
np.int64(25)
```

```python
a.max(0)
```

```
array([21, 22, 23, 24, 25])
```

```python
a.max(1)
```

```
array([ 5, 10, 15, 20, 25])
```

```python
np.maximum(a, b)
```

```
array([[ 2,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10],
       [11, 12, 13, 14, 15],
       [16, 17, 18, 19, 20],
       [21, 22, 23, 24, 25]])
```

```python
np.sqrt(v @ v)
```

```
np.float64(1.1)
```

```python
a & b
```

```
array([[ 0,  2,  1,  0,  5],
       [ 4,  6,  8,  1,  8],
       [ 2,  4,  8, 12,  2],
       [ 0, 17,  2,  2,  4],
       [ 1,  2, 18,  0,  8]])
```

```python
a | b
```

```
array([[ 3, -1,  3,  7,  5],
       [ 6,  7, -2,  9, -2],
       [11, 15, 13, -1, 15],
       [25, -5, 19, 27, 22],
       [21, 22, -1, 28, 25]])
```

```python
linalg.inv(a)
```

```
array([[ 9.88788504e+14, -1.27093574e+15,  3.44610070e+15,
        -7.03454818e+15,  3.87059472e+15],
       [ 5.14308655e+15, -9.88505579e+15,  2.90320598e+16,
```

```
            -4.89812984e+16,  2.46912079e+16],
        [-8.80212389e+15,  1.58160893e+16, -4.98904081e+16,
          8.75410440e+16, -4.46646013e+16],
        [-1.78016589e+15,  3.10673182e+15, -1.09976595e+15,
          0.00000000e+00, -2.26799981e+14],
        [ 4.45041473e+15, -7.76682955e+15,  1.85120136e+16,
         -3.15251974e+16,  1.63295986e+16]])
```

```python
linalg.pinv(a)
```

```
    array([[-1.52000000e-01, -9.60000000e-02, -4.00000000e-02,
             1.60000000e-02,  7.20000000e-02],
           [-8.00000000e-02, -5.00000000e-02, -2.00000000e-02,
             1.00000000e-02,  4.00000000e-02],
           [-8.00000000e-03, -4.00000000e-03, -5.56659644e-17,
             4.00000000e-03,  8.00000000e-03],
           [ 6.40000000e-02,  4.20000000e-02,  2.00000000e-02,
            -2.00000000e-03, -2.40000000e-02],
           [ 1.36000000e-01,  8.80000000e-02,  4.00000000e-02,
            -8.00000000e-03, -5.60000000e-02]])
```

```python
np.linalg.matrix_rank(a)
```

```
    np.int64(2)
```

```python
```

```
U, S, Vh = linalg.svd(a); V = Vh.T
print(U, S, Vh)
```

```
    [[-0.09359323 -0.76892152  0.6227015   0.10627514 -0.0307967 ]
     [-0.24363203 -0.49055421 -0.61581665  0.00856407  0.56630072]
     [-0.39367083 -0.2121869  -0.39176853  0.18697875 -0.78202069]
     [-0.54370962  0.06618041  0.14018102 -0.82475027 -0.01167397]
     [-0.69374842  0.34454772  0.24470267  0.52293231  0.25819064]] [7.42540539e+01 3.36
     8.37261245e-17] [[-0.39262281 -0.41913118 -0.44563955 -0.47214793 -0.4986563 ]
     [ 0.667718    0.35260325  0.0374885  -0.27762625 -0.592741  ]
     [ 0.49049199 -0.33928644 -0.67863404  0.41315946  0.11426904]
     [-0.36621757  0.34035142 -0.11468843  0.67319286 -0.53263828]
     [ 0.15906697 -0.68486905  0.57122885  0.27588156 -0.32130833]]
```

```python
D,V = linalg.eig(a)
print(D, V)
```

```
    [ 6.86420807e+01+0.j -3.64208074e+00+0.j -7.20005968e-16+0.j
     -2.30386567e-15+0.j  1.92781376e-16+0.j] [[-0.10797496 -0.67495283 -0.00083256 -0.3
     [-0.25277499 -0.3603897  -0.4202587   0.76783803  0.12604466]
     [-0.39757502 -0.04582657  0.82293922 -0.55648658  0.48174126]
     [-0.54237506  0.26873656 -0.38177209  0.04823335 -0.79322814]
     [-0.68717509  0.58329969 -0.02007587  0.05010877  0.32253931]]
```

```python
D,V = linalg.eig(a, b)
print(D, V)
```

```
    [-4.79851812e+01+0.j -3.13824094e+00+0.j -4.33685013e-16+0.j
      1.22748796e-16+0.j -2.58332354e-16+0.j] [[-0.61426766  0.67034071 -0.1624923   0.3
     [ 0.23779462 -0.13891522 -0.23220322 -0.2601021   0.33272424]
     [ 0.23389451 -0.23936079  0.34460357  0.09133315  0.4643816 ]
     [ 0.69235254 -0.64351202  0.65737172 -0.71351133  0.28904318]
     [-0.17908176  0.24483584 -0.60727977  0.55449245 -0.53215425]]
```

```python
D,V = eigs(a, k=3)
print(D, V)
```

```
[ 6.86420807e+01+0.j -3.64208074e+00+0.j  9.11570922e-16+0.j] [[ 0.10797496+0.j -0.6
 [ 0.25277499+0.j -0.3603897 +0.j -0.27745473+0.j]
 [ 0.39757502+0.j -0.04582657+0.j  0.88515721+0.j]
 [ 0.54237506+0.j  0.26873656+0.j -0.10982316+0.j]
 [ 0.68717509+0.j  0.58329969+0.j -0.29084755+0.j]]
```

```python
Q,R = linalg.qr(a)
print(Q, R)
```

```
[[-0.03419928 -0.77384133  0.57288178  0.26761922 -0.0136536 ]
 [-0.20519567 -0.50783338 -0.46201924 -0.47980472  0.50628613]
 [-0.37619206 -0.24182542 -0.58621584  0.48348797 -0.47179485]
 [-0.54718845  0.02418254  0.26696229 -0.59803865 -0.5206543 ]
 [-0.71818485  0.2901905   0.20839101  0.32673619  0.49981662]] [[-2.92403830e+01 -3
  -3.67642243e+01]
 [ 0.00000000e+00 -1.20912708e+00 -2.41825417e+00 -3.62738125e+00
  -4.83650833e+00]
 [ 0.00000000e+00  0.00000000e+00  1.49036927e-15  6.68353138e-15
   8.39640579e-15]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  1.52677905e-15
   1.29373286e-15]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
  -5.87033973e-16]]
```

```python
cg
```

```
<function scipy.sparse.linalg._isolve.iterative.cg(A, b, x0=None, *, rtol=1e-05, ato
```

```python
np.fft.fft(a)
```

```
array([[ 15. +0.j        ,  -2.5+3.4409548j ,  -2.5+0.81229924j,
         -2.5-0.81229924j,  -2.5-3.4409548j ],
       [ 40. +0.j        ,  -2.5+3.4409548j ,  -2.5+0.81229924j,
         -2.5-0.81229924j,  -2.5-3.4409548j ],
       [ 65. +0.j        ,  -2.5+3.4409548j ,  -2.5+0.81229924j,
         -2.5-0.81229924j,  -2.5-3.4409548j ],
       [ 90. +0.j        ,  -2.5+3.4409548j ,  -2.5+0.81229924j,
         -2.5-0.81229924j,  -2.5-3.4409548j ],
       [115. +0.j        ,  -2.5+3.4409548j ,  -2.5+0.81229924j,
         -2.5-0.81229924j,  -2.5-3.4409548j ]])
```

```python
np.fft.ifft(a)
```

```
array([[ 3. +0.j        , -0.5-0.68819096j, -0.5-0.16245985j,
         -0.5+0.16245985j, -0.5+0.68819096j],
       [ 8. +0.j        , -0.5-0.68819096j, -0.5-0.16245985j,
         -0.5+0.16245985j, -0.5+0.68819096j],
       [13. +0.j        , -0.5-0.68819096j, -0.5-0.16245985j,
         -0.5+0.16245985j, -0.5+0.68819096j],
       [18. +0.j        , -0.5-0.68819096j, -0.5-0.16245985j,
         -0.5+0.16245985j, -0.5+0.68819096j],
       [23. +0.j        , -0.5-0.68819096j, -0.5-0.16245985j,
         -0.5+0.16245985j, -0.5+0.68819096j]])
```

```python
```

```python
np.sort(a)
```

```
array([[ 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]])
```

```python
np.sort(a, axis=1)
```

```
array([[ 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]])
```

```python
I = np.argsort(a[:, 0]); b = a[I,:]
b
```

```
array([[ 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]])
```

```python
np.unique(a)
```

```
array([ 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])
```
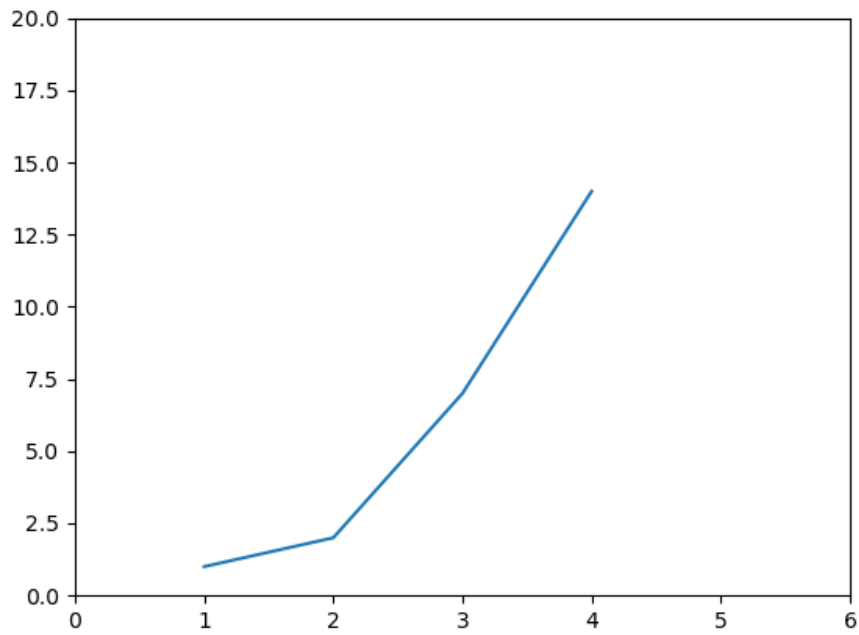
```python
a.squeeze()
```

```
array([[ 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]])
```

```python

```

Task 3

```python
import matplotlib.pyplot as plt
plt.plot([1,2,3,4], [1,2,7,14])
plt.axis([0, 6, 0, 20])
plt.show()
```
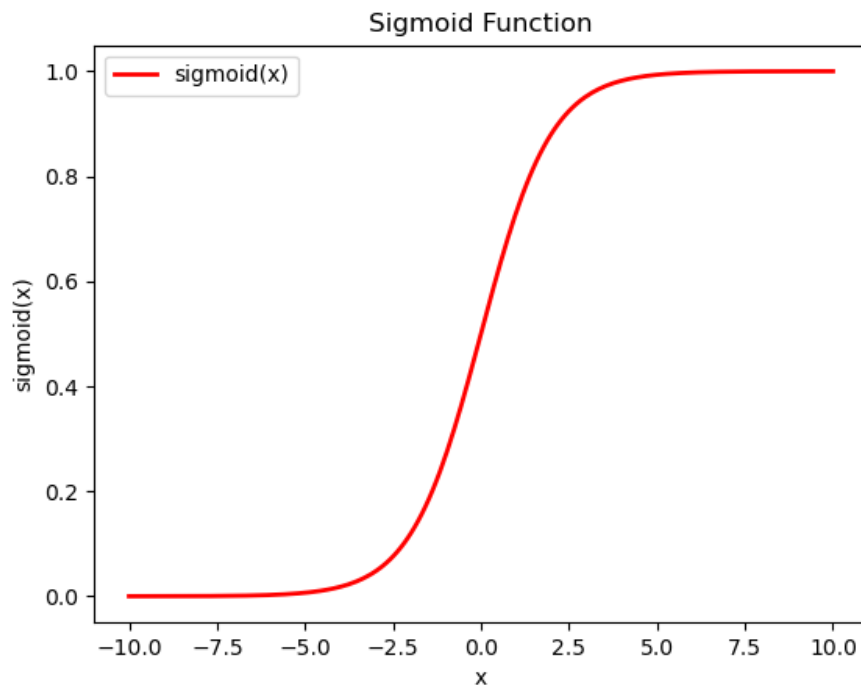
Task 4

import numpy **as** np

```
x = np.linspace(-10, 10, 200)
y = 1 / (1 + np.exp(-x))

plt.plot(x, y, label="sigmoid(x)", color="red", linewidth=2)

plt.title("Sigmoid-Function")
plt.xlabel("x")
plt.ylabel("sigmoid(x)")

plt.legend()

plt.savefig("Task4.png")
```

Sigmoid Function

—

Task 5

https://github.com/PinfengH

—

Task 6

https://github.com/PinfengH/576-intro-dl