**numpy**

A reference to NumPy, the Python package for scientific computing

**Load NumPy**

|  |  |
| --- | --- |
| import numpy as np | Conveniently imports the package with all the available NumPy modules |

**Array Initialization**

|  |  |
| --- | --- |
| np.array([2, 3, 4]) | Direct initialization of array |
| np.empty(20, dtype=np.float32) | Single precision array of size 20 |
| np.zeros(200) | Initializes 200 with value zero |
| np.ones((3,3), dtype=np.int32) | Creates a 3 x 3 integer matrix with all ones |
| np.zeros\_like(a) | Creates an array with zeros and the shape of a |
| np.copy(a) | Copy array to new memory |

**Indexing**

|  |  |
| --- | --- |
| a = np.arange(100) | Initialization with 0 - 99 |
| a[:3] = 0 | Set the first three indices to zero |
| a[2:5] = 1 | Set indices 2-4 to 1 |
| a[:-3] = 2 | Set all but last three elements to 2 |
| a[[1, 1, 3, 8]] | Return array with values of the indices |
| a = a.reshape(10, 10) | Transform to 10 x 10 matrix |
| a.T | Return transposed view of array 'a' |

**Array Properties and Operations**

|  |  |
| --- | --- |
| len(a) | Returns length of axis 0 |
| a.shape | Returns a tuple with the lengths of each axis |
| a.ndim | Number of dimensions (axes) |
| a.sort(axis=1) | Sort array along axis |
| a.flatten() | Collapse array to one dimension |
| a.astype(np.int16) | Cast to integer |
| a.tolist() | Convert (possibly multidimensional) array to list |
| np.argmax(a, axis=1) | Return index of maximum along a given axis |
| np.cumsum(a) | Return cumulative sum |

**Elementwise Operations and Math Functions**

|  |  |
| --- | --- |
| a \* 5 | Multiplication with scalar and same goes for addition, substraction and division |
| a + b | Addition with array b |
| np.exp(a) | Exponential (complex and real) |
| np.power(a, b) | a to the power b |
| np.sin(a) | Returns sine of a |
| np.cos(a) | Returns cosine of a |
| np.std(a, axis=1) | Returns standard deviation for given axis |
| np.radians(a) | Convert degrees to radians |
| np.degrees(a) | Convert radians to degrees |
| np.var(a) | Variance of array |

**Linear Algebra**

|  |  |
| --- | --- |
| evals, evecs = np.linalg.eig(a) | Find eigenvalues and eigenvectors |
| evals, evecs = np.linalg.eigh(a) | p.linalg.eig for hermitian matrix |

**Reading/ Writing Files**

|  |  |
| --- | --- |
| np.loadtxt(fname/fobject, skiprows=2, delimiter=',') | Loads ascii data from specified file |
| np.savetxt(fname/fobject, array, fmt='%.5f') | Write ascii data to specified file |

**Rounding**

|  |  |
| --- | --- |
| np.ceil(a) | Rounds to nearest upper int |
| np.floor(a) | Rounds to nearest lower int |

**Interpolation, Integration and Optimization**

|  |  |
| --- | --- |
| np.trapz(a, x=x, axis=1) | Integrate along axis 1 |
| np.interp(x, xp, yp) | Interpolate function xp, yp at points x |
| np.linalg.lstsq(a, b) | Solve a x = b in least square sense |