

# Numpy & Scientific Computing in Python

Jonathan Street

# Outline

- Numpy / Scipy
  - Background
  - Tour and example use
- Related projects

# Numpy & Scipy

- Numpy – Building on Numeric with some improvements from numarray
- Scipy – Travis Oliphant and Eric Jones each built libraries during their PhD's and decided to merge their collections in 2001

# Numpy

- N-dimensional array object
- Universal function object

# Numpy Slicing

```
>>> l1 = [ [1,2,3,1], [4,5,6,1], [7,8,9,1], ]
>>> l1[1], l1[1:], l1[1][1]
([4, 5, 6], [[4, 5, 6], [7, 8, 9]], 5)
>>> l1[1:][1:]
[[7, 8, 9]]
>>> import numpy as np
>>> a1 = np.array(l1)
>>> a1[1:,1:]
array([[5, 6],
       [8, 9]])
>>>
```

```
>>> a2
array([[0, 1, 2],
       [3, 4, 5],
       [6, 7, 8]])
>>> a2[1:,1:] = 9
>>> a2
array([[0, 1, 2],
       [3, 9, 9],
       [6, 9, 9]])
>>> a2[a2<9] = 0
>>> a2
array([[0, 0, 0],
       [0, 9, 9],
       [0, 9, 9]])
```

# Numpy basic operations

```
>>> a2 = np.arange(10).reshape(2,5)
>>> a2
array([[0, 1, 2, 3, 4],
       [5, 6, 7, 8, 9]])
>>> a2.shape
(2, 5)
>>> a2.ndim
2
>>> [i for i in dir(a2) if '__' not in i]
['T', 'all', 'any', 'argmax', 'argmin', 'argsort', 'astype', 'base', 'byteswap',
 'choose', 'clip', 'compress', 'conj', 'conjugate', 'copy', 'ctypes', 'cumprod',
 'cumsum', 'data', 'diagonal', 'dtype', 'dump', 'dumps', 'fill', 'flags', 'flat',
 'flatten', 'getfield', 'imag', 'item', 'itemset', 'itemsize', 'max', 'mean', '
min', 'nbytes', 'ndim', 'newbyteorder', 'nonzero', 'prod', 'ptp', 'put', 'ravel',
 'real', 'repeat', 'reshape', 'resize', 'round', 'searchsorted', 'setfield', 's
etflags', 'shape', 'size', 'sort', 'squeeze', 'std', 'strides', 'sum', 'swapaxes',
 'take', 'tofile', 'tolist', 'tostring', 'trace', 'transpose', 'var', 'view']
>>> a2.max(), a2.max(axis=0), a2.max(axis=1)
(9, array([5, 6, 7, 8, 9]), array([4, 9]))
>>>
```

# Basic modules

- Linear Algebra
- Discrete Fourier Transforms
- Random Numbers
- Matrix-specific functions
- Ctypes utility functions



Scipy

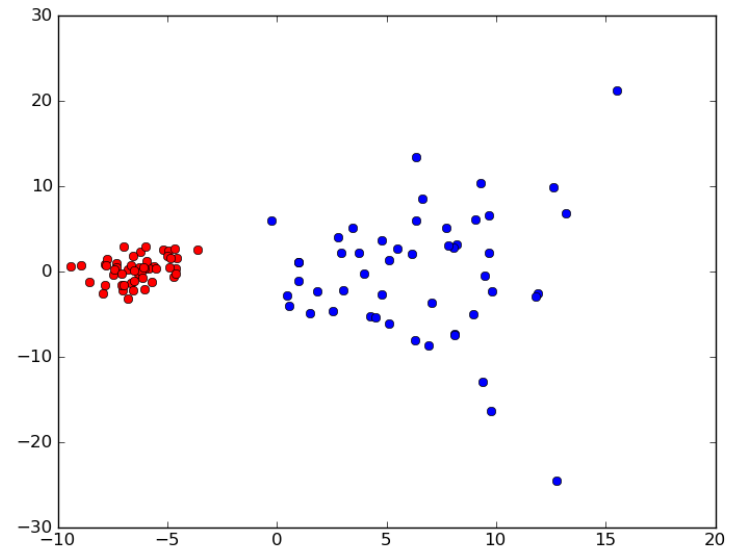
```
>>> import scipy
>>> len(dir(scipy))
533
>>> from scipy import stats
>>> len(dir(stats)) # Over 80 distributions and 70 statistical tests
229
>>>
>>> a3 = stats.norm.rvs(loc=5, scale=10, size=500)
>>> a4 = stats.norm.rvs(loc=5, scale=10, size=500)
>>> stats.ttest_ind(a3, a4)
(0.84230882215903091, 0.39981691731046398)
>>> a5 = stats.norm.rvs(loc=8, scale=10, size=500)
>>> stats.ttest_ind(a3, a5)
(-5.0490128841816402, 5.2770696812309836e-07)
>>>
```

# Scientific computing

- Images
  - Scipy.ndimage, mahotas, scikits.image, pyopencv
- Machine Learning
  - Scikit.learn, pybrain
- Data processing
  - MDP
- Domain-specific packages
  - BioPython, PyClimate, SpacePy
- Data visualisation
  - Matplotlib, Mayavi

# MDP & Matplotlib

```
>>> data = np.random.normal(loc=1., scale=1., size=(100,1000))
>>> data.shape
(100, 1000)
>>> data[:50,500:600] = data[:50,500:600] ** 2
>>> pcan = mdp.nodes.PCANode(output_dim=0.95)
>>> pcar = pcan.execute(data)
>>> pcar.shape
(100, 89)
>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)
>>> ax.plot(pcar[:50,0], pcar[:50,1], 'bo')
[<matplotlib.lines.Line2D object at 0x03225F30>]
>>> ax.plot(pcar[50:,0], pcar[50:,1], 'ro')
[<matplotlib.lines.Line2D object at 0x031E39F0>]
>>> plt.show()
```



# Useful Links

- <http://www.tramy.us/>
- [http://www.rexx.com/~dkuhlman/scipy\\_course\\_01.html](http://www.rexx.com/~dkuhlman/scipy_course_01.html)
- [http://scipy.org/Tentative NumPy Tutorial](http://scipy.org/Tentative_NumPy_Tutorial)
- [http://scipy.org/Numpy Example List](http://scipy.org/Numpy_Example_List)
- [http://www.scipy.org/Topical Software](http://www.scipy.org/Topical_Software)