INTRODUCTION TO DATA ANALYTICS

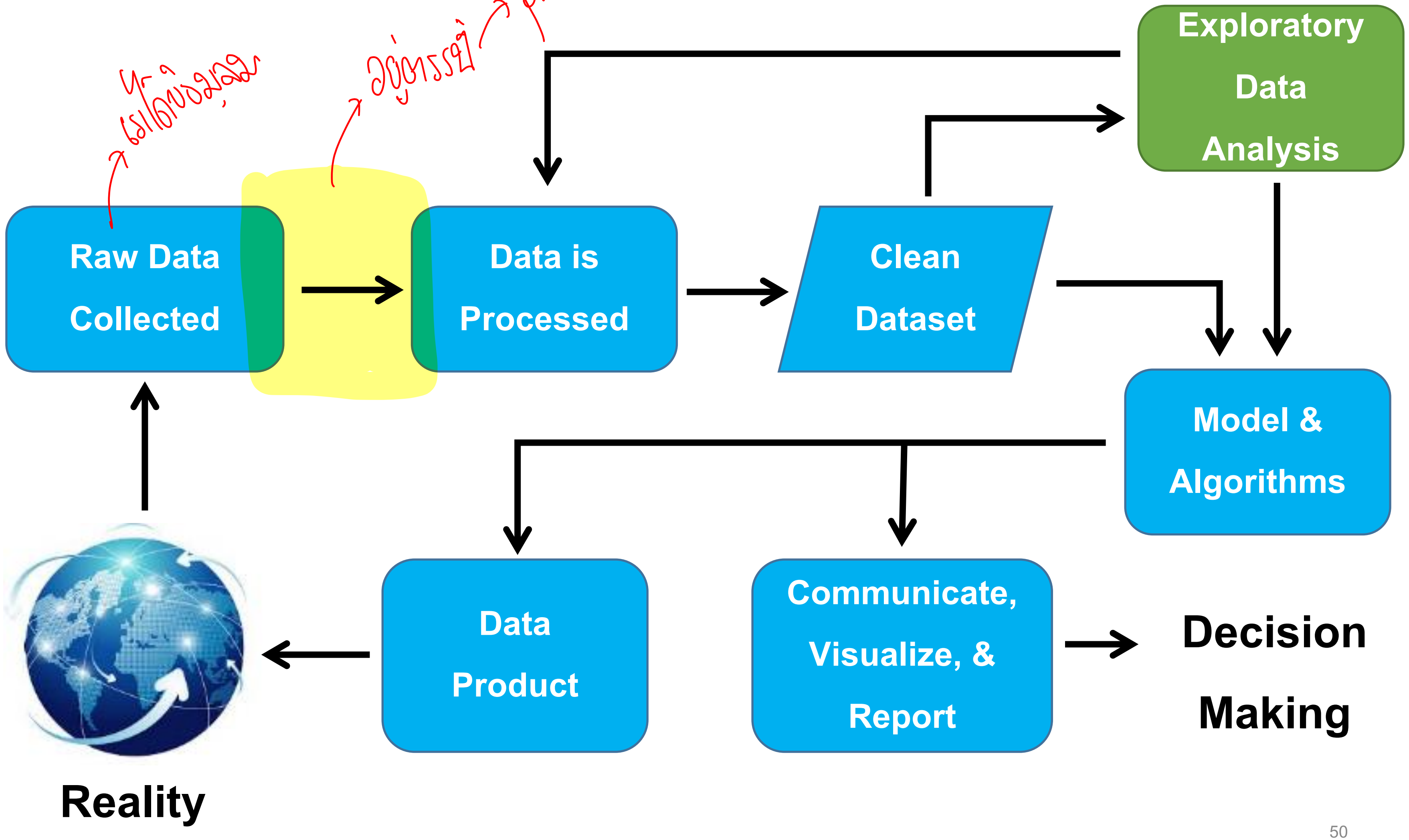
Data Manipulation

Using Python

Dr. Rathachai Chawuthai

Department of Computer Engineering
Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang

Data Science Process



เราได้ข้อมูลมา

อยู่ตรงนี้ → process ข้อมูลยังไง

Python

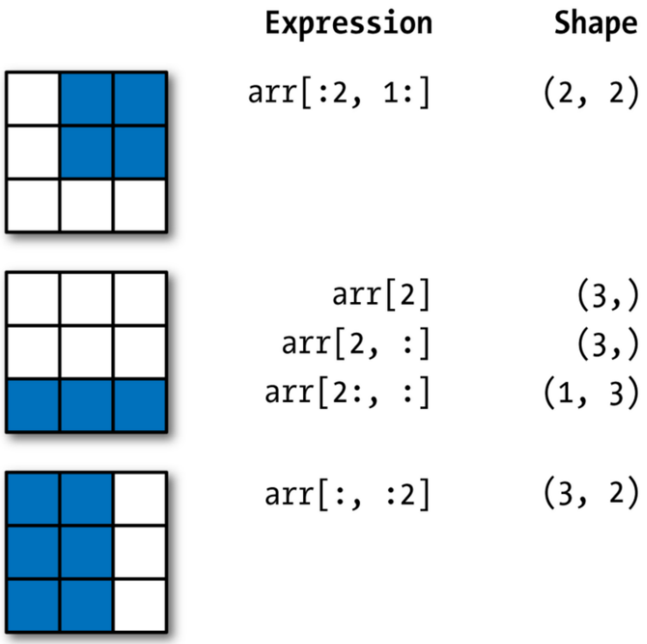
python basic

- List
  - [1, 2, "a"]
  - [1, 2, 3] + [10, 10, 10] = ? = msconcat
  - [1, 2, 3] * 10 = ? ms dup con
- Dictionary
  - { "a":1, "b":2 }
- Tuples
  - (1, 2, "a") immutable

Numpy

NumPy (or Numpy) is a Linear Algebra Library for Python, the reason it is so important for Data Science with Python is that almost all of the libraries in the PyData Ecosystem rely on NumPy as one of their main building blocks.

- Array
- Matrix
- Index
- Selection
- Operation

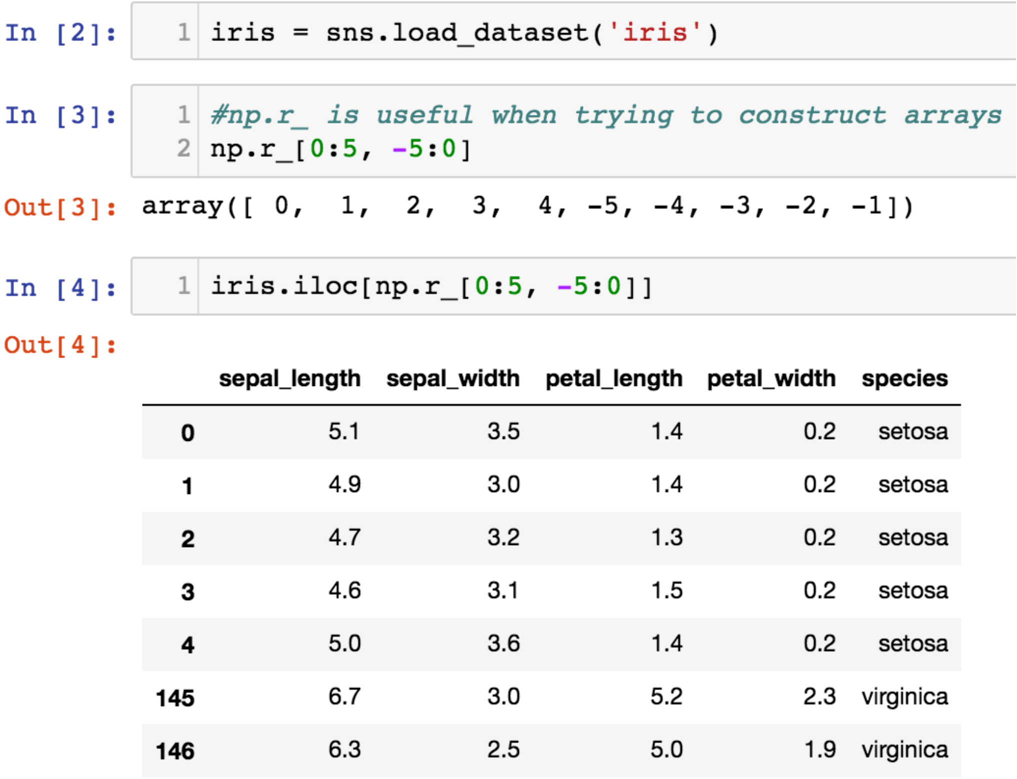


Expression	Shape
arr[:2, 1:]	(2, 2)
arr[2]	(3,)
arr[2, :]	(3,)
arr[2:, :]	(1, 3)
arr[:, :2]	(3, 2)

Ref: • (image) https://www.safaribooksonline.com/library/view/python-for-data/9781449323592/ch04.html

Pandas

- pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.
    - Series
    - DataFrame
    - Operation
    - Join DataFrames
    - Input and Output
    - GroupBy

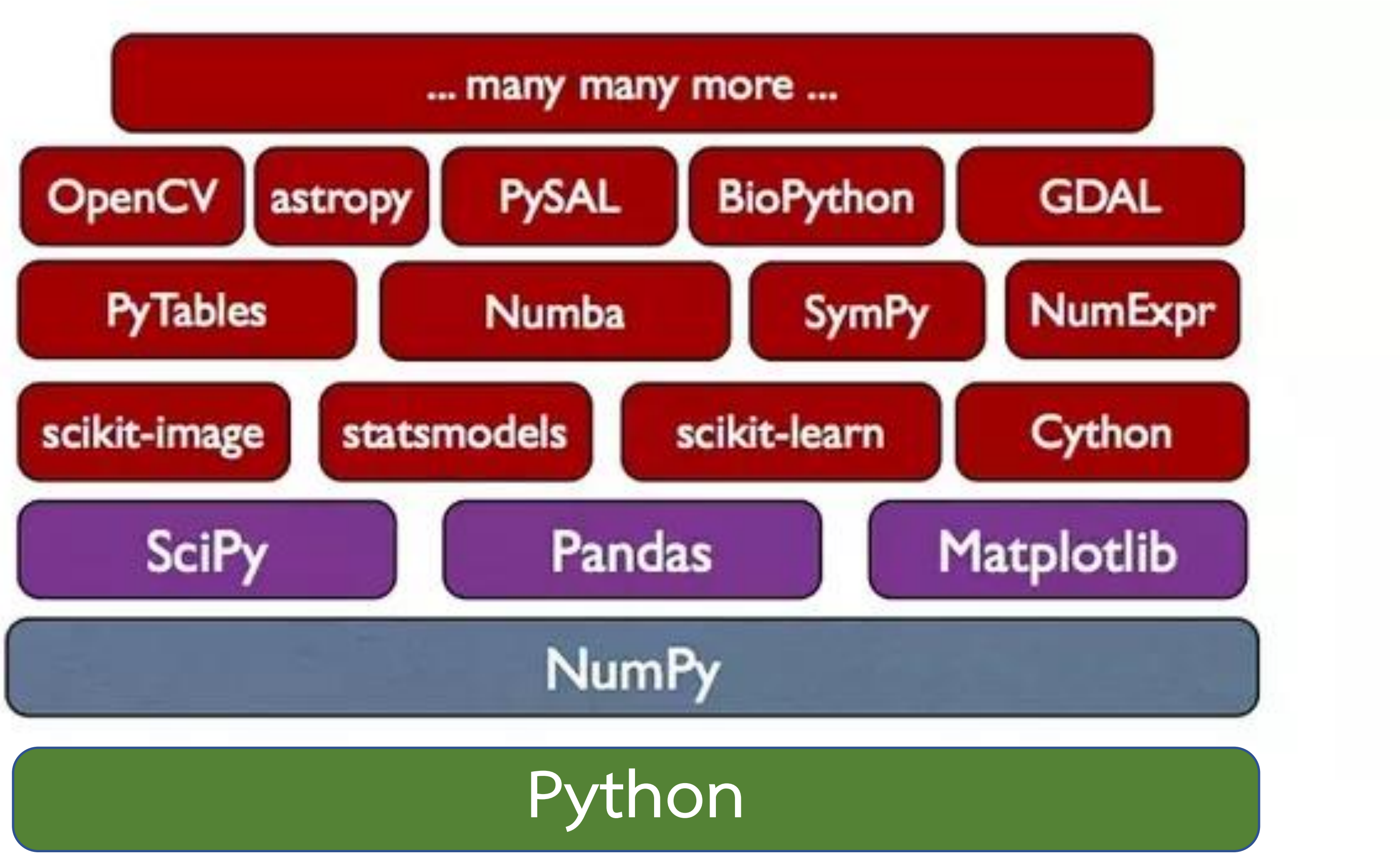


```
In [2]:  iris = sns.load_dataset('iris')

In [3]:  #np.r_ is useful when trying to construct arrays
         np.r_[0:5, -5:0]

Out[3]: array([ 0,  1,  2,  3,  4, -5, -4, -3, -2, -1])

In [4]:  iris.iloc[np.r_[0:5, -5:0]]

Out[4]:
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica

Ref: • (image) https://medium.com/@sean.turner026/week-2-and-useful-pandas-techniques-2f5dd78a5a59

Data Analytics Ecosystem



... many many more ...

OpenCV astropy PySAL BioPython GDAL

PyTables Numba SymPy NumExpr

scikit-image statsmodels scikit-learn Cython

SciPy Pandas Matplotlib

NumPy

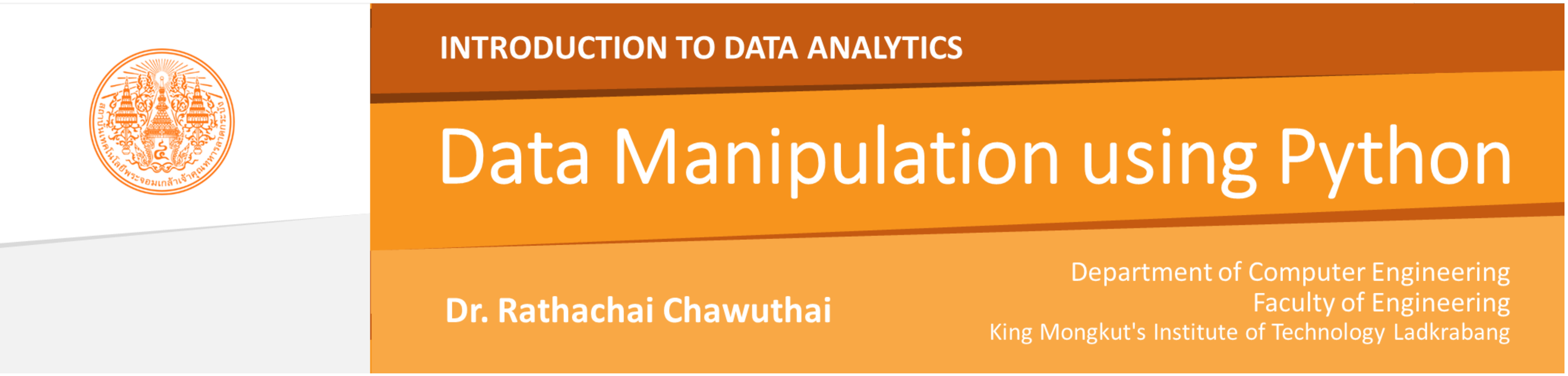
Python

Ref: • (image) https://www.quora.com/What-is-the-relationship-among-NumPy-SciPy-Pandas-and-Scikit-learn-and-when-should-I-use-each-one-of-them

Next

วันนี้สอนในแง่เปิดแค่ jupyter notebook
งงหมดเลยจ้า

↳ เนื้อหาจากความยาก
จะจดแต่ function ที่ต้องรู้เท่านั้น



INTRODUCTION TO DATA ANALYTICS

Data Manipulation using Python

Dr. Rathachai Chawuthai

Department of Computer Engineering
Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang

Demonstration

“

Data is the sword of the 21st century,
those who wield it the samurai.

”

Jonathan Rosenberg

つづく

การเลือกเอา column ออกมา df['column'] #series

df[['column', ...]] # dataframe

ใช้ string if type(index) == 'string' ได้

การเลือก row ออกมา df.loc[เลขแถว] #series

df.loc[[เลขแถว, ...]] #dataframe

df.loc[start:stop] #dataframe

↳ ไม่เหมือน iloc → iloc → ใช้เหมือน array slicing

อย่าสับสน

ใช้ addr เดียวกันแต่เลือก

↓

ทำอะไรตัวใหม่

ทำด้วย

combination df.loc[[เลขแถว, ...], ['column', ...]] #dataframe

↳ ใช้ใน edit

↳ df[['column', ...]].loc[[เลขแถว, ...]]

เหมือนเอา df ไปเลือก index ต่อ

การเลือกโดย iloc → df.iloc[[index, ...]]#df index location หรือ integer location

↳ เช่น df.iloc[[0,1]][['column', ...]] #df

df[df['column'] operator value] → ได้แถวที่เป็น true ออกมา

$$\begin{bmatrix} t \\ t \\ f \\ \vdots \end{bmatrix}$$

gb_obj= df.groupby(['column', ...])

gb_obj.mean(), min(), max(), ต่างๆนาๆ

ถ้าอยากแสดงค่าทีละหลายค่า gb_obj.agg({"mean", "count"})

ทำกับทุก column A → mean, count

ถ้าอยากเลือกcolumn เดียว

```
gb_obj['age'].agg({...})
```

ถ้าอยากเลือก column

```
gb_obj.agg({"column": 'function', ... })

gb_obj.agg({'column': ['function', ...], ... })
```

↳ การเข้าถึง → แปลกไปแบ่งขอบ → multi index

```
cols= ['_'.join(index) for index in gb_obj.columns.ravel]

gb_obj.columns = cols
```