

Programming with NumPy

Computational Design Laboratory

Department of Aerospace Engineering
Iowa State University

September 7, 2020

Outline

- Installing NumPy
- NumPy arrays
- NumPy matrices
- Basic operations using NumPy
- Further reading

Installing NumPy

- Linux: open terminal and type 'pip install numpy'
- Windows: open Anaconda prompt from start menu and type 'pip install numpy' (always install packages through Anaconda prompt)

NumPy arrays

- Open the folder you have downloaded from GitHub
- Open the 03_programming_with_numpy folder
- Open arrays.py from Spyder and run it
- This code can also be found here : [▶ Link](#)

This tutorial covers:

- Defining a NumPy array in Python
- Printing out its values, type and shape
- Printing out individual element values
- Changing individual element values
- Reshaping the array
- Defining a NumPy array of zeros and ones

NumPy arrays

The image shows the Spyder Python IDE interface. The main editor window displays a Python script titled 'numpy_array.py' with the following code:

```
1 #-*- coding: utf-8 -*-
2 """
3 Created on Mon Aug 31 09:05:32 2020
4
5 @author:
6 """
7
8 #Numpy tutorials
9 #Arrays
10
11 #Import the numpy library
12 import numpy as np
13
14
15 a = np.array([1, 2, 3])
16 #print the values of 'a'
17 print(a)
18 #returns the type of 'a'
19 print(type(a))
20 #returns the shape of 'a'
21 print(a.shape)
22 #prints the value of 'a'
23 print(a[0], a[1], a[2])
24 # Change an element of the array
25 a[0] = 5
26 #print the values of 'a'
27 print(a)
28
29 #reshape a
30 a = a.reshape(-1,1)
31 #returns the shape of 'a'
32 print(a.shape)
33 #Note (3,) means it is a 1d matrix
34 #(3,1) means it is a 2d matrix
35 #In python this matters
```

The Variable explorer on the right shows the state of the program:

Name	Type	Size	Value
a	int32	(1, 3)	[[5 2 3]]
b	float64	(3,)	[0. 0. 0.]
c	float64	(3,)	[1. 1. 1.]

The IPython console shows the execution output:

```
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.4.0 -- An enhanced Interactive Python.

In [1]: runfile('V:/ISU/ISU_optimization_course_prep/
numpy_array.py', wdir='V:/ISU/
ISU_optimization_course_prep')
[1 2 3]
<class 'numpy.ndarray'>
(3,)
1 2 3
[5 2 3]
(3, 1)
(1, 3)
[0. 0. 0.]
[1. 1. 1.]

In [2]:
```

The status bar at the bottom indicates: Permissions: RW, End-of-lines: CRLF, Encoding: UTF-8, Line: 1, Column: 1, Memory: 70 %.

NumPy matrices

- Open the folder you have downloaded from GitHub
- Open the 03_programming_with_numpy folder
- Open matrices.py from Spyder and run it
- This code can also be found here : [▶ Link](#)

This tutorial covers:

- Defining a NumPy matrix
- Printing out individual row values
- Transposing the matrix
- Reshaping the matrix

NumPy matrices

Spyder (Python 3.6)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Editor - V:\ISU\ISU_optimization_course_prep\numpy_slide.py

```
1
2 #Numpy tutorials
3 #Matrices
4
5 #Import the numpy library
6 import numpy as np
7
8 a = np.array([[1,2,3,4], [5,6,7,8], [9,10,11,12]])
9 #print the values of 'a'
10 print(a)
11 #returns the shape of 'a'
12 print(a.shape)
13 #prints the 1st row values
14 print(a[0, :])
15
16 #transpose the matrix
17 b = np.transpose(a)
18 #print the values of 'b'
19 print(b)
20 #returns the shape of 'b'
21 print(b.shape)
22
23 #reshape into different size
24 c = a.reshape(2,-1)
25 print(c)
```

Variable explorer

Name	Type	Size	Value
a	int32	(3, 4)	$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix}$
b	int32	(4, 3)	$\begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}$
c	int32	(2, 6)	$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 7 & 8 & 9 & 10 & 11 & 12 \end{bmatrix}$

Variable explorer File explorer Help Profiler Static code analysis

Python console

Console 1/A

```
In [1]: runfile('V:/ISU/ISU_optimization_course_prep/numpy_slide.py',
wdir='V:/ISU/ISU_optimization_course_prep')
[[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]]
(3, 4)
[[ 1  2  3  4]
 [ 1  5  9]
 [ 2  6 10]
 [ 3  7 11]
 [ 4  8 12]]
(4, 3)
[[ 1  2  3  4  5  6]
 [ 7  8  9 10 11 12]]

In [2]:
```

Python console History log

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 25 Column: 9 Memory: 76 %

Basic operations using NumPy

- Open the folder you have downloaded from GitHub
- Open the 03_programming_with_numpy folder
- Open matrix_operations.py from spyder and run it
- This code can also be found here : [▶ Link](#)

This tutorial covers:

- Defining a matrix with random integer values
- Adding and subtracting two matrices
- Matrix multiplication
- Element-wise multiplication
- Inverting a matrix

Basic operations using NumPy

Spyder (Python 3.6)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Editor - V:\ISU\ISU_optimization_course_prep\numpy_slide.py

```
1 #Numpy tutorials
2 #Matrix operations
3
4 #Import the numpy Library
5 import numpy as np
6
7 a = np.random.randint(10, size = (3,3))
8 print(a)
9 print("\n")
10 b = np.random.randint(10, size = (3,3))
11 print(b)
12 print("\n")
13
14 #Add
15 c = np.add(a,b)
16 print(c)
17 c = a + b
18 print(c)
19 print("\n")
20
21 #subtract
22 d = np.subtract(a,b)
23 print(d)
24 d = a - b
25 print(d)
26 print("\n")
27
28 #multiply (element-wise)
29 e = np.multiply(a,b)
30 print(e)
31 e = a * b
32 print(e)
33
34 #Matrix multiplication
35 e = np.dot(a,b)
36 print(e)
37 print("\n")
```

Variable explorer

Name	Type	Size	Value
a	int32	(3, 3)	[[4 1 3] [8 7 3] [9 9 7]]
b	int32	(3, 3)	[[0 1 0] [9 7 3] [4 4 9]]
c	int32	(3, 3)	[[4 2 3] [17 14 6] [13 13 16]]

Variable explorer | File explorer | Help | Profiler | Static code analysis

IPython console

```
Console 1/A
[[4 1 3]
 [8 7 3]
 [9 9 7]]

[[0 1 0]
 [9 7 3]
 [4 4 9]]

[[4 2 3]
 [17 14 6]
 [13 13 16]]

[[4 0 3]
 [-1 0 0]
 [5 5 -2]]
[[4 0 3]
 [-1 0 0]
 [5 5 -2]]
```

IPython console | History log

Permissions: RW | End-of-lines: CRLF | Encoding: UTF-8 | Line: 48 | Column: 9 | Memory: 79 %

Further reading

- Additional NumPy tutorials [▶ Link](#)