

Library Overview

September 16, 2019

0.1 Library Overview

Much of this is adapted from “Python for Data Analysis : Data Wrangling with Pandas, NumPy, and IPython”. It is written by Wes McKinney the creator of Pandas

I didn't understand what the Python libraries were and in my online courses it wasn't really explained which to use when. I was simply asked to trust and incorporate these which resulted in my confusion. From my C programming days, they pretty much were equivalent to the *include* statement so using them wasn't really an issue.

numpy Short for Numerica Python. It is responsible for most numerical computations and provides the necessary data structures and algorithms. It has a multidimensional array called **ndarray** and associated operators

The syntax is:

```
[ ]: import numpy as np
```

np is the common shorthand for it.

pandas Comes from “panel data”, an econometrics term for multidimensional structured datasets or a play of the words “Python Data Analysis”. **DataFrames**, tabular and column-centric, and **Series**, a one-dimensional array, are the primary data structures.

The operations tend to resemble either a spreadsheet or **SQL**

The syntax is:

```
[ ]: import pandas as pd
```

pd is the common shorthand for it.

matplotlib Used for charting “basic” visualisations.

The syntax is:

```
[ ]: import matplotlib as mp
```

mp is the common shorthand for it.

scipy Used for scientific applications like integration. linear algebra, sparse matrices, function optimisation, signal processing, statistics, etc.

The syntax is:

```
[2]: import scipy
```

scikit-learn Used for “basic” Machine Learning (ML)

The syntax is:

```
[6]: import sklearn
```

You can find out more from: <https://scikit-learn.org/stable/tutorial/basic/tutorial.html>

```
[ ]:
```