EXPLORATORY DATA ANALYSIS

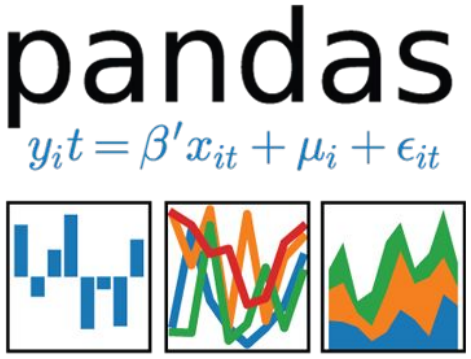
CHRISTELLE SCHARFF

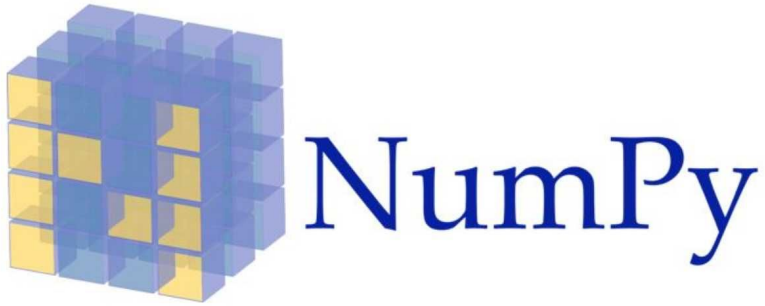
IAN CARVAHALO

KALEEMUNNISA

KRISHNA BATHULA

CONTENTS



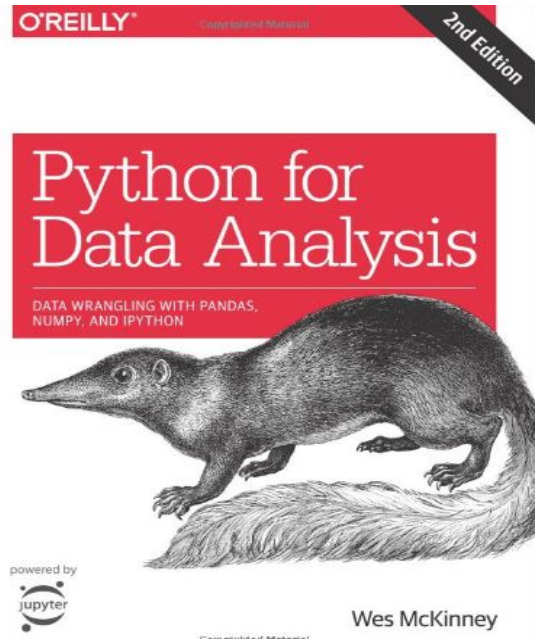


PART 1 - PANDAS

INTRODUCTION

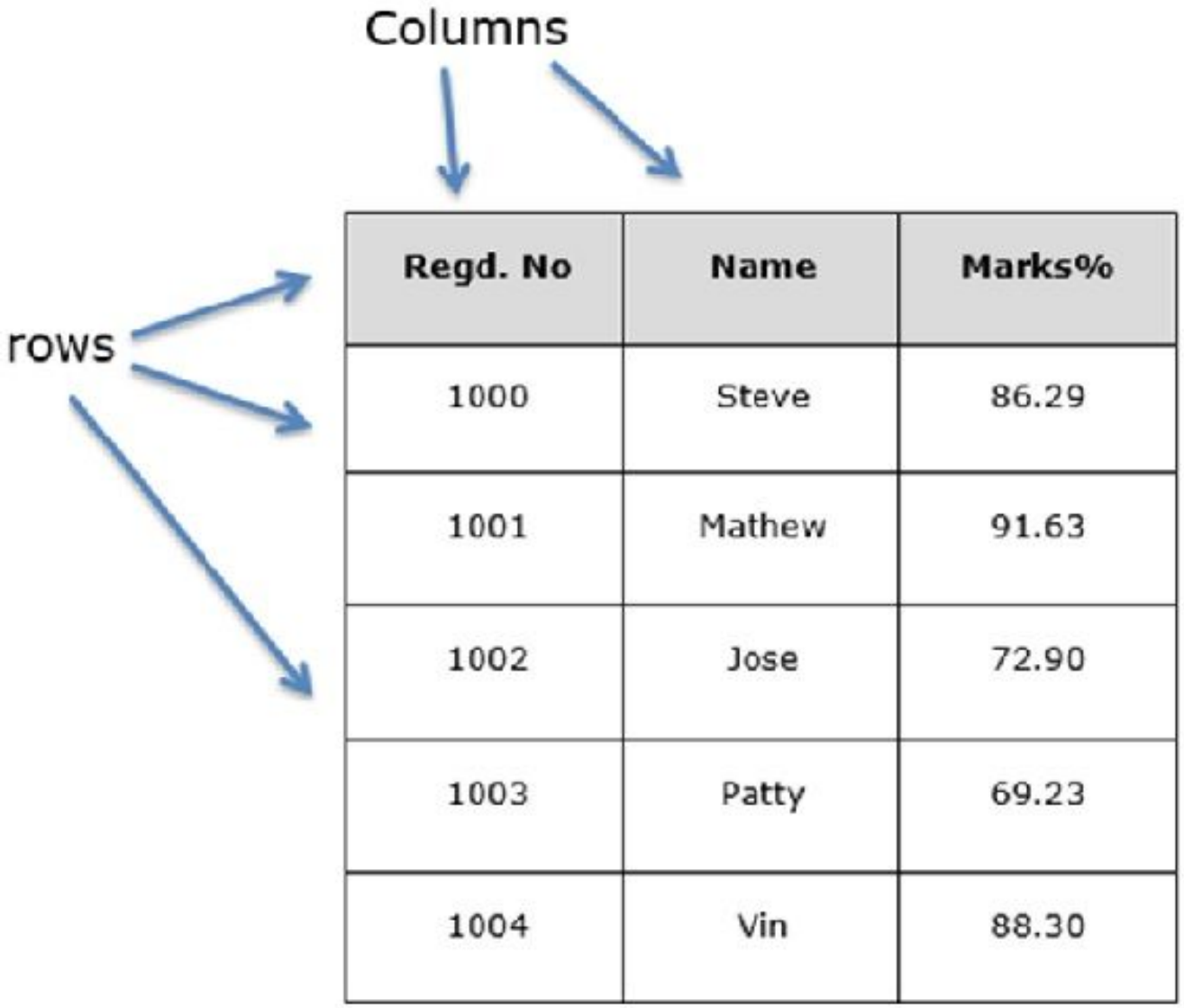
pandas

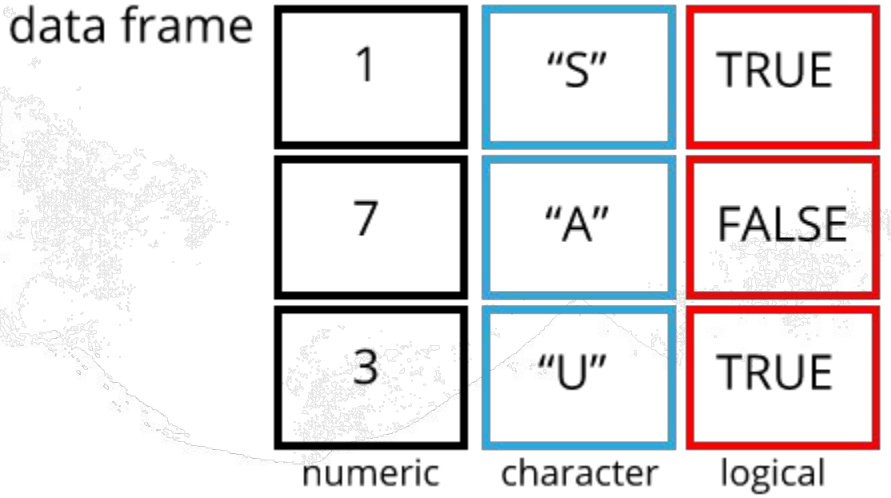
Explore Data




```
import pandas as pd
```

DATA FRAMES


Columns
rows
Regd. No
Name
Marks%
1000
Steve
86.29
1001
Mathew
91.63
1002
Jose
72.90
1003
Patty
69.23
1004
Vin
88.30


data frame
1
"S"
TRUE
7
"A"
FALSE
3
"U"
TRUE
numeric
character
logical

INDEXING IN DF

pandas

```
> dfList[[1]]
```

a	b	c
g	1.2724293	-0.005767173
j	0.4146414	2.404653389
o	-1.53995	0.763593461
x	-0.928567	-0.799009249
f	-0.2947204	-1.147657009

```
> dfList[[2]]
```

a	b	c
k	-0.04493361	0.91897737
a	-0.01619026	0.7821363
j	0.94383621	0.07456498
w	0.8212212	-1.9893517
i	0.59390132	0.61982575

```
> dfList[[3]]
```

a	b	c
m	-1.28459935	-0.6494716
w	0.04672617	0.7267507
l	-0.23570656	1.1519118
g	-0.54288826	0.9921604
b	-0.43331032	-0.4295131

index	a	b	c
1	g	1.2724293	-0.005767173
1	j	0.4146414	2.404653389
1	o	-1.53995	0.763593461
1	x	-0.928567	-0.799009249
1	f	-0.2947204	-1.147657009
2	k	-0.04493361	0.91897737
2	a	-0.01619026	0.7821363
2	j	0.94383621	0.07456498
2	w	0.8212212	-1.9893517
2	i	0.59390132	0.61982575
3	m	-1.28459935	-0.6494716
3	w	0.04672617	0.7267507
3	l	-0.23570656	1.1519118
3	g	-0.54288826	0.9921604
3	b	-0.43331032	-0.4295131

SELECTIONS IN DF

pandas

Selecting Some columns and all rows

	state	color	food	age	height	score
Jane	NY	blue	Steak	30	165	4.6
Niko	TX	green	Lamb	2	70	8.3
Aaron	FL	red	Mango	12	120	9
Penelope	AL	white	Apple	4	80	3.3
Dean	AK	gray	Cheese	32	180	1.8
Christina	TX	black	Melon	33	172	9.5
Cornelia	TX	red	Beans	69	150	2.2

	color	age	height
Jane	blue	30	165
Niko	green	2	70
Aaron	red	12	120
Penelope	white	4	80
Dean	gray	32	180
Christina	black	33	172
Cornelia	red	69	150

Column Selection

SELECTIONS IN DF

pandas

Selecting Some rows and all columns.

	state	color	food	age	height	score
Jane	NY	blue	Steak	30	165	4.6
Niko	TX	green	Lamb	2	70	8.3
Aaron	FL	red	Mango	12	120	9
Penelope	AL	white	Apple	4	80	3.3
Dean	AK	gray	Cheese	32	180	1.8
Christina	TX	black	Melon	33	172	9.5
Cornelia	TX	red	Beans	69	150	2.2

	state	color	food	age	height	score
Aaron	FL	red	Mango	12	120	9.0
Dean	AK	gray	Cheese	32	180	1.8

Row Selection

SELECTIONS IN DF

pandas

Selecting Some rows and some columns.

	state	color	food	age	height	score
Jane	NY	blue	Steak	30	165	4.6
Niko	TX	green	Lamb	2	70	8.3
Aaron	FL	red	Mango	12	120	9
Penelope	AL	white	Apple	4	80	3.3
Dean	AK	gray	Cheese	32	180	1.8
Christina	TX	black	Melon	33	172	9.5
Cornelia	TX	red	Beans	69	150	2.2

	color	age	height
Aaron	red	12	120
Dean	gray	32	180

Slicing and Dicing

SELECTIONS IN DF

pandas

Selecting Some rows and some columns.

	state	color	food	age	height	score
Jane	NY	blue	Steak	30	165	4.6
Niko	TX	green	Lamb	2	70	8.3
Aaron	FL	red	Mango	12	120	9
Penelope	AL	white	Apple	4	80	3.3
Dean	AK	gray	Cheese	32	180	1.8
Christina	TX	black	Melon	33	172	9.5
Cornelia	TX	red	Beans	69	150	2.2

	color	age	height
Aaron	red	12	120
Dean	gray	32	180

Slicing and Dicing

SORTING ON DF

pandas

Sorting by score on the DF.

	Age	Name	Score
0	26	Alisa	89
1	27	Bobby	87
2	25	Cathrine	67
3	24	Madonna	55
4	31	Rocky	47
5	27	Sebastian	72
6	25	Jaqluine	76
7	33	Rahul	79
8	42	David	44
9	32	Andrew	92
10	51	Ajay	99
11	47	Teresa	69

Sorted in ascending order of the score

	Age	Name	Score
8	42	David	44
4	31	Rocky	47
3	24	Madonna	55
2	25	Cathrine	67
11	47	Teresa	69
5	27	Sebastian	72
6	25	Jaqluine	76
7	33	Rahul	79
1	27	Bobby	87
0	26	Alisa	89
9	32	Andrew	92
10	51	Ajay	99

ADDING DF

pandas

Merging two DF's

DF - 1

Name	City	Country
Lenna	San Francisco	US
Malcom	New York	US
Akiko	Tokyo	Japan

+

DF - 2

Name	City	Country
Lenna	San Francisco	US
Thomas	London	UK
Diane	Chicago	US

→

Name	City	Country
Lenna	San Francisco	US
Malcom	New York	US
Akiko	Tokyo	Japan
Lenna	San Francisco	US
Thomas	London	UK
Diane	Chicago	US

Resulting DF

AGGREGATIONS ON DF

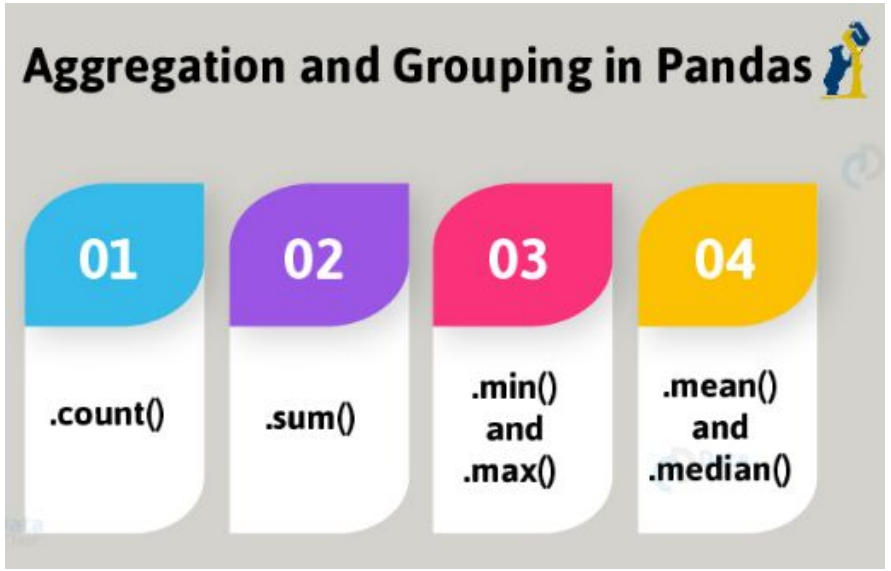
pandas

animal	water_need
zebra	10
zebra	15
lion	100
elephant	320
zebra	20
lion	120
lion	140
zebra	15

zebra → mean: 15

lion → mean: 120

elephant → mean: 320



PIVOT TABLE IN DF

pandas

Pivot

```
df.pivot(index='foo',
         columns='bar',
         values='baz')
```

df

	foo	bar	baz	zoo
0	one	A	1	x
1	one	B	2	y
2	one	C	3	z
3	two	A	4	q
4	two	B	5	w
5	two	C	6	t

bar	A	B	C
foo			
one	1	2	3
two	4	5	6

CORRELATION

pandas

Correlation between the columns in the DF

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0
5	Amir Johnson	Boston Celtics	90.0	PF	29.0	6-9	240.0	NaN	12000000.0
6	Jordan Mickey	Boston Celtics	55.0	PF	21.0	6-8	235.0	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41.0	C	25.0	7-0	238.0	Gonzaga	2165160.0
8	Terry Rozier	Boston Celtics	12.0	PG	22.0	6-2	190.0	Louisville	1824360.0
9	Marcus Smart	Boston Celtics	36.0	PG	22.0	6-4	220.0	Oklahoma State	3431040.0

Output :

	Number	Age	Weight	Salary
Number	1.000000	0.028724	0.206921	-0.112386
Age	0.028724	1.000000	0.087183	0.213459
Weight	0.206921	0.087183	1.000000	0.138321
Salary	-0.112386	0.213459	0.138321	1.000000

PANDAS (EDA PART-I) EXERCISE

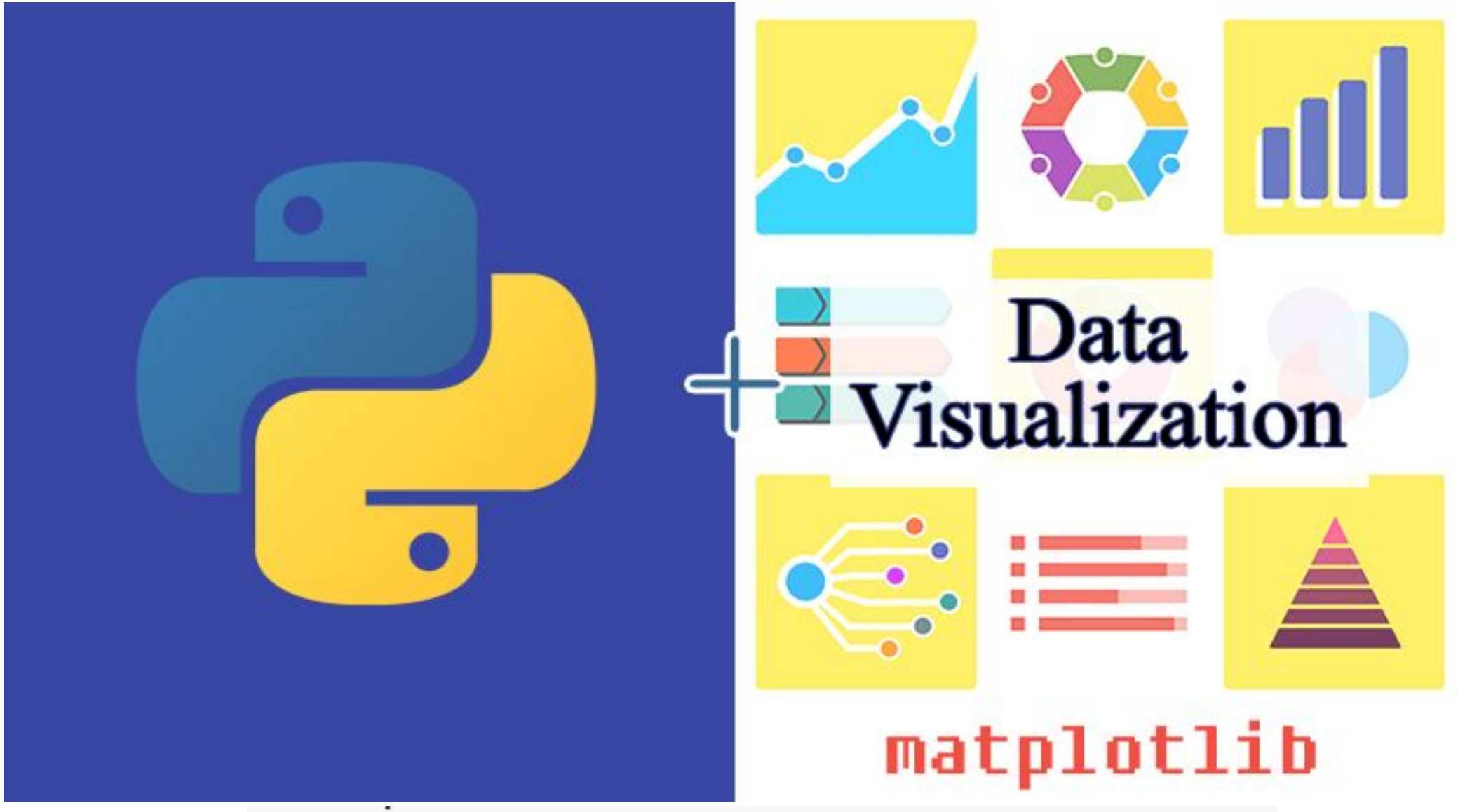
The Exercises using **Pandas Library** is showns in the **Part-I** of the **Exploratory Data Analysis** Python Notebook.

The relevant Data has to be loaded.

END OF SECTION

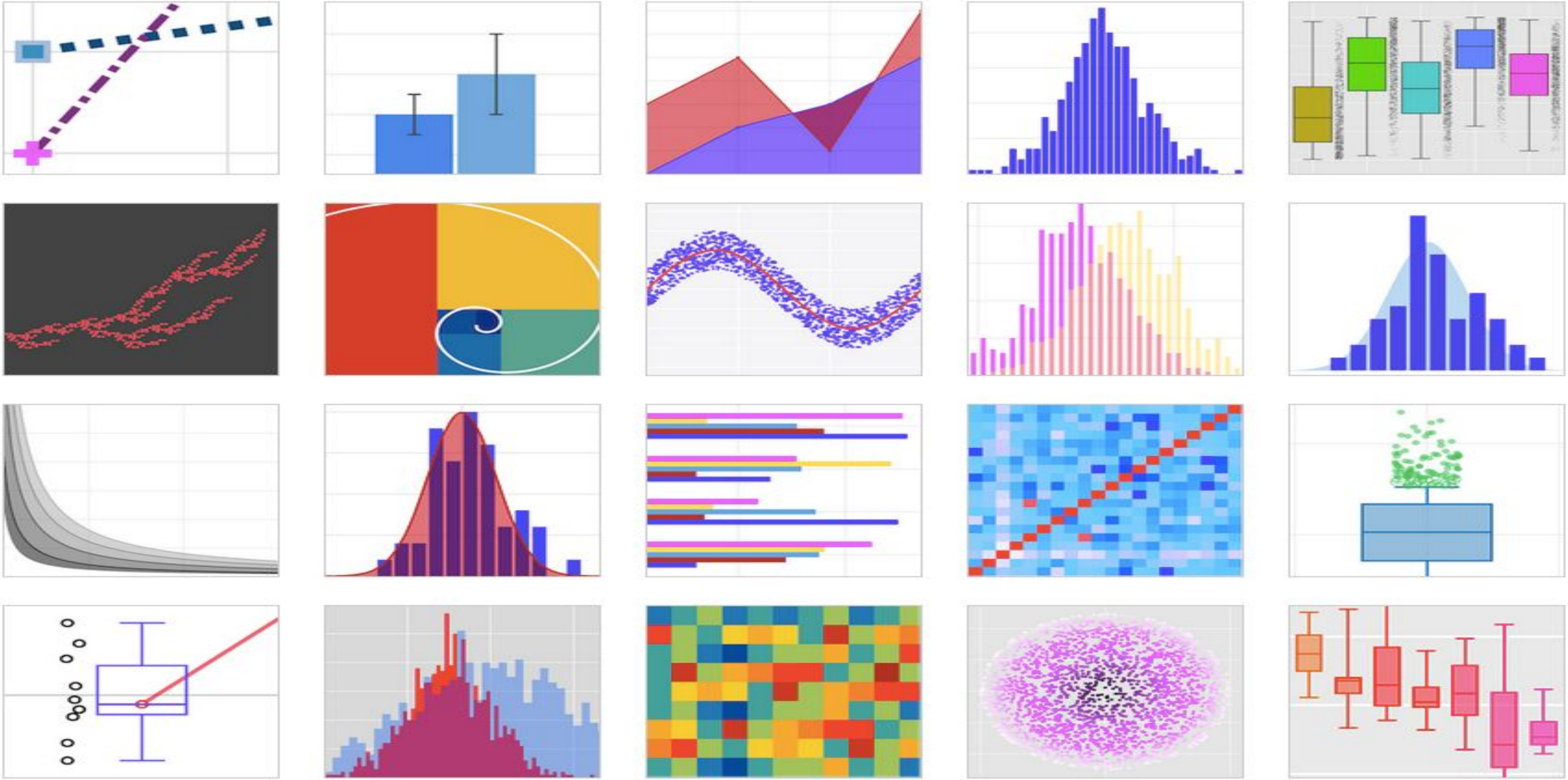
PART 2 - MATPLOTLIB

INTRODUCTION

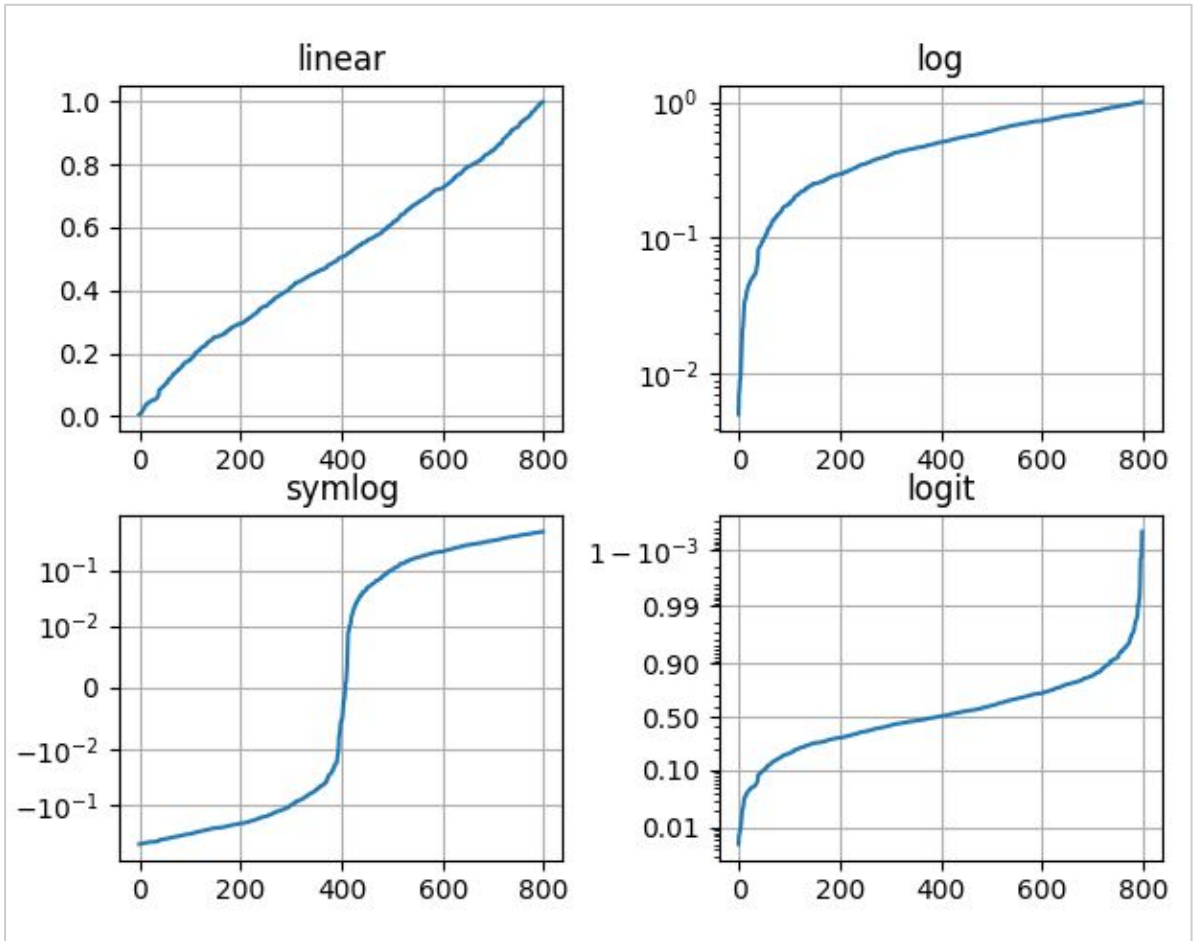



```
import matplotlib.pyplot as plt
```

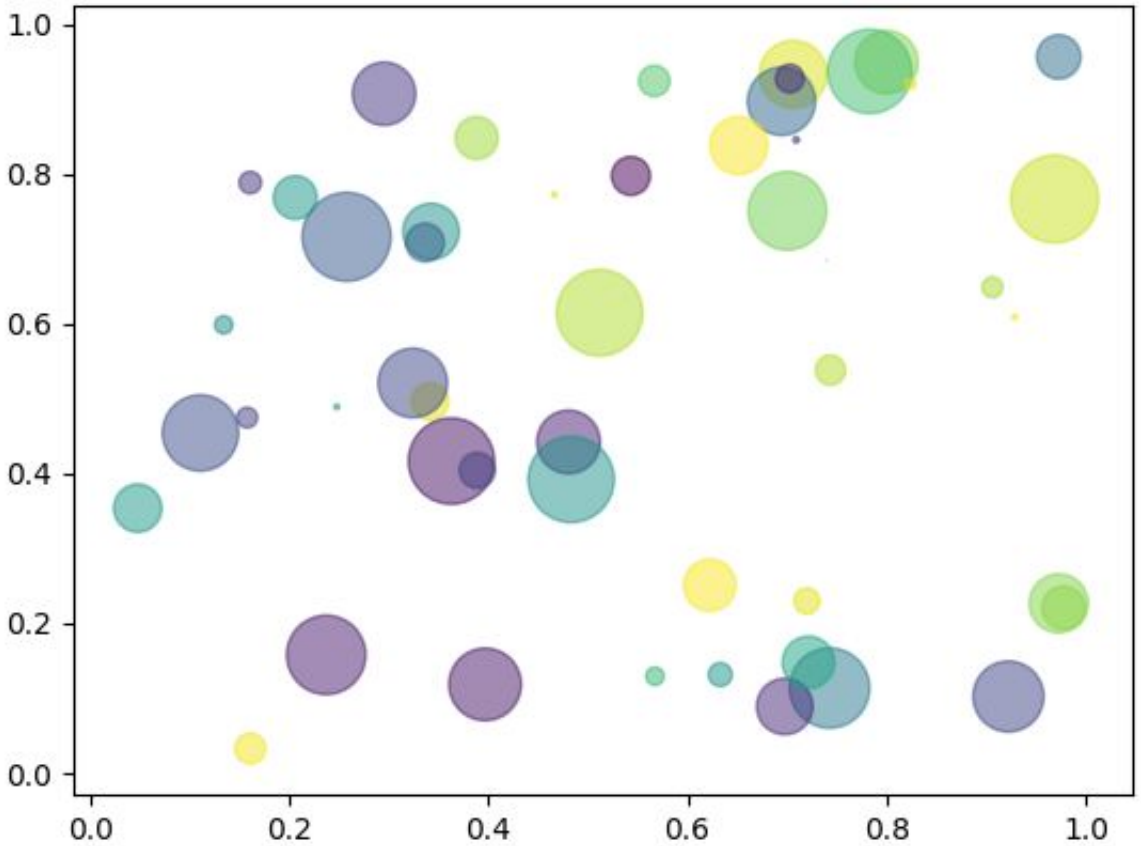
PLOTTING DATA



PLOTS

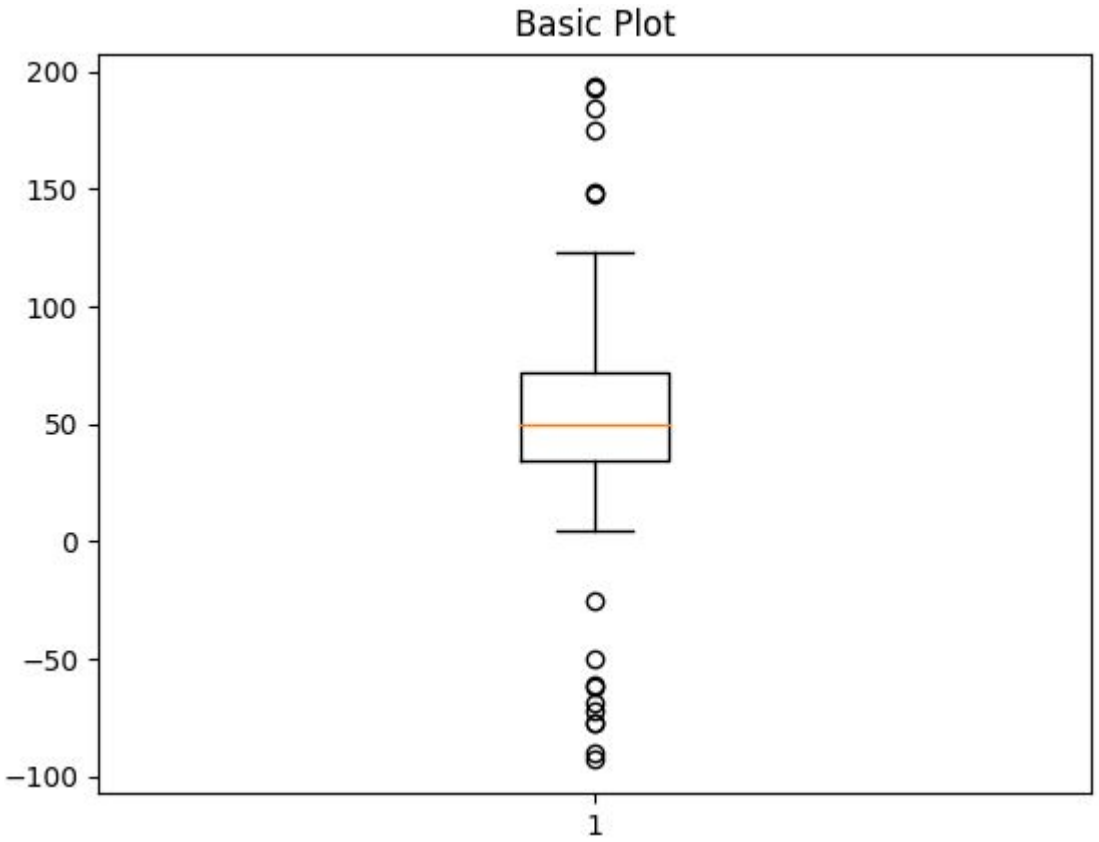


Line Plot

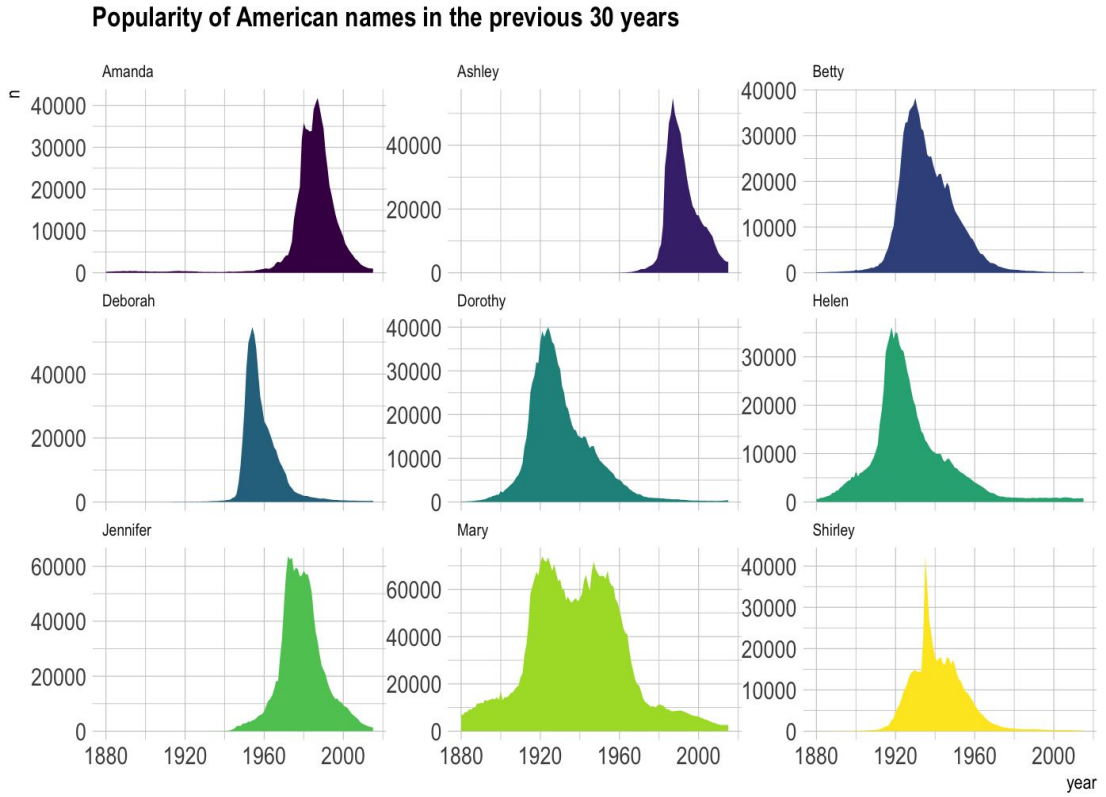


Scatter Plot

PLOTS

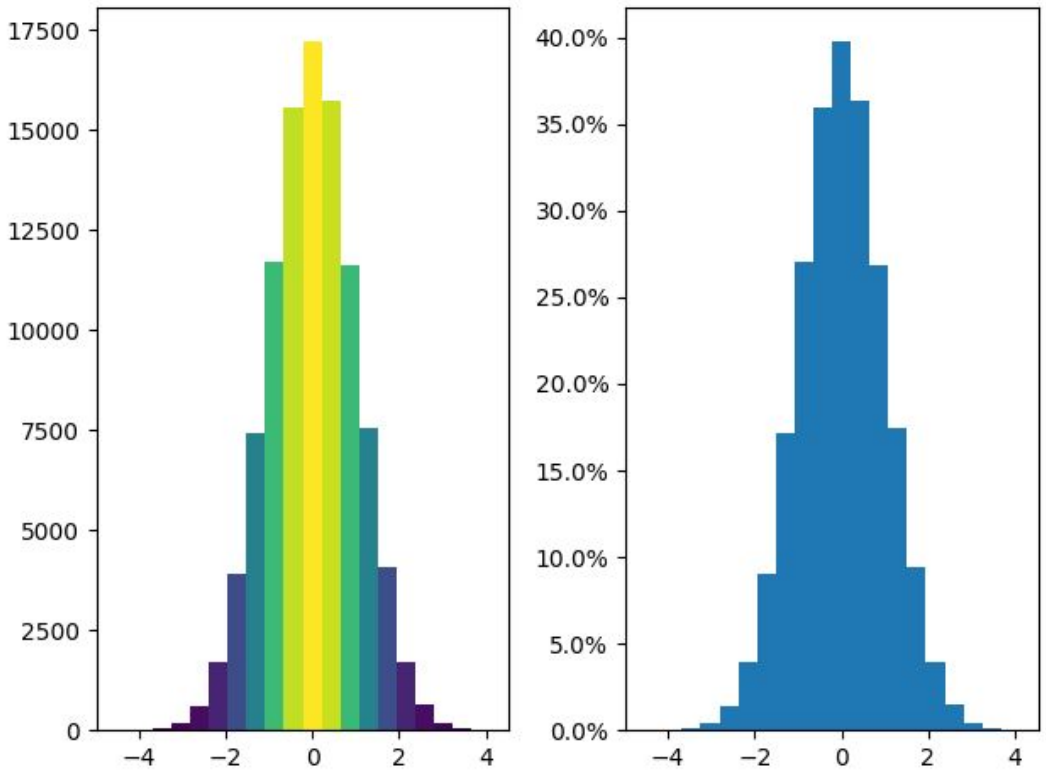


Line Plot

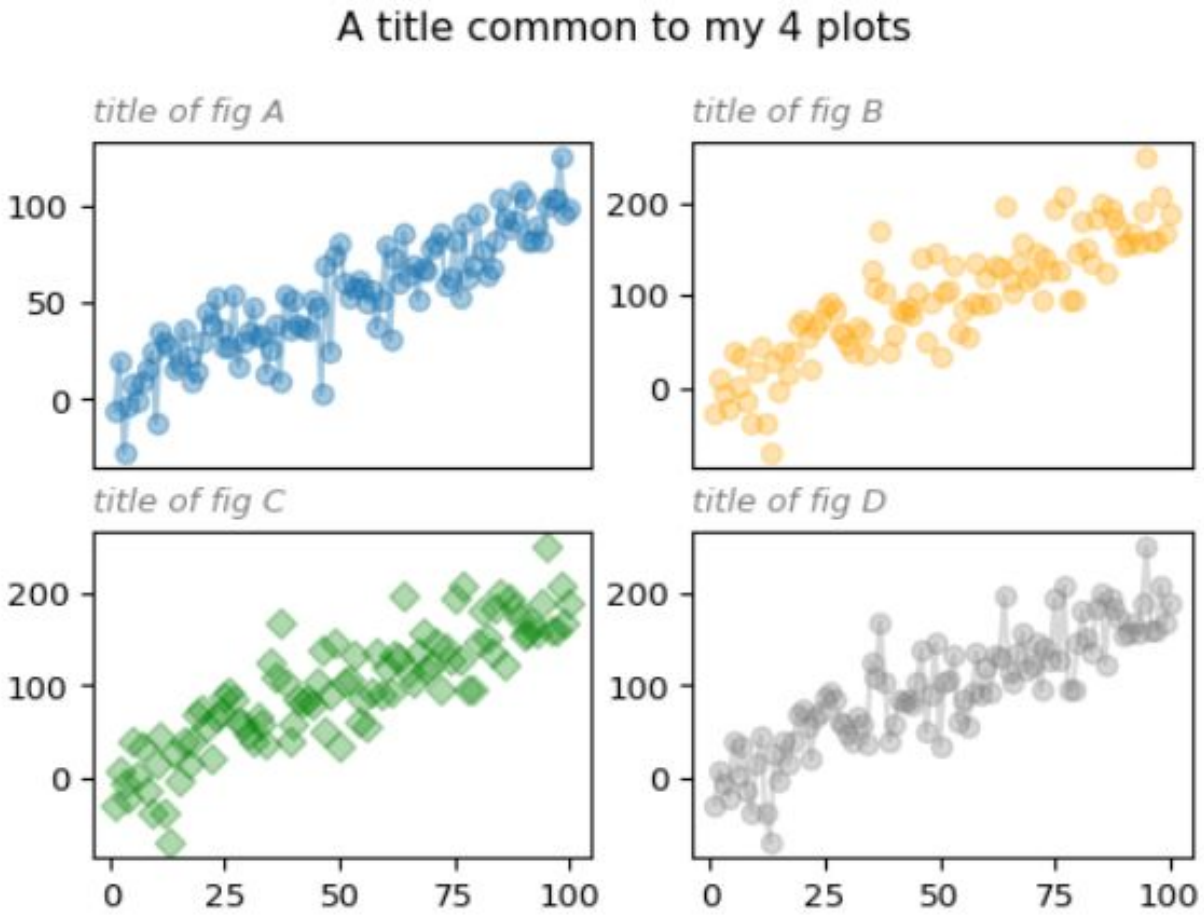


Area Plot

PLOTS

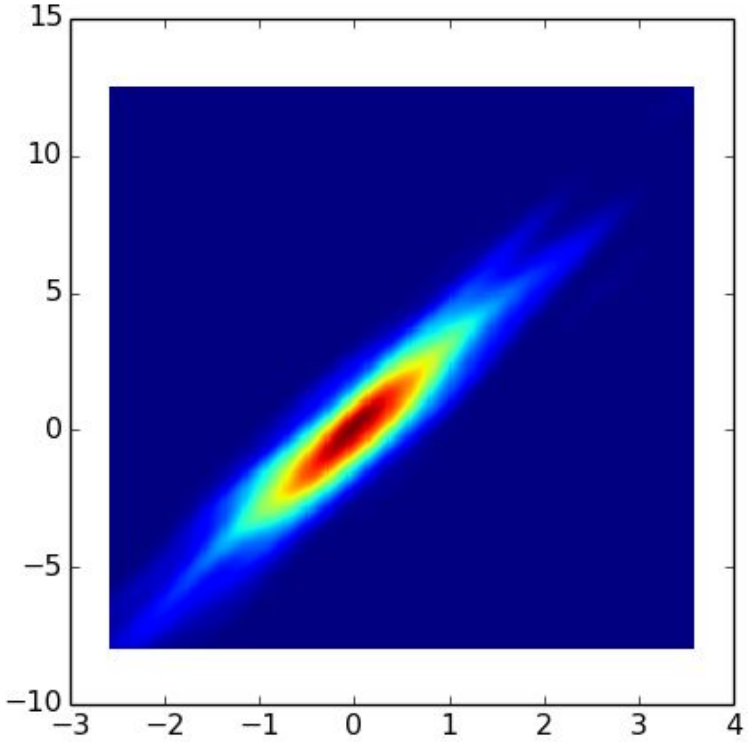


Histogram

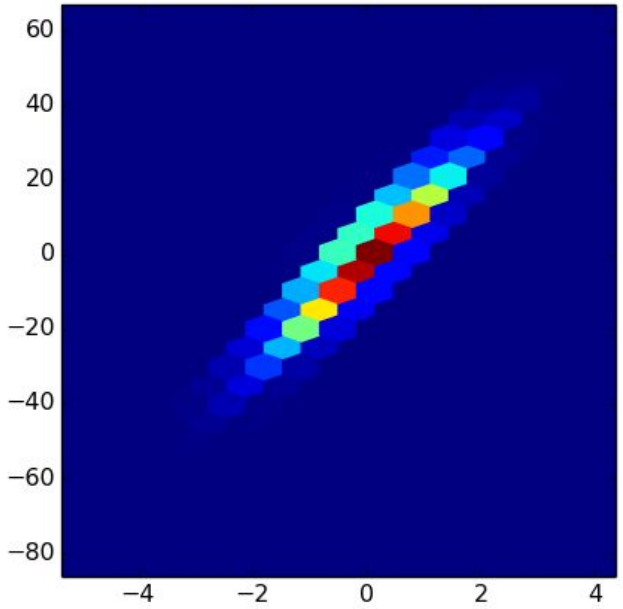


Sub Plot

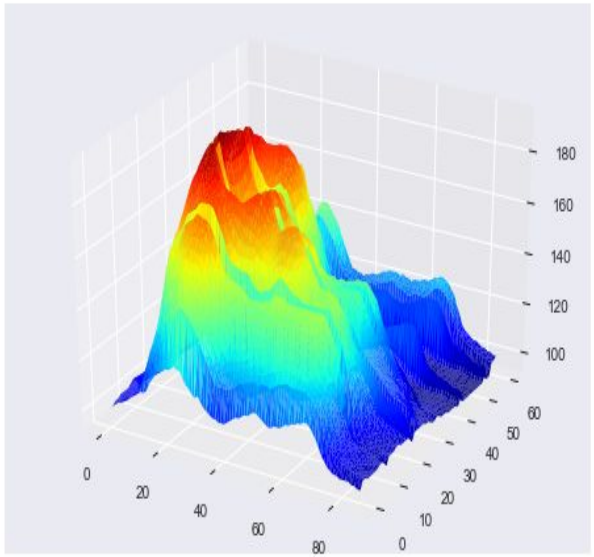
PLOTS



Density Plot

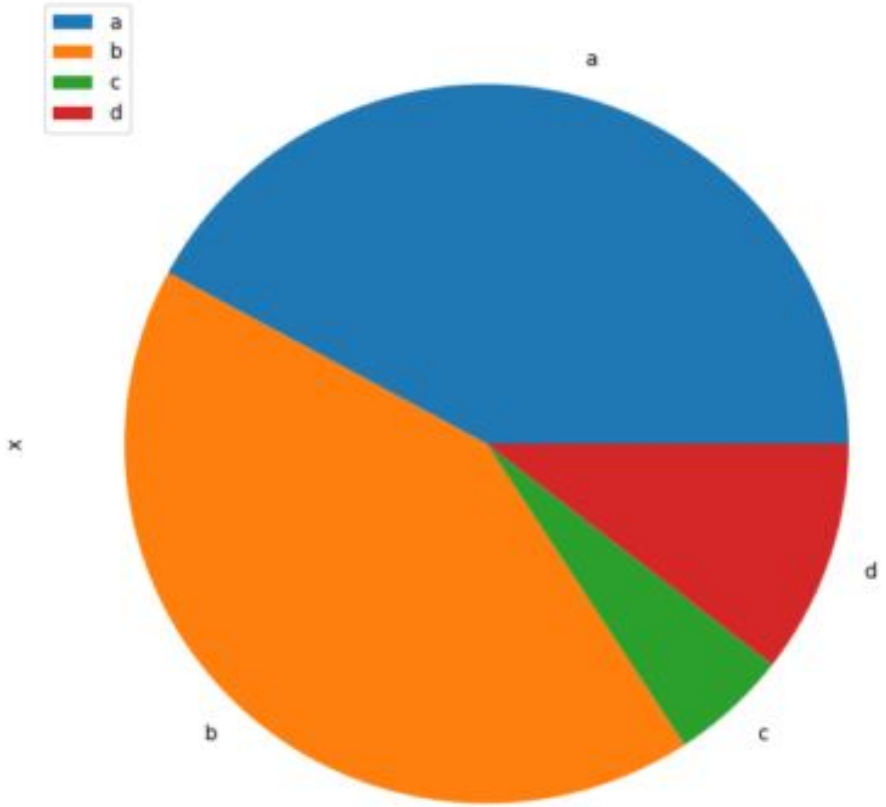


Hexbin Plot

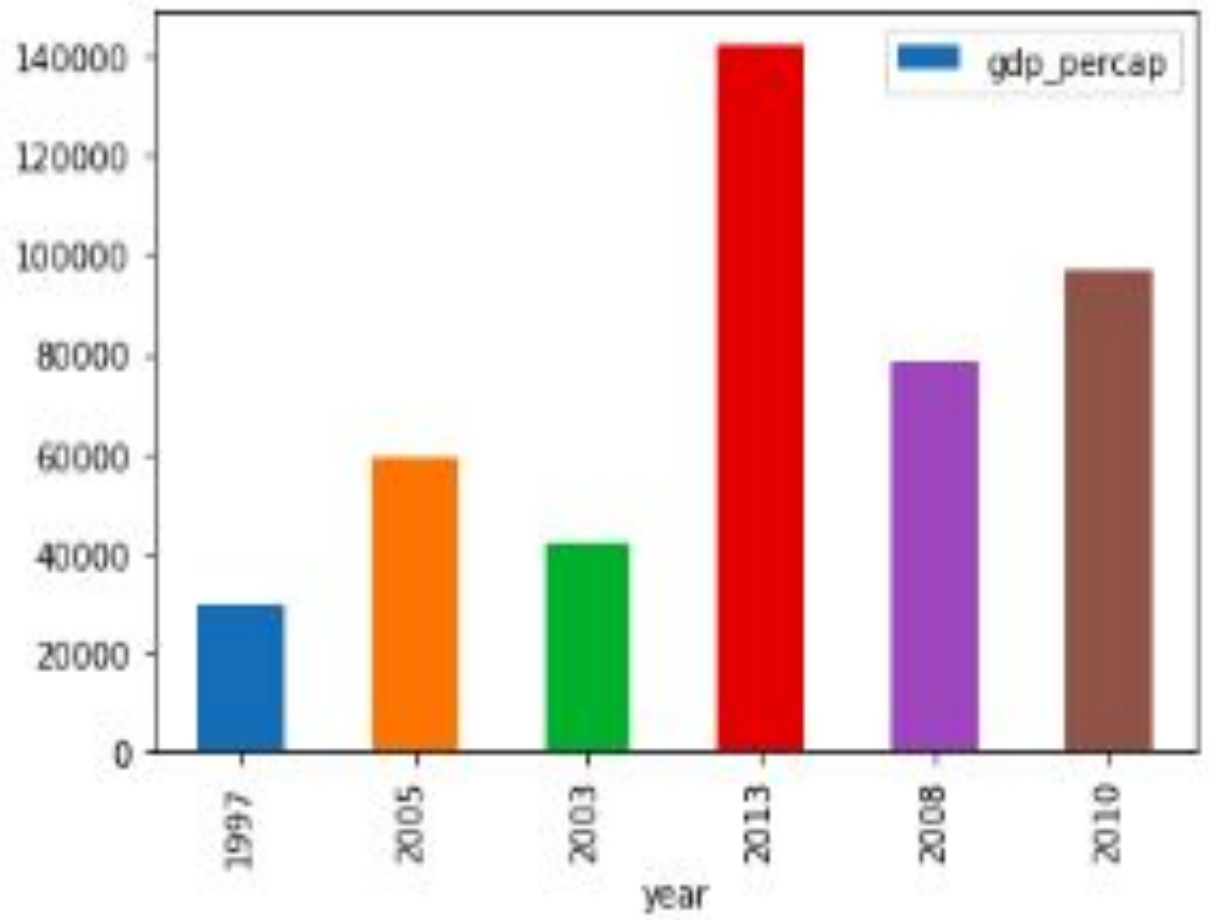


Surface Plot

CHARTS



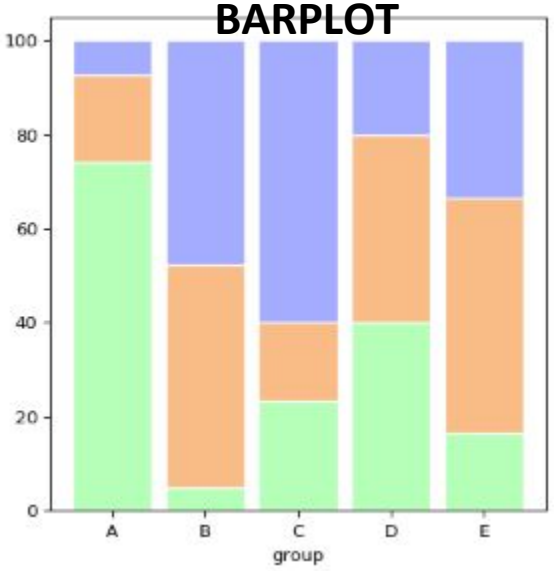
Pie Chart



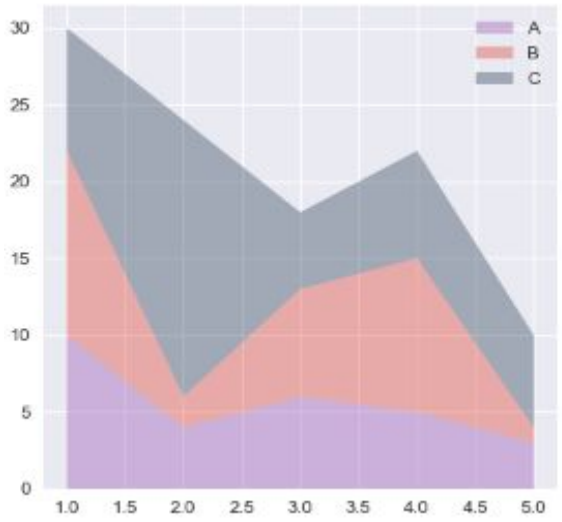
Column Chart

MORE VISUALIZATIONS

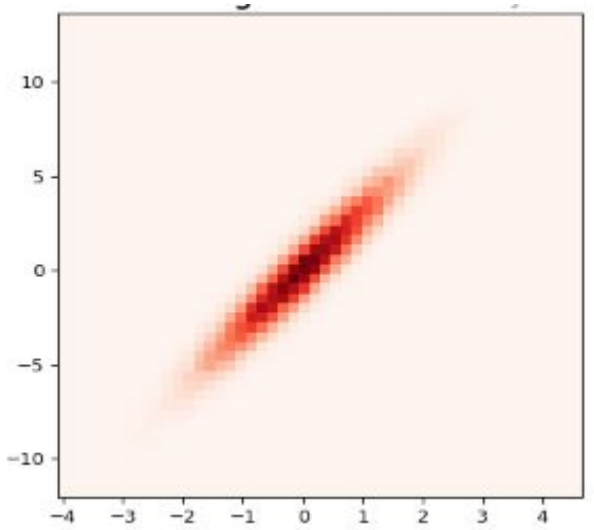
PERCENTAGE STACKED BARPLOT



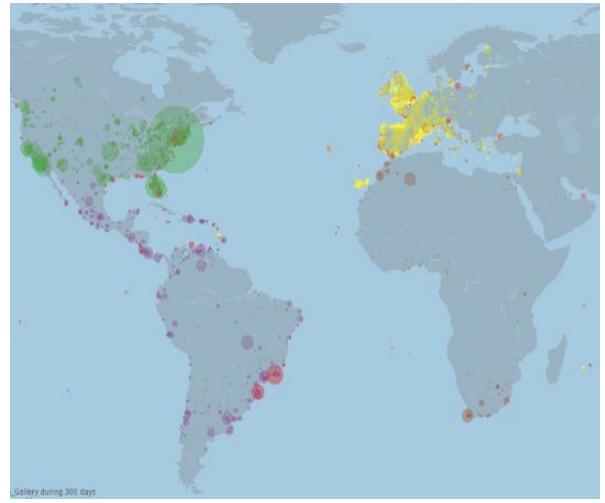
STACKED AREA PLOT



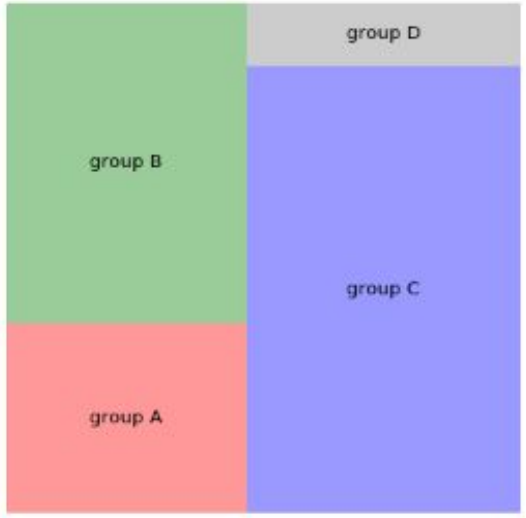
2D HISTOGRAM



MAP

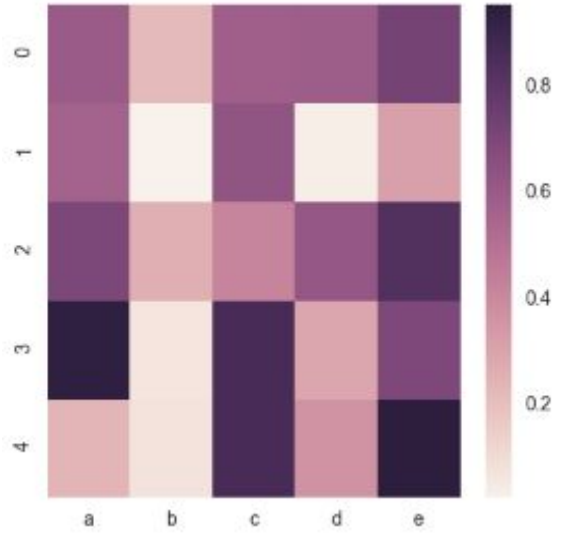


TREE MAP

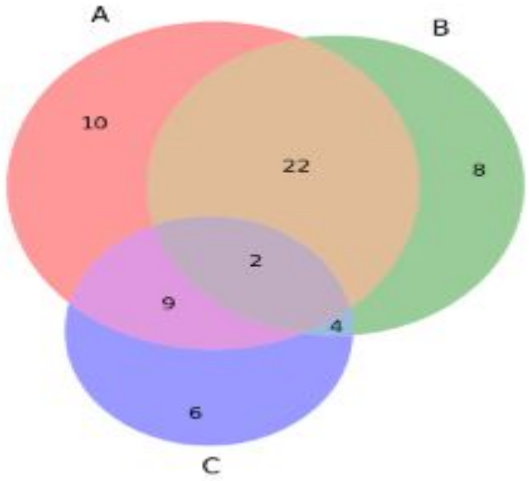


WORDCLOUD

HEATMAP



VENN DIAGRAM



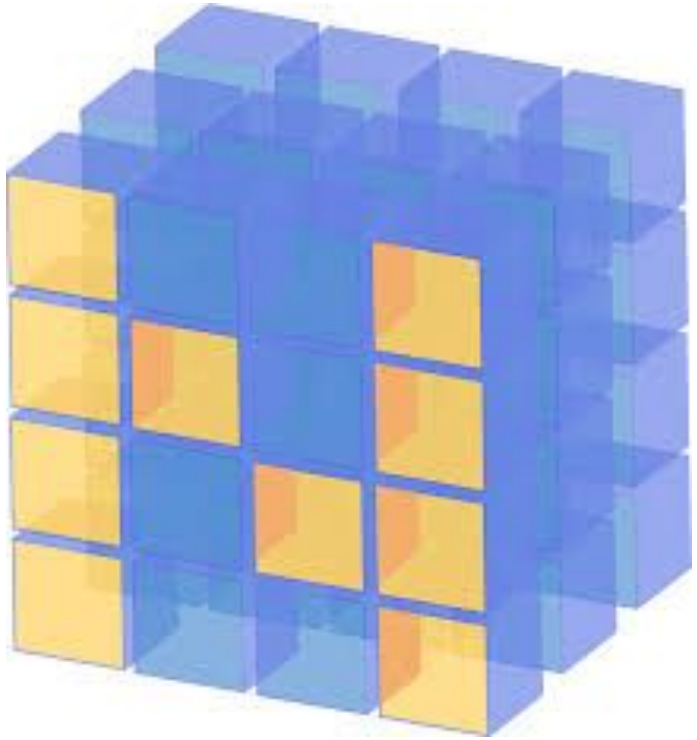
MATPLOTLIB (EDA PART-II) EXERCISE

The Exercises using Matplot library is showns in the **Part-II** of the **Exploratory Data Analysis** Python Notebook.

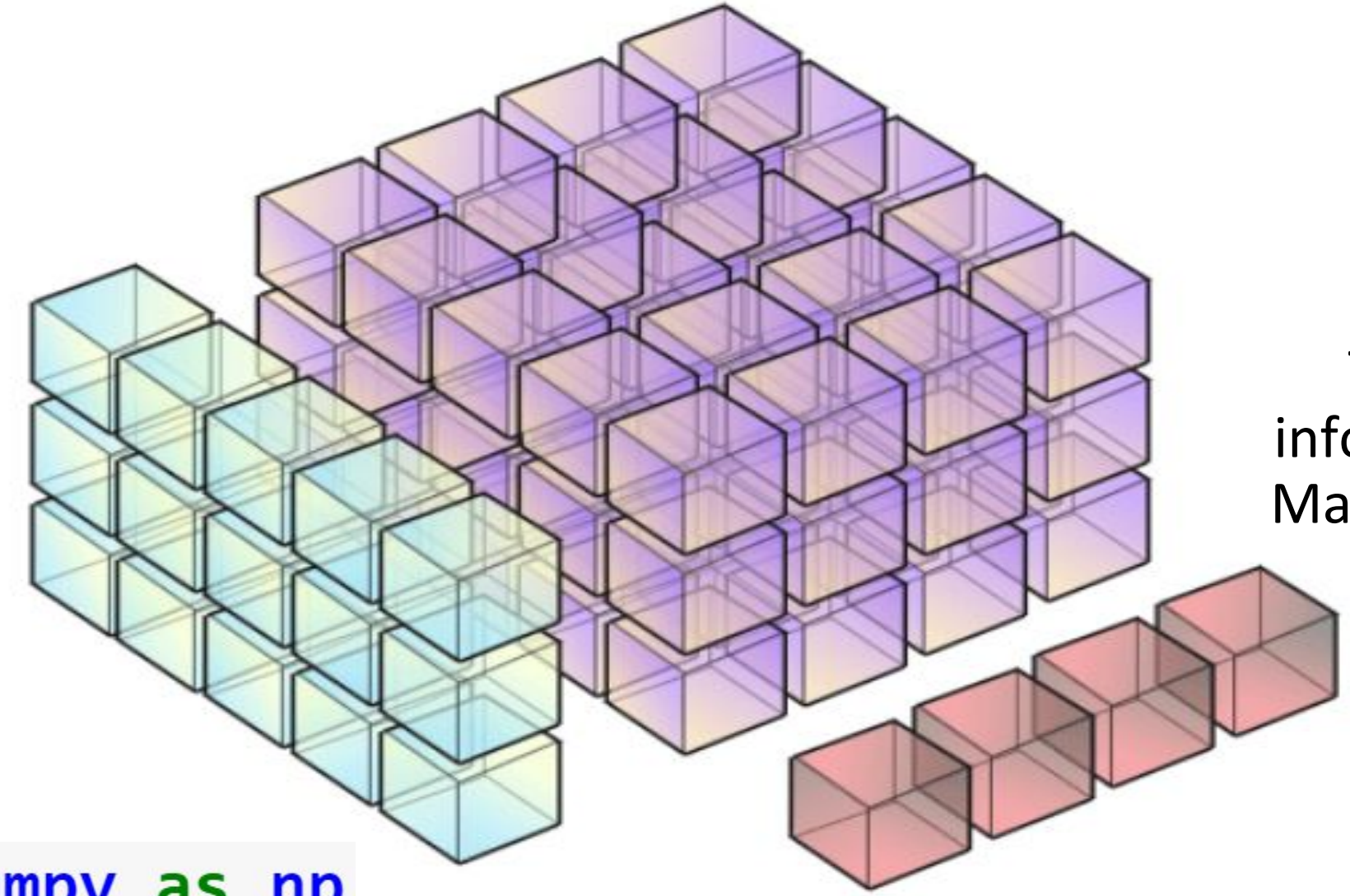
The various **Matplot Library** methods are used to depict the different **Data Visualizations** in the Notebook.

END OF SECTION

PART 3 - NUMPY



INTRODUCTION



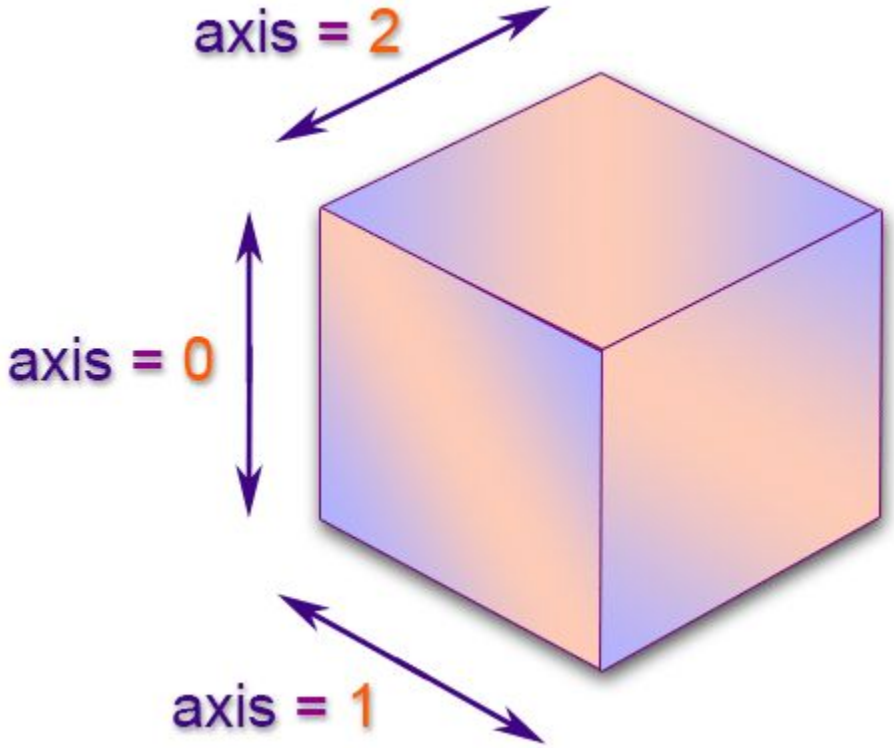
The image information as a Matrix and Cube

```
import numpy as np
```

NumPy arrays

ARRAY TYPES

NumPy



1D Array

2	5	6	9

axis 0

shape : (4,)

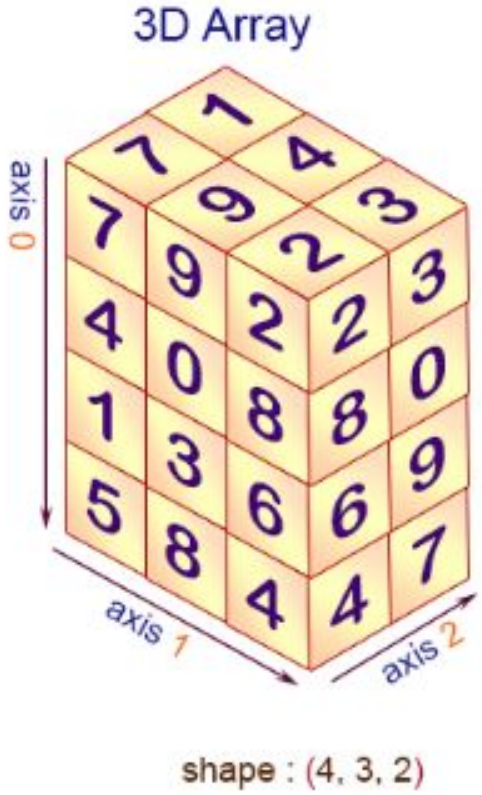
2D Array

axis 0

3.5	4.0	6.5
0.4	0.9	4.7

axis 1

shape : (2, 3)



shape : (4, 3, 2)

N (MULTI) DIMENSIONAL ARRAY

NumPy

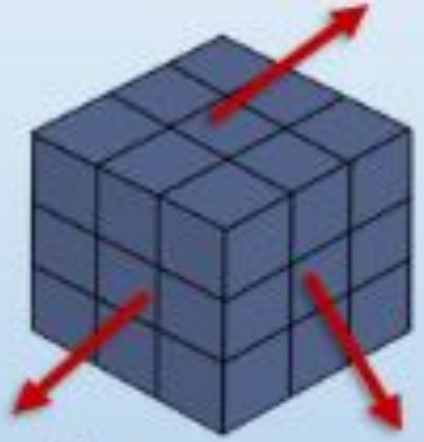
NumPy Ndarray

10	15	13	8	25

1D-Array

	Column 0	Column 1	Column 2
Row 0	X[0][0]	X[0][1]	X[0][2]
Row 1	X[1][0]	X[1][1]	X[1][2]
Row 2	X[2][0]	X[2][1]	X[2][2]

2D-Array



3D-Array

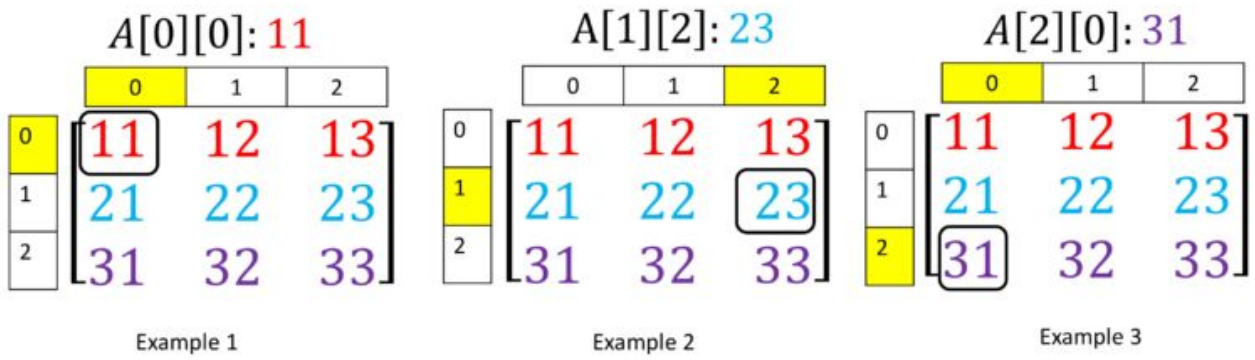
ARRAY REPRESENTATION

NumPy

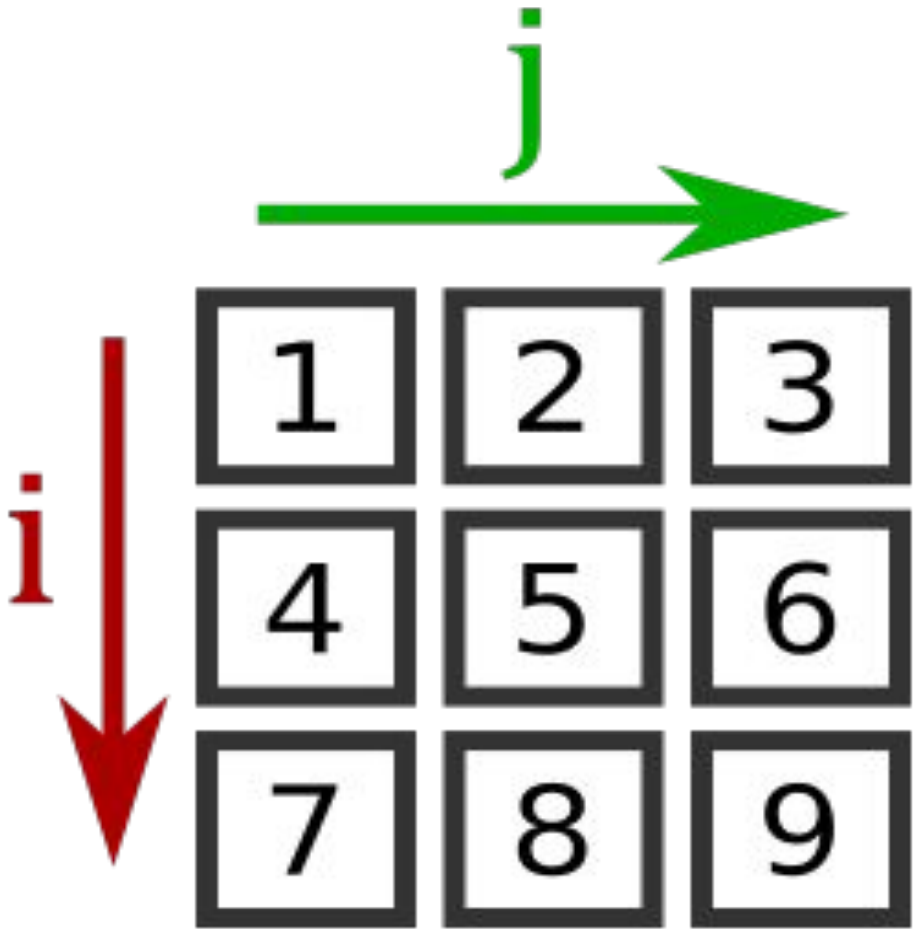
$$A = [[A[0][0], A[0][1], A[0][2]], [A[1][0], A[1][1], A[1][2]][A[2][0], A[2][1], A[2][2]]]$$

$$\begin{bmatrix} A[0][0] & A[0][1] & A[0][2] \\ A[1][0] & A[1][1] & A[1][2] \\ A[2][0] & A[2][1] & A[2][2] \end{bmatrix}$$

$$A = [[11, 12, 13], [21, 22, 23], [31, 32, 33]]$$

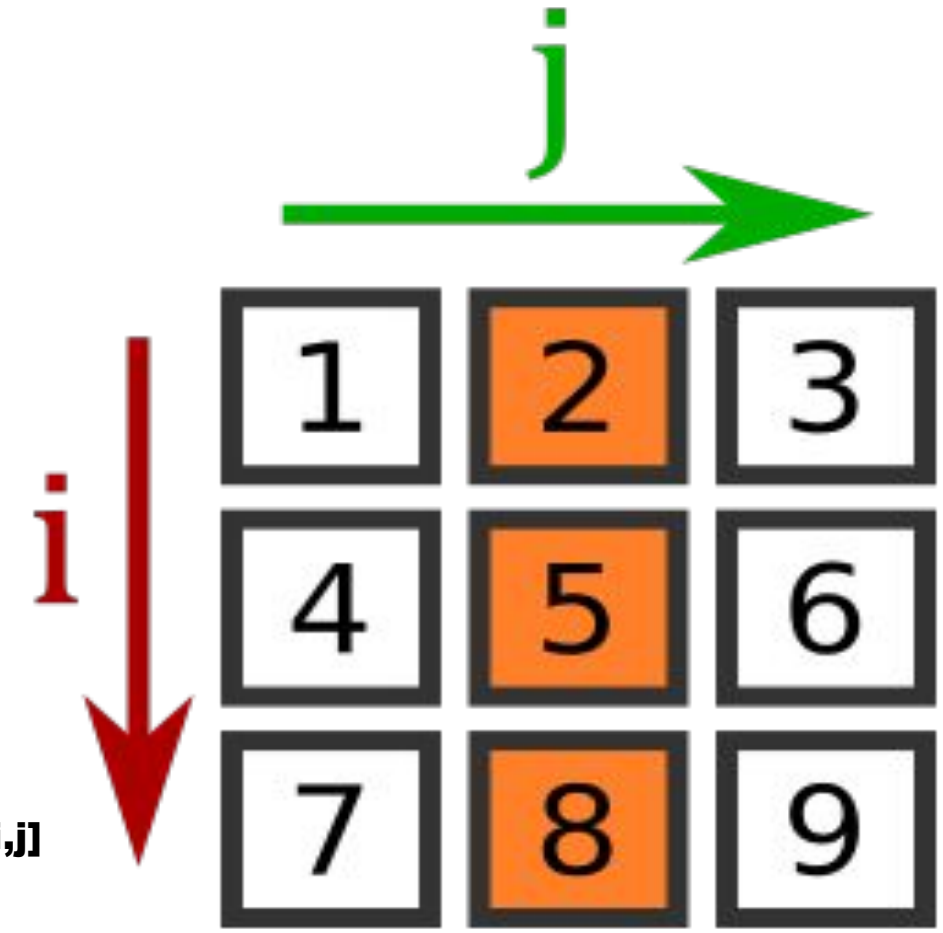


INDEXING IN 2D NUMPY ARRAYS



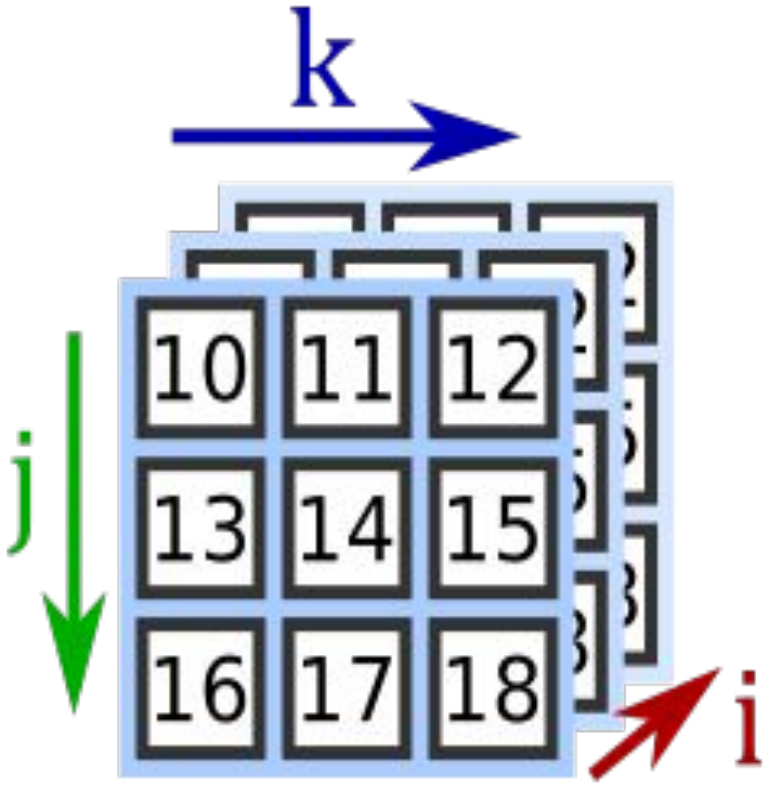
‘i’ selects the row, and ‘j’ selects the column

Arrary[i,j]

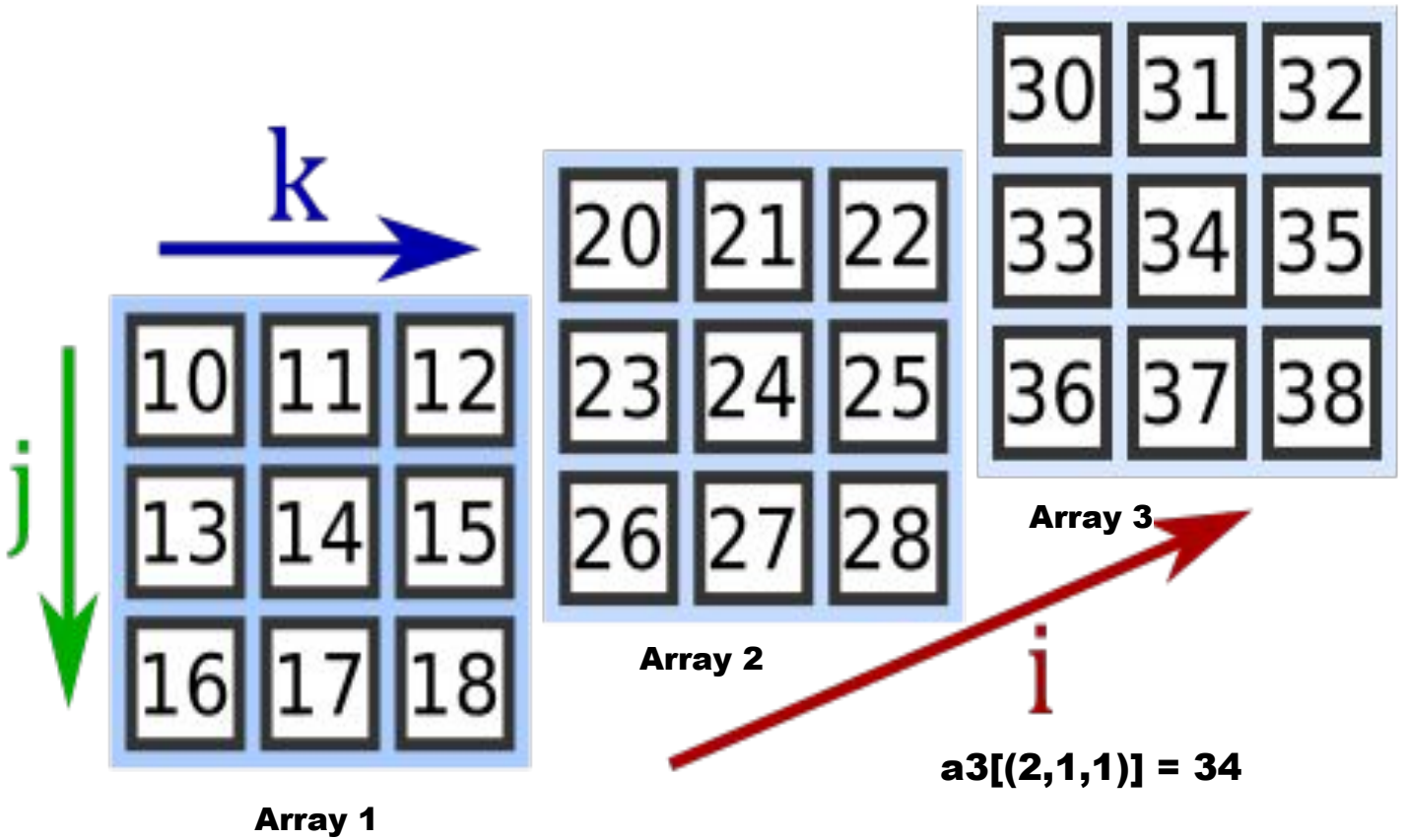


Elements selected from 3 rows(index - 0,1,2) and 1st column(index - 1).

INDEXING IN 3D NUMPY ARRAYS



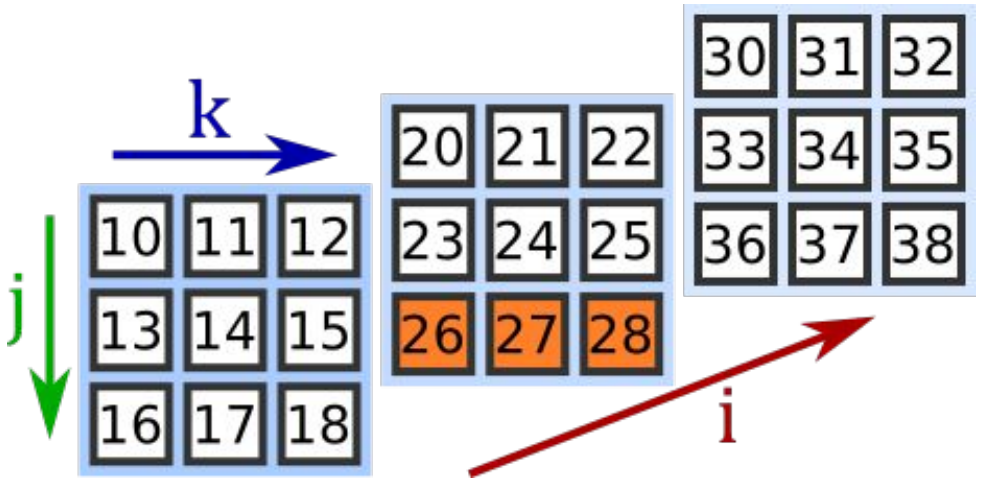
‘i’ - array(matrix),
‘j’ - row and
‘k’ - column




```
a3 = np.array([[[10, 11, 12], [13, 14, 15], [16, 17, 18]],
               [[20, 21, 22], [23, 24, 25], [26, 27, 28]],
               [[30, 31, 32], [33, 34, 35], [36, 37, 38]]])
```

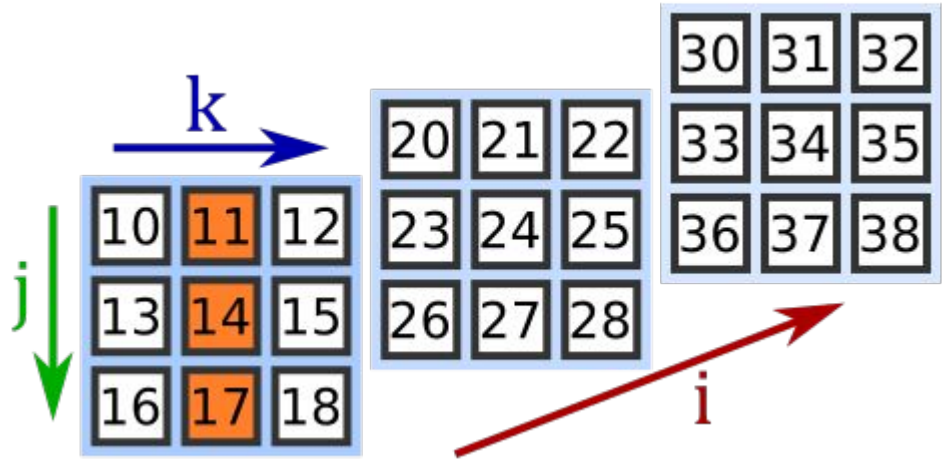
SELECTING ROW OR COLUMN IN 3D NUMPY ARRAY

```
a3 = np.array([[[10, 11, 12], [13, 14, 15], [16, 17, 18]],
               [[20, 21, 22], [23, 24, 25], [26, 27, 28]],
               [[30, 31, 32], [33, 34, 35], [36, 37, 38]]])
```

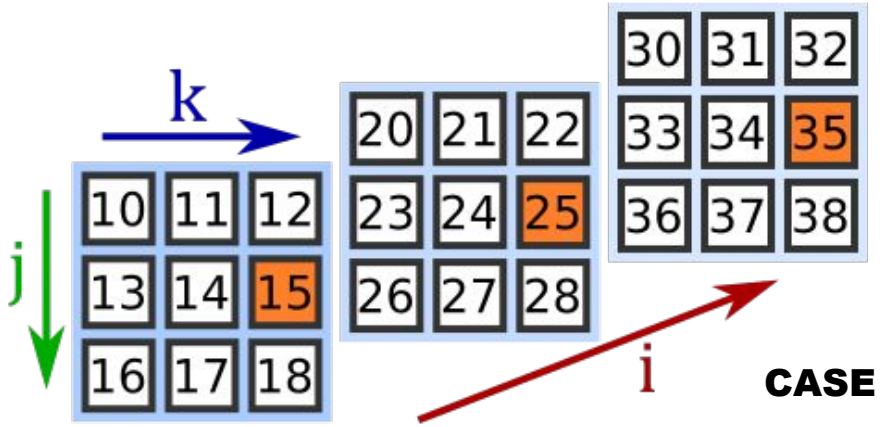


CASE 1: a3[1,2,:] = [26 27 28]

: -> Selects all



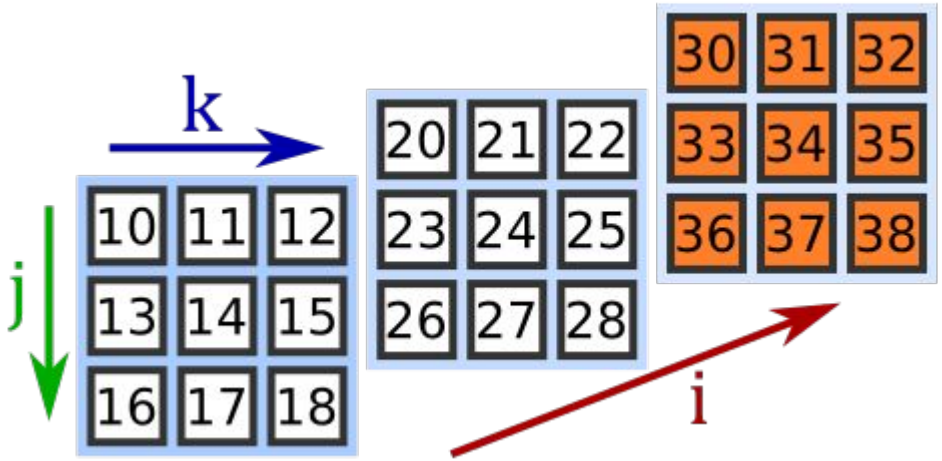
CASE 2: a3[0,:,1] = [11 14 17]



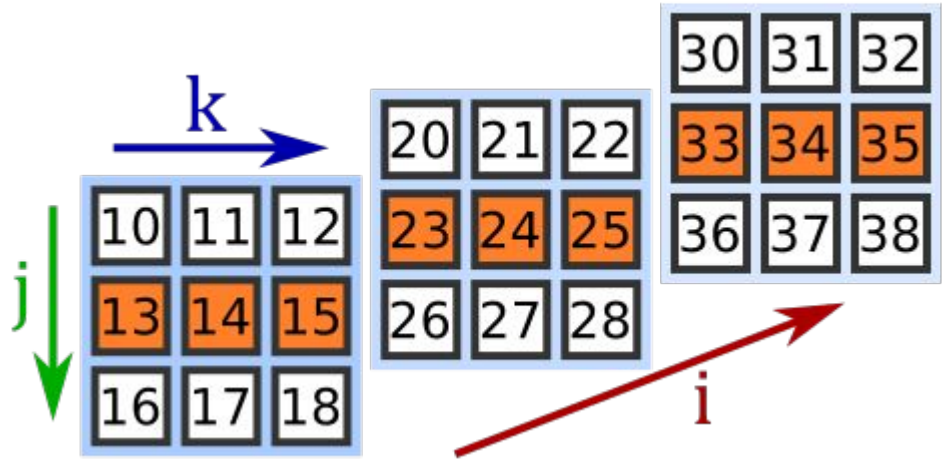
CASE 3: a3[:,1,2] = [15 25 35]

SELECTING MATRIX IN 3D NUMPY ARRAY

```
a3 = np.array([[[10, 11, 12], [13, 14, 15], [16, 17, 18]],
               [[20, 21, 22], [23, 24, 25], [26, 27, 28]],
               [[30, 31, 32], [33, 34, 35], [36, 37, 38]]])
```

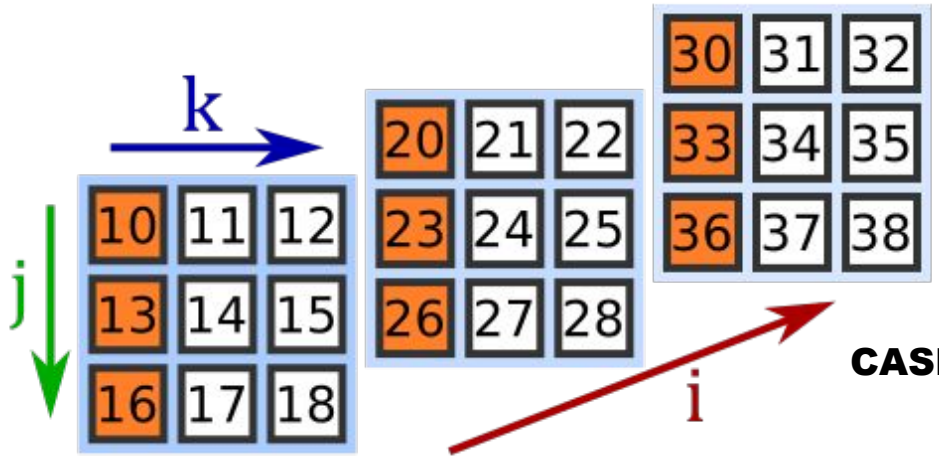


CASE 1: a3[2] = [[30 31 32]
[33 34 35]
[36 37 38]]



CASE 2: a3[:, 1]) = [[13 14 15]
[23 24 25]
[33 34 35]]

: -> Selects all

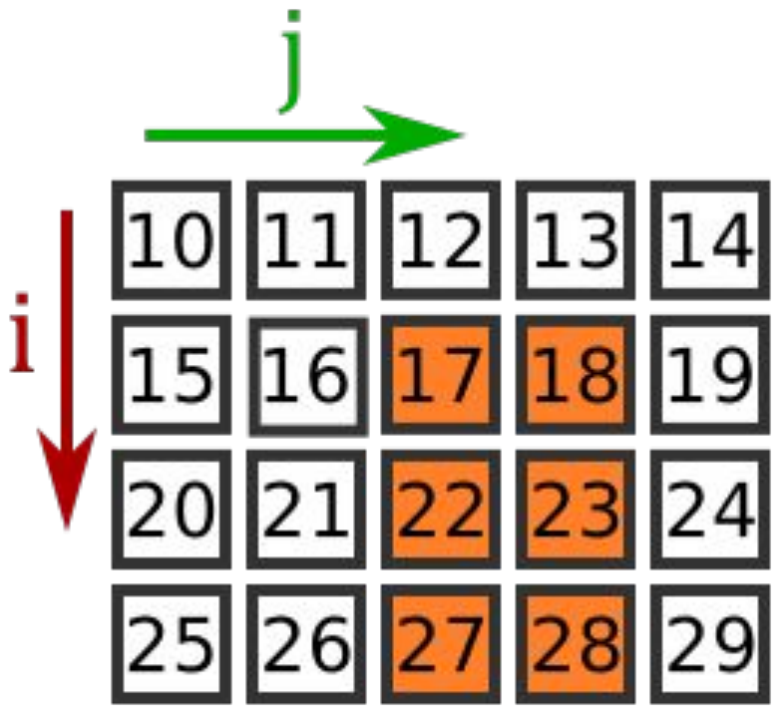


CASE 3: a3[:, :, 0]) # [[10 13 16]
[20 23 26]
[30 33 36]]

SLICING - NUMPY ARRAY

NumPy

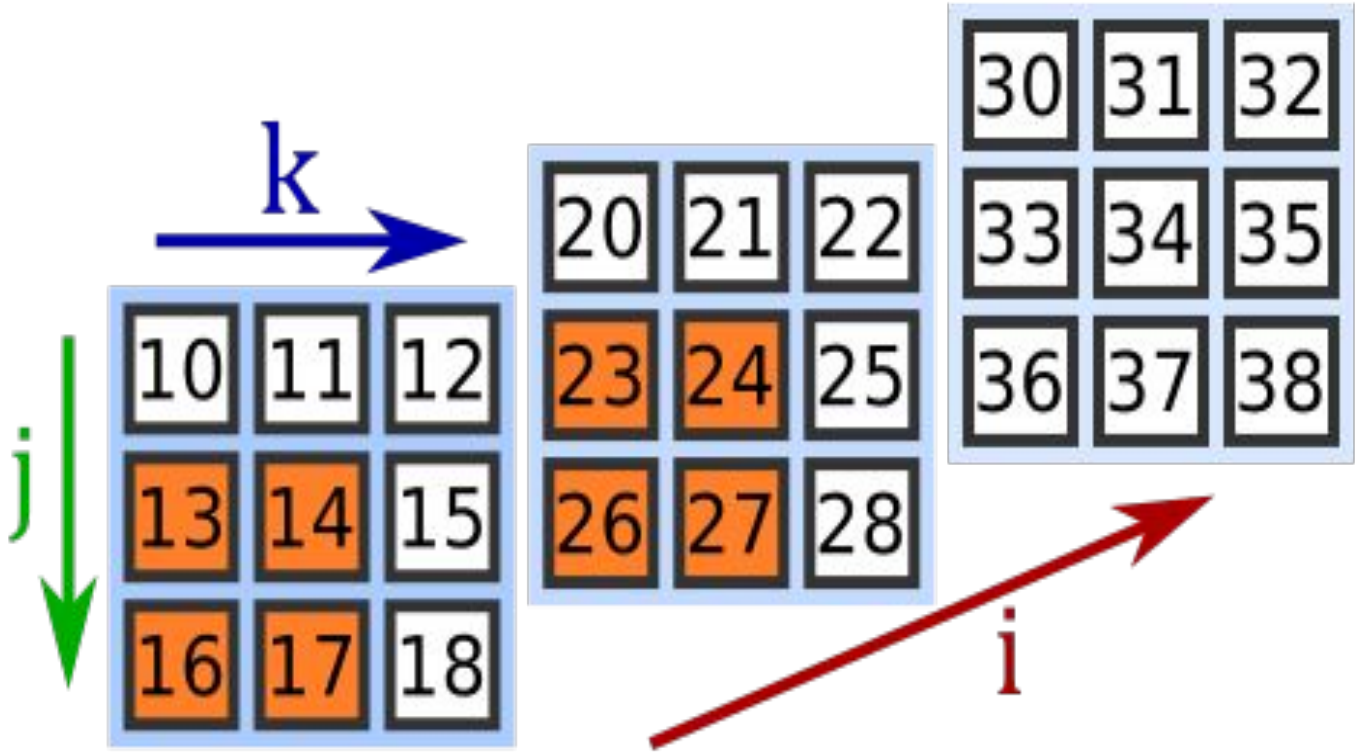
Slicing a 2D Array




```
a2 = np.array([[10, 11, 12, 13, 14],
               [15, 16, 17, 18, 19],
               [20, 21, 22, 23, 24],
               [25, 26, 27, 28, 29]])
```

```
SLICE a2[1:,2:4]) = [[17 18]
                     [22 23]
                     [27 28]]
```

Slicing a 3D Array

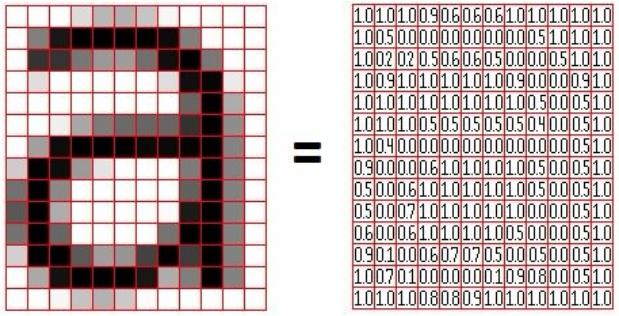



```
a3 = np.array([[[10, 11, 12], [13, 14, 15], [16, 17, 18]],
               [[20, 21, 22], [23, 24, 25], [26, 27, 28]],
               [[30, 31, 32], [33, 34, 35], [36, 37, 38]]])
```

```
SLICE a3[:2,1:,:2]) = [[ [13 14] [16 17] ]
                       [ [23 24] [26 27] ]]
```

IMAGE AS ARRAYS

NumPy



GREYSCALE IMAGE

pixel image

imread

3-channel matrix

COLOR IMAGE

NUMPY(EDA PART-III) EXERCISE

The Exercises using **Numpy** library is showns in the **Part-III** of the **Exploratory Data Analysis** Python Notebook.

The various **Numpy Library** methods are used to handle the **Image and Sound Data** (unstructured) in the Notebook.

EDA EXERCISE-1

Steps to be done:

1. Load the dataset: "international-airline-passengers.csv" File is stored in Google Drive.
2. Check the csv file using the ".info()" and ".head()" functions and write down your observations.
3. Use the function "pd.to_datetime()" to change the column type of 'Month' to a datatime type.
4. Set the index of data frame to be a datetime index using the column 'Month' and the "df.set_index()" function.
5. Choose the appropriate plot and display the data.
6. Choose appropriate scale.
7. Label the axes.

EDA EXERCISE-2

Steps to be done:

1. Load the dataset: "weight-height.csv"
2. Inspect it
3. Plot it using a scatter plot with Weight as a function of Height
4. Plot the male and female populations with 2 different colors on a new scatter plot
5. Remember to label the axes

EDA EXERCISE-3

Steps to be done:

1. 1.Plot the histogram of the heights for males and for females on the same plot.
2. 2.Use alpha to control transparency in the plot comand
3. 3.Plot a vertical line at the mean of each population using plt.axvline()

END OF CHAPTER