

```
In [1]: import numpy as np
import scipy.linalg
```

```
In [2]: a = np.random.rand(10,2)
```

```
In [3]: a.ndim
```

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

```
In [4]: a.size
```

```
Out[4]: 20
```

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

```
Out[6]: 2
```

```
In [7]: np.array([[1,2,3],[4,5,6]])
```

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

```
In [8]: np.vstack([np.hstack([a,a]),np.hstack([a,a])])
```

```
Out[8]: array([[ 0.49926506,  0.65065265,  0.49926506,  0.65065265],
               [ 0.39066572,  0.77851938,  0.39066572,  0.77851938],
               [ 0.72041562,  0.32988807,  0.72041562,  0.32988807],
               [ 0.42046973,  0.79006924,  0.42046973,  0.79006924],
               [ 0.53983844,  0.23675587,  0.53983844,  0.23675587],
               [ 0.43063407,  0.87103111,  0.43063407,  0.87103111],
               [ 0.89189145,  0.29332695,  0.89189145,  0.29332695],
               [ 0.70529784,  0.70913641,  0.70529784,  0.70913641],
               [ 0.7662732 ,  0.93814996,  0.7662732 ,  0.93814996],
               [ 0.81161707,  0.61294193,  0.81161707,  0.61294193],
               [ 0.49926506,  0.65065265,  0.49926506,  0.65065265],
               [ 0.39066572,  0.77851938,  0.39066572,  0.77851938],
               [ 0.72041562,  0.32988807,  0.72041562,  0.32988807],
               [ 0.42046973,  0.79006924,  0.42046973,  0.79006924],
               [ 0.53983844,  0.23675587,  0.53983844,  0.23675587],
               [ 0.43063407,  0.87103111,  0.43063407,  0.87103111],
               [ 0.89189145,  0.29332695,  0.89189145,  0.29332695],
               [ 0.70529784,  0.70913641,  0.70529784,  0.70913641],
               [ 0.7662732 ,  0.93814996,  0.7662732 ,  0.93814996],
               [ 0.81161707,  0.61294193,  0.81161707,  0.61294193]])
```

```
In [9]: a[-1]
a[1]
a[0:5]
a[-5:]
a[0:3][:,4:9]
```

```
Out[9]: array([], shape=(3, 0), dtype=float64)
```

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

```
Out[10]: array([ 0.81161707,  0.61294193])
```

```
In [11]: a[1]
```

```
Out[11]: array([ 0.39066572,  0.77851938])
```

```
In [12]: a[0:5]
```

```
Out[12]: array([[ 0.49926506,  0.65065265],
                 [ 0.39066572,  0.77851938],
                 [ 0.72041562,  0.32988807],
                 [ 0.42046973,  0.79006924],
                 [ 0.53983844,  0.23675587]])
```

```
In [13]: a[-5:]
```

```
Out[13]: array([[ 0.43063407,  0.87103111],
                 [ 0.89189145,  0.29332695],
                 [ 0.70529784,  0.70913641],
                 [ 0.7662732 ,  0.93814996],
                 [ 0.81161707,  0.61294193]])
```

```
In [14]: a[0:3]
```

```
Out[14]: array([[ 0.49926506,  0.65065265],
                 [ 0.39066572,  0.77851938],
                 [ 0.72041562,  0.32988807]])
```

```
In [15]: a[0:3][:,1:2]
```

```
File "<ipython-input-15-102999f40f29>", line 1
```

```
a[0:3][:,1:2]
```

```
^
```

```
SyntaxError: invalid syntax
```

```
In [16]: a[0:3][:,1:2]
```

```
Out[16]: array([[ 0.65065265],
                 [ 0.77851938],
                 [ 0.32988807]])
```

```
In [19]: a[np.ix_([1,3,4],[0,1])]
```

```
Out[19]: array([[ 0.39066572,  0.77851938],
                 [ 0.42046973,  0.79006924],
                 [ 0.53983844,  0.23675587]])
```

```
In [20]: a[2:7:2,:]
```

```
Out[20]: array([[ 0.72041562,  0.32988807],
                 [ 0.53983844,  0.23675587],
                 [ 0.89189145,  0.29332695]])
```

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

```
Out[21]: array([[ 0.49926506,  0.65065265],
                [ 0.72041562,  0.32988807],
                [ 0.53983844,  0.23675587],
                [ 0.89189145,  0.29332695],
                [ 0.7662732 ,  0.93814996]])
```

```
In [22]: a[::-1,:]
```

```
Out[22]: array([[ 0.81161707,  0.61294193],
                [ 0.7662732 ,  0.93814996],
                [ 0.70529784,  0.70913641],
                [ 0.89189145,  0.29332695],
                [ 0.43063407,  0.87103111],
                [ 0.53983844,  0.23675587],
                [ 0.42046973,  0.79006924],
                [ 0.72041562,  0.32988807],
                [ 0.39066572,  0.77851938],
                [ 0.49926506,  0.65065265]])
```

```
In [23]: a[np.r_[:len(a),0]]
```

```
Out[23]: array([[ 0.49926506,  0.65065265],
                [ 0.39066572,  0.77851938],
                [ 0.72041562,  0.32988807],
                [ 0.42046973,  0.79006924],
                [ 0.53983844,  0.23675587],
                [ 0.43063407,  0.87103111],
                [ 0.89189145,  0.29332695],
                [ 0.70529784,  0.70913641],
                [ 0.7662732 ,  0.93814996],
                [ 0.81161707,  0.61294193],
                [ 0.49926506,  0.65065265]])
```

```
In [24]: a.T
```

```
Out[24]: array([[ 0.49926506,  0.39066572,  0.72041562,  0.42046973,  0.53983844,
                  0.43063407,  0.89189145,  0.70529784,  0.7662732 ,  0.81161707],
                [ 0.65065265,  0.77851938,  0.32988807,  0.79006924,  0.23675587,
                  0.87103111,  0.29332695,  0.70913641,  0.93814996,  0.61294193]])
```

```
In [25]: a.conj().transpose()
```

```
Out[25]: array([[ 0.49926506,  0.39066572,  0.72041562,  0.42046973,  0.53983844,
                  0.43063407,  0.89189145,  0.70529784,  0.7662732 ,  0.81161707],
                [ 0.65065265,  0.77851938,  0.32988807,  0.79006924,  0.23675587,
                  0.87103111,  0.29332695,  0.70913641,  0.93814996,  0.61294193]])
```

```
In [27]: a.dot(a.T)
```

```
Out[27]: array([[ 0.67261447,  0.70159144,  0.5743209 ,  0.72398649,  0.4235683 ,
                  0.78173924,  0.63614419,  0.81353205,  0.99298319,  0.80402434],
                [ 0.70159144,  0.75871213,  0.53826595,  0.77934732,  0.3952154 ,
                  0.84634857,  0.57679213,  0.82761213,  1.0297246 ,  0.79425815],
                [ 0.5743209 ,  0.53826595,  0.62782481,  0.56354738,  0.46701098,
                  0.59757828,  0.73929759,  0.74204322,  0.86151966,  0.78690385],
                [ 0.72398649,  0.77934732,  0.56354738,  0.80100419,  0.41403925,
                  0.86924347,  0.60676195,  0.85682326,  1.0633981 ,  0.82552698],
                [ 0.4235683 ,  0.3952154 ,  0.46701098,  0.41403925,  0.34747888,
                  0.43869455,  0.55092416,  0.54863909,  0.63577623,  0.58325969],
                [ 0.78173924,  0.84634857,  0.59757828,  0.86924347,  0.43869455,
                  0.94414089,  0.63957574,  0.92140515,  1.14714114,  0.88340145],
                [ 0.63614419,  0.57679213,  0.73929759,  0.60676195,  0.55092416,
                  0.63957574,  0.88151105,  0.83705793,  0.95861717,  0.90366671],
                [ 0.81353205,  0.82761213,  0.74204322,  0.85682326,  0.54863909,
                  0.92140515,  0.83705793,  1.00031949,  1.20572713,  1.00709121],
                [ 0.99298319,  1.0297246 ,  0.86151966,  1.0633981 ,  0.63577623,
                  1.14714114,  0.95861717,  1.20572713,  1.46729995,  1.19695186],
                [ 0.80402434,  0.79425815,  0.78690385,  0.82552698,  0.58325969,
                  0.88340145,  0.90366671,  1.00709121,  1.19695186,  1.03442008]])
```

```
In [28]: a * a
```

```
Out[28]: array([[ 0.2492656 ,  0.42334887],
                [ 0.15261971,  0.60609243],
                [ 0.51899867,  0.10882614],
                [ 0.17679479,  0.6242094 ],
                [ 0.29142554,  0.05605334],
                [ 0.1854457 ,  0.75869519],
                [ 0.79547035,  0.0860407 ],
                [ 0.49744504,  0.50287445],
                [ 0.58717461,  0.88012534],
                [ 0.65872227,  0.37569781]])
```

```
In [29]: a/b
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-29-7a64888ccf6e> in <module>()
----> 1 a/b

NameError: name 'b' is not defined
```

```
In [30]: a/a
```

```
Out[30]: array([[ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.],
 [ 1.,  1.]])
```

```
In [31]: a**3
```

```
Out[31]: array([[ 0.12444961,  0.27545306],
 [ 0.05962329,  0.4718547 ],
 [ 0.37389475,  0.03590045],
 [ 0.07433686,  0.49316865],
 [ 0.15732271,  0.01327096],
 [ 0.07985924,  0.66084711],
 [ 0.7094732 ,  0.02523806],
 [ 0.35084692,  0.35660658],
 [ 0.44993617,  0.82568955],
 [ 0.53463024,  0.23028094]])
```

```
In [32]: (a>.5)
```

```
Out[32]: array([[False,  True],
 [False,  True],
 [ True, False],
 [False,  True],
 [ True, False],
 [False,  True],
 [ True, False],
 [ True,  True],
 [ True,  True],
 [ True,  True]], dtype=bool)
```

```
In [34]: np.nonzero(a>.5)
```

```
Out[34]: (array([0, 1, 2, 3, 4, 5, 6, 7, 7, 8, 8, 9, 9], dtype=int64),
 array([1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1], dtype=int64))
```

```
In [38]: v = np.random.rand(10,2)
```

```
In [41]: a[np.nonzero(v>.5)[0],:]
```

```
Out[41]: array([[ 0.49926506,  0.65065265],
 [ 0.39066572,  0.77851938],
 [ 0.42046973,  0.79006924],
 [ 0.53983844,  0.23675587],
 [ 0.43063407,  0.87103111],
 [ 0.89189145,  0.29332695],
 [ 0.89189145,  0.29332695],
 [ 0.70529784,  0.70913641],
 [ 0.7662732 ,  0.93814996]])
```

```
In [47]: a[np.nonzero(v.T>.5)[0],:]
```

```
Out[47]: array([[ 0.49926506,  0.65065265],
 [ 0.49926506,  0.65065265],
 [ 0.49926506,  0.65065265],
 [ 0.39066572,  0.77851938],
 [ 0.39066572,  0.77851938],
 [ 0.39066572,  0.77851938],
 [ 0.39066572,  0.77851938],
 [ 0.39066572,  0.77851938],
 [ 0.39066572,  0.77851938]])
```

```
In [48]: a[a<.5] = 0
```

```
In [49]: a * (a>.5)
```

```
Out[49]: array([[ 0.          ,  0.65065265],
 [ 0.          ,  0.77851938],
 [ 0.72041562,  0.          ],
 [ 0.          ,  0.79006924],
 [ 0.53983844,  0.          ],
 [ 0.          ,  0.87103111],
 [ 0.89189145,  0.          ],
 [ 0.70529784,  0.70913641],
 [ 0.7662732 ,  0.93814996],
 [ 0.81161707,  0.61294193]])
```

```
In [50]: a[:] = 3
```

```
In [51]: b = a.copy()
```

```
In [53]: b = a[1,:].copy()
```

```
In [54]: b = a.flatten()
```

```
In [55]: np.arange(1,11)
```

```
Out[55]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

```
In [56]: np.arange(10)
```

```
Out[56]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

```
In [62]: np.arange(1,11)[1]
```

```
Out[62]: 2
```

```
In [64]: np.zeros((3,4))
```

```
Out[64]: array([[ 0.,  0.,  0.,  0.],
                [ 0.,  0.,  0.,  0.],
                [ 0.,  0.,  0.,  0.]])
```

```
In [65]: np.ones((3,4))
```

```
Out[65]: array([[ 1.,  1.,  1.,  1.],
                [ 1.,  1.,  1.,  1.],
                [ 1.,  1.,  1.,  1.]])
```

```
In [66]: np.eye(3)
```

```
Out[66]: array([[ 1.,  0.,  0.],
                [ 0.,  1.,  0.],
                [ 0.,  0.,  1.]])
```

```
In [67]: np.diag(a)
```

```
Out[67]: array([ 3.,  3.])
```

```
In [68]: np.diag(a,0)
```

```
Out[68]: array([ 3.,  3.])
```

```
In [69]: np.random.rand(3,4)
```

```
Out[69]: array([[ 0.51083469,  0.86564706,  0.46202883,  0.62333548],
                [ 0.86541398,  0.19583144,  0.2552639 ,  0.01399741],
                [ 0.59949781,  0.71741526,  0.37002147,  0.66033718]])
```

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

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

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

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

```
In [72]: np.meshgrid([1,2,4],[2,4,5])
```

```
Out[72]: [array([1, 2, 4],
                [1, 2, 4],
                [1, 2, 4]]), array([[2, 2, 2],
                [4, 4, 4],
                [5, 5, 5]])]
```

```
In [73]: np.ix_([1,2,4],[2,4,5])
```

```
Out[73]: (array([[1],
                [2],
                [4]]), array([[2, 4, 5]]))
```



```
In [74]: np.tile(a,(2,1))
```

```
Out[74]: 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.]])
```

```
In [75]: np.concatenate((a,a),1)
```

```
Out[75]: 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.]])
```

```
In [76]: np.concatenate((a,a))
```

```
Out[76]: 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.]])
```

```
In [77]: a.max()
```

```
Out[77]: 3.0
```

```
In [78]: a.max(0)
```

```
Out[78]: array([ 3.,  3.] )
```

```
In [79]: a.max(1)
```

```
Out[79]: array([ 3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.] )
```

```
In [80]: np.maximum(a,a)
```

```
Out[80]: array([[ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.],
 [ 3.,  3.]])
```

```
In [83]: np.sqrt(np.dot(a.T,a))
```

```
Out[83]: array([[ 9.48683298,  9.48683298],
 [ 9.48683298,  9.48683298]])
```

```
In [84]: np.linalg.norm(a)
```

```
Out[84]: 13.416407864998739
```

```
In [85]: np.logical_and(a,a)
```

```
Out[85]: array([[ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True]], dtype=bool)
```

```
In [86]: np.logical_or(a,a)
```

```
Out[86]: array([[ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True],
                [ True,  True]], dtype=bool)
```

```
In [89]: a = np.random.rand(10,1)
```

```
In [90]: b = np.random.rand(10,1)
```

```
In [94]: np.linalg.inv(a*a.T)
```

```
Out[94]: array([[ 1.33934293e+16,  7.02892292e+16, -8.36740935e+16,
-8.24973038e+15, -7.24873990e+16,  6.41353814e+16,
 2.81005165e+15,  2.30543639e+15, -1.13844140e+16,
-3.33310042e+15],
 [ -3.05396523e+16, -1.14304814e+17,  1.17980040e+16,
 1.39146603e+17, -1.25208313e+17,  3.95639093e+15,
-1.09492277e+17,  1.02803410e+17,  5.47291040e+16,
-2.50178794e+16],
 [ -3.98531338e+16, -3.88146151e+17,  3.89599274e+17,
 9.83851771e+16,  3.53725356e+17, -2.63612435e+17,
-9.57247260e+16, -2.84554735e+16,  5.92130112e+16,
 4.90590802e+15],
 [ 4.42663638e+16,  3.52647165e+17, -2.02712128e+17,
-4.14845484e+17, -5.64456954e+16,  2.01063121e+17,
 2.03513882e+17, -1.13459614e+17, -1.26683178e+17,
 6.23509103e+16],
 [ 1.56581107e+16, -2.08319472e+17,  4.02113094e+16,
 1.73486334e+17,  2.14441705e+16, -9.66249669e+16,
-3.99101709e+16,  4.74324229e+16,  2.75091341e+16,
-3.06853645e+15],
 [ 3.22946211e+16,  3.19681077e+17, -2.74530267e+17,
-3.08014727e+16, -3.46388631e+17,  1.84159463e+17,
 6.71113504e+16,  3.33091959e+16, -3.30124125e+16,
-1.80214922e+16],
 [ -4.45741671e+16, -3.37011739e+17,  1.91109385e+17,
 2.21682051e+17,  1.98047208e+17, -1.53110871e+17,
-9.69358567e+16,  5.23446412e+16,  5.96844111e+16,
-2.40337074e+16],
 [ -1.76499374e+15,  9.87975034e+16, -1.05688584e+16,
-1.19992113e+17,  5.42895933e+16,  1.92984645e+16,
 5.96469669e+16, -6.54666888e+16, -3.41552539e+16,
 3.37565275e+16],
 [ 4.59890154e+15,  5.29056128e+16, -3.39332151e+15,
-9.18894817e+16,  1.59285305e+16,  1.64601778e+16,
 2.27612594e+16, -3.48638808e+16, -7.32987248e+15,
 1.00000246e+16],
 [ -8.60141170e+15, -7.77585411e+16,  4.87244713e+16,
 7.86651461e+16,  4.73803815e+16, -5.11730917e+16,
-3.69311201e+16,  3.18367123e+16,  1.84557106e+16,
-1.46915415e+16]])
```

```
In [95]: np.linalg.pinv(a*a.T)
```

```
Out[95]: array([[ 0.06778746,  0.0174019 ,  0.03740171,  0.01997325,  0.02040382,
                  0.05693705,  0.03615308,  0.04214191,  0.08462533,  0.07342765],
                [ 0.0174019 ,  0.00446729,  0.00960149,  0.00512739,  0.00523792,
                  0.01461646,  0.00928095,  0.01081836,  0.0217244 ,  0.01884981],
                [ 0.03740171,  0.00960149,  0.02063639,  0.01102023,  0.0112578 ,
                  0.031415 ,  0.01994745,  0.02325179,  0.04669201,  0.04051369],
                [ 0.01997325,  0.00512739,  0.01102023,  0.00588502,  0.00601189,
                  0.01677623,  0.01065233,  0.01241691,  0.02493444,  0.0216351 ],
                [ 0.02040382,  0.00523792,  0.0112578 ,  0.00601189,  0.00614149,
                  0.01713788,  0.01088197,  0.01268459,  0.02547196,  0.0221015 ],
                [ 0.05693705,  0.01461646,  0.031415 ,  0.01677623,  0.01713788,
                  0.04782341,  0.03036623,  0.03539646,  0.07107976,  0.06167445],
                [ 0.03615308,  0.00928095,  0.01994745,  0.01065233,  0.01088197,
                  0.03036623,  0.01928152,  0.02247554,  0.04513322,  0.03916117],
                [ 0.04214191,  0.01081836,  0.02325179,  0.01241691,  0.01268459,
                  0.03539646,  0.02247554,  0.02619866,  0.05260963,  0.04564829],
                [ 0.08462533,  0.0217244 ,  0.04669201,  0.02493444,  0.02547196,
                  0.07107976,  0.04513322,  0.05260963,  0.1056456 ,  0.09166651],
                [ 0.07342765,  0.01884981,  0.04051369,  0.0216351 ,  0.0221015 ,
                  0.06167445,  0.03916117,  0.04564829,  0.09166651,  0.07953714]])
```

```
In [96]: np.linalg.matrix_rank(a*a.T)
```

```
Out[96]: 1
```

```
In [98]: np.linalg.solve(a*a.T,a)
```

```
Out[98]: array([[ 0.4771612 ],
                [ 1.46781649],
                [-2.2144041 ],
                [ 1.12071186],
                [-0.15162013],
                [ 2.43220663],
                [-1.09254932],
                [ 0.12379368],
                [ 0.16268069],
                [-0.17024588]])
```

```
In [99]: U,S,Vh = np.linalg.svd(a*a.T)
```

```
In [100]: V = Vh.T
```

```
In [101]: np.linalg.cholesky(a*a.T).T
```

```
-----
LinAlgError                                Traceback (most recent call last)
<ipython-input-101-bade6eb7c488> in <module>()
----> 1 np.linalg.cholesky(a*a.T).T

C:\Users\Abhijit\AppData\Local\Continuum\Anaconda3\lib\site-packages\numpy\linalg\linalg.py in cholesky(a)
    610     t, result_t = _commonType(a)
    611     signature = 'D->D' if isComplexType(t) else 'd->d'
--> 612     r = gufunc(a, signature=signature, extobj=extobj)
    613     return wrap(r.astype(result_t, copy=False))
    614

C:\Users\Abhijit\AppData\Local\Continuum\Anaconda3\lib\site-packages\numpy\linalg\linalg.py in _raise_linalgerror_nonposdef(err, flag)
    91
    92 def _raise_linalgerror_nonposdef(err, flag):
--> 93     raise LinAlgError("Matrix is not positive definite")
    94
    95 def _raise_linalgerror_eigenvalues_nonconvergence(err, flag):

LinAlgError: Matrix is not positive definite
```

```
In [104]: a = np.random.rand(10,1) + 3
```

```
In [107]: a = np.eye(3)
```

```
In [108]: np.linalg.cholesky(a).T
```

```
Out[108]: array([[ 1.,  0.,  0.],
                 [ 0.,  1.,  0.],
                 [ 0.,  0.,  1.]])
```

```
In [109]: D,V = np.linalg.eig(a)
```

```
In [ ]:
```

```
In [111]: D,V = scipy.linalg.eig(a,a)
```

```
In [112]: Q,R = scipy.linalg.qr(a)
```

```
In [114]: L,U = scipy.linalg.lu_factor(a)
```

```
In [115]: scipy.sparse.linalg.cg
```

```
-----  
AttributeError                                Traceback (most recent call last)  
<ipython-input-115-8717a5e877a7> in <module>()  
----> 1 scipy.sparse.linalg.cg  
  
AttributeError: module 'scipy.sparse' has no attribute 'linalg'
```

```
In [119]: scipy.fft(a)
```

```
Out[119]: array([[ 1.0+0.j          ,  1.0+0.j          ,  1.0+0.j          ],  
                [ 1.0+0.j          , -0.5-0.8660254j, -0.5+0.8660254j],  
                [ 1.0+0.j          , -0.5+0.8660254j, -0.5-0.8660254j]])
```

```
In [120]: scipy.ifft(a)
```

```
Out[120]: array([[ 0.33333333+0.j          ,  0.33333333+0.j          ,  0.33333333+0.j  
                  ],  
                [ 0.33333333+0.j          , -0.16666667+0.28867513j,  
                  -0.16666667-0.28867513j],  
                [ 0.33333333+0.j          , -0.16666667-0.28867513j,  
                  -0.16666667+0.28867513j]])
```

```
In [121]: a.sort()
```

```
In [123]: I = np.argsort(a[:,1])
```

```
In [124]: b = a[I,:]
```

```
In [125]: np.linalg.lstsq(a,b)
```

```
Out[125]: (array([[ 0.,  0.,  0.],  
                [ 0.,  0.,  0.],  
                [ 0.,  0.,  1.]]),  
          array([], dtype=float64),  
          1,  
          array([ 1.73205081,  0.          ,  0.          ]))
```

```
In [129]: np.unique(a)
```

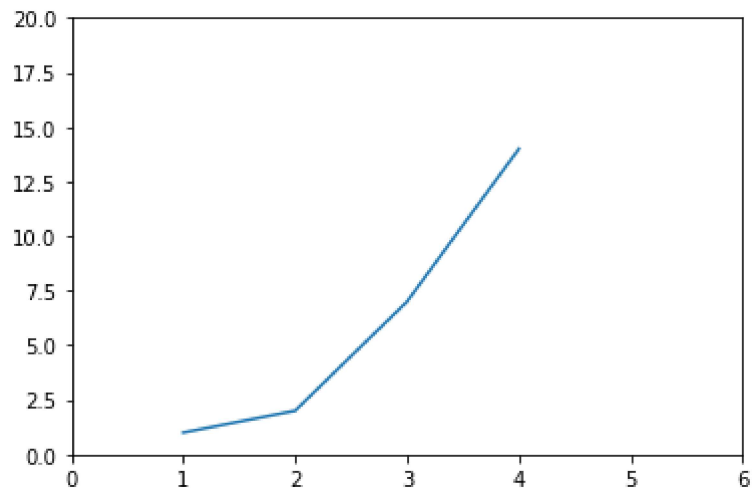
```
Out[129]: array([ 0.,  1.])
```

```
In [130]: a.squeeze()
```

```
Out[130]: array([[ 0.,  0.,  1.],  
                [ 0.,  0.,  1.],  
                [ 0.,  0.,  1.]])
```

```
In [131]: import matplotlib.pyplot as plt
```

```
In [132]: plt.plot([1,2,3,4], [1,2,7,14])  
plt.axis([0, 6, 0, 20])  
plt.show()
```



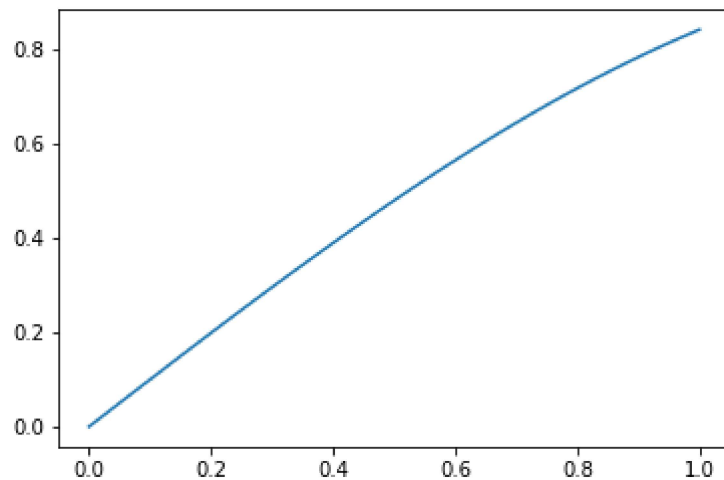
```
In [133]: x = np.linspace(0,1,100)
```

```
In [134]: y = np.sin(x)
```

```
In [135]: plt.plot(x,y)
```

```
Out[135]: [<matplotlib.lines.Line2D at 0x2293f3d32b0>]
```

```
In [136]: plt.show()
```



```
In [ ]:
```