UECS3213 / UECS3453 Data Mining

SESSION: January 2019

Lab 2: Introduction to Python Programming & Python Data Science Libraries: numpy, scipy, pandas, matplotlib, scikit-learn

Part 1: Introduction to Python Programming

Objectives

At the end of this lab, you are expected to know and able to program with Python in the following (but not limited to) topics such as:

- Variables and Types
- Lists
- Basic Operators
- String Formatting
- Basic String Operations
- Conditions
- Loops
- Functions
- Classes and Objects
- Dictionaries
- Modules and Packages

Introduction

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

- **Python is Object-Oriented:** Python supports object-oriented style or technique of programming that encapsulates code within objects.

Today, Python is a general-purpose programming language that is becoming more and more popular for doing data science.

Instruction

1. Please attempt the Python tutorials on any of the following websites for practice:
   - https://www.w3schools.com/python
   - https://www.tutorialspoint.com/python
   - https://www.learnpython.org

2. Also, attempt the "7-Day Learn Python Challenge" (https://www.kaggle.com/page/7-day-python-challenge) on Kaggle. (Note: Kaggle is an AirBnB for Data Scientists. It is a crowd-sourced platform to attract, nurture, train and challenge data scientists from all around the world to solve data science, machine learning and predictive analytics problems.)

You are recommended to use the Python (Anaconda distribution) and Jupyter or Spyder Notebook IDE that we have already downloaded, install and set up though you may also execute Python using online compiler tool such as https://www.tutorialspoint.com/execute_python_online.php

The End

Part 2: Introduction to Python Data Science Libraries

Objectives

At the end of this lab, you are expected to understand the following Python data analysis libraries:

a) numpy
b) scipy
c) pandas
d) matplotlib
e) scikit-learn

Introduction

a) **numpy** adds Python support for large, multi-dimensional arrays and matrices, along with a large library of high-level mathematical functions to operate on these arrays.

b) **scipy** is a collection of mathematical algorithms and convenience functions built on the numpy extension of Python. It adds significant power to the interactive Python session by providing the user with high-level commands and classes for manipulating and visualizing data.

c) **pandas** is the **s**oftware library written for data manipulation and analysis in Python. Pandas is like spreadsheets for Python (something like R). It is able to describe the data for you. It can do grouping and pivot tables on larger data.

d) **matplotlib** is a multi-platform data visualization library built on numpy arrays, and designed to work with the broader scipy stack. Matplotlib supports dozens of backends and output types, which means you can count on it to work regardless of which operating system you are using or which output format you wish.

e) **scikit-learn** is a a free software machine learning library for the Python. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

The data science libraries of Python is summarized in Figures 1 and 2 below.

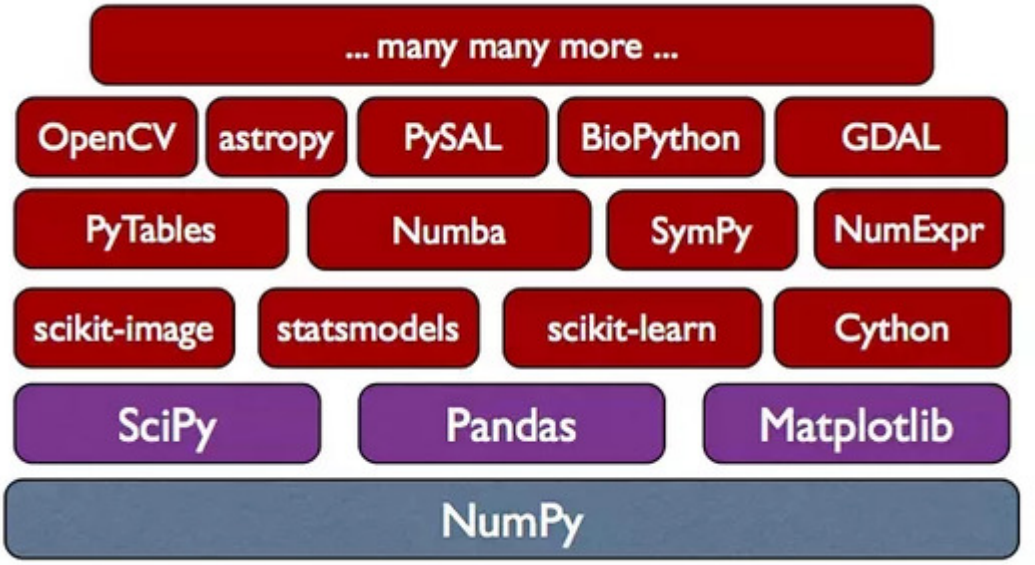

Figure 1: Python data science libraries

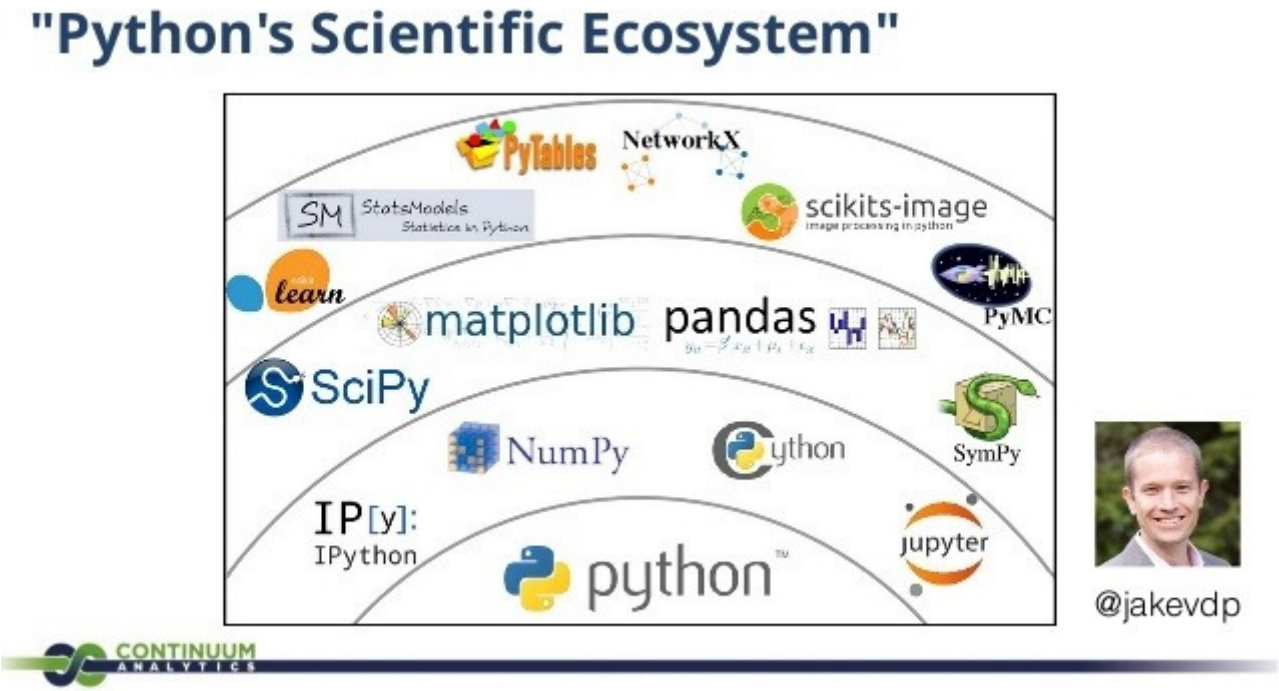

Figure 2: Python scientific ecosystem

Instruction

How to Install Numpy, Scipy, Matplotlib, Pandas & Scikit-Learn on Windows

Python comes loaded with powerful packages that make machine learning tasks easier. This is why it is the language of choice among data scientists. Of the vast collection of libraries that you can choose from, there are a set of basic libraries you should be familiar with as a beginner.

In this tutorial we are going to install these basic libraries on our system using Python's built in package manager PIP.

Numpy:

NumPy (stands for Numerical Python) provides useful features for operations on n-arrays and matrices in Python. It provides vectorization of mathematical operations on the NumPy array type.

Installation:

1. In the terminal type the command **pip install numpy**
2. For security reasons, you will be asked to enter your password.
3. Installation will take only a few seconds.

Numpy is now installed on your system.

Testing:

1. In the terminal, start Python by typing the command **python**
2. Use the following error handling block:

```
try:
   import numpy
except ImportError:
   print ("numpy is not installed")
```

3. If numpy is installed successfully, then you will not get any message in the terminal. Otherwise you will get an error message saying "numpy is not installed".

Troubleshooting:

If you get the error message, try this command **pip install -U numpy**

Scipy:

SciPy contains modules for linear algebra, optimization, integration, and statistics. It is built upon NumPy. It provides efficient numerical routines as numerical integration, optimization, and more via specific submodules.

Installation:

1. In the terminal type the command **pip install scipy**
2. For security reasons, you will be asked to enter your password.
3. Installation will take only a few seconds.

Scipy is now installed on your system.

Testing:

1. In the terminal, start Python by typing the command **python**
2. Use the following error handling block:

```
try:
   import scipy
except ImportError:
   print ("scipy is not installed")
```

3. If scipy is installed successfully, then you will not get any message in the terminal. Otherwise you will get an error message saying "scipy is not installed".

Troubleshooting:

If you get the error message, try this command **pip install -U scipy**

Matplotlib:

It is used for the generation of simple and powerful visualizations.
You can make just about any visualizations such as bar charts, histograms & pie charts. There are facilities for creating labels, grids and other formatting elements.

Installation:

1. In the terminal type the command **pip install matplotlib**
2. For security reasons, you will be asked to enter your password.
3. Installation will take only a few seconds.

Matplotlib is now installed on your system.

Testing:

1. In the terminal, start Python by typing the command **python**
2. Use the following error handling block:

```
try:
   import matplotlib
except ImportError:
   print ("matplotlib is not installed")
```

3. If matplotlib is installed successfully, then you will not get any message in the terminal. Otherwise you will get an error message saying "matplotlib is not installed".

Troubleshooting:

If you get the error message, try this command **pip install -U matplotlib**

Pandas:

Pandas works with "labeled" and "relational" data. Pandas is primarily used for data wrangling. It was designed for quick and easy data manipulation, aggregation, and visualization.

Installation:

1. In the terminal type the command **pip install pandas**
2. For security reasons, you will be asked to enter your password.
3. Installation will take only a few seconds.

Pandas is now installed on your system.

Testing:

1. In the terminal, start Python by typing the command **python**
2. Use the following error handling block:

```
try:
   import pandas
except ImportError:
   print ("pandas is not installed")
```

3. If pandas is installed successfully, then you will not get any message in the terminal. Otherwise you will get an error message saying "pandas is not installed".

Troubleshooting:

If you get the error message, try this command **pip install -U pandas**

Scikit-Learn:

This package is built on the top of SciPy and makes heavy use of its mathematical operations. It provides access to common machine learning algorithms, making it simple to bring machine learning into any project. It is easy to use and is great for playing around with machine learning concepts.

Installation:

1. In the terminal type the command **pip install scikit-learn**
2. For security reasons, you will be asked to enter your password.
3. Installation will take only a few seconds.

Scikit-Learn is now installed on your system.

Testing:

1. In the terminal, start Python by typing the command **python**
2. Use the following error handling block:

```
try:
   import scikit-learn
except ImportError:
   print ("scikit-learn is not installed")
```

3. If scikit-learn is installed successfully, then you will not get any message in the terminal. Otherwise you will get an error message saying “scikit-learn is not installed”.

Troubleshooting:

If you get the error message, try this command **pip install -U scikit-learn**

You are now armed with the basic tools that you need to begin your data science journey.

The End