Assignment 0

Work Summary:

I completed all the tasks shown below.

Task 1:

In your terminal, run: `conda info`

Result:

I performed this task on my Windows PC. This was before I configured Anaconda to my liking, but hopefully shows what you were looking for.

```
(base) C:\Users\Gundam>conda info

     active environment : base
    active env location : D:\anaconda3
            shell level : 1
       user config file : C:\Users\Gundam\.condarc
 populated config files : C:\Users\Gundam\.condarc
          conda version : 23.7.2
    conda-build version : 3.26.0
         python version : 3.11.4.final.0
       virtual packages : __archspec=1=x86_64
                          __cuda=12.2=0
                          __win=0=0
       base environment : D:\anaconda3  (writable)
      conda av data dir : D:\anaconda3\etc\conda
  conda av metadata url : None
           channel URLs : https://repo.anaconda.com/pkgs/main/win-64
                          https://repo.anaconda.com/pkgs/main/noarch
                          https://repo.anaconda.com/pkgs/r/win-64
                          https://repo.anaconda.com/pkgs/r/noarch
                          https://repo.anaconda.com/pkgs/msys2/win-64
                          https://repo.anaconda.com/pkgs/msys2/noarch
          package cache : D:\anaconda3\pkgs
       envs directories : D:\anaconda3\envs
                          C:\Users\Gundam\.conda\envs
                          C:\Users\Gundam\AppData\Local\conda\conda\envs
               platform : win-64
             user-agent : conda/23.7.2 requests/2.31.0 CPython/3.11.4 Windows/10 Windows/10.0.19045
          administrator : True
             netrc file : None
           offline mode : False


(base) C:\Users\Gundam>
```

Task 2:

1. *Run **all** of the python commands in the table "Linear Algebra Equivalents"*

2. *Perform this task in IPython/Jupyter and paste the results into my report

Jupyter Notebook:

There were so many commands in that section.

Task_2

September 12, 2023

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

       a = np.array((1,1,23,21,23,123,12))
```

```python
[5]: np.ndim(a)
```

```
[5]: 1
```

```python
[8]: np.size(a)
```

```
[8]: 7
```

```python
[9]: a.shape
```

```
[9]: (7,)
```

```python
[10]: np.block([[a,a],[a,a]])
```

```
[10]: array([[  1,   1,  23,  21,  23, 123,  12,   1,   1,  23,  21,  23, 123,
                12],
              [  1,   1,  23,  21,  23, 123,  12,   1,   1,  23,  21,  23, 123,
                12]])
```

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

```
[11]: 12
```

```python
[28]: B = np.array((1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)).reshape((4,4))
       B
```

```
[28]: array([[ 1,  2,  3,  4],
              [ 5,  6,  7,  8],
              [ 9, 10, 11, 12],
              [13, 14, 15, 16]])
```

```python
[29]: B[1,1]
```

```
[29]: 6
```

```
[26]: B[-3:]
```

```
[26]: array([[ 5,  6,  7,  8],
             [ 9, 10, 11, 12],
             [13, 14, 15, 16]])
```

```
[27]: B[0:3,1:2]
```

```
[27]: array([[ 2],
             [ 6],
             [10]])
```

```
[31]: B[np.ix_([1,2,3],[0,2])]
```

```
[31]: array([[ 5,  7],
             [ 9, 11],
             [13, 15]])
```

```
[33]: C = np.random.rand(21,21)
```

```
[34]: C[2:21:2,:]
```

```
[34]: array([[0.50795545, 0.48936508, 0.54067876, 0.3877504 , 0.80515658,
              0.09005673, 0.7860626 , 0.78982655, 0.06836634, 0.07917667,
              0.54096591, 0.63693553, 0.11297633, 0.22294527, 0.68505408,
              0.03129167, 0.64712718, 0.65773018, 0.04004832, 0.19742811,
              0.66979499],
             [0.36540886, 0.85378718, 0.33451775, 0.70047621, 0.86586032,
              0.94262554, 0.17621397, 0.13278768, 0.27242085, 0.63859063,
              0.69569402, 0.59574855, 0.11666465, 0.39917953, 0.92778887,
              0.24767024, 0.89866584, 0.53402894, 0.44943926, 0.12791323,
              0.04244098],
             [0.3156326 , 0.25628462, 0.57912897, 0.2885751 , 0.72893306,
              0.7482645 , 0.12890675, 0.76897558, 0.644847  , 0.17760294,
              0.615192  , 0.23460024, 0.54182636, 0.24070467, 0.41319899,
              0.25978578, 0.22849712, 0.06512782, 0.80682495, 0.54597361,
              0.75766397],
             [0.04832052, 0.43765049, 0.31369769, 0.53767639, 0.67677963,
              0.00610449, 0.15507278, 0.1168632 , 0.12087029, 0.89153942,
              0.49992397, 0.35936763, 0.32153875, 0.56689468, 0.21077418,
              0.64822014, 0.38455403, 0.77736567, 0.52788009, 0.77798752,
              0.67517354],
             [0.25932408, 0.22168514, 0.98573892, 0.53340685, 0.46022071,
              0.80362839, 0.44997946, 0.44523415, 0.83118348, 0.48272407,
              0.08545704, 0.39027295, 0.70768485, 0.45499212, 0.42929322,
              0.21836007, 0.6137215 , 0.09078424, 0.06465765, 0.86069554,
```

```
        0.92471019],
       [0.38259821, 0.8134914 , 0.56900485, 0.83573219, 0.80576633,
        0.06502091, 0.13705673, 0.39074202, 0.13382944, 0.78657798,
        0.18562449, 0.51465965, 0.83759434, 0.25976978, 0.68537889,
        0.50675771, 0.81652188, 0.84394796, 0.62765541, 0.77699853,
        0.04896942],
       [0.18892511, 0.09578368, 0.51885701, 0.16234459, 0.70087325,
        0.44716754, 0.66323531, 0.14955779, 0.02719985, 0.06699379,
        0.35640079, 0.19507003, 0.66281473, 0.77415986, 0.83173451,
        0.52912313, 0.33085883, 0.92503473, 0.6700173 , 0.71896037,
        0.2112904 ],
       [0.76127489, 0.05859464, 0.02737959, 0.24182748, 0.66490087,
        0.1380581 , 0.09028234, 0.97205854, 0.58328348, 0.78524923,
        0.09932692, 0.95116439, 0.92136103, 0.68790983, 0.28094438,
        0.46263586, 0.86330351, 0.34751444, 0.38661191, 0.22994917,
        0.75824505],
       [0.97160582, 0.21003744, 0.06975635, 0.46236794, 0.94414214,
        0.72099979, 0.62184149, 0.63313746, 0.10901849, 0.64110355,
        0.25723196, 0.93792264, 0.00844011, 0.57442003, 0.18453634,
        0.06088297, 0.59450565, 0.45279661, 0.94871528, 0.23382246,
        0.56501549],
       [0.35903316, 0.56371838, 0.27109753, 0.09264331, 0.19383362,
        0.16165441, 0.13867873, 0.96635108, 0.53393402, 0.64520554,
        0.69629756, 0.40287136, 0.48388163, 0.88754486, 0.39644624,
        0.75021415, 0.85506252, 0.43879764, 0.2276983 , 0.55798028,
        0.45007882]])
```

[35]: `C[::2,:]`

```
[35]: array([[0.66765167, 0.70327315, 0.71019957, 0.61384896, 0.07877105,
        0.0948342 , 0.38565003, 0.93872098, 0.10146884, 0.535532  ,
        0.8972091 , 0.13652558, 0.84987938, 0.26354035, 0.37128465,
        0.08154511, 0.60327041, 0.88705769, 0.16169807, 0.58512215,
        0.04572998],
       [0.50795545, 0.48936508, 0.54067876, 0.3877504 , 0.80515658,
        0.09005673, 0.7860626 , 0.78982655, 0.06836634, 0.07917667,
        0.54096591, 0.63693553, 0.11297633, 0.22294527, 0.68505408,
        0.03129167, 0.64712718, 0.65773018, 0.04004832, 0.19742811,
        0.66979499],
       [0.36540886, 0.85378718, 0.33451775, 0.70047621, 0.86586032,
        0.94262554, 0.17621397, 0.13278768, 0.27242085, 0.63859063,
        0.69569402, 0.59574855, 0.11666465, 0.39917953, 0.92778887,
        0.24767024, 0.89866584, 0.53402894, 0.44943926, 0.12791323,
        0.04244098],
       [0.3156326 , 0.25628462, 0.57912897, 0.2885751 , 0.72893306,
        0.7482645 , 0.12890675, 0.76897558, 0.644847  , 0.17760294,
        0.615192  , 0.23460024, 0.54182636, 0.24070467, 0.41319899,
```

```
        0.25978578, 0.22849712, 0.06512782, 0.80682495, 0.54597361,
        0.75766397],
       [0.04832052, 0.43765049, 0.31369769, 0.53767639, 0.67677963,
        0.00610449, 0.15507278, 0.1168632 , 0.12087029, 0.89153942,
        0.49992397, 0.35936763, 0.32153875, 0.56689468, 0.21077418,
        0.64822014, 0.38455403, 0.77736567, 0.52788009, 0.77798752,
        0.67517354],
       [0.25932408, 0.22168514, 0.98573892, 0.53340685, 0.46022071,
        0.80362839, 0.44997946, 0.44523415, 0.83118348, 0.48272407,
        0.08545704, 0.39027295, 0.70768485, 0.45499212, 0.42929322,
        0.21836007, 0.6137215 , 0.09078424, 0.06465765, 0.86069554,
        0.92471019],
       [0.38259821, 0.8134914 , 0.56900485, 0.83573219, 0.80576633,
        0.06502091, 0.13705673, 0.39074202, 0.13382944, 0.78657798,
        0.18562449, 0.51465965, 0.83759434, 0.25976978, 0.68537889,
        0.50675771, 0.81652188, 0.84394796, 0.62765541, 0.77699853,
        0.04896942],
       [0.18892511, 0.09578368, 0.51885701, 0.16234459, 0.70087325,
        0.44716754, 0.66323531, 0.14955779, 0.02719985, 0.06699379,
        0.35640079, 0.19507003, 0.66281473, 0.77415986, 0.83173451,
        0.52912313, 0.33085883, 0.92503473, 0.6700173 , 0.71896037,
        0.2112904 ],
       [0.76127489, 0.05859464, 0.02737959, 0.24182748, 0.66490087,
        0.1380581 , 0.09028234, 0.97205854, 0.58328348, 0.78524923,
        0.09932692, 0.95116439, 0.92136103, 0.68790983, 0.28094438,
        0.46263586, 0.86330351, 0.34751444, 0.38661191, 0.22994917,
        0.75824505],
       [0.97160582, 0.21003744, 0.06975635, 0.46236794, 0.94414214,
        0.72099979, 0.62184149, 0.63313746, 0.10901849, 0.64110355,
        0.25723196, 0.93792264, 0.00844011, 0.57442003, 0.18453634,
        0.06088297, 0.59450565, 0.45279661, 0.94871528, 0.23382246,
        0.56501549],
       [0.35903316, 0.56371838, 0.27109753, 0.09264331, 0.19383362,
        0.16165441, 0.13867873, 0.96635108, 0.53393402, 0.64520554,
        0.69629756, 0.40287136, 0.48388163, 0.88754486, 0.39644624,
        0.75021415, 0.85506252, 0.43879764, 0.2276983 , 0.55798028,
        0.45007882]])
```

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

```
[36]: array([[ 1,  2,  3,  4],
             [ 5,  6,  7,  8],
             [ 9, 10, 11, 12],
             [13, 14, 15, 16],
             [ 1,  2,  3,  4]])
```

```python
[37]: B.transpose()
```

```
[37]: array([[ 1,  5,  9, 13],
             [ 2,  6, 10, 14],
             [ 3,  7, 11, 15],
             [ 4,  8, 12, 16]])
```

```
[38]: B.conj().T
```

```
[38]: array([[ 1,  5,  9, 13],
             [ 2,  6, 10, 14],
             [ 3,  7, 11, 15],
             [ 4,  8, 12, 16]])
```

```
[40]: B @ B
```

```
[40]: array([[ 90, 100, 110, 120],
             [202, 228, 254, 280],
             [314, 356, 398, 440],
             [426, 484, 542, 600]])
```

```
[41]: B * B
```

```
[41]: array([[  1,   4,   9,  16],
             [ 25,  36,  49,  64],
             [ 81, 100, 121, 144],
             [169, 196, 225, 256]])
```

```
[42]: B / B
```

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

```
[43]: B **3
```

```
[43]: array([[   1,    8,   27,   64],
             [ 125,  216,  343,  512],
             [ 729, 1000, 1331, 1728],
             [2197, 2744, 3375, 4096]], dtype=int32)
```

```
[45]: (B > 4)
```

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

```
[46]: np.nonzero(B > 4)
```

```
[46]: (array([1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3], dtype=int64),
       array([0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3], dtype=int64))
```

```
[47]: C[:, np.nonzero(a > 8)[0]]
```

```
[47]: array([[0.71019957, 0.61384896, 0.07877105, 0.0948342 , 0.38565003],
             [0.99964595, 0.83588417, 0.73720922, 0.36751443, 0.95219909],
             [0.54067876, 0.3877504 , 0.80515658, 0.09005673, 0.7860626 ],
             [0.49694401, 0.73623091, 0.33167801, 0.56177818, 0.4154183 ],
             [0.33451775, 0.70047621, 0.86586032, 0.94262554, 0.17621397],
             [0.68216186, 0.87984085, 0.87529541, 0.72252974, 0.94559731],
             [0.57912897, 0.2885751 , 0.72893306, 0.7482645 , 0.12890675],
             [0.38643026, 0.6155913 , 0.27644926, 0.69047203, 0.5766595 ],
             [0.31369769, 0.53767639, 0.67677963, 0.00610449, 0.15507278],
             [0.06468991, 0.97739081, 0.34953018, 0.66689364, 0.5662745 ],
             [0.98573892, 0.53340685, 0.46022071, 0.80362839, 0.44997946],
             [0.19539218, 0.95886336, 0.18287204, 0.9723965 , 0.2306296 ],
             [0.56900485, 0.83573219, 0.80576633, 0.06502091, 0.13705673],
             [0.33839171, 0.77048957, 0.66956242, 0.81707666, 0.73450392],
             [0.51885701, 0.16234459, 0.70087325, 0.44716754, 0.66323531],
             [0.85061293, 0.46462982, 0.20948858, 0.61939412, 0.84629468],
             [0.02737959, 0.24182748, 0.66490087, 0.1380581 , 0.09028234],
             [0.44813288, 0.2777773 , 0.71652776, 0.60497513, 0.81186188],
             [0.06975635, 0.46236794, 0.94414214, 0.72099979, 0.62184149],
             [0.89612215, 0.40789089, 0.4485911 , 0.33195765, 0.52459968],
             [0.27109753, 0.09264331, 0.19383362, 0.16165441, 0.13867873]])
```

```
[50]: C[:, C[:,0].T > 0.5]
```

```
[50]: array([[0.66765167, 0.71019957, 0.93872098, 0.13652558, 0.60327041,
               0.16169807],
             [0.01974482, 0.99964595, 0.62909784, 0.49422326, 0.30861164,
               0.33333274],
             [0.50795545, 0.54067876, 0.78982655, 0.63693553, 0.64712718,
               0.04004832],
             [0.42763192, 0.49694401, 0.11997263, 0.83028259, 0.72595403,
               0.87682461],
             [0.36540886, 0.33451775, 0.13278768, 0.59574855, 0.89866584,
               0.44943926],
             [0.18569337, 0.68216186, 0.24469452, 0.17996015, 0.46018781,
               0.3036383 ],
             [0.3156326 , 0.57912897, 0.76897558, 0.23460024, 0.22849712,
               0.80682495],
             [0.85058719, 0.38643026, 0.95571536, 0.51630769, 0.09872904,
               0.87184936],
             [0.04832052, 0.31369769, 0.1168632 , 0.35936763, 0.38455403,
               0.52788009],
```

```
      [0.28662007, 0.06468991, 0.7311157 , 0.7281835 , 0.78800516,
       0.28118054],
      [0.25932408, 0.98573892, 0.44523415, 0.39027295, 0.6137215 ,
       0.06465765],
      [0.81778012, 0.19539218, 0.04414333, 0.38248309, 0.77151403,
       0.44774193],
      [0.38259821, 0.56900485, 0.39074202, 0.51465965, 0.81652188,
       0.62765541],
      [0.32219784, 0.33839171, 0.92032249, 0.45106637, 0.39143511,
       0.18588854],
      [0.18892511, 0.51885701, 0.14955779, 0.19507003, 0.33085883,
       0.6700173 ],
      [0.18886701, 0.85061293, 0.24704175, 0.18121727, 0.40706745,
       0.8552289 ],
      [0.76127489, 0.02737959, 0.97205854, 0.95116439, 0.86330351,
       0.38661191],
      [0.23316187, 0.44813288, 0.73425247, 0.36153839, 0.70215769,
       0.87752115],
      [0.97160582, 0.06975635, 0.63313746, 0.93792264, 0.59450565,
       0.94871528],
      [0.21990272, 0.89612215, 0.20298869, 0.690036  , 0.7212259 ,
       0.7863145 ],
      [0.35903316, 0.27109753, 0.96635108, 0.40287136, 0.85506252,
       0.2276983 ]])
```

[51]: `C*(C[:,0].T > 0.5)`

```
[51]: array([[0.66765167, 0.        , 0.71019957, 0.        , 0.        ,
        0.        , 0.        , 0.93872098, 0.        , 0.        ,
        0.        , 0.13652558, 0.        , 0.        , 0.        ,
        0.        , 0.60327041, 0.        , 0.16169807, 0.        ,
        0.        ],
       [0.01974482, 0.        , 0.99964595, 0.        , 0.        ,
        0.        , 0.        , 0.62909784, 0.        , 0.        ,
        0.        , 0.49422326, 0.        , 0.        , 0.        ,
        0.        , 0.30861164, 0.        , 0.33333274, 0.        ,
        0.        ],
       [0.50795545, 0.        , 0.54067876, 0.        , 0.        ,
        0.        , 0.        , 0.78982655, 0.        , 0.        ,
        0.        , 0.63693553, 0.        , 0.        , 0.        ,
        0.        , 0.64712718, 0.        , 0.04004832, 0.        ,
        0.        ],
       [0.42763192, 0.        , 0.49694401, 0.        , 0.        ,
        0.        , 0.        , 0.11997263, 0.        , 0.        ,
        0.        , 0.83028259, 0.        , 0.        , 0.        ,
        0.        , 0.72595403, 0.        , 0.87682461, 0.        ,
        0.        ],
```

```
[0.36540886, 0.        , 0.33451775, 0.        , 0.        ,
 0.        , 0.        , 0.13278768, 0.        , 0.        ,
 0.        , 0.59574855, 0.        , 0.        , 0.        ,
 0.        , 0.89866584, 0.        , 0.44943926, 0.        ,
 0.        ],
[0.18569337, 0.        , 0.68216186, 0.        , 0.        ,
 0.        , 0.        , 0.24469452, 0.        , 0.        ,
 0.        , 0.17996015, 0.        , 0.        , 0.        ,
 0.        , 0.46018781, 0.        , 0.3036383 , 0.        ,
 0.        ],
[0.3156326 , 0.        , 0.57912897, 0.        , 0.        ,
 0.        , 0.        , 0.76897558, 0.        , 0.        ,
 0.        , 0.23460024, 0.        , 0.        , 0.        ,
 0.        , 0.22849712, 0.        , 0.80682495, 0.        ,
 0.        ],
[0.85058719, 0.        , 0.38643026, 0.        , 0.        ,
 0.        , 0.        , 0.95571536, 0.        , 0.        ,
 0.        , 0.51630769, 0.        , 0.        , 0.        ,
 0.        , 0.09872904, 0.        , 0.87184936, 0.        ,
 0.        ],
[0.04832052, 0.        , 0.31369769, 0.        , 0.        ,
 0.        , 0.        , 0.1168632 , 0.        , 0.        ,
 0.        , 0.35936763, 0.        , 0.        , 0.        ,
 0.        , 0.38455403, 0.        , 0.52788009, 0.        ,
 0.        ],
[0.28662007, 0.        , 0.06468991, 0.        , 0.        ,
 0.        , 0.        , 0.7311157 , 0.        , 0.        ,
 0.        , 0.7281835 , 0.        , 0.        , 0.        ,
 0.        , 0.78800516, 0.        , 0.28118054, 0.        ,
 0.        ],
[0.25932408, 0.        , 0.98573892, 0.        , 0.        ,
 0.        , 0.        , 0.44523415, 0.        , 0.        ,
 0.        , 0.39027295, 0.        , 0.        , 0.        ,
 0.        , 0.6137215 , 0.        , 0.06465765, 0.        ,
 0.        ],
[0.81778012, 0.        , 0.19539218, 0.        , 0.        ,
 0.        , 0.        , 0.04414333, 0.        , 0.        ,
 0.        , 0.38248309, 0.        , 0.        , 0.        ,
 0.        , 0.77151403, 0.        , 0.44774193, 0.        ,
 0.        ],
[0.38259821, 0.        , 0.56900485, 0.        , 0.        ,
 0.        , 0.        , 0.39074202, 0.        , 0.        ,
 0.        , 0.51465965, 0.        , 0.        , 0.        ,
 0.        , 0.81652188, 0.        , 0.62765541, 0.        ,
 0.        ],
[0.32219784, 0.        , 0.33839171, 0.        , 0.        ,
 0.        , 0.        , 0.92032249, 0.        , 0.        ,
```

```
       0.        , 0.45106637, 0.        , 0.        , 0.        ,
       0.        , 0.39143511, 0.        , 0.18588854, 0.        ,
       0.        ],
      [0.18892511, 0.        , 0.51885701, 0.        , 0.        ,
       0.        , 0.        , 0.14955779, 0.        , 0.        ,
       0.        , 0.19507003, 0.        , 0.        , 0.        ,
       0.        , 0.33085883, 0.        , 0.6700173 , 0.        ,
       0.        ],
      [0.18886701, 0.        , 0.85061293, 0.        , 0.        ,
       0.        , 0.        , 0.24704175, 0.        , 0.        ,
       0.        , 0.18121727, 0.        , 0.        , 0.        ,
       0.        , 0.40706745, 0.        , 0.8552289 , 0.        ,
       0.        ],
      [0.76127489, 0.        , 0.02737959, 0.        , 0.        ,
       0.        , 0.        , 0.97205854, 0.        , 0.        ,
       0.        , 0.95116439, 0.        , 0.        , 0.        ,
       0.        , 0.86330351, 0.        , 0.38661191, 0.        ,
       0.        ],
      [0.23316187, 0.        , 0.44813288, 0.        , 0.        ,
       0.        , 0.        , 0.73425247, 0.        , 0.        ,
       0.        , 0.36153839, 0.        , 0.        , 0.        ,
       0.        , 0.70215769, 0.        , 0.87752115, 0.        ,
       0.        ],
      [0.97160582, 0.        , 0.06975635, 0.        , 0.        ,
       0.        , 0.        , 0.63313746, 0.        , 0.        ,
       0.        , 0.93792264, 0.        , 0.        , 0.        ,
       0.        , 0.59450565, 0.        , 0.94871528, 0.        ,
       0.        ],
      [0.21990272, 0.        , 0.89612215, 0.        , 0.        ,
       0.        , 0.        , 0.20298869, 0.        , 0.        ,
       0.        , 0.690036  , 0.        , 0.        , 0.        ,
       0.        , 0.7212259 , 0.        , 0.7863145 , 0.        ,
       0.        ],
      [0.35903316, 0.        , 0.27109753, 0.        , 0.        ,
       0.        , 0.        , 0.96635108, 0.        , 0.        ,
       0.        , 0.40287136, 0.        , 0.        , 0.        ,
       0.        , 0.85506252, 0.        , 0.2276983 , 0.        ,
       0.        ]])
```

```
[53]: G = C.copy()
      G[:] = 3
      G
```

```
[53]: 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., 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.,
        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., 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., 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., 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., 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., 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.]])
```

[55]: `C[1,:].copy()`

[55]:
```
array([0.01974482, 0.64398191, 0.99964595, 0.83588417, 0.73720922,
       0.36751443, 0.95219909, 0.62909784, 0.55998114, 0.47259722,
       0.42441957, 0.49422326, 0.44192151, 0.92646357, 0.8736134 ,
       0.48038851, 0.30861164, 0.58253552, 0.33333274, 0.42675823,
       0.59618863])
```

```
[56]: C.flatten()
```

```
[56]: array([0.66765167, 0.70327315, 0.71019957, 0.61384896, 0.07877105,
             0.0948342 , 0.38565003, 0.93872098, 0.10146884, 0.535532  ,
             0.8972091 , 0.13652558, 0.84987938, 0.26354035, 0.37128465,
             0.08154511, 0.60327041, 0.88705769, 0.16169807, 0.58512215,
             0.04572998, 0.01974482, 0.64398191, 0.99964595, 0.83588417,
             0.73720922, 0.36751443, 0.95219909, 0.62909784, 0.55998114,
             0.47259722, 0.42441957, 0.49422326, 0.44192151, 0.92646357,
             0.8736134 , 0.48038851, 0.30861164, 0.58253552, 0.33333274,
             0.42675823, 0.59618863, 0.50795545, 0.48936508, 0.54067876,
             0.3877504 , 0.80515658, 0.09005673, 0.7860626 , 0.78982655,
             0.06836634, 0.07917667, 0.54096591, 0.63693553, 0.11297633,
             0.22294527, 0.68505408, 0.03129167, 0.64712718, 0.65773018,
             0.04004832, 0.19742811, 0.66979499, 0.42763192, 0.8077131 ,
             0.49694401, 0.73623091, 0.33167801, 0.56177818, 0.4154183 ,
             0.11997263, 0.32594569, 0.3462954 , 0.09437829, 0.83028259,
             0.81123039, 0.46139246, 0.99435563, 0.14300766, 0.72595403,
             0.80833704, 0.87682461, 0.93907868, 0.88110827, 0.36540886,
             0.85378718, 0.33451775, 0.70047621, 0.86586032, 0.94262554,
             0.17621397, 0.13278768, 0.27242085, 0.63859063, 0.69569402,
             0.59574855, 0.11666465, 0.39917953, 0.92778887, 0.24767024,
             0.89866584, 0.53402894, 0.44943926, 0.12791323, 0.04244098,
             0.18569337, 0.71869917, 0.68216186, 0.87984085, 0.87529541,
             0.72252974, 0.94559731, 0.24469452, 0.09083119, 0.12253008,
             0.68240709, 0.17996015, 0.21773992, 0.75203189, 0.72815613,
             0.16790109, 0.46018781, 0.18516958, 0.3036383 , 0.86488328,
             0.77023496, 0.3156326 , 0.25628462, 0.57912897, 0.2885751 ,
             0.72893306, 0.7482645 , 0.12890675, 0.76897558, 0.644847  ,
             0.17760294, 0.615192  , 0.23460024, 0.54182636, 0.24070467,
             0.41319899, 0.25978578, 0.22849712, 0.06512782, 0.80682495,
             0.54597361, 0.75766397, 0.85058719, 0.67581048, 0.38643026,
             0.6155913 , 0.27644926, 0.69047203, 0.5766595 , 0.95571536,
             0.7403472 , 0.71837528, 0.8632412 , 0.51630769, 0.06620236,
             0.75975391, 0.01020282, 0.12728421, 0.09872904, 0.27054806,
             0.87184936, 0.60308585, 0.19036307, 0.04832052, 0.43765049,
             0.31369769, 0.53767639, 0.67677963, 0.00610449, 0.15507278,
             0.1168632 , 0.12087029, 0.89153942, 0.49992397, 0.35936763,
             0.32153875, 0.56689468, 0.21077418, 0.64822014, 0.38455403,
             0.77736567, 0.52788009, 0.77798752, 0.67517354, 0.28662007,
             0.74725434, 0.06468991, 0.97739081, 0.34953018, 0.66689364,
             0.5662745 , 0.7311157 , 0.97803475, 0.32666199, 0.50720416,
             0.7281835 , 0.19606144, 0.40465897, 0.27577268, 0.81512318,
             0.78800516, 0.60529236, 0.28118054, 0.32507877, 0.68342808,
             0.25932408, 0.22168514, 0.98573892, 0.53340685, 0.46022071,
             0.80362839, 0.44997946, 0.44523415, 0.83118348, 0.48272407,
             0.08545704, 0.39027295, 0.70768485, 0.45499212, 0.42929322,
```

```
        0.21836007, 0.6137215 , 0.09078424, 0.06465765, 0.86069554,
        0.92471019, 0.81778012, 0.365743  , 0.19539218, 0.95886336,
        0.18287204, 0.9723965 , 0.2306296 , 0.04414333, 0.76730757,
        0.17145096, 0.13484781, 0.38248309, 0.0022059 , 0.79333267,
        0.27530156, 0.62944613, 0.77151403, 0.2463938 , 0.44774193,
        0.39905209, 0.24168986, 0.38259821, 0.8134914 , 0.56900485,
        0.83573219, 0.80576633, 0.06502091, 0.13705673, 0.39074202,
        0.13382944, 0.78657798, 0.18562449, 0.51465965, 0.83759434,
        0.25976978, 0.68537889, 0.50675771, 0.81652188, 0.84394796,
        0.62765541, 0.77699853, 0.04896942, 0.32219784, 0.96809382,
        0.33839171, 0.77048957, 0.66956242, 0.81707666, 0.73450392,
        0.92032249, 0.3047297 , 0.15896727, 0.65920102, 0.45106637,
        0.40984918, 0.74634819, 0.44347315, 0.98183026, 0.39143511,
        0.07967018, 0.18588854, 0.5539278 , 0.88165363, 0.18892511,
        0.09578368, 0.51885701, 0.16234459, 0.70087325, 0.44716754,
        0.66323531, 0.14955779, 0.02719985, 0.06699379, 0.35640079,
        0.19507003, 0.66281473, 0.77415986, 0.83173451, 0.52912313,
        0.33085883, 0.92503473, 0.6700173 , 0.71896037, 0.2112904 ,
        0.18886701, 0.0810226 , 0.85061293, 0.46462982, 0.20948858,
        0.61939412, 0.84629468, 0.24704175, 0.38971302, 0.98760284,
        0.82928162, 0.18121727, 0.87199287, 0.76147414, 0.95421424,
        0.81432047, 0.40706745, 0.68306045, 0.8552289 , 0.21845963,
        0.56201457, 0.76127489, 0.05859464, 0.02737959, 0.24182748,
        0.66490087, 0.1380581 , 0.09028234, 0.97205854, 0.58328348,
        0.78524923, 0.09932692, 0.95116439, 0.92136103, 0.68790983,
        0.28094438, 0.46263586, 0.86330351, 0.34751444, 0.38661191,
        0.22994917, 0.75824505, 0.23316187, 0.63123325, 0.44813288,
        0.2777773 , 0.71652776, 0.60497513, 0.81186188, 0.73425247,
        0.30213704, 0.71567392, 0.29041554, 0.36153839, 0.40539386,
        0.88064201, 0.41436388, 0.71116361, 0.70215769, 0.45861722,
        0.87752115, 0.61539366, 0.78548926, 0.97160582, 0.21003744,
        0.06975635, 0.46236794, 0.94414214, 0.72099979, 0.62184149,
        0.63313746, 0.10901849, 0.64110355, 0.25723196, 0.93792264,
        0.00844011, 0.57442003, 0.18453634, 0.06088297, 0.59450565,
        0.45279661, 0.94871528, 0.23382246, 0.56501549, 0.21990272,
        0.11380696, 0.89612215, 0.40789089, 0.4485911 , 0.33195765,
        0.52459968, 0.20298869, 0.52611763, 0.0957067 , 0.81560728,
        0.690036  , 0.9484254 , 0.28968705, 0.07277301, 0.48230064,
        0.7212259 , 0.04632913, 0.7863145 , 0.10389162, 0.17197998,
        0.35903316, 0.56371838, 0.27109753, 0.09264331, 0.19383362,
        0.16165441, 0.13867873, 0.96635108, 0.53393402, 0.64520554,
        0.69629756, 0.40287136, 0.48388163, 0.88754486, 0.39644624,
        0.75021415, 0.85506252, 0.43879764, 0.2276983 , 0.55798028,
        0.45007882])
```

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

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

```
[59]:  np.arange(10)
```

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

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

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

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

```
[61]:  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.]]])
```

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

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

```
[63]:  np.eye(3)
```

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

```
[64]: np.diag(a)
```

```
[64]: array([[  1,   0,   0,   0,   0,   0,   0],
              [  0,   1,   0,   0,   0,   0,   0],
              [  0,   0,  23,   0,   0,   0,   0],
              [  0,   0,   0,  21,   0,   0,   0],
              [  0,   0,   0,   0,  23,   0,   0],
              [  0,   0,   0,   0,   0, 123,   0],
              [  0,   0,   0,   0,   0,   0,  12]])
```

```
[66]: np.diag(a,1)
```

```
[66]: array([[  0,   1,   0,   0,   0,   0,   0,   0],
              [  0,   0,   1,   0,   0,   0,   0,   0],
              [  0,   0,   0,  23,   0,   0,   0,   0],
              [  0,   0,   0,   0,  21,   0,   0,   0],
              [  0,   0,   0,   0,   0,  23,   0,   0],
              [  0,   0,   0,   0,   0,   0, 123,   0],
              [  0,   0,   0,   0,   0,   0,   0,  12],
              [  0,   0,   0,   0,   0,   0,   0,   0]])
```

```
[69]: np.random.rand(2,3)
```

```
[69]: array([[0.64637182, 0.05697136, 0.76095196],
              [0.98705373, 0.51300603, 0.19795049]])
```

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

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

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

```
[71]: 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.]]])
```

[73]: `np.ogrid[0:9.,0:6.]`

[73]:
```
[array([[0.],
        [1.],
        [2.],
        [3.],
        [4.],
        [5.],
        [6.],
        [7.],
        [8.]]),
 array([[0., 1., 2., 3., 4., 5.]])]
```

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

[74]:
```
[array([[1, 2, 4],
        [1, 2, 4],
        [1, 2, 4]]),
 array([[2, 2, 2],
        [4, 4, 4],
        [5, 5, 5]])]
```

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

[75]:
```
(array([[1],
        [2],
        [4]]),
 array([[2, 4, 5]]))
```

[77]: `np.tile(np.arange(3), (4, 5))`

[77]:
```
array([[0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2],
       [0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2],
       [0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2],
       [0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 2]])
```

[83]: `np.concatenate((np.arange(10).reshape(2,5),np.arange(10,20).reshape(2,5)), 1)`

[83]:
```
array([[ 0,  1,  2,  3,  4, 10, 11, 12, 13, 14],
       [ 5,  6,  7,  8,  9, 15, 16, 17, 18, 19]])
```

[84]: `np.concatenate((np.arange(10),np.arange(10,20)))`

```
[84]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
             17, 18, 19])
```

```
[85]: a.max()
```

```
[85]: 123
```

```
[86]: np.maximum(a, a**2)
```

```
[86]: array([    1,     1,   529,   441,   529, 15129,   144])
```

```
[87]: np.sqrt(B @ B)
```

```
[87]: array([[ 9.48683298, 10.        , 10.48808848, 10.95445115],
             [14.2126704 , 15.09966887, 15.93737745, 16.73320053],
             [17.72004515, 18.86796226, 19.94993734, 20.97617696],
             [20.63976744, 22.        , 23.28089345, 24.49489743]])
```

```
[88]: np.logical_and(C>0.5, C<0.75)
```

```
[88]: array([[ True,  True,  True,  True, False, False, False, False, False,
               True, False, False, False, False, False, False,  True, False,
              False,  True, False],
             [False,  True, False, False,  True, False, False,  True,  True,
              False, False, False, False, False, False, False, False,  True,
              False, False,  True],
             [ True, False,  True, False, False, False, False, False, False,
              False,  True,  True, False, False,  True, False,  True,  True,
              False, False,  True],
             [False, False, False,  True, False,  True, False, False, False,
              False, False, False, False, False, False, False,  True, False,
              False, False, False],
             [False, False, False,  True, False, False, False, False, False,
               True,  True,  True, False, False, False, False, False,  True,
              False, False, False],
             [False,  True,  True, False, False,  True, False, False, False,
              False,  True, False, False, False,  True, False, False, False,
              False, False, False],
             [False, False,  True, False,  True,  True, False, False,  True,
              False,  True, False,  True, False, False, False, False, False,
              False,  True, False],
             [False,  True, False,  True, False,  True,  True, False,  True,
               True, False,  True, False, False, False, False, False, False,
              False,  True, False],
             [False, False, False,  True,  True, False, False, False, False,
              False, False, False, False,  True, False,  True, False, False,
               True, False,  True],
             [False,  True, False, False, False,  True,  True,  True, False,
```

```
             False,  True,   True, False, False, False, False, False,   True,
             False, False,   True],
           [False, False, False,   True, False, False, False, False, False,
             False, False, False,   True, False, False, False,   True, 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, False,   True, False, False, False, False, False, False,
             False, False,   True, False, False,   True,   True, False, False,
               True, False, False],
           [False, False, False, False,   True, False,   True, False, False,
             False,   True, False, False,   True, False, False, False, False,
             False,   True, False],
           [False, False,   True, False,   True, False,   True, False, False,
             False, False, False,   True, False, False,   True, False, False,
               True,   True, False],
           [False, False, False, False, False,   True, False, False, False,
             False, False, False, False, False, False, False, False,   True,
             False, False,   True],
           [False, False, False, False,   True, False, False, False,   True,
             False, False, False, False,   True, False, False, False, False,
             False, False, False],
           [False,   True, False, False,   True,   True, False,   True, False,
               True, False, False, False, False, False,   True,   True, False,
             False,   True, False],
           [False, False, False, False, False,   True,   True,   True, False,
               True, False, False, False,   True, False, False,   True, False,
             False, False,   True],
           [False, False, False, False, False, False,   True, False,   True,
             False, False,   True, False, False, False, False,   True, False,
             False, False, False],
           [False,   True, False, False, False, False, False, False,   True,
               True,   True, False, False, False, False, False, False, False,
             False,   True, False]])
```

```
[90]: np.logical_or(C < 0.1, C > 0.9)
```

```
[90]: array([[False, False, False, False,   True,   True, False,   True, False,
             False, False, False, False, False, False,   True, False, False,
             False, False,   True],
           [ True, False,   True, False, False, False,   True, False, False,
             False, False, False, False,   True, False, False, False, False,
             False, False, False],
           [False, False, False, False, False,   True, False, False,   True,
               True, False, False, False, False, False,   True, False, False,
               True, False, False],
```

```
[False, False, False, False, False, False, False, False, False,
 False,  True, False, False, False,  True, False, False, False,
 False,  True, False],
[False, False, False, False, False,  True, False, False, False,
 False, False, False, False, False,  True, False, False, False,
 False, False,  True],
[False, False, False, False, False, False,  True, False,  True,
 False, False, False, False, False, False, False, False, False,
 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, False, False, False, False,  True, False,
 False, False, False,  True, False,  True, False,  True, False,
 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,  True,  True, False, False, False, False,  True,
 False, False, False, False, False, False, False, False, False,
 False, False, False],
[False, False,  True, False, False, False, False, False, False,
 False,  True, False, False, False, False, False, False,  True,
  True, False,  True],
[False, False, False,  True, False,  True, False,  True, False,
 False, False, False,  True, False, False, False, False, False,
 False, False, False],
[False, False, False, False, False,  True, False, False, False,
 False, False, False, False, False, False, False, False, False,
 False, False,  True],
[False,  True, False, False, False, False, False,  True, False,
 False, False, False, False, False, False,  True, False,  True,
 False, False, False],
[False,  True, False, False, False, False, False, False,  True,
  True, False, False, False, False, False, False, False,  True,
 False, False, False],
[False,  True, False, False, False, False, False, False, False,
  True, False, False, False, False,  True, False, False, False,
 False, False, False],
[False,  True,  True, False, False, False,  True,  True, False,
 False,  True,  True,  True, False, False, False, False, False,
 False, False, False],
[False, False, False, False, False, False, False, False, False,
 False, False, False, False, False, False, False, False, False,
 False, False, False],
[ True, False,  True, False,  True, False, False, False, False,
 False, False,  True,  True, False, False,  True, False, False,
```

```
             True, False, False],
       [False, False, False, False, False, False, False, False, False,
         True, False, False,  True, False,  True, False, False,  True,
        False, False, False],
       [False, False, False,  True, False, False, False,  True, False,
        False, False, False, False, False, False, False, False, False,
        False, False, False]])
```

```
[95]: J = np.arange(0,256, dtype='int')
      K = np.arange(0,256, dtype='int') + 1

      K & J
```

```
[95]: array([  0,   0,   2,   0,   4,   4,   6,   0,   8,   8,  10,   8,  12,
             12,  14,   0,  16,  16,  18,  16,  20,  20,  22,  16,  24,  24,
             26,  24,  28,  28,  30,   0,  32,  32,  34,  32,  36,  36,  38,
             32,  40,  40,  42,  40,  44,  44,  46,  32,  48,  48,  50,  48,
             52,  52,  54,  48,  56,  56,  58,  56,  60,  60,  62,   0,  64,
             64,  66,  64,  68,  68,  70,  64,  72,  72,  74,  72,  76,  76,
             78,  64,  80,  80,  82,  80,  84,  84,  86,  80,  88,  88,  90,
             88,  92,  92,  94,  64,  96,  96,  98,  96, 100, 100, 102,  96,
            104, 104, 106, 104, 108, 108, 110,  96, 112, 112, 114, 112, 116,
            116, 118, 112, 120, 120, 122, 120, 124, 124, 126,   0, 128, 128,
            130, 128, 132, 132, 134, 128, 136, 136, 138, 136, 140, 140, 142,
            128, 144, 144, 146, 144, 148, 148, 150, 144, 152, 152, 154, 152,
            156, 156, 158, 128, 160, 160, 162, 160, 164, 164, 166, 160, 168,
            168, 170, 168, 172, 172, 174, 160, 176, 176, 178, 176, 180, 180,
            182, 176, 184, 184, 186, 184, 188, 188, 190, 128, 192, 192, 194,
            192, 196, 196, 198, 192, 200, 200, 202, 200, 204, 204, 206, 192,
            208, 208, 210, 208, 212, 212, 214, 208, 216, 216, 218, 216, 220,
            220, 222, 192, 224, 224, 226, 224, 228, 228, 230, 224, 232, 232,
            234, 232, 236, 236, 238, 224, 240, 240, 242, 240, 244, 244, 246,
            240, 248, 248, 250, 248, 252, 252, 254,   0])
```

```
[96]: K | J
```

```
[96]: array([  1,   3,   3,   7,   5,   7,   7,  15,   9,  11,  11,  15,  13,
             15,  15,  31,  17,  19,  19,  23,  21,  23,  23,  31,  25,  27,
             27,  31,  29,  31,  31,  63,  33,  35,  35,  39,  37,  39,  39,
             47,  41,  43,  43,  47,  45,  47,  47,  63,  49,  51,  51,  55,
             53,  55,  55,  63,  57,  59,  59,  63,  61,  63,  63, 127,  65,
             67,  67,  71,  69,  71,  71,  79,  73,  75,  75,  79,  77,  79,
             79,  95,  81,  83,  83,  87,  85,  87,  87,  95,  89,  91,  91,
             95,  93,  95,  95, 127,  97,  99,  99, 103, 101, 103, 103, 111,
            105, 107, 107, 111, 109, 111, 111, 127, 113, 115, 115, 119, 117,
            119, 119, 127, 121, 123, 123, 127, 125, 127, 127, 255, 129, 131,
            131, 135, 133, 135, 135, 143, 137, 139, 139, 143, 141, 143, 143,
```

```
       159, 145, 147, 147, 151, 149, 151, 151, 159, 153, 155, 155, 159,
       157, 159, 159, 191, 161, 163, 163, 167, 165, 167, 167, 175, 169,
       171, 171, 175, 173, 175, 175, 191, 177, 179, 179, 183, 181, 183,
       183, 191, 185, 187, 187, 191, 189, 191, 191, 255, 193, 195, 195,
       199, 197, 199, 199, 207, 201, 203, 203, 207, 205, 207, 207, 223,
       209, 211, 211, 215, 213, 215, 215, 223, 217, 219, 219, 223, 221,
       223, 223, 255, 225, 227, 227, 231, 229, 231, 231, 239, 233, 235,
       235, 239, 237, 239, 239, 255, 241, 243, 243, 247, 245, 247, 247,
       255, 249, 251, 251, 255, 253, 255, 255, 511]])
```

[97]: `np.linalg.inv(C)`

[97]:
```
array([[-0.26672426, -1.08147473,  1.6819471 , -0.03266171,  0.16098261,
        -0.49944951, -0.31140883,  1.19052311,  0.36627422, -0.98524635,
        -0.02013484,  1.01620187,  0.01110087,  0.43544334,  0.11230539,
         0.22619511,  0.9034426 ,  0.61301652, -1.67652619, -0.08237231,
        -1.32202516],
       [ 0.12292603, -0.38321704,  0.94855221,  0.41142747,  0.90124711,
        -0.67850681, -0.4083396 ,  1.04675666,  0.5158068 , -0.70814768,
        -0.25743684,  0.40006358, -0.58185912,  0.45502567, -0.08797323,
        -0.53857071,  0.84124406,  1.64729373, -2.1421718 ,  0.13604767,
        -1.55460387],
       [ 0.25038896,  0.7603968 ,  0.22614706,  0.03850001, -0.08494815,
        -0.93724537,  0.210459  , -0.42333658,  0.3034296 , -0.77739081,
         0.43836998,  0.40936231, -0.08954172,  0.24984884, -0.34600119,
        -0.16906395, -0.90455167, -0.06034046,  0.51895515,  0.31728564,
         0.34723017],
       [ 0.44480822,  0.92754689, -1.1925983 , -0.10372324, -0.68928393,
         0.82975288,  0.461998  , -1.04940103, -0.29318575,  0.6301641 ,
        -0.40461344,  0.00969567,  0.49302051, -0.23765882, -0.58195126,
         0.17395162, -0.47467821, -1.1187739 ,  1.52565727, -0.13223703,
         0.67951174],
       [-0.19933718, -0.2851706 ,  0.60128938, -0.526033  ,  0.5339578 ,
         0.07036033,  0.10089186,  0.48973592,  0.42849212, -0.21858836,
        -0.07223409,  0.14634263, -0.0184308 , -0.05371738,  0.40475719,
        -0.38118385,  0.8946147 ,  0.66012451, -1.09285399,  0.02715852,
        -1.16026532],
       [ 0.52599788,  0.15882208, -0.91435001,  0.00617047,  0.46433683,
        -0.41841334,  0.05248467, -0.52894569, -0.16665823,  0.21524335,
         0.56840347, -0.30555296, -0.50471148,  0.40551903,  0.32314852,
        -0.14685969, -0.52632849, -0.18046767,  0.9786002 , -0.10835216,
         0.11865309],
       [-0.44885277, -0.81467335,  0.68524956, -0.22003771, -0.15032398,
         0.49013847, -0.8244487 ,  0.85734936, -0.42896644,  0.40674355,
         0.17728829, -0.29058412,  0.29098323, -0.1260903 ,  0.31228219,
         0.38412862,  0.40892377,  0.66503888, -0.78533557,  0.06761315,
        -0.74637476],
```

```
[ 0.37984222,  0.88763262, -1.05182112, -0.18763854, -0.66727803,
   0.15854584,  0.66612599, -1.01320877, -0.8211883 ,  0.56113212,
  -0.04938359, -0.54482654,  0.50370057, -0.06903333, -0.25505038,
  -0.00401078, -0.91078091, -0.80551757,  1.89988994, -0.30067225,
   1.46607207],
[-0.53443968, -0.54716683,  1.01581567, -0.16203298,  0.40646317,
  -0.06172515, -0.05909419,  1.28508677,  0.17360675,  0.22308841,
   0.02281501,  0.34993864, -0.15920114, -0.57092305,  0.32270533,
  -0.0905909 ,  1.08048649,  0.85045717, -2.10765209, -0.00824061,
  -1.18430342],
[-0.21782464, -0.29905169, -0.03712635, -0.22521753,  0.19721736,
   0.0533268 , -0.39460502,  0.36434429,  0.09809499, -0.01561084,
   0.40247866, -0.34826079,  0.33491007, -0.06444782, -0.30053987,
   0.49048922,  0.12833235,  0.06891283,  0.00684105, -0.25845256,
  -0.07648818],
[-0.01430405, -0.39814727,  0.0941094 , -0.08837548,  0.16714896,
   0.45975568,  0.07464974,  0.20976141,  0.28759731,  0.14706696,
  -0.19734003, -0.24960353, -0.28583777, -0.05359745,  0.05323833,
   0.20502266,  0.00399286, -0.57033217, -0.09862311,  0.23265628,
   0.3082288 ],
[-0.50107637,  0.43957781, -0.49819532,  0.43212709, -0.16295851,
  -0.41706058, -0.45690409, -0.18380445, -0.14709702,  0.18927778,
   0.47605851, -0.76615507,  0.21696996,  0.50985381,  0.10714568,
  -0.07117994, -0.81781247, -1.46898182,  1.53106566,  0.35204347,
   1.20777168],
[ 0.48254284, -0.28977496, -0.21555111,  0.12232758,  0.39638451,
   0.07381135, -0.23342671,  0.3020209 ,  0.04739244, -0.0993007 ,
  -0.10684876, -0.12336008, -0.41978598,  0.2182048 ,  0.34874802,
  -0.14644608,  1.18733956,  0.56863289, -1.07463421,  0.20187361,
  -1.20029662],
[ 0.50068516,  1.00596536, -0.73903876,  0.13880497,  0.19294646,
   0.17544169, -0.0658116 , -0.30991772,  0.25294355, -0.44772397,
  -0.43484346,  0.35476345, -0.89620425, -0.15380397,  0.08361532,
  -0.52583441,  0.41672898,  0.17347391,  0.10959838,  0.06892734,
   0.21883466],
[-0.78136707, -0.09163163,  0.28157435,  0.15915035, -0.20880689,
   0.43810826,  0.22473806,  0.03312496, -0.75839149,  0.0395796 ,
  -0.14985812, -0.17406943,  0.81641391, -0.0919171 , -0.07206693,
   0.71737164, -0.18435989, -0.84758437,  0.155113  , -0.439596  ,
   0.81906622],
[-0.59524472, -0.39007521,  0.4845506 , -0.24165173, -0.33101919,
  -0.62678558, -0.10470606,  0.02965492,  0.10035356, -0.12102471,
   0.15495458,  0.25876607,  0.653946  ,  0.77303687,  0.12239624,
   0.4126215 , -0.48012328, -0.27132233,  0.00635716,  0.04933208,
   0.27005525],
[ 0.20685924, -0.11971234, -0.26796382, -0.12232921, -0.14238509,
   0.64916119,  0.05363218, -0.82499515, -0.42032487,  0.39460071,
```

```
        -0.0460244 ,  0.08680794,  0.19368634, -0.70578371, -0.35915007,
        -0.05870142, -0.33683062,  0.28172126,  0.60461303,  0.19879038,
         0.86045482],
       [ 0.7939062 ,  0.33417963,  0.03845416,  0.08826019,  0.35317995,
        -0.79815565,  0.04938653, -0.22279288,  0.65511538,  0.20989258,
        -0.0124584 ,  0.17526749, -0.83980816, -0.15569322,  0.54685071,
        -0.42146634,  0.01926495,  0.41281219, -0.14203278, -0.13057575,
        -0.63841267],
       [-0.06761684,  0.25470965, -0.33698653,  0.26435594, -0.40812583,
         0.15867737,  0.62397519, -0.26283869, -0.24511713,  0.07711171,
        -0.61890859,  0.08612379,  0.28011314, -0.4159097 , -0.336132  ,
         0.10192807, -0.3964215 ,  0.23595402,  0.50247343,  0.13796742,
         0.34440348],
       [-0.61153919, -0.56796465, -0.27496919, -0.02778463, -0.64776118,
         0.61857811, -0.16516254,  0.19774383, -0.31616153,  0.40294293,
         0.59941521, -0.54529738,  0.93341566, -0.0664683 ,  0.37006518,
         0.11591439, -0.73887877, -0.98147525,  0.77145798, -0.1556654 ,
         1.15755296],
       [ 0.25270508, -0.05937898,  0.46102418,  0.37334241,  0.00522267,
        -0.13939431,  0.32513357, -0.22601992,  0.67696835, -0.2774059 ,
        -0.17310167,  0.3535176 , -0.79146824,  0.08852253, -0.44038546,
         0.02003731,  0.36367269,  0.51419792, -0.48040908, -0.08655192,
        -0.56380545]])
```

[101]: `np.linalg.pinv(B)`

[101]:
```
array([[-0.285 , -0.145 , -0.005 ,  0.135 ],
       [-0.1075, -0.0525,  0.0025,  0.0575],
       [ 0.07  ,  0.04  ,  0.01  , -0.02  ],
       [ 0.2475,  0.1325,  0.0175, -0.0975]])
```

[103]: `np.linalg.matrix_rank(B)`

[103]: 2

[108]: `np.linalg.solve(B, B[2,:])`

[108]: `array([-6.75,  4.25,  3.75, -1.  ])`

[109]: `np.linalg.svd(B)`

[109]:
```
SVDResult(U=array([[-0.13472212, -0.82574206,  0.38124745, -0.39325613],
       [-0.3407577 , -0.4288172 , -0.21521411,  0.80850658],
       [-0.54679327, -0.03189234, -0.71331411, -0.43724476],
       [-0.75282884,  0.36503251,  0.54728078,  0.02199431]]),
    S=array([3.86226568e+01, 2.07132307e+00, 7.60977226e-16, 3.86063773e-16]),
    Vh=array([[-0.4284124 , -0.47437252, -0.52033264, -0.56629275],
       [ 0.71865348,  0.27380781, -0.17103786, -0.61588352],
```

```
        [ 0.28255951, -0.72647617,  0.6052738 , -0.16135714],
        [-0.46921223,  0.41500889,  0.57761893, -0.52341558]]))
```

[120]:
```
np.linalg.cholesky(np.array((2,-1,0,-1,2,-1,0,-1,2)).reshape(3,3))
```

[120]:
```
array([[ 1.41421356,  0.        ,  0.        ],
       [-0.70710678,  1.22474487,  0.        ],
       [ 0.        , -0.81649658,  1.15470054]])
```

[126]:
```
D, V = np.linalg.eig(B)
print(D)
print(V)
```

```
[ 3.62093727e+01 -2.20937271e+00  2.80575969e-16 -8.29646454e-16]
[[-0.15115432 -0.72704996 -0.01128698 -0.250341  ]
 [-0.34923733 -0.28320876  0.42321091  0.69689908]
 [-0.54732033  0.16063243 -0.81256087 -0.64277516]
 [-0.74540333  0.60447363  0.40063694  0.19621708]]
```

[150]:
```
D, V = linalg.eig(B, np.random.rand(B.shape[0], B.shape[1]))
print(D)
print(V)
```

```
[-5.13011045e+01+0.j  1.02614934e+01+0.j  4.69505866e-15+0.j
   1.56664418e-15+0.j]
[[ 0.78126873  0.68870424 -0.36104822  0.42786893]
 [-0.03516474 -0.0116871   0.17440125 -0.82536462]
 [-0.61764839 -0.72494302  0.73434214  0.36712246]
 [ 0.08302458 -0.00273762 -0.54769518  0.03037323]]
```

[151]:
```
D,V = eigs(B, k=3)
print(D)
print(V)
```

```
[ 3.62093727e+01+0.j -2.20937271e+00+0.j  2.80575969e-16+0.j
 -8.29646454e-16+0.j]
[[-0.15115432 -0.72704996 -0.01128698 -0.250341  ]
 [-0.34923733 -0.28320876  0.42321091  0.69689908]
 [-0.54732033  0.16063243 -0.81256087 -0.64277516]
 [-0.74540333  0.60447363  0.40063694  0.19621708]]
```

[134]:
```
Q,R = np.linalg.qr(B)
print(Q,R)
```

```
[[-0.06019293 -0.83449195 -0.19896747 -0.51030574]
 [-0.30096463 -0.45762462  0.6456494   0.53210606]
 [-0.54173634 -0.08075729 -0.69439639  0.4667051 ]
 [-0.78250805  0.29611005  0.24771446 -0.48850542]] [[-1.66132477e+01
 -1.82986497e+01 -1.99840516e+01 -2.16694536e+01]
 [ 0.00000000e+00 -1.07676380e+00 -2.15352761e+00 -3.23029141e+00]
```

```
 [ 0.00000000e+00  0.00000000e+00  1.69104133e-15 -2.91558850e-16]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00 -1.57441779e-15]]
```

```
[144]: P,L,U = linalg.lu(B)
       print(P)
       print(L)
       print(U)
```

```
[[0. 1. 0. 0.]
 [0. 0. 0. 1.]
 [0. 0. 1. 0.]
 [1. 0. 0. 0.]]
[[ 1.          0.          0.          0.        ]
 [ 0.07692308  1.          0.          0.        ]
 [ 0.69230769  0.33333333  1.          0.        ]
 [ 0.38461538  0.66666667 -0.5         1.        ]]
[[ 1.30000000e+01  1.40000000e+01  1.50000000e+01  1.60000000e+01]
 [ 0.00000000e+00  9.23076923e-01  1.84615385e+00  2.76923077e+00]
 [ 0.00000000e+00  0.00000000e+00 -1.77635684e-15 -1.77635684e-15]
 [ 0.00000000e+00  0.00000000e+00  0.00000000e+00 -2.22044605e-16]]
```

```
[152]: cg
```

```
[152]: <function scipy.sparse.linalg._isolve.iterative.cg(A, b, x0=None, tol=1e-05,
       maxiter=None, M=None, callback=None, atol=None)>
```

```
[153]: np.fft.fft(a)
```

```
[153]: array([ 204.         +0.j         ,  -63.02531912+106.960705j  ,
               -106.00067557 -34.22782984j,   70.52599469 -71.46057129j,
                 70.52599469 +71.46057129j, -106.00067557 +34.22782984j,
                -63.02531912-106.960705j  ])
```

```
[154]: np.fft.ifft(a)
```

```
[154]: array([ 29.14285714 +0.j        ,  -9.00361702-15.28010071j,
               -15.14295365 +4.88968998j,  10.0751421 +10.20865304j,
                10.0751421 -10.20865304j, -15.14295365 -4.88968998j,
                -9.00361702+15.28010071j])
```

```
[155]: np.sort(a)
```

```
[155]: array([  1,    1,  12,  21,  23,  23, 123])
```

```
[156]: np.sort(B,axis=1)
```

```
[156]: array([[ 1,  2,  3,  4],
              [ 5,  6,  7,  8],
              [ 9, 10, 11, 12],
```

```
                    [13, 14, 15, 16]])
```

[158]:
```
I = np.argsort(B[:,0])
b = B[I,:]
print(b)
```

```
[[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]
 [13 14 15 16]]
```

[160]:
```
linalg.lstsq(B, np.random.rand(B.shape[0]))
```

[160]:
```
(array([0.00211429, 0.00845879, 0.01480329, 0.02114778]),
 array([], dtype=float64),
 2,
 array([3.86226568e+01, 2.07132307e+00, 7.60977226e-16, 3.86063773e-16]))
```

[163]:
```
signal.resample(a, int(np.ceil(len(a)/2)))
```

[163]:
```
array([-19.1502842 ,  28.86856302,  16.86418387,  89.98896588])
```

[164]:
```
np.unique(a)
```

[164]:
```
array([  1,  12,  21,  23, 123])
```

[165]:
```
a.squeeze()
```

[165]:
```
array([  1,   1,  23,  21,  23, 123,  12])
```

[ ]:

Task 3:

Run the following script in IPython and paste the figure created by the script into your report

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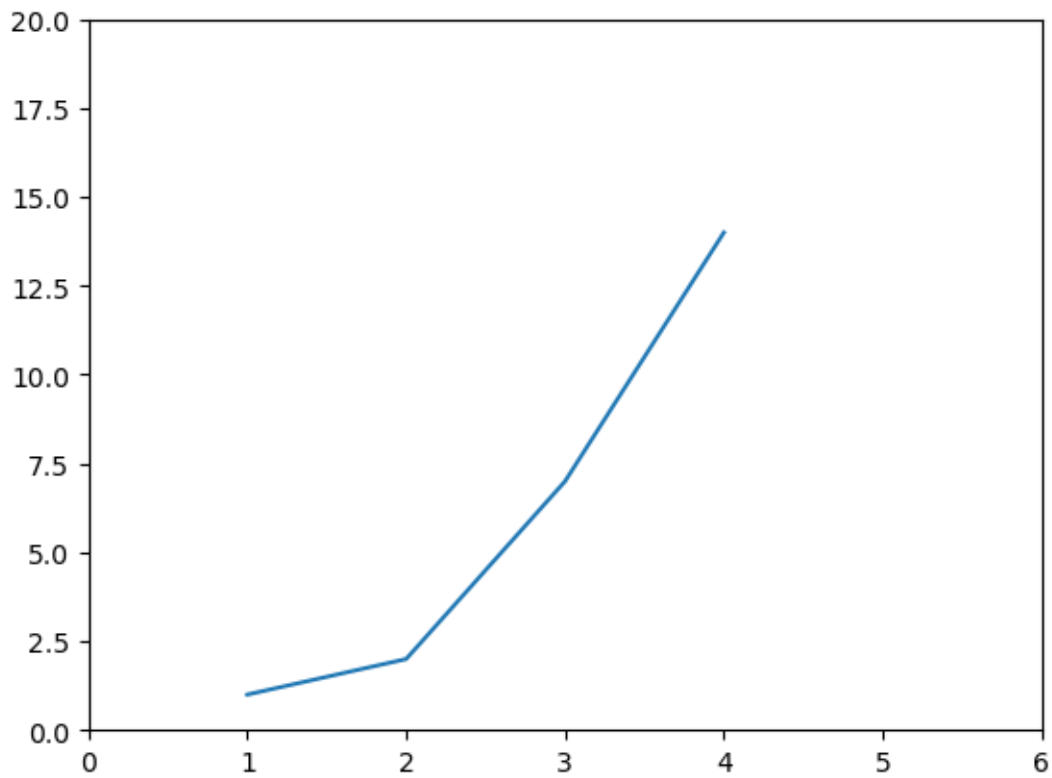import matplotlib.pyplot as plt
plt.plot([1,2,3,4], [1,2,7,14])
plt.axis([0, 6, 0, 20])
plt.show()
```

Jupyter Notebook:

Task_3

September 12, 2023

```
[2]: 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:

Use Matplotlib to create a figure of your choice in IPython. Paste your code and figure into your report

Jupyter Notebook:

Task_4

September 12, 2023

```python
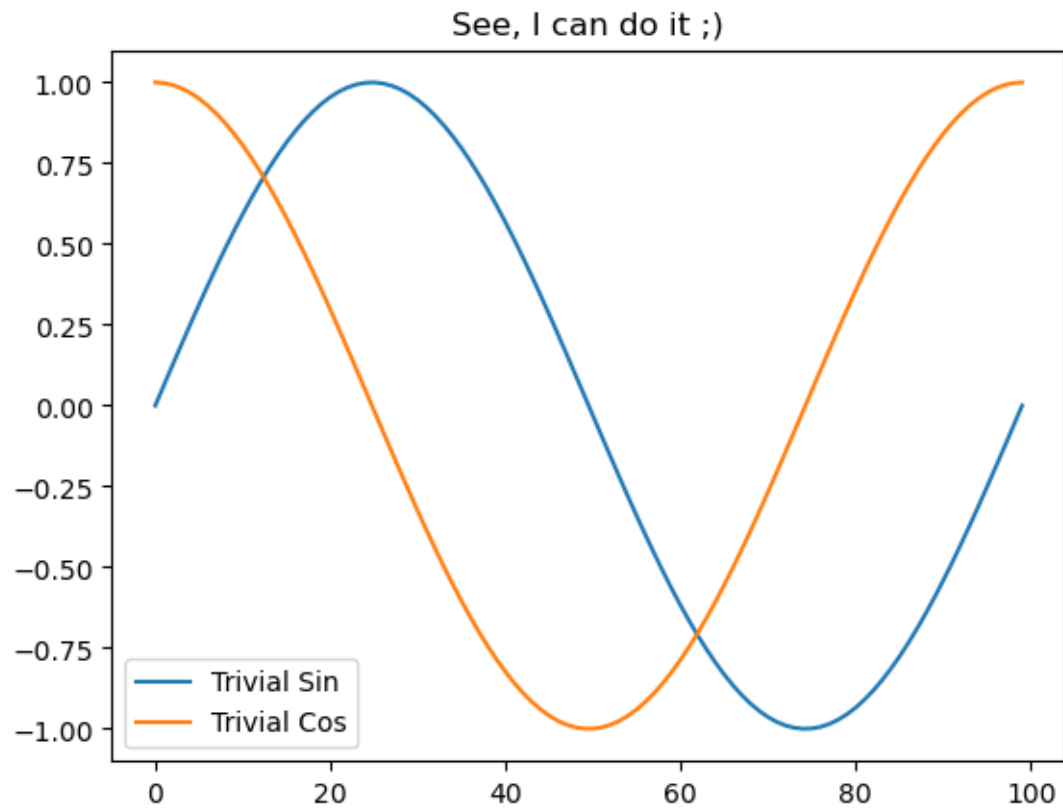[9]: import matplotlib.pyplot as plt
     import numpy as np

     vals_1 = np.sin(np.linspace(0,np.pi*2, 100))
     vals_2 = np.cos(np.linspace(0,np.pi*2, 100))

     plt.plot(vals_1, label = "Trivial Sin")
     plt.plot(vals_2, label = "Trivial Cos")

     plt.title("See, I can do it ;)")
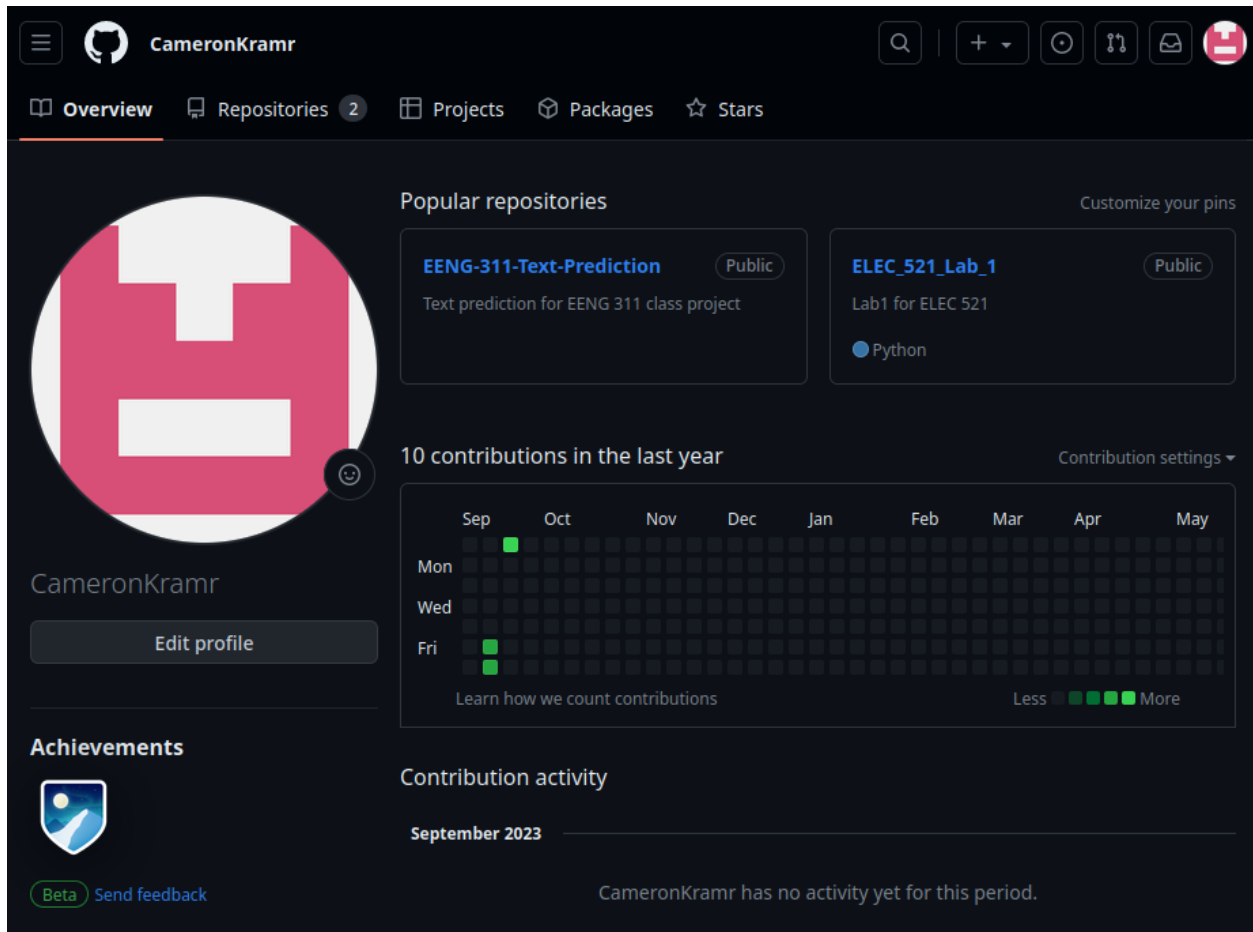
     plt.legend()


     plt.show()
```

[ ]:

Task 5:

Paste your VCS account into your report

Result:

Here is my Github account as requested. This was before I pushed this project to Github.

Task 6:

Start a new project in PyCharm. Commit and push your project to VCS as a public project and share a link of your project in your report

Result:

I did not use PyCharm to make the repository. Find link to the repository for this task here. There, you will find all of the Jupyter notebooks as well as the build and Markdown files used to generate this report. Instead of PyCharm, I used the CLI interface for GIT.