

Creating NumPy Arrays

Thomas Schwarz, SJ

Creating NumPy Arrays

- Use `np.array` on a Python list
 - You can explicitly set the type

- ```
import numpy as np
my_arr = np.array([[1,2], [3,4]], dtype='float32')
my_arr
array([[1., 2.],
 [3., 4.]], dtype=float32)
```

- Use `np.zeros` (SIC!)
- Use `np.ones`

```
>>> my_arr = np.zeros((3,5), dtype = int)
>>> my_arr
array([[0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0]])
```

```
>>> my_arr = np.ones((3,5), dtype = float)
>>> my_arr
array([[1., 1., 1., 1., 1.],
 [1., 1., 1., 1., 1.],
 [1., 1., 1., 1., 1.]])
```

# Creating NumPy Arrays

- Use linspace
  - First and second variable are bounds
  - Third argument is the number of values

- ```
>>> np.linspace(1,3,5)
array([1. , 1.5, 2. , 2.5, 3. ])
```

- Use arange (close to range)

- ```
>>> np.arange(0,20,3)
array([0, 3, 6, 9, 12, 15, 18])
```

# Creating NumPy Arrays

- Use a probability distribution

- ```
>>> np.random.normal(10,2,(3,3))  
array([[ 9.71266683, 10.1511357 , 11.59357302],  
       [10.6661394 ,  9.75288609,  9.45306968],  
       [10.87938216, 10.7632545 ,  8.73987421]])
```

- Use eye for the Identity matrix

```
>>> np.eye(4)  
array([[1., 0., 0., 0.],  
       [0., 1., 0., 0.],  
       [0., 0., 1., 0.],  
       [0., 0., 0., 1.]])
```

Creating NumPy Arrays

- From CSV files
 - CSV (Comma separated) files are the most common way to exchange data
 - Two-dimensional data
 - In each row, data is separated by a comma or white-space

Creating NumPy Arrays

- Use `genfromtxt`

```
testdata = np.genfromtxt('StatewiseTestingDetails.csv',  
                        delimiter = ',',  
                        skip_header=1,  
                        usecols=(0,1,2,4)  
                        )
```

- Unfortunately, only easy for numeric data
 - Needs “converters”
 - nan means not a number

```
>>> testdata  
array([[ nan,          nan,  1.4030000e+03,  1.2000000e+01],  
       [ nan,          nan,  2.6790000e+03,  2.7000000e+01],  
       [ nan,          nan,  2.8480000e+03,  3.3000000e+01],  
       ...,  
       [ nan,          nan,  1.6092192e+07,          nan],  
       [ nan,          nan,  1.6122345e+07,          nan],  
       [ nan,          nan,  1.6162814e+07,          nan]])  
>>>
```

Pandas Data Frames

- Pandas is written on top of numpy
- Has a data frame that can be filled from a file

```
testframe = pd.read_csv('StatewiseTestingDetails.csv')
```

```
>>> testframe
```

	Date	State	TotalSamples	Negative	Positive
0	2020-04-17	Andaman and Nicobar Islands	1403.0	1210	12.0
1	2020-04-24	Andaman and Nicobar Islands	2679.0	NaN	27.0
2	2020-04-27	Andaman and Nicobar Islands	2848.0	NaN	33.0
3	2020-05-01	Andaman and Nicobar Islands	3754.0	NaN	33.0
4	2020-05-16	Andaman and Nicobar Islands	6677.0	NaN	33.0
...
16331	2021-08-06	West Bengal	15999961.0	NaN	NaN
16332	2021-08-07	West Bengal	16045662.0	NaN	NaN
16333	2021-08-08	West Bengal	16092192.0	NaN	NaN
16334	2021-08-09	West Bengal	16122345.0	NaN	NaN
16335	2021-08-10	West Bengal	16162814.0	NaN	NaN

```
[16336 rows x 5 columns]
```