# Intro to Python for Data Science

Anthony FAUSTINE

May 2017

# Outline

# Learning goal

- Understand python programming language and different python libraries for data science.
- Explore Python language fundamentals, including basic syntax, variables, control flow, data structure and functions.
- Build Numpy arrays, and perform basic and some linear algebra calculations.
- Create and customize plots using matplotlib.

# Presenter Bio

- PhD student at Nelson Mandela African Institution of Science and Technology,
- **Research :** Applied machine learning and signal processing for computational sustainability.
  - Develop probabilistic-deep learning algorithm (Hybrid HMM-DNN) for energy dis-aggregation problem.
- Co-founder Pythontz
- Assistant Lecturer (UDOM), Researcher (Vicres, Hakikidawa).

# Pythontz


python™
Tanzania
We aim to create a vibrant and diverse python community in Tanzania

# Pythontz

## About Pythontz

- A postive peer learning community for interested Python users in Tanzania.

**Vision**

- To create a vibrant and diverse python community in Tanzania.

**Mission**

- To foster the application of python programming across industries, learning centers, schools and community in Tanzania.

# Outline

# Introduction

## What is Python ?

A very popular general-purpose programming language.

- Open source general-purpose language
- Dynamically semantics (rather than statically typed like Java or C/C++)
- Interpreted (rather than compiled like Java or C/C++)
- Object Oriented,

# What can you use Python for ?

- Web development (Django)
- Web Scraping (Beautiful Soup)
- Scripting Language.
- Scientific programming and Numeric Computing.
- Automation and Embedded Sytstem.
- Desktop GUIs and 3D modelling.

# But Why Python ?

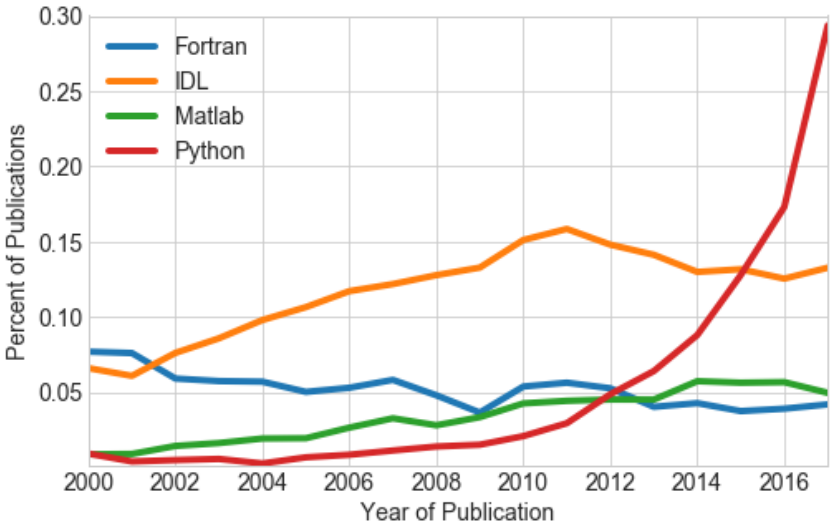



FIGURE – Jake VanderPlas PyCon 2017

- Python is a “teaching language”
- ....created to “bridge the gap between the shell and C
- “never intended. . . to be the primary language for programmers.”

# Why is Python such an effective tool in science ?

1. Interoperability with Other Languages : You can use it in the shell on microtasks, or interactively, or in scripts, or build enterprise software with GUIs.
2. “Batteries Included” + Third-Party Modules : Python has built-in libraries and third-party liabraies for nearly everything.
3. Simplicity & Dynamic Nature : You can run your Python code on any architecture.
4. Open ethos well-fit to science : Easy to reproduce results with python
5. Python is the future of Machine Learning and AI.

*Jake VanderPlas PyCon 2017*

# Why is Python such an effective tool in science ?

1. Interoperability with Other Languages : You can use it in the shell on microtasks, or interactively, or in scripts, or build enterprise software with GUIs.
2. “Batteries Included” + Third-Party Modules : Python has built-in libraries and third-party liabraies for nearly everything.
3. Simplicity & Dynamic Nature : You can run your Python code on any architecture.
4. Open ethos well-fit to science : Easy to reproduce results with python
5. Python is the future of Machine Learning and AI.

*Jake VanderPlas PyCon 2017*

# Why is Python such an effective tool in science ?

1. Interoperability with Other Languages : You can use it in the shell on microtasks, or interactively, or in scripts, or build enterprise software with GUIs.
2. “Batteries Included” + Third-Party Modules : Python has built-in libraries and third-party liabraies for nearly everything.
3. Simplicity & Dynamic Nature : You can run your Python code on any architecture.
4. Open ethos well-fit to science : Easy to reproduce results with python
5. Python is the future of Machine Learning and AI.

*Jake VanderPlas PyCon 2017*

# Why is Python such an effective tool in science ?

1. Interoperability with Other Languages : You can use it in the shell on microtasks, or interactively, or in scripts, or build enterprise software with GUIs.
2. “Batteries Included” + Third-Party Modules : Python has built-in libraries and third-party liabraies for nearly everything.
3. Simplicity & Dynamic Nature : You can run your Python code on any architecture.
4. Open ethos well-fit to science : Easy to reproduce results with python
5. Python is the future of Machine Learning and AI.

*Jake VanderPlas PyCon 2017*

# Why is Python such an effective tool in science ?

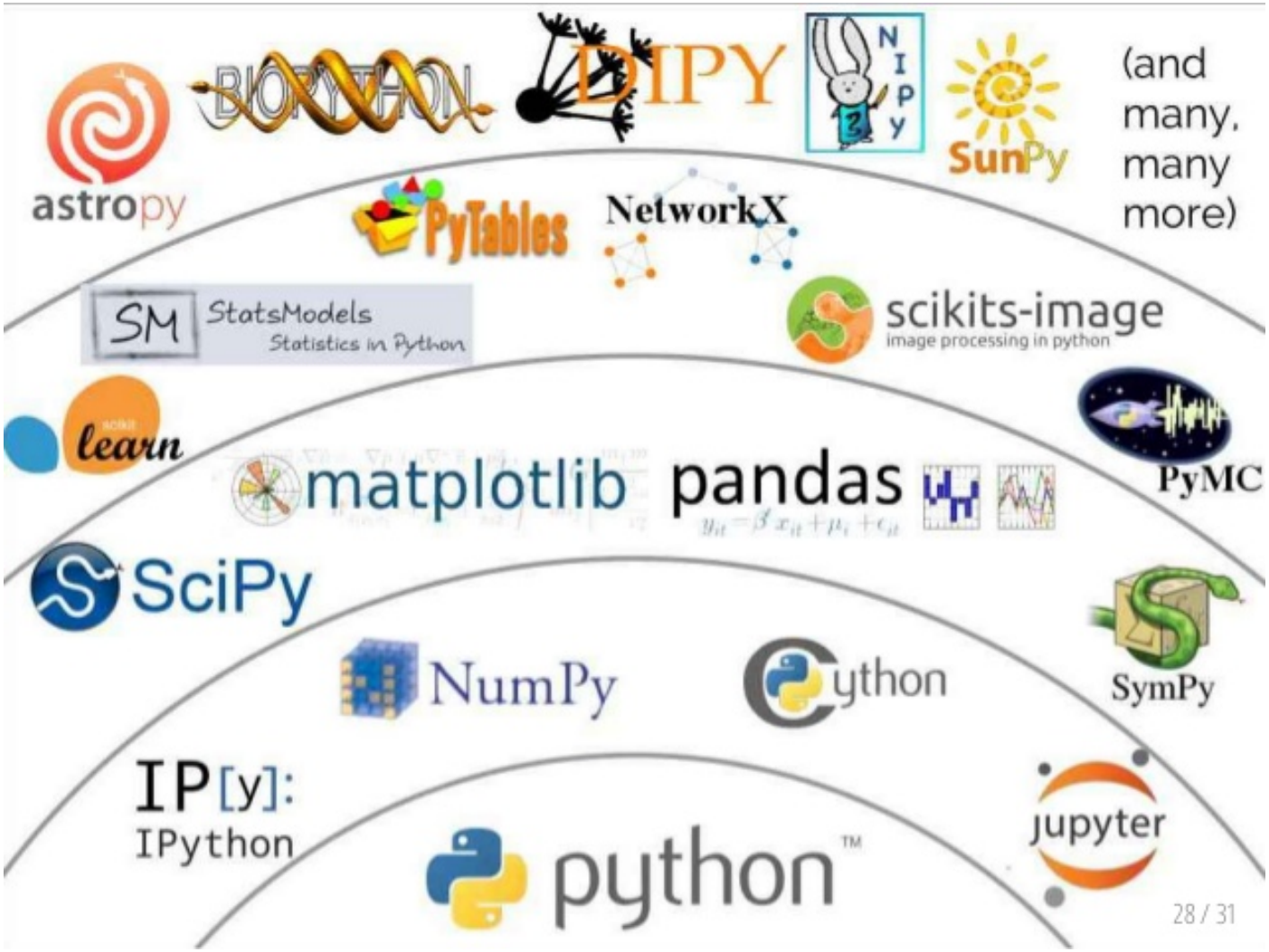
1. Interoperability with Other Languages : You can use it in the shell on microtasks, or interactively, or in scripts, or build enterprise software with GUIs.
2. “Batteries Included” + Third-Party Modules : Python has built-in libraries and third-party liabraies for nearly everything.
3. Simplicity & Dynamic Nature : You can run your Python code on any architecture.
4. Open ethos well-fit to science : Easy to reproduce results with python
5. Python is the future of Machine Learning and AI.

*Jake VanderPlas PyCon 2017*

# Resource to learn Python

## 10 Resources to Get Started Learning Python

# Python's Scientific Stack



# Outline

# Jupyter

**Jupyter** : Open-source web application for interactive and exploratory computing.

- Allows to create and share documents that contain live code, equations, visualizations and explanatory text.


- It is a platform for Data Science at scale.
- Covers all the life-cycle of scientific ideas :ideas to publications.
- Demo

# Numpy and Sci-py

**Numpy** : the fundamental Python package for scientific computing.


- Provide high-performance vector, matrix and higher-dimensional data structures.
- Offers Matlab-ish capabilities within Python.

**Sci-py** : Collections of high level mathematical operations

- linear algebra.
- Optimization
- Integration etc.

# Numpy Arrays vs Python list

A numpy array is a grid of values, all of the same type, and is indexed by a tuple of nonnegative integers.

## Numpy vs List

- The essential difference between lists and NumPy arrays is functionality and speed.
  - Lists give you basic operation, but NumPy adds FFTs, convolutions, fast searching, basic statistics, linear algebra, histograms etc.
- Thus Numpy array is memory-efficient container that provides fast numerical operations.

# Matplotlib

Matplotlib is an excellent 2D and 3D graphics library for generating scientific figures.

- It provides both a very quick way to visualize data from Python and publication-quality figures in many formats.

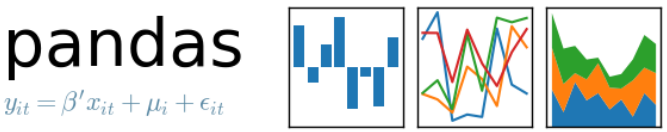

Other data visualization packages : Seaborn and Bokeh.

# Other Python Library for Visualization

# Pandas

Panda : a python package providing fast, flexible, and expressive data structures for data analysis.

- A fundamental high-level building block for doing practical, real world data analysis in Python.
- Designed to work with relational or labeled data or both.



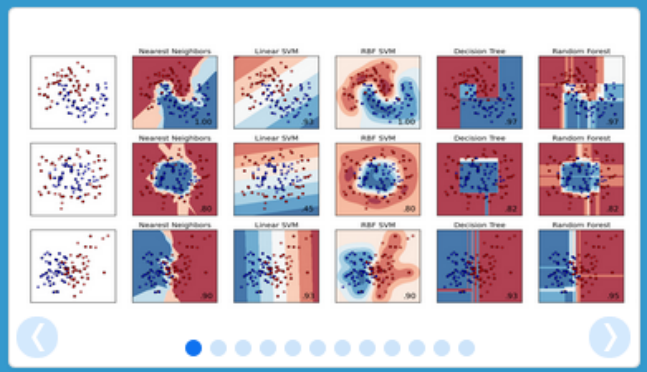
# Scikit-Learn for ML

Scikit-Learn (sklearn) is Python's premier general-purpose machine learning library.

scikit learn

Home Installation Documentation Examples

Google™ Custom Search Search



## scikit-learn

*Machine Learning in Python*

- Simple and efficient tools for data mining and data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

### Classification

Identifying to which set of categories a new observation belong to.

**Applications**: Spam detection, Image recognition.
**Algorithms**: *SVM*, *nearest neighbors*, *random forest*, ... — *Examples*

### Regression

Predicting a continuous value for a new example.

**Applications**: Drug response, Stock prices.
**Algorithms**: *SVR*, *ridge regression*, *Lasso*, ... — *Examples*

### Clustering

Automatic grouping of similar objects into sets.

**Applications**: Customer segmentation, Grouping experiment outcomes
**Algorithms**: *k-Means*, *spectral clustering*, *mean-shift*, ... — *Examples*

### Dimensionality reduction

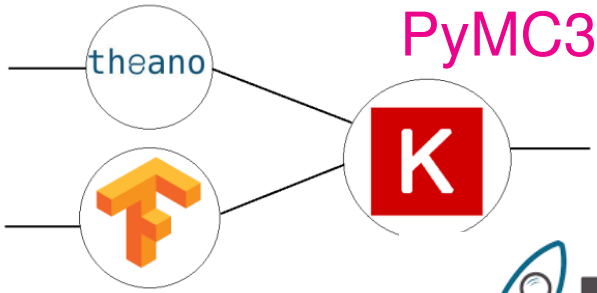
### Model selection

### Preprocessing

# Python ML and AI libraries

Tensorflow


Pytorch

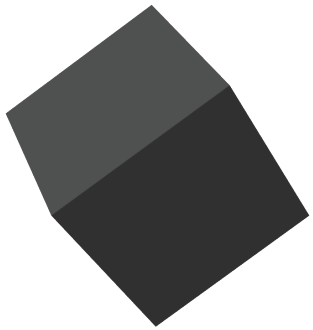




PyMC3


Theano


Keras

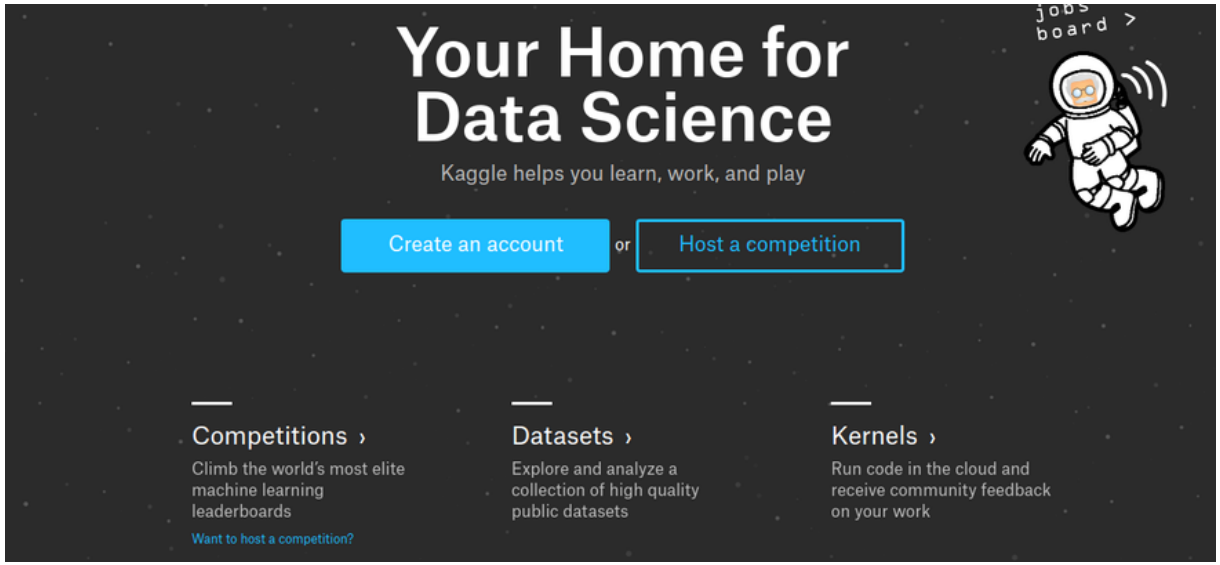
Edward



NLTK

# Data Science Platform

Kaggle : helps you learn, work, and play.




Data set :

- Academic Torrents
- UCI Machine learning repository

**THANK YOU**

# Practical Session

Practical Session