

Getting Started with Analysis in Python: NumPy, Pandas and Plotting

Prat Thiru

Bioinformatics and Research Computing (BaRC)

http://barc.wi.mit.edu/hot_topics/

Python Review*

- Objects

- Attributes
- Methods

#String example

```
foo = "Hello World" #string object
```

```
foo.upper() #method
```

#numpy example

```
x = np.array([1, 2, 3]) #create a numpy object
```

```
x.ndim #attribute
```

- Data Structures

- lists ["a", "b", ...]
- tuples (1,2,3,...)
- dictionary {<key>:<value>, ...}

* See previous Hot Topics, Intro to Python I, for more information
http://barc.wi.mit.edu/education/hot_topics/

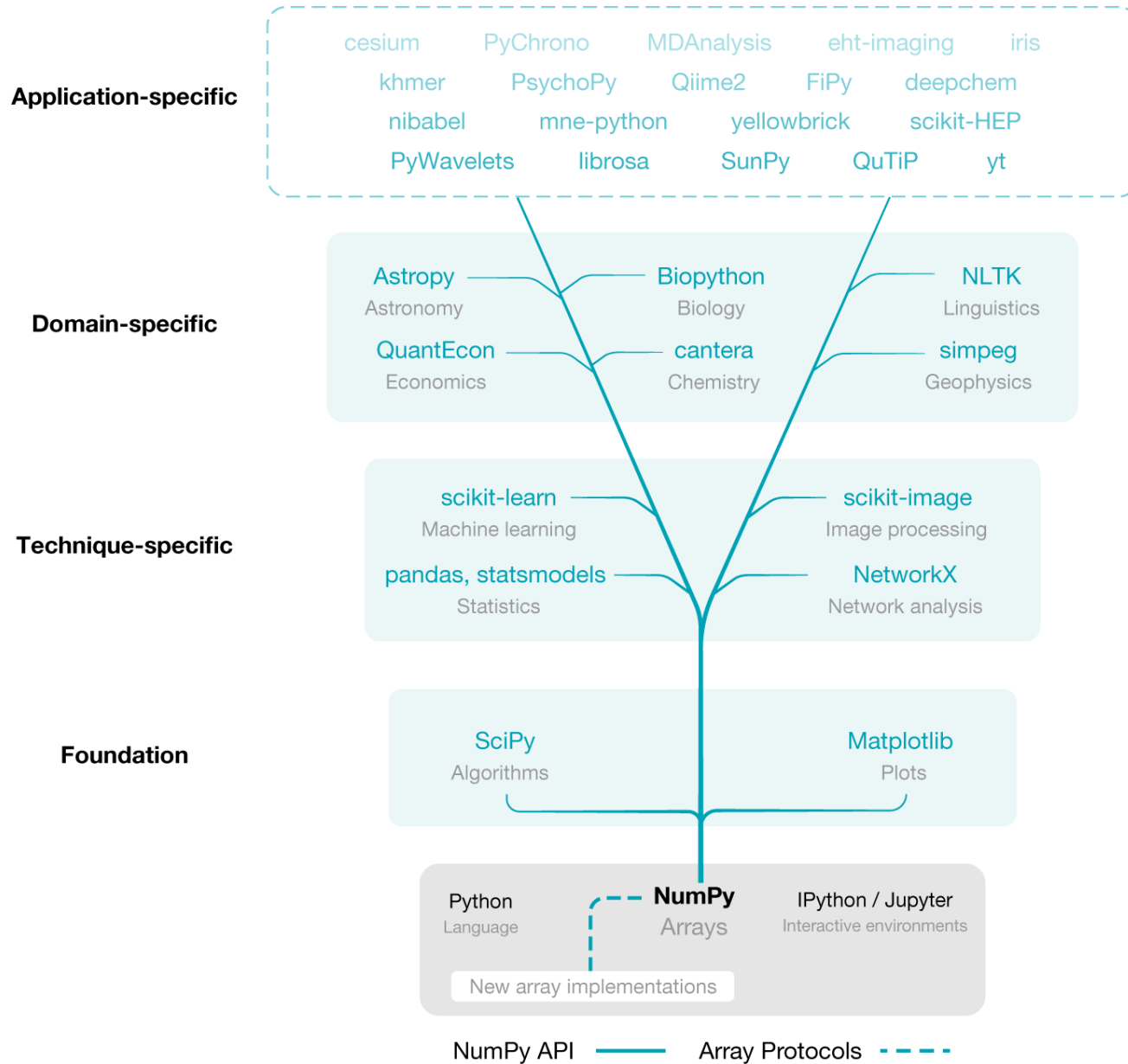


Python Packages

- Efficient and reusable
 - Avoid re-writing code
 - More flexibility
- Use the “import” command to use a package

```
import numpy as np
```

- Packages covered in this workshop:
 - NumPy
 - Pandas
 - Graphical: matplotlib, plotly and seaborn



NumPy

- Numerical Python
- Efficient multidimensional array processing and operations
 - Linear algebra (matrix operations)
 - Mathematical functions
- An array is a type of *data structure*
- Array (objects) must be of the **same type**

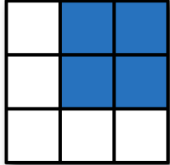
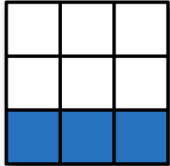
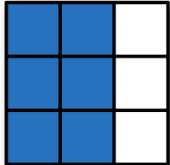
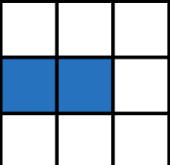
```
>>>import numpy as np
>>>np.array([1,2,3,4],float)
```


(NumPy) Array Concepts

- Index: refers to individual elements, or subarrays, that allows users to interact with arrays
 - slices
- Shape: number of elements along each *axis*, which determines the dimensions
- Vectorization: array programming, operations on the entire array than individual elements



NumPy: Slicing

| | Expression | Shape |
|--|--|---|
|  | <code>arr[:2, 1:]</code> | <code>(2, 2)</code> |
|  | <code>arr[2]</code> <code>arr[2, :]</code> <code>arr[2:, :]</code> | <code>(3,)</code> <code>(3,)</code> <code>(1, 3)</code> |
|  | <code>arr[:, :2]</code> | <code>(3, 2)</code> |
|  | <code>arr[1, :2]</code> <code>arr[1:2, :2]</code> | <code>(2,)</code> <code>(1, 2)</code> |

Pandas

- Efficient for processing tabular, or panel, data
- Built on top of NumPy
- Data structures: Series and DataFrame (DF)
 - Series: one-dimensional , same data type
 - DataFrame: two-dimensional, columns of different data types
 - index can be integer (0,1,...) or non-integer ('GeneA','GeneB',...)

index
↓

Series

| Gene | Expression |
|-------|------------|
| GeneA | 3.51 |
| GeneB | 0.44 |
| GeneC | 5.21 |
| GeneD | 4.55 |
| GeneE | 6.78 |

index
↓

DataFrame

| | Gene | GTEX-1117F | GTEX-111CU | GTEX-111FC |
|---|------------|------------|------------|------------|
| 0 | DDX11L1 | 0.1082 | 0.1158 | 0.02104 |
| 1 | WASH7P | 21.4 | 11.03 | 16.75 |
| 2 | MIR1302-11 | 0.1602 | 0.06433 | 0.04674 |
| 3 | FAM138A | 0.05045 | 0 | 0.02945 |
| 4 | OR4G4P | 0 | 0 | 0 |
| 5 | OR4F5 | 0 | 0 | 0 |

axis = 1

axis = 0

What can you do with a Pandas DataFrame?

- Filter
 - Select rows/columns
- Sort
- Numerical or Mathematical operations (e.g. mean)
- Group by column(s)
- Many others!

<https://pandas.pydata.org/pandas-docs/stable/>

DataFrame Slicing: Selecting Data

| Ensembl ID | Gene | GTEX-1117F | GTEX-111CU | GTEX-111FC |
|-----------------|------------|------------|------------|------------|
| ENSG00000223972 | DDX11L1 | 0.1082 | 0.1158 | 0.02104 |
| ENSG00000227232 | WASH7P | 21.4 | 11.03 | 16.75 |
| ENSG00000243485 | MIR1302-11 | 0.1602 | 0.06433 | 0.04674 |
| ENSG00000237613 | FAM138A | 0.05045 | 0 | 0.02945 |
| ENSG00000268020 | OR4G4P | 0 | 0 | 0 |
| ENSG00000186092 | OR4F5 | 0 | 0 | 0 |

- loc by row or column names
e.g. "Gene", "GTEX-117F"
- iloc by integer location,
i.e. column or row number
e.g. 1,2,3

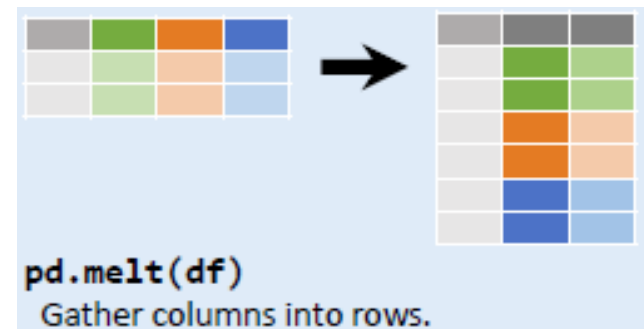
Data Formatting/Organizing

- By default, Pandas, and other packages, expect your data formatted such that each column represents a variable, and each row to represent an observation



Each **variable** is saved in its own **column**

Each **observation** is saved in its own **row**



Data Format Example

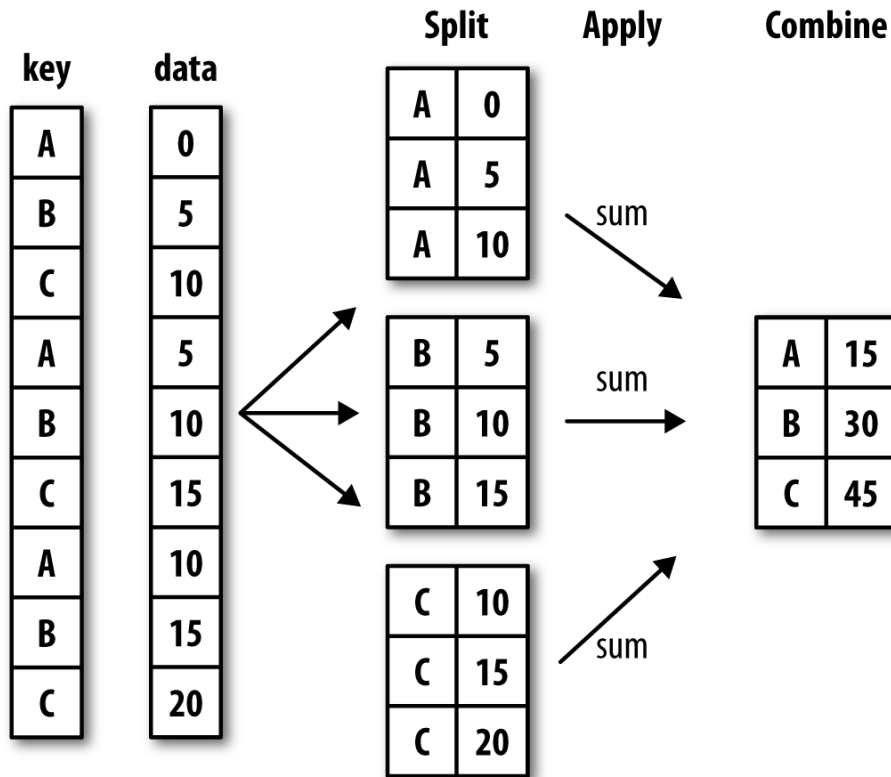
| Gene | Adipose | Adipose | Blood | Blood | Heart | Heart |
|---------|---------|---------|---------|---------|---------|-------|
| DDX11L1 | 0.1082 | 0.1158 | 0.05103 | 0.03214 | 0.04833 | 0.144 |
| WASH7P | 21.4 | 11.03 | 10.7 | 11.62 | 9.953 | 10.35 |
| FAM138A | 0.05045 | 0 | 0 | 0 | 0.09018 | 0.144 |



| Gene | Tissue | Expression |
|---------|---------|------------|
| DDX11L1 | Adipose | 0.1082 |
| WASH7P | Adipose | 21.4 |
| FAM138A | Adipose | 0.05045 |
| DDX11L1 | Adipose | 0.1158 |
| WASH7P | Adipose | 11.03 |
| FAM138A | Adipose | 0 |
| DDX11L1 | Blood | 0.05103 |
| WASH7P | Blood | 10.7 |
| FAM138A | Blood | 0 |
| DDX11L1 | Blood | 0.03214 |
| WASH7P | Blood | 11.62 |
| FAM138A | Blood | 0 |
| DDX11L1 | Heart | 0.04833 |
| WASH7P | Heart | 9.953 |
| FAM138A | Heart | 0.09018 |
| DDX11L1 | Heart | 0.144 |
| WASH7P | Heart | 10.35 |
| FAM138A | Heart | 0.144 |

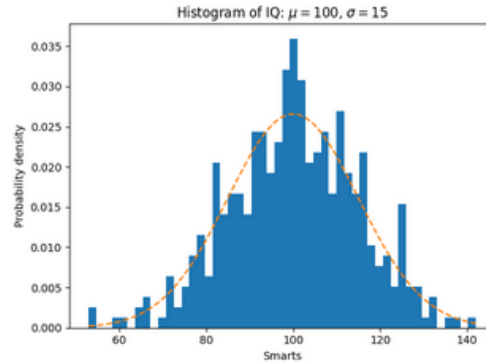
Pandas - groupby

- Split, Apply and Combine

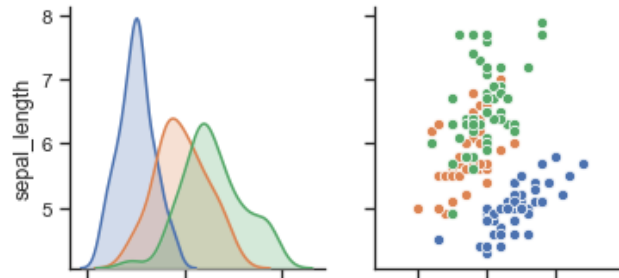


Plotting

- Matplotlib



- Seaborn



- Plotly

