NumPy

Prof. Carlos J. Costa, PhD

2020

Learning Objectives

Understand what is NumPy

Know the main features of Numpy

Solve mathematical problems using NumPy

Numpy

Numerical Python

Is an open source Python library

Is a fundamental Python library for scientific computing.

Provides array related functionality

Has higher level of performance

Numpy

```
import numpy as np
c = np.array([1,2,3,4])
print(type(c))
```

```
<class 'numpy.ndarray'>
```

Shape, Rank and Size

Create array

```
a = np.array([1,2,3,4])
```

Shape, rank and size:

```
shape = a.shape
rank = np.ndim(a)
size = a.size
```

Shape, Rank and Size

The bidimensional array (matrix):

```
b = np.array([[1,2,3],[4,5,6]])
```

What information can be obtained about this array:

```
shape = b.shape
rank = np.ndim(b)
size = b.size
```

Access an Array Element

Change value to array:

```
a[2]=50
print(a)
```

Array with Zeros Only

Create an array with zeros only

```
a = np.zeros((2,2))
print(a)
```

Array with Only “ones”

Create array with only “one”

```
b = np.ones((1,2))
print(b)
```

Identity Array

create 3x3 identity array

```
d = np.eye(3)
print(d)
```

Array filled with random numbers

Create array filled with random numbers

```
e = np.random.random((4,4))
print(e)
```

Slicing Arrays

```
a = np.array([[1,2,3,4], [5,6,7,8], [9,10,11,12]])
print(a)
```

```
[[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]]
```

Slicing Arrays

```
b = a[:3, 1:3]

print(b)
```

Slicing Arrays

```
b = a[:3, 1:3]

print(b)
```

```
[[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]]
```

Slicing Arrays

```
b[0, 0] = 99
```

what happens to array a?

Slicing Arrays

```
b[0, 0] = 99
```

what happens to array a?

```
b = a[:3, 1:3]
```

bCopyA= `a[:3, 1:3].copy()`

Handling Indexes

Another example handles indexes

```
import numpy as np
#create a new array
a = np.array([[1,2,3], [4,5,6], [7,8,9], [10, 11, 12]])
print(a)
```

Handling Indexes

create an array of indexes

```
b = np.array([0, 2, 0, 1])

print(a[np.arange(4), b])
```

Handling Indexes

```
a[np.arange(4), b] += 10


print(a)
```

Handling Indexes

```
b=1
b+=10
B
```

```
b=1
b=b+10
b
```

Searching Arrays

```
grades = np.array([14, 12, 13, 14, 15, 14, 14])
x = np.where(grades == 14)

print(x)
```

Searching Arrays

```
grades = np.array([14, 12, 13, 14, 15, 14, 14])
x = np.where(grades == 14)

print(x)
```

```
(array([0, 3, 5, 6], dtype=int64),)
```

Conclusions

NumPy (Numerical Python) is an open source Python library

Array related functionality, including searches, index handling

References

http://www.numpy.org/