

## Task 1:

```
Last login: Tue Sep 12 22:24:04 on ttys000
(base) lixinran@lixinrandeMacBook-Pro ~ % conda list
# packages in environment at /Users/lixinran/anaconda3:
#
# Name                                Version                                Build      Channel
_anaconda_depends                     2023.07                               py311_1
abseil-cpp                            20211102.0                           hc377ac9_0
aiobotocore                           2.5.0                                py311hca03da5_0
aiofiles                              22.1.0                                py311hca03da5_0
aiohttp                               3.8.3                                py311h80987f9_0
aioitertools                          0.7.1                                pyhd3eb1b0_0
aiosignal                             1.2.0                                pyhd3eb1b0_0
aiosqlite                             0.18.0                                py311hca03da5_0
alabaster                             0.7.12                                pyhd3eb1b0_0
anaconda-catalogs                     0.2.0                                py311hca03da5_0
anaconda-client                       1.12.0                                py311hca03da5_0
anaconda-navigator                    2.4.2                                py311hca03da5_0
anaconda-project                      0.11.1                                py311hca03da5_0
anyio                                  3.5.0                                py311hca03da5_0
appdirs                               1.4.4                                pyhd3eb1b0_0
applaunchservices                     0.3.0                                py311hca03da5_0
appnope                               0.1.2                                py311hca03da5_1001
appscript                             1.1.2                                py311h80987f9_0
argon2-cffi                           21.3.0                                pyhd3eb1b0_0
```

```
Last login: Tue Sep 12 22:56:34 on ttys000
(base) lixinran@lixinrandeMacBook-Pro ~ % conda info

      active environment : base
      active env location : /Users/lixinran/anaconda3
            shell level : 1
       user config file : /Users/lixinran/.condarc
 populated config files : /Users/lixinran/.condarc
         conda version : 23.7.2
    conda-build version : 3.26.0
         python version : 3.11.4.final.0
       virtual packages : __archspec=1=arm64
                          __osx=12.5.1=0
                          __unix=0=0
      base environment : /Users/lixinran/anaconda3  (writable)
      conda av data dir : /Users/lixinran/anaconda3/etc/conda
      conda av metadata url : None
            channel URLs : 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/lixinran/anaconda3/pkgs
                       /Users/lixinran/.conda/pkgs
      envs directories : /Users/lixinran/anaconda3/envs
                       /Users/lixinran/.conda/envs
                platform : osx-arm64
            user-agent : conda/23.7.2 requests/2.31.0 CPython/3.11.4 Darwin/21.6.0 OSX/12.5.1
                UID:GID : 501:20
                netrc file : None
        offline mode : False
```

## Task 2:

```
In [1]: import numpy as np
        from scipy import io, integrate, linalg, signal
        from scipy.sparse.linalg import cg, eigs
```

```
In [2]: a = np.array([[1., 2., 3.], [4., 5., 6.]])
```

```
In [3]: np.ndim(a)
```

```
Out[3]: 2
```

```
In [4]: np.size(a)
```

```
Out[4]: 6
```

```
In [5]: np.shape(a)
```

```
Out[5]: (2, 3)
```

```
In [6]: n=2
        a.shape[n-1]
```

```
Out[6]: 3
```

```
In [7]: A = np.eye(2) * 2
        B = np.zeros((2, 3))
        C = np.ones((3, 2))
        D = np.eye(3) * 3
        np.block([[A, B],[C, D]])
```

```
Out[7]: array([[2., 0., 0., 0., 0.],
               [0., 2., 0., 0., 0.],
               [1., 1., 3., 0., 0.],
               [1., 1., 0., 3., 0.],
               [1., 1., 0., 0., 3.]])
```

```
In [8]: a = np.block([[A, B],[C, D]])
        a[-1]
```

```
Out[8]: array([1., 1., 0., 0., 3.])
```

```
In [9]: a[1, 4]
```

```
Out[9]: 0.0
```

```
In [10]: a[1]
```

```
Out[10]: array([0., 2., 0., 0., 0.])
```

```
In [11]: a[0: 5]
```

```
Out[11]: array([[2., 0., 0., 0., 0.],
               [0., 2., 0., 0., 0.],
               [1., 1., 3., 0., 0.],
               [1., 1., 0., 3., 0.],
               [1., 1., 0., 0., 3.]])
```

```
In [12]: a[-5: ]
```

```
Out[12]: array([[2., 0., 0., 0., 0.],
               [0., 2., 0., 0., 0.],
               [1., 1., 3., 0., 0.],
               [1., 1., 0., 3., 0.],
               [1., 1., 0., 0., 3.]])
```

```
In [13]: a = np.block([[A, B, A, B], [C, D, C, D]])
        a
```

```
Out[13]: array([[2., 0., 0., 0., 0., 2., 0., 0., 0., 0.],
               [0., 2., 0., 0., 0., 0., 2., 0., 0., 0.],
               [1., 1., 3., 0., 0., 1., 1., 3., 0., 0.],
               [1., 1., 0., 3., 0., 1., 1., 0., 3., 0.],
               [1., 1., 0., 0., 3., 1., 1., 0., 0., 3.]])
```

```
In [14]: np.shape(a)
```

```
Out[14]: (5, 10)
```

```
In [15]: a[0:3, 4:9]
```

```
Out[15]: array([[0., 2., 0., 0., 0.],
               [0., 0., 2., 0., 0.],
               [0., 1., 1., 3., 0.]])
```

```
In [16]: a[np.ix_([1, 3, 4], [0, 2])]
```

```
Out[16]: array([[0., 0.],
               [1., 0.],
               [1., 0.]])
```

```
In [17]: a[2:21:2,:]
```

```
Out[17]: array([[1., 1., 3., 0., 0., 1., 1., 3., 0., 0.],
               [1., 1., 0., 0., 3., 1., 1., 0., 0., 3.]])
```

```
In [18]: a[:,2, :]
```

```
Out[18]: array([[2., 0., 0., 0., 0., 2., 0., 0., 0., 0.],
               [1., 1., 3., 0., 0., 1., 1., 3., 0., 0.],
               [1., 1., 0., 0., 3., 1., 1., 0., 0., 3.]])
```

```
In [19]: a[:,:-1,:]
```

```
Out[19]: array([[1., 1., 0., 0., 3., 1., 1., 0., 0., 3.],
               [1., 1., 0., 3., 0., 1., 1., 0., 3., 0.],
               [1., 1., 3., 0., 0., 1., 1., 3., 0., 0.],
               [0., 2., 0., 0., 0., 0., 2., 0., 0., 0.],
               [2., 0., 0., 0., 0., 2., 0., 0., 0., 0.]])
```

```
In [20]: a[np.r_[:len(a),0]]
```

```
Out[20]: array([[2., 0., 0., 0., 0., 2., 0., 0., 0., 0.],
               [0., 2., 0., 0., 0., 0., 2., 0., 0., 0.],
               [1., 1., 3., 0., 0., 1., 1., 3., 0., 0.],
               [1., 1., 0., 3., 0., 1., 1., 0., 3., 0.],
               [1., 1., 0., 0., 3., 1., 1., 0., 0., 3.],
               [2., 0., 0., 0., 0., 2., 0., 0., 0., 0.]])
```

```
In [21]: a.transpose()
```

```
Out[21]: array([[2., 0., 1., 1., 1.],
               [0., 2., 1., 1., 1.],
               [0., 0., 3., 0., 0.],
               [0., 0., 0., 3., 0.],
               [0., 0., 0., 0., 3.],
               [2., 0., 1., 1., 1.],
               [0., 2., 1., 1., 1.],
               [0., 0., 3., 0., 0.],
               [0., 0., 0., 3., 0.],
               [0., 0., 0., 0., 3.]])
```

```
In [22]: a.conj().transpose()
```

```
Out[22]: array([[2., 0., 1., 1., 1.],
               [0., 2., 1., 1., 1.],
               [0., 0., 3., 0., 0.],
               [0., 0., 0., 3., 0.],
               [0., 0., 0., 0., 3.],
               [2., 0., 1., 1., 1.],
               [0., 2., 1., 1., 1.],
               [0., 0., 3., 0., 0.],
               [0., 0., 0., 3., 0.],
               [0., 0., 0., 0., 3.]])
```

```
In [23]: b = a.transpose()
```

```
In [24]: a@b
```

```
Out[24]: array([[ 8.,  0.,  4.,  4.,  4.],
               [ 0.,  8.,  4.,  4.,  4.],
               [ 4.,  4., 22.,  4.,  4.],
               [ 4.,  4.,  4., 22.,  4.],
               [ 4.,  4.,  4.,  4., 22.]])
```

```
In [25]: b=a-1
```

```
In [26]: a*b
```

```
Out[26]: array([[ 2., -0., -0., -0., -0.,  2., -0., -0., -0., -0.],
               [-0.,  2., -0., -0., -0., -0.,  2., -0., -0., -0.],
               [ 0.,  0.,  6., -0., -0.,  0.,  0.,  6., -0., -0.],
               [ 0.,  0., -0.,  6., -0.,  0.,  0., -0.,  6., -0.],
               [ 0.,  0., -0., -0.,  6.,  0.,  0., -0., -0.,  6.]])
```

```
In [27]: a/b
```

```
/var/folders/js/_pjj81gd59qf43_p9d8hlyvc0000gn/T/ipykernel_4422/1348051284.py:1: RuntimeWarning: divide by zero encountered in divide
a/b
```

```
Out[27]: array([[ 2., -0., -0., -0., -0.,  2., -0., -0., -0., -0.],
               [-0.,  2., -0., -0., -0., -0.,  2., -0., -0., -0.],
               [ inf,  inf, 1.5, -0., -0.,  inf,  inf, 1.5, -0., -0.],
               [ inf,  inf, -0., 1.5, -0.,  inf,  inf, -0., 1.5, -0.],
               [ inf,  inf, -0., -0., 1.5,  inf,  inf, -0., -0., 1.5.]])
```

```
In [28]: a**3
Out[28]: array([[ 8.,  0.,  0.,  0.,  0.,  8.,  0.,  0.,  0.,  0.],
 [ 0.,  8.,  0.,  0.,  0.,  0.,  8.,  0.,  0.,  0.],
 [ 1.,  1., 27.,  0.,  0.,  1.,  1., 27.,  0.,  0.],
 [ 1.,  1.,  0., 27.,  0.,  1.,  1.,  0., 27.,  0.],
 [ 1.,  1.,  0.,  0., 27.,  1.,  1.,  0.,  0., 27.]])
```

```
In [29]: (a > 0.5)
Out[29]: array([[ True, False, False, False, False,  True, False, False, False,
        False],
 [False,  True, False, False, False, False,  True, False, False,
        False],
 [ True,  True,  True, False, False,  True,  True,  True, False,
        False],
 [ True,  True, False,  True, False,  True,  True, False,  True,
        False],
 [ True,  True, False, False,  True,  True,  True, False, False,
        True]])
```

```
In [30]: a
Out[30]: array([[2.,  0.,  0.,  0.,  0.,  2.,  0.,  0.,  0.,  0.],
 [0.,  2.,  0.,  0.,  0.,  0.,  2.,  0.,  0.,  0.],
 [1.,  1.,  3.,  0.,  0.,  1.,  1.,  3.,  0.,  0.],
 [1.,  1.,  0.,  3.,  0.,  1.,  1.,  0.,  3.,  0.],
 [1.,  1.,  0.,  0.,  3.,  1.,  1.,  0.,  0.,  3.]])
```

```
In [31]: v = a[0:1, :]
v
Out[31]: array([[2.,  0.,  0.,  0.,  0.,  2.,  0.,  0.,  0.,  0.]])
```

```
In [32]: a[:,np.nonzero(v > 0.5)[0]]
Out[32]: array([[2.,  2.],
 [0.,  0.],
 [1.,  1.],
 [1.,  1.],
 [1.,  1.]])
```

```
In [33]: a[a < 0.5]=0
```

```
In [34]: a * (a > 0.5)
Out[34]: array([[2.,  0.,  0.,  0.,  0.,  2.,  0.,  0.,  0.,  0.],
 [0.,  2.,  0.,  0.,  0.,  0.,  2.,  0.,  0.,  0.],
 [1.,  1.,  3.,  0.,  0.,  1.,  1.,  3.,  0.,  0.],
 [1.,  1.,  0.,  3.,  0.,  1.,  1.,  0.,  3.,  0.],
 [1.,  1.,  0.,  0.,  3.,  1.,  1.,  0.,  0.,  3.]])
```

```
In [35]: a[:,] = 3
```

```
In [36]: x=a
y = x.copy()
```

```
In [37]: np.arange(1., 11.)
Out[37]: array([ 1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9., 10.])
```

```
In [38]: np.arange(10.)
Out[38]: array([0.,  1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9.])
```

```
In [39]: np.arange(1.,11.)[:, np.newaxis]
```

```
Out[39]: array([[ 1.],
 [ 2.],
 [ 3.],
 [ 4.],
 [ 5.],
 [ 6.],
 [ 7.],
 [ 8.],
 [ 9.],
 [10.]])
```

```
In [40]: np.zeros((3, 4))
Out[40]: array([[0.,  0.,  0.,  0.],
 [0.,  0.,  0.,  0.],
 [0.,  0.,  0.,  0.]])
```

```
In [41]: np.zeros((3, 4, 5))
```

```
Out[41]: 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.]])
```

```
In [42]: np.ones((3, 4))
```

```
Out[42]: array([[1., 1., 1., 1.],
                [1., 1., 1., 1.],
                [1., 1., 1., 1.]])
```

```
In [43]: np.eye(3)
```

```
Out[43]: array([[1., 0., 0.],
               [0., 1., 0.],
               [0., 0., 1.]])
```

```
In [44]: np.diag(a)
```

```
Out[44]: array([3., 3., 3., 3., 3.])
```

```
In [45]: np.diag(v, 0)
```

```
Out[45]: array([3.])
```

```
In [46]: np.linspace(1,3,4)
```

```
Out[46]: array([1.          , 1.66666667, 2.33333333, 3.          ])
```

```
In [47]: np.mgrid[0:9.,0:6.]
```

[illegible]

```
In [48]: from numpy.random import default_rng
```

```
In [49]: np.mgrid[0:9.,0:6.]
```

[illegible]

```
In [50]: np.ix_(np.r_[0:9.], np.r_[0:6.])
```

```
Out[50]: (array([[0.],
                [1.],
                [2.],
                [3.],
                [4.],
                [5.],
                [6.],
                [7.],
                [8.]])
         array([0., 1., 2., 3., 4., 5.])))
```

```
In [51]: np.meshgrid([1,2,4],[2,4,5])
```

```
Out[51]: [array([[1, 2, 4],
                [1, 2, 4],
                [1, 2, 4]]),
          array([[2, 2, 2],
                [4, 4, 4],
                [5, 5, 5]])]
```

```
In [52]: np.ix_([1,2,4],[2,4,5])
```

```
Out[52]: (array([[1],
                [2],
                [4]]),
          array([[2, 4, 5]]))
```

```
In [53]: m=3
          np.tile(a, (m, n))
```

[illegible]

```
In [54]: np.concatenate((a,b),1)
Out[54]: array([[ 3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  1., -1., -1.,
        -1., -1.,  1., -1., -1., -1., -1.],
       [ 3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3., -1.,  1., -1.,
        -1., -1., -1.,  1., -1., -1., -1.],
       [ 3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  0.,  0.,  2.,
        -1., -1.,  0.,  0.,  2., -1., -1.],
       [ 3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  0.,  0., -1.,
         2., -1.,  0.,  0., -1.,  2., -1.],
       [ 3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  0.,  0., -1.,
        -1.,  2.,  0.,  0., -1., -1.,  2.]])
```

```
In [55]: np.concatenate((a,b))
Out[55]: 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.,  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.],
       [ 1., -1., -1., -1., -1.,  1., -1., -1., -1., -1.],
       [-1.,  1., -1., -1., -1., -1.,  1., -1., -1., -1.],
       [ 0.,  0.,  2., -1., -1.,  0.,  0.,  2., -1., -1.],
       [ 0.,  0., -1.,  2., -1.,  0.,  0., -1.,  2., -1.],
       [ 0.,  0., -1., -1.,  2.,  0.,  0., -1., -1.,  2.]])
```

```
In [56]: a.max()
```

```
Out[56]: 3.0
```

```
In [57]: a.max(0)
```

```
Out[57]: array([3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.])
```

```
In [58]: a.max(1)
```

```
Out[58]: array([3.,  3.,  3.,  3.,  3.])
```

```
In [59]: np.maximum(a, b)
```

```
Out[59]: 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.,  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.]])
```

```
In [60]: v = v[0:5, 0:5]
```

```
In [61]: np.logical_or(a,b)
```

```
Out[61]: 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,  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]])
```

```
In [62]: a = np.array([2, 5, 255])
b = np.array([3, 14, 16])
a&b
```

```
Out[62]: array([ 2,  4, 16])
```

```
In [63]: a|b
```

```
Out[63]: array([ 3, 15, 255])
```

```
In [64]: a = np.array([[1., 2.], [3., 4.]])
linalg.inv(a)
```

```
Out[64]: array([[ -2. ,  1. ],
       [  1.5, -0.5]])
```

```
In [65]: linalg.pinv(a)
```

```
Out[65]: array([[ -2. ,  1. ],
       [  1.5, -0.5]])
```

```

In [66]: np.linalg.matrix_rank(a)
Out[66]: 2

In [67]: b = np.array([1, 2])
          linalg.solve(a, b)
Out[67]: array([0. , 0.5])

In [68]: U, S, Vh = linalg.svd(a);
          V = Vh.T

In [69]: a = np.array([[0., -1.], [1., 0.]])
          b = np.array([[0., 1.], [1., 1.]])
          linalg.cholesky(np.matrix(A))
Out[69]: array([[1.41421356, 0.          ],
                [0.          , 1.41421356]])

In [70]: D,V = linalg.eig(a)
          D,V
Out[70]: (array([0.+1.j, 0.-1.j]),
          array([[0.70710678+0.j, 0.70710678-0.j],
                [0.          -0.70710678j, 0.          +0.70710678j]]))

In [71]: D,V = linalg.eig(a, b)
          D,V
Out[71]: (array([ 1.+0.j, -1.+0.j]),
          array([[ 1.          , -0.4472136 ],
                [ 0.          ,  0.89442719]]))

In [72]: D,V = eigs(a, k=3)
          D,V
/Users/lixinran/anaconda3/lib/python3.11/site-packages/scipy/sparse/linalg/_eigen/arnoldi.py:1272: RuntimeWarning: k >= N - 1 for N * N square matrix. Attempting to use scipy.linalg.eig instead.
  warnings.warn("k >= N - 1 for N * N square matrix. ")
Out[72]: (array([0.+1.j, 0.-1.j]),
          array([[0.70710678+0.j, 0.70710678-0.j],
                [0.          -0.70710678j, 0.          +0.70710678j]]))

In [73]: Q,R = linalg.qr(a)
          Q,R
Out[73]: (array([[ 0., -1.],
                [-1.,  0.]]),
          array([[ -1.,  0.],
                [ 0.,  1.]])

In [74]: P,L,U = linalg.lu(a)
          P,L,U
Out[74]: (array([[0., 1.],
                [1., 0.]]),
          array([[1., 0.],
                [0., 1.]]),
          array([[ 1.,  0.],
                [ 0., -1.]])

In [75]: np.fft.fft(a)
Out[75]: array([[ -1.+0.j,  1.+0.j],
                [ 1.+0.j,  1.+0.j]])

In [76]: np.fft.ifft(a)
Out[76]: array([[ -0.5+0.j,  0.5+0.j],
                [ 0.5+0.j,  0.5+0.j]])

In [77]: np.sort(a)
Out[77]: array([[ -1.,  0.],
                [ 0.,  1.]])

In [78]: np.sort(a, axis=1)
Out[78]: array([[ -1.,  0.],
                [ 0.,  1.]])

In [79]: I = np.argsort(a[:, 0]);
          I
Out[79]: array([0, 1])

```



```

In [80]: b = a[I,:]
          b

Out[80]: array([[ 0., -1.],
                [ 1.,  0.]])

In [81]: z = np.array([[ 0., 1.],
                       [ 1., 1.],
                       [ 2., 1.],
                       [ 3., 1.]])
          y = np.array([-1, 0.2, 0.9, 2.1])
          x = linalg.lstsq(z, y)
          x

Out[81]: (array([ 1. , -0.95]), 0.05, 2, array([4.10003045, 1.09075677]))

In [82]: x = np.linspace(0, 10, 20, endpoint=False)
          np.linspace(0, 10, 20, endpoint=False)
          y = np.cos(-x**2/6.0)
          f = signal.resample(y, 100)
          f

Out[82]: array([ 1.          ,  1.23283311,  1.31727019,  1.27302765,  1.14740269,
                 0.99913207,  0.88078847,  0.82497964,  0.83806509,  0.90249922,
                 0.98614323,  1.05486   ,  1.08402435,  1.06536954,  1.00747904,
                 0.93050762,  0.85758366,  0.80620995,  0.78263328,  0.78081837,
                 0.78588726,  0.78033048,  0.75048699,  0.69094819,  0.60552319,
                 0.50478221,  0.40143727,  0.30548407,  0.22092394,  0.14511765,
                 0.0707372 , -0.01068719, -0.10512733, -0.21370903, -0.33240564,
                -0.45366208, -0.56910426, -0.67215691, -0.75955883, -0.83131707,
                -0.88932657, -0.93540864, -0.9696874 , -0.98998822, -0.9924354 ,
                -0.97288395, -0.92847886, -0.85863231, -0.76502853, -0.65074937,
                -0.51903563, -0.37236513, -0.21236454, -0.04065068,  0.13978552,
                 0.3233596 ,  0.50154373,  0.66375109,  0.79927642,  0.89960584,
                 0.96017029,  0.98077748,  0.96447814,  0.91530208,  0.8358487 ,
                 0.7258814 ,  0.58274561,  0.40370101,  0.18940426, -0.0528479 ,
                -0.30761458, -0.55253797, -0.76231625, -0.91434367, -0.99416302,
                -0.99876133, -0.93639432, -0.82285784, -0.67549774, -0.50723015,
                -0.3230094 , -0.12038944,  0.1057018 ,  0.35559088,  0.61779519,
                 0.86547229,  1.0586815 ,  1.15286077,  1.11163922,  0.9202326 ,
                 0.59492066,  0.18484931, -0.23549411, -0.5819343 , -0.78050802,
                -0.78611604, -0.59448055, -0.24372704,  0.19511472,  0.63613281])

```

```

In [83]: np.unique(a)

Out[83]: array([-1.,  0.,  1.])

In [84]: a.squeeze()

Out[84]: array([[ 0., -1.],
                [ 1.,  0.]])

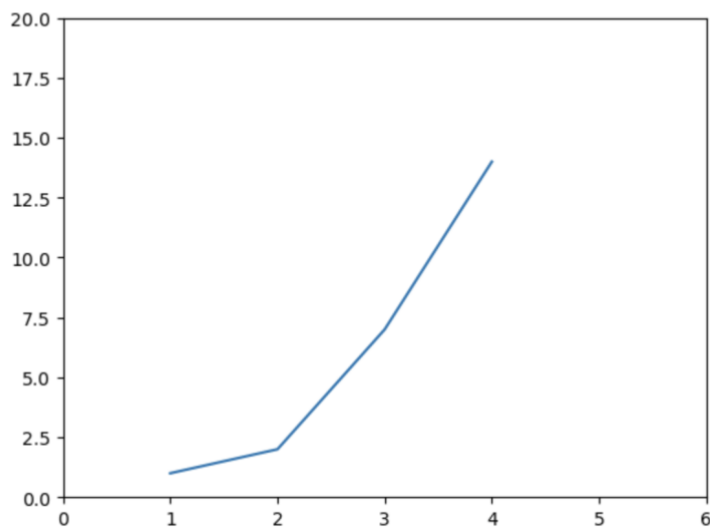
```

## Task 3:

```

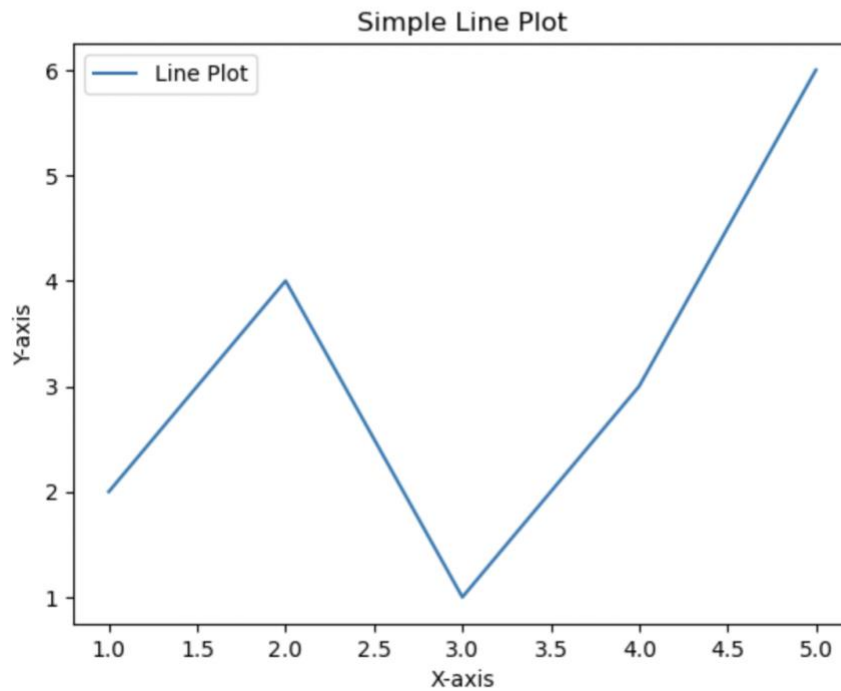
In [1]: import matplotlib.pyplot as plt
          plt.plot([1,2,3,4], [1,2,7,14])
          plt.axis([0, 6, 0, 20])
          plt.show()

```



## Task 4:

```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [2, 4, 1, 3, 6]
fig, ax = plt.subplots()
ax.plot(x, y, label='Line Plot')
ax.set_xlabel('X-axis')
ax.set_ylabel('Y-axis')
ax.set_title('Simple Line Plot')
ax.legend()
plt.show()
```



## Task 5:

Github account: [xl153@rice.edu](mailto:xl153@rice.edu)

## Task 6:

<https://github.com/XinranLi123/comp576.git>