

NumPy & Python Ecosystem

TensorFlow -
Pandas -
scikit-learn -
matplotlib -

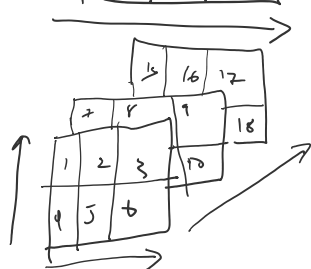
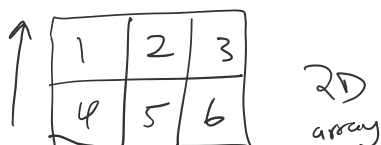
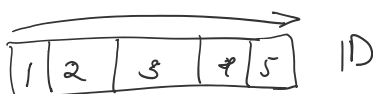
NumPy

NumPy Arrays

1D

2D

3D



import numpy as np

np.

ipython

Creating 1D arrays from
List

students_list = [1, 4, 6, 7, 8]

student_array = np.array(students_list)

student_array

Create a 2D array using lists
2 lists

student_marks = [[1, 3, 6, 7],
[...]]

7

2D dimensional array

Numpy Array

- Can only contain a single value
- Uses less space memory

Creating Arrays From Scratch

2. np.random.random()

3. np. arrange()

NP-Zeros

Creates an array with zeros

$np.zeros(\text{C_5, 3})$ (shape)

$\text{np.zeros}(5)$
 shape

Create an array of zeros
with three rows and two columns

$\text{zero_array} = \text{np.zeros}(3, 2)$

2. np.random.random

import random
random module

$\text{np.random.random}(5, 3)$
 random module random function

import random
 + 2
 random.

3. $\text{np.arange}()$

$\text{np.arange}(-3, 4)$

-3, -2, -1, 0, 1, 2, 3

`np.arange(-3, 4, 3)`

-3, 0, 3 ↓ step value

from matplotlib import pyplot as plt

`plt.scatter(np.arange(0, 7),
np.arange(-3, 4))`

`plt.show`

ARRAY DIMENSIONALITY

3D Array

2 array-1-2D = `np.array([[2, 2]
[6, 7]])`

3 array-2-2D =

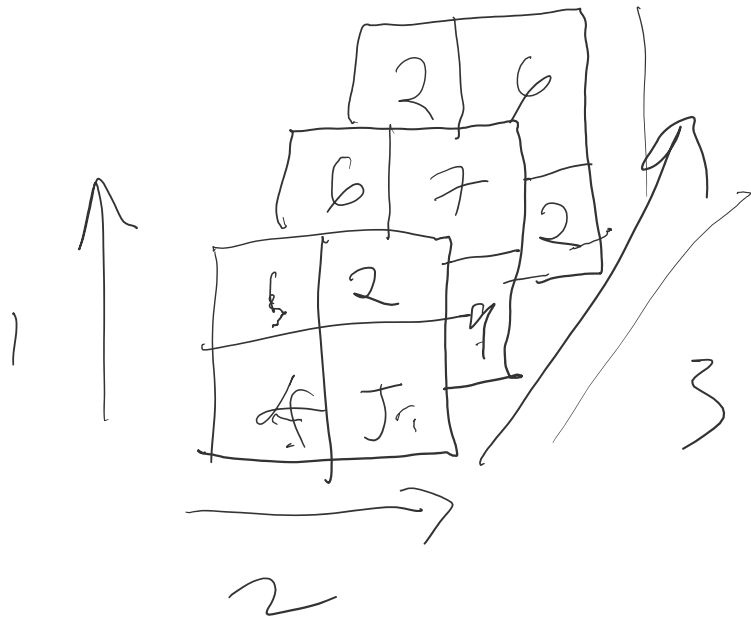
`[2, 6]`

3 array-3-2D =

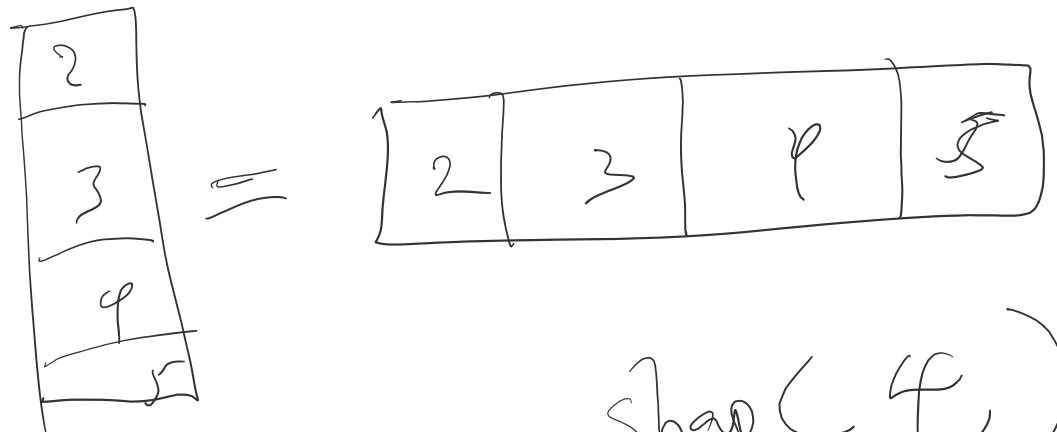
array-3D = `np.array([`

[4, 5]]
[3, 9]
[9, 2]

array-1 2D, array-2 2D, array-3 2D]



Vector Arrays



shape(4,)

shape(4,)

2
3
4
5

\neq

2	3	4	5
---	---	---	---

shape = (1, 4)

shape = (4, 1)

~~A~~

- div
- m

np.array (type
dtype)



)

B

ation

magnitude

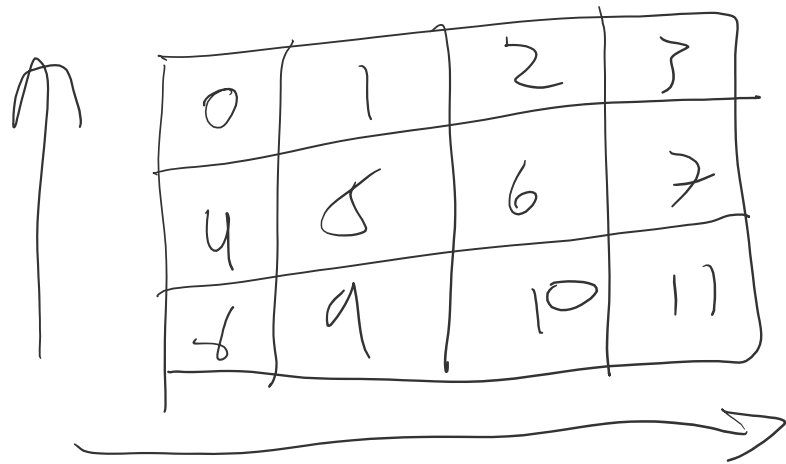
re:

y₁₇ - 0

Matrix } Tensor

Matrix Array

A matrix has two dimensions



0	1	2	3
4	5	6	7
8	9	10	11

A Tensor

- A tensor has 3 or

gnd n

move dimension

SHAPE SHIFTING

A `array` attribute
• `shape ()`

`np.shape (student_list_a)`

`students_array.shape`

5,
1 1 1
3, 2, 2
1 1
3, 2

Rows } Columns
m 2D Array

- Row 0 the first dimension
- Columns is the second

15

rays

enon

and dimensions

1st dimension

9	3	6
7	8	5
10	11	4

second dimension

Flattening An

array = np.array(LC

array.flatten()



1 dimension

Array

[1, 2], [3, 4], [5, 6])

array.flatten()

>>

Assignment

Look into the data
types in Numpy
to what we have in

- Bits } Bytes
- Type conversion

thent data

in comparison

python

~ Type Coercion

SELECTING

DATA In

Indexing: is a

hierarchy

3 UPDATION

Arrays

n order - based

normal data

9

method

— of a
Num

array = np. arra

1D Array

array [3]

crossing data
py indexing is

g C [2 , 4 , 6 , 8
0 1 2 3

y

Zero based

100
4

1

>> 8

2D Avrc

1	2	
5	6	
5	4	
3	4	

ys

3	25
9	2
2	1
6	7

array [ro

array [

w ind, (column is 6)

2, 2]

1 2 1 ~~1 2 1~~

array[2]

~~~~~

))) x

array [ 3

) > 4

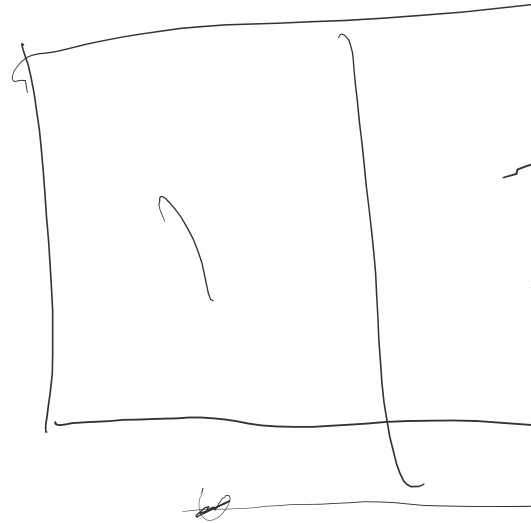
array [ 0

3, 2]

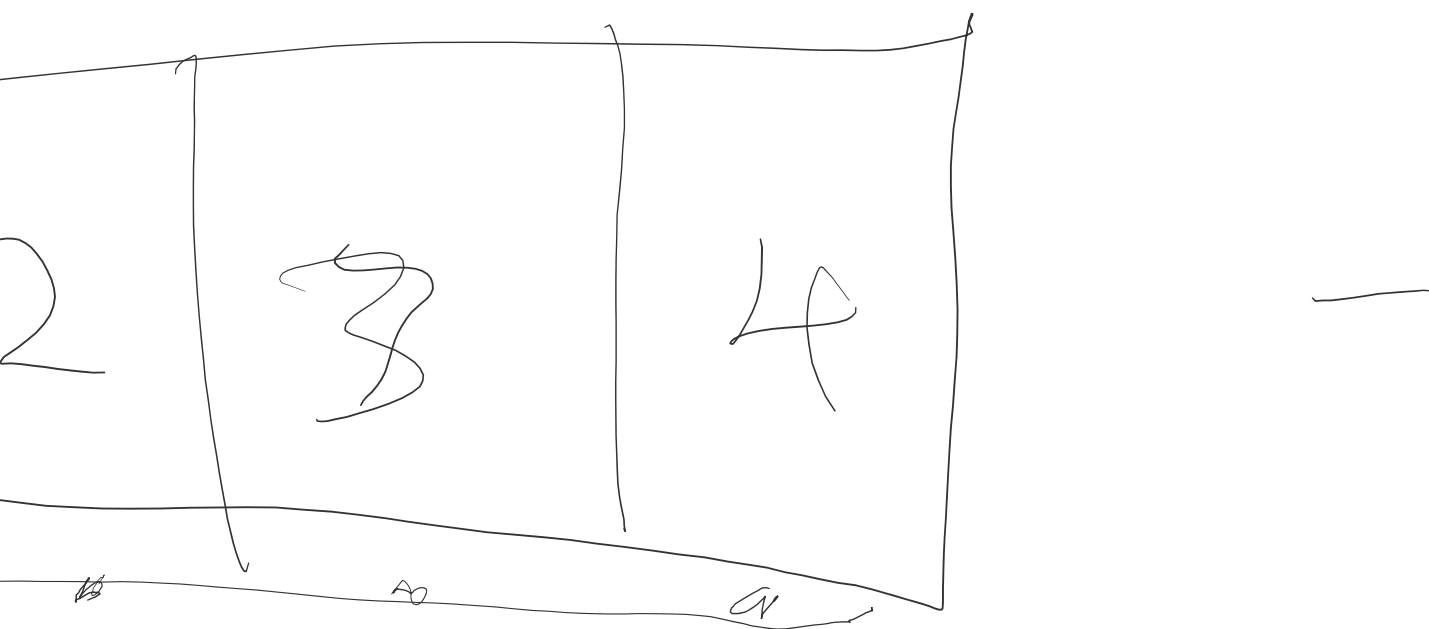
0, 3]



# Vectors



Shan



$$pe = (1, 0)$$

1D - any





1D

Array

array

=

np. a

↑

y

array C [1, 2, 3

|   |   |   |   |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
|---|---|---|---|

145)



Vectors = Mat

Shape = CLT

yn inde

} Direction

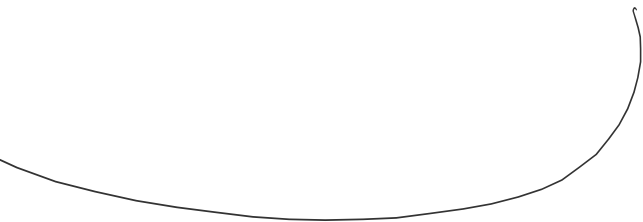
NO  
↑

+

10

Row

no claim







$$arr = np.$$

any ( L L 1, 2

[ 5, 6,

[ 9, 10

[ 6, 2

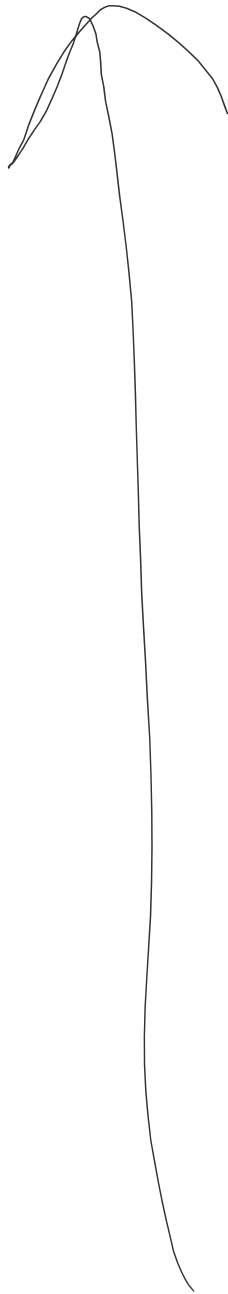
13, 14],

7, 8],

, 11, 12],

, 3, 4])





|   |    |
|---|----|
| 1 | 2  |
| 5 | 6  |
| 9 | 10 |
| 6 | 2  |

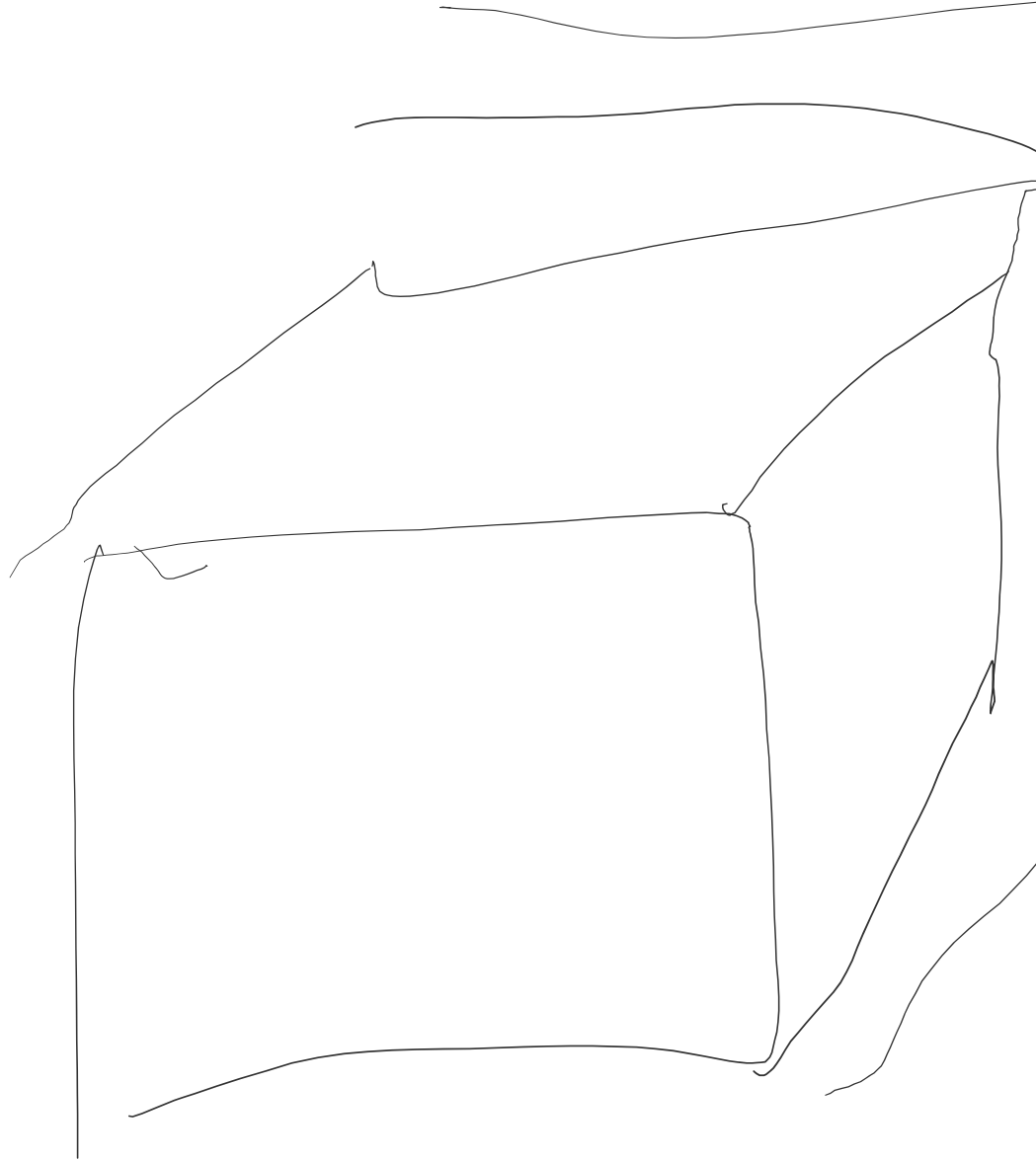
|  |    |    |
|--|----|----|
|  | 3  | 4  |
|  | 7  | 8  |
|  | 11 | 12 |
|  | 3  | 4  |

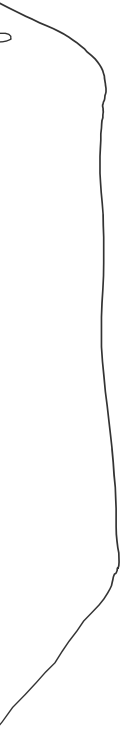
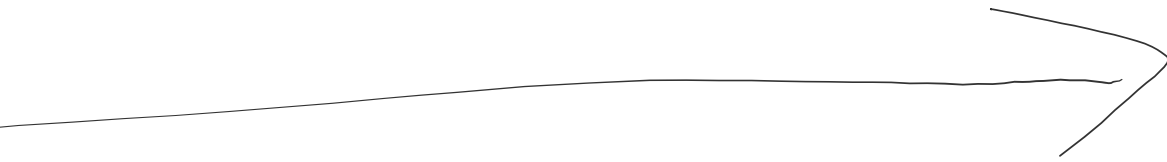
5

hype = (1st, 4th, 2nd)









Shop

pe (8 2, 2)  
11  
204  
3



SLLIC1

- Extract

based on g

array

and

11

11

111



data

from one

array





with the sliced

1D Array

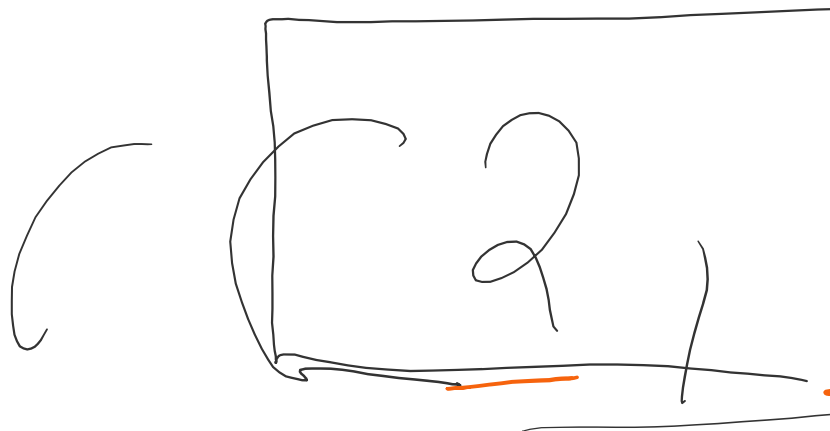
array = np.

array [0; 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28; 29; 30; 31; 32; 33; 34; 35; 36; 37; 38; 39; 40; 41; 42; 43; 44; 45; 46; 47; 48; 49; 50; 51; 52; 53; 54; 55; 56; 57; 58; 59; 60; 61; 62; 63; 64; 65; 66; 67; 68; 69; 70; 71; 72; 73; 74; 75; 76; 77; 78; 79; 80; 81; 82; 83; 84; 85; 86; 87; 88; 89; 90; 91; 92; 93; 94; 95; 96; 97; 98; 99]

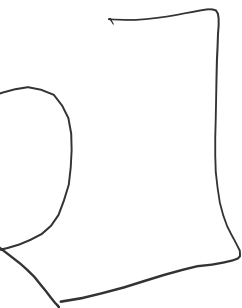
1 dafon

y

ax ray



o



4, 9, 10, 12]

1 2



inductive

2D

Ar

Run

Row

0

1

2

~~1~~

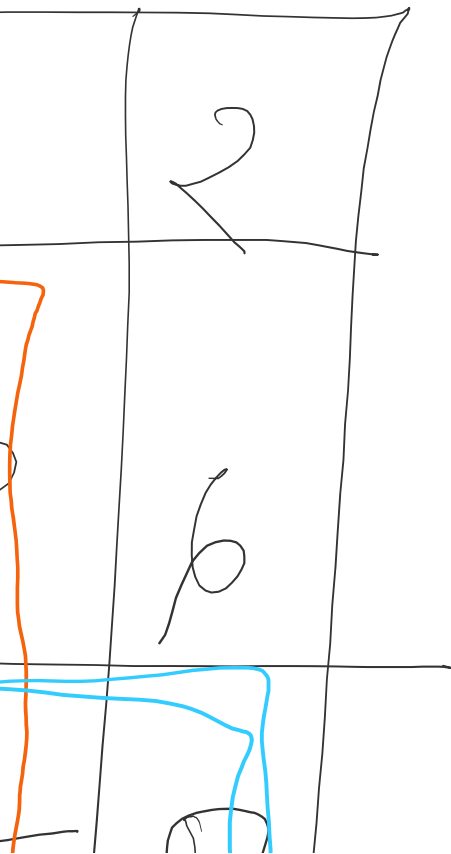
2

3

|   | 0 | 1 | 2 |
|---|---|---|---|
| 0 | 6 | 7 | 9 |
| 1 | 1 | 4 | 5 |
| 2 | 8 | 0 | 7 |

Exclusive

ay  
3



(2,

1/21 need  
provide nv  
inf nn

1)

to

happy  
hand

hah





|   |   |   |   |
|---|---|---|---|
| 2 | 9 | 6 | 7 |
| 3 | 2 | 6 | 1 |

- Row start

- Allowed by



info on  
row and col  
sliced

and stop index  
column start

how both  
turns will be

cer provided  
? App indlog



Schinken = [ 1 ]

0

1 - 6

3, 137

|   |   |   |
|---|---|---|
| 2 | 3 | 4 |
| 7 | 8 | 9 |

50 durb

=

[

$$\mu = [1:3, 4:6]$$

$$[1:3:4, 4:6:7]$$





1

0

2

1

3

2

4

array

[

2

2

i

>

|   |   |   |
|---|---|---|
|   |   |   |
| 2 | 3 | 6 |
| 5 | 6 | 7 |

f 1 , 1 : 4 : 1

j k

Step value

$\frac{1}{4}$

1 2 3

Ops



i

i = start

j = stop

k = step

U      ↖

i = 2

=

7

i + k = 2

=

8

i + k = 3

=

9

i + k = 4

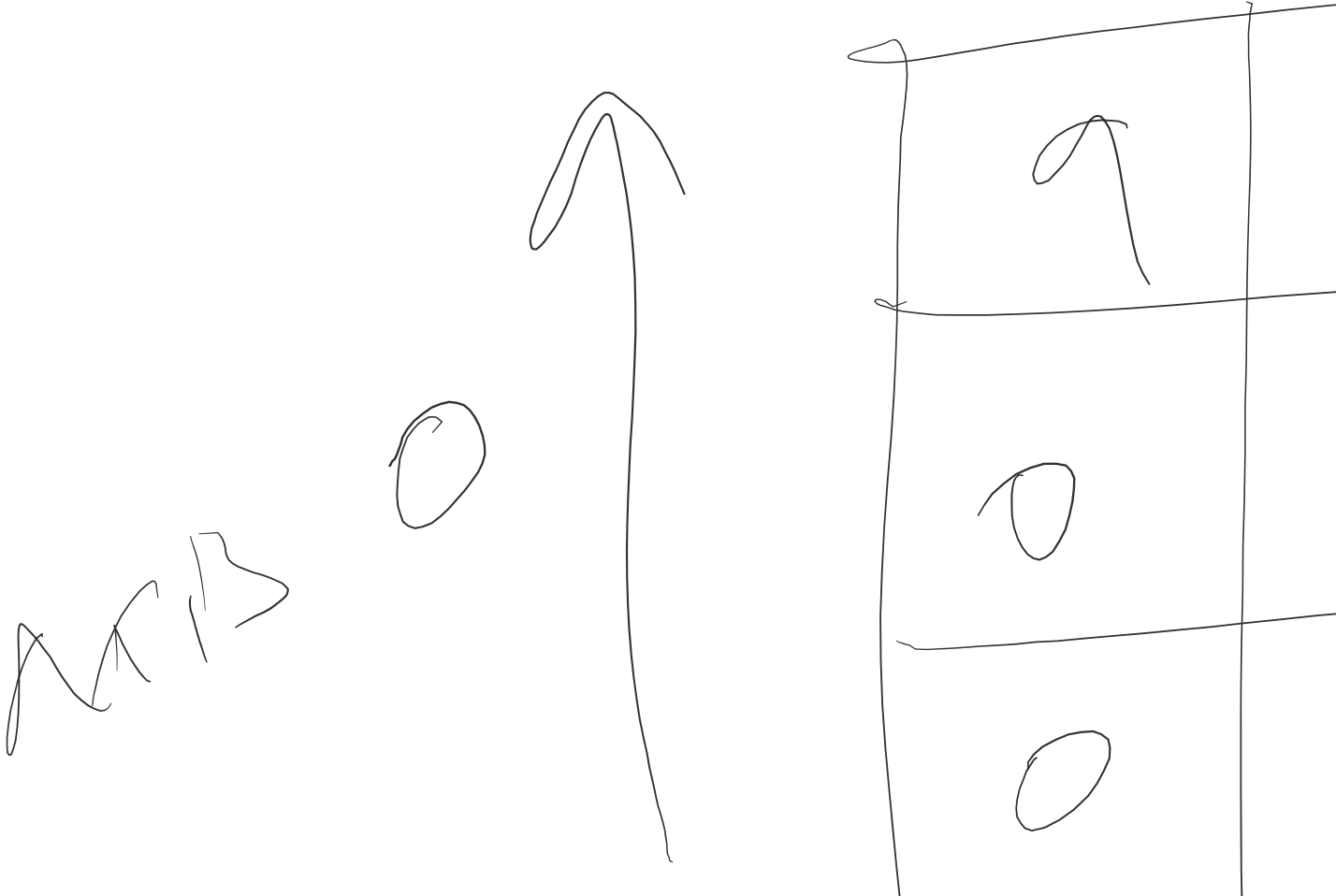
7





# Sortine

np.sort C



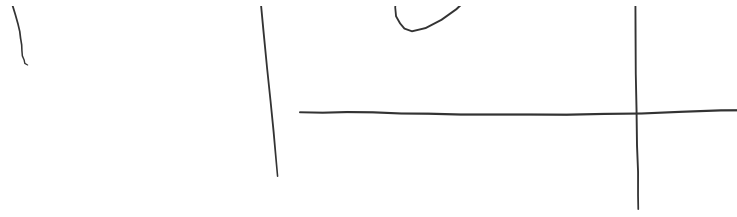
7

main\_drive)

|    |    |
|----|----|
| 10 | 13 |
| 1  | 2  |
| 0  | 1  |







ax

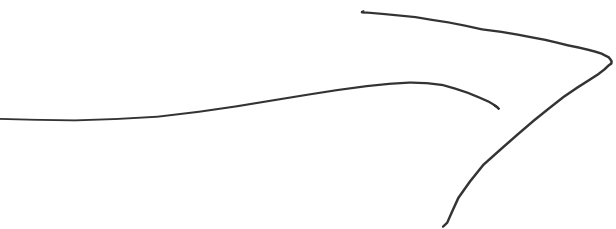
Abdullahi

- Fikheria

- an

u

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|



is 2

g Arrays

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

1





— W I

John

— np, w

Nathan

asks } Fancy

where -

nd exing



- Conr

Reagan

- Sher

th

— concatenating Array

— Rows

— Columns

pe Compatibility  
main

sin NumPy

y in concave Ar  
Hilbert

my Array



- Di  
- Co

Bellamy

- M. Lettine

1  
mention (impe  
eaking compatible  
→ Coordinating

with np. d. e

11/11/11

11/11/11

new

dimensions

lete ( )

ms

Deleting

- Delete

- Delete

Abdullahi

ing

ing

columns

without

array

and Do

an axi





✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓  
- - - - -  
- - - - -

6

7

8

ring Array

3 Fancy

Boo lean m

filtering w/

2D fancy

Indexing

asks

fancy Indexing

Indexing



John

-

Filter

-

an ink

Fancy Index

with m.

non element

ing vs. NP, where

where

technical

e



- np. wk
- tnples
- find.

more different

of indices

ing ? replace

U U | 0 1 ~

ding

