

Artificial Intelligence Environment setup

Group 5 AI Year 3 African Leadership University Computer Science

January 2021

1 Group Member Details

- Francis Hayward Walker <https://github.com/Francis-Walker>
- Wenseslaus Raphael Mwtia <https://github.com/Wenseslaus>
- Jean Damascene Uwingeneye <https://github.com/jeandamas>
- Rachel Christelle Sarah Ayateke <https://github.com/KChristelle>

2 Introduction

During this assignment, the main purpose was to setup the environment and test libraries required in Artificial Intelligence course. Starting from python installation to package installation, below are the processes it took to set-up the AI development environment.

3 Demo Link

<https://www.youtube.com/watch?v=cElrZSl1W1cfeature=youtu.be>

4 Installation Process

This section describes in details the steps of installing and setting up, the Artificial Intelligence Working environment. **NOTE:** The steps taken here, are specifically for Linux and Unix operating systems.

4.1 Anaconda individual edition installation

1. Download Anaconda installer and follow instructions.
2. Install Anaconda for Python 3.8 on linux/Unix
3. Accept the license agreements.
4. Accept to run conda init.
5. Installation is completed.

4.2 Installing packages

To install packages, there are three possibilities.

1. Using Conda command or
2. Using Pip command.
3. Using Anaconda Graphical user interface

Anaconda has ability to manage packages installed via Pip, Conda or the Anaconda navigator.

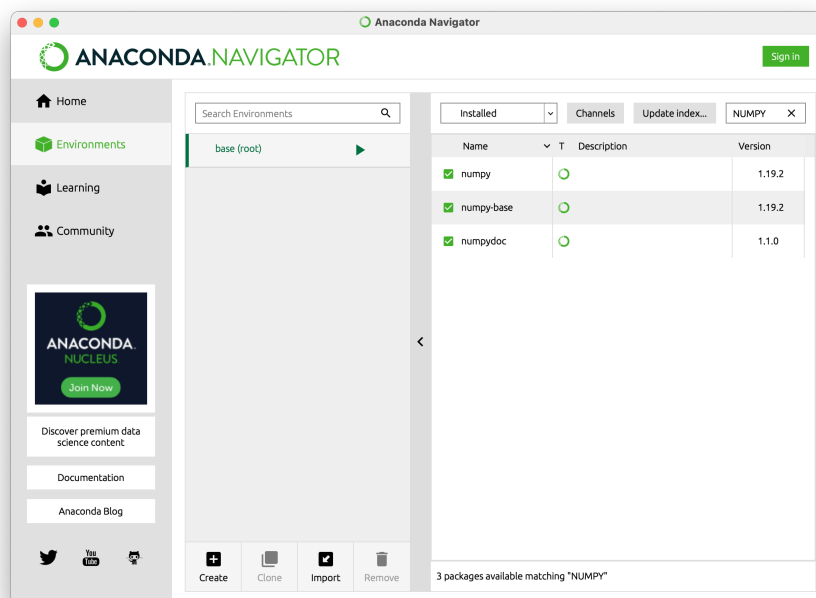


Figure 1: Using Anaconda Navigator to install packages

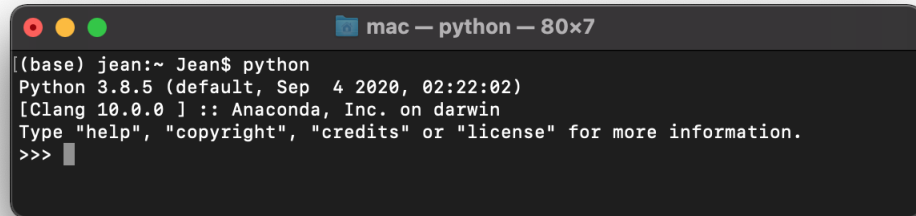
5 Anaconda

5.1 Overview

Anaconda is a multi platform distribution of R and Python which is intended to increase the ease of use of package-management in the field of data science. With over 250 packages installed by default, it supports a over 7500 open-source packages. Utilizing the Conda package management System it removes user configuration integration's by accounting for, installing and deploying all required dependencies while also considering the current operating environment for any user specified requirements such as a preferred version type or alternatively already installed packages.

5.2 Practical Example

The version of anaconda is demonstrated here which serves to confirm the integrity of it's installed package and functionality.



```
((base) jean:~ Jean$ python
Python 3.8.5 (default, Sep 4 2020, 02:22:02)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Figure 2: Python Conda based Version screen-shoot

6 Python

6.1 Overview

Python is a high-level programming language with an emphasise on code readability and dynamic usage. It's use of white space and indentation for classification and separation of atomic commands it allows for increased readability of understanding of commands. Further Python supports multiple programming paradigms such as structural and object orientated thus providing it's user with a incredibly versatile language. Further, this has wide and varied support for thousands of packages and vast public support for multiple different skill groups. These benefits have made Python the go-to language for data-science and some would say essential language.

6.2 Practical Example

All code is done in the python language on Jupyter Notebook.

7 Jupyter Notebook

7.1 Overview

Jupyter Notebook is an open source web-app aimed at providing a platform for collaboration of live code, visualisations and text. Based on the IPython project it implements a code segmentation timeline model in order to record inputted code and it's related output, as code is sent to Ipython kernel it is recorded in a time-series fashion allowing for post run analysis of code.

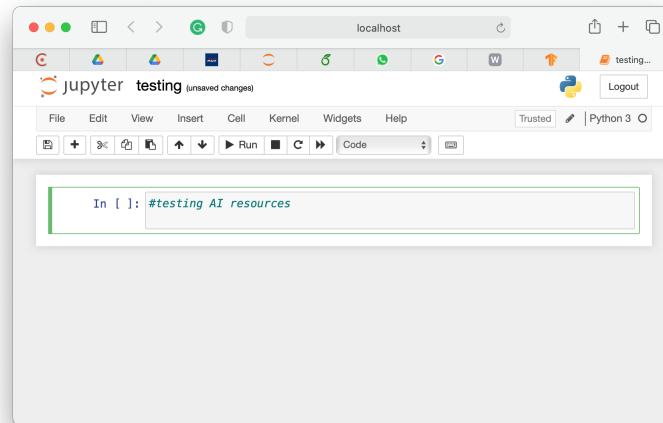


Figure 3: The Jupyter Notebook screen-shoot

7.2 Practical Example

The Jupyter Notebook was used as an operating environment.

8 TensorFlow

8.1 Overview

Tensor flow is an open-source library used in machine learning. While being able to utilize vast functionality it is specifically used for it's machine-learning and training functionality. Primarily the library contains large amount of optimised math functions (specifically data flow and calculus) which both aim and increase the efficiency of developing, implementing and training machine learning algorithms.

```
In [52]: # Multiplication of two different contacts with tensorflow
x = tf.random.normal((2, 2))
y = tf.random.normal((2, 2))
solution = tf.multiply(x, y)
print(solution)

tf.Tensor(
[[ 2.022013    0.31564692]
 [ 2.5037603 -0.089178  ]], shape=(2, 2), dtype=float32)
```

Figure 4: Tensor Creation

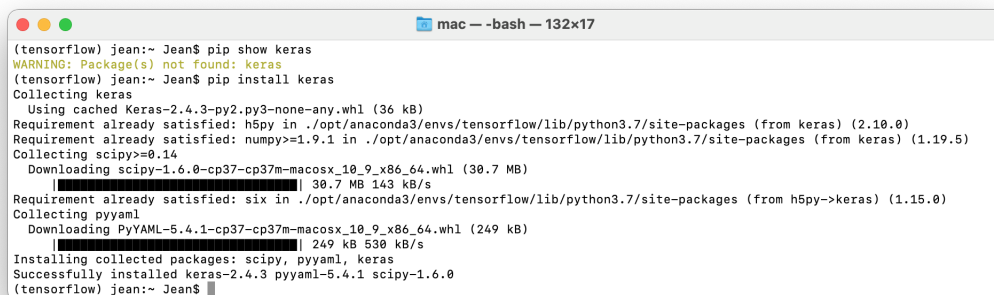
A basic tensor was created and printed.

9 Keras

9.1 Overview

Keras is a simple, clean and easy way to implement machine-learning based on the tensorflow package. It's main aim is to simplify the creation, training and learning of machine learning algorithms coded in python that use tensorflow as it's base. The need for keras arises from the complex and hard to understand nature of tensorflow which keras simplifies allowing for quick and simple implementation of machine learning algorithms.

9.2 Practical Example



```
(tensorflow) jean:~ Jean$ pip show keras
WARNING: Package(s) not found: keras
(tensorflow) jean:~ Jean$ pip install keras
Collecting keras
  Using cached Keras-2.4.3-py2.py3-none-any.whl (36 kB)
Requirement already satisfied: h5py in ./opt/anaconda3/envs/tensorflow/lib/python3.7/site-packages (from keras) (2.10.0)
Requirement already satisfied: numpy>=1.9.1 in ./opt/anaconda3/envs/tensorflow/lib/python3.7/site-packages (from keras) (1.19.5)
Collecting scipy>=0.14
  Downloading scipy-1.6.0-cp37-cp37m-macosx_10_9_x86_64.whl (30.7 MB)
    |#####| 30.7 MB 143 kB/s
Requirement already satisfied: six in ./opt/anaconda3/envs/tensorflow/lib/python3.7/site-packages (from h5py->keras) (1.15.0)
Collecting pyyaml
  Downloading PyYAML-5.4.1-cp37-cp37m-macosx_10_9_x86_64.whl (249 kB)
    |#####| 249 kB 530 kB/s
Installing collected packages: scipy, pyyaml, keras
Successfully installed keras-2.4.3 pyyaml-5.4.1 scipy-1.6.0
(tensorflow) jean:~ Jean$
```

Figure 5: keras installation into conda using pip command

10 NumPy

10.1 Overview

NumPy is a python library that implements a number of complex math related data structures and its basic operations. For the focus of machine learning NumPy implements large multi-dimensional array and matrices and their respective operations which are dependencies of tensorflow thus making it essential for the use in machine-learning.

10.2 Practical Example

The Ebola data set was read and cleaned.

```
In [51]: # One of the use of numpy is in data cleaning. Here I will use numpy to remove all the
# empty columns and replace them with nan
data = read_csv('ebola_dataset.csv')
replace_data = data.replace(r'^\s*$', np.nan, regex=True)
```

Figure 6: The Numpy screen-shoot

11 SciPy

11.1 Overview

SciPy is open-source python package used for scientific and technical computing. It focuses on optimization, linear algebra, integration, interpolation and other special functions. It builds on NumPy data structures and operations and further extends the ability of data-science engineers to integrate mathematical functions into machine-learning.

12 Matplotlib

12.1 Overview

The Matplotlib package provides production level visualisation capabilities for python programmers. It enable them to produce high quality graphs and plots to demonstrate pragmatically enumerated data-sets in an easily disgust-able medium.

12.2 Practical Example

```
In [53]: # Sklearn is used for data preprocessing. I will use a combination of both
# matplotlib and sklearn to convert digits into a hand written images.
from sklearn.datasets import load_digits
my_digits = load_digits()
my_digits.images.shape
my_digits.images[0]

# Now we import matplotlib to draw the graph which is going to hold our array of numbers
import matplotlib.pyplot as plt
fig, axes = plt.subplots(5,5, figsize=(4,4))
for i,ax in enumerate(axes.flat):
    ax.imshow(my_digits.images[i])
```

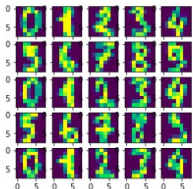


Figure 7: The Matplotlib screen-shoot 3

A Plot was created which demonstrates the various formatting options provided by Matplotlib

13 Pandas

13.1 Overview

Pandas expands and simplifies the use of the data-structures of Numpy, while also including many data-analysis functions. Another tool to simplify and expand the ease of use, data-analysis and algorithm creation for data-scientists.

13.2 Practical Example

```
In [47]: print(dataset.describe())
```

Cumulative no. of confirmed, probable and suspected cases \	
count	2477.000000
mean	2553.678644
std	4427.118148
min	0.000000
25%	1.000000
50%	8.000000
75%	3657.000000
max	14122.000000

Cumulative no. of confirmed, probable and suspected deaths	
count	2485.000000
mean	1028.347686
std	1656.064372
min	0.000000
25%	0.000000
50%	6.000000
75%	2386.000000
max	4806.000000


```
In [48]: print(dataset.groupby('Country').size())
```

Country	259
Guinea	141
Italy	365
Liberia	243
Mali	255
Nigeria	254
Senegal	259
Sierra Leone	243
Spain	221
United Kingdom	245
United States of America	
dtype: int64	

Figure 8: The Pandas screen-shoot

Pandas was used to calculate the descriptive data for a data set.

14 Scikit-Learn

14.1 Overview

Scikit-Learn is a package that expands the predictive-analysis of the data-scientist being based on the NumPy, SciPy, and matplotlib packages, it expands and further simplifies their functionality.

14.2 Practical Example

15 Conclusion

Basically, this was just an introduction to how we went from installing these libraries to actually using them for various use cases. They have more deep


```
mac — -bash — 140x26
(tensorflow) jean:~ Jean$ pip show sklearn
WARNING: Package(s) not found: sklearn
(tensorflow) jean:~ Jean$ pip install sklearn
Collecting sklearn
  Downloading sklearn-0.0.tar.gz (1.1 kB)
Collecting scikit-learn
  Downloading scikit_learn-0.24.1-cp37-cp37m-macosx_10_13_x86_64.whl (7.2 MB)
    |██████████████████████████████| 7.2 MB 997 kB/s
Requirement already satisfied: scipy>=0.19.1 in ./opt/anaconda3/envs/tensorflow/lib/python3.7/site-packages (from scikit-learn->sklearn) (1.6.0)
Requirement already satisfied: numpy>=1.13.3 in ./opt/anaconda3/envs/tensorflow/lib/python3.7/site-packages (from scikit-learn->sklearn) (1.19.5)
Collecting joblib>=0.11
  Downloading joblib-1.0.0-py3-none-any.whl (302 kB)
    |██████████████████████████████| 302 kB 421 kB/s
Collecting threadpoolctl>=2.0.0
  Downloading threadpoolctl-2.1.0-py3-none-any.whl (12 kB)
Building wheels for collected packages: sklearn
  Building wheel for sklearn (setup.py) ... done
  Created wheel for sklearn: filename=sklearn-0.0-py2.py3-none-any.whl size=1316 sha256=c7ace0a1da2b81024cda79630fd16c2d01bdfc852bdec884e41067d2f2435aa1
  Stored in directory: /Users/mac/Library/Caches/pip/wheels/46/ef/c3/157e41f5ee1372d1be90b09f74f82b10e391eaacca8f22d33e
Successfully built sklearn
Installing collected packages: threadpoolctl, joblib, scikit-learn, sklearn
Successfully installed joblib-1.0.0 scikit-learn-0.24.1 sklearn-0.0 threadpoolctl-2.1.0
(tensorflow) jean:~ Jean$
```

Figure 9: sklearn

application in Machine Learning and Artificial Learning that we are going to cover in this course.

References

- <https://docs.anaconda.com/anacondaorg/faq/what-is-anaconda-inc>
- <https://numpy.org/doc/stable/user/whatisnumpy.html>
- https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html
- https://www.w3schools.com/python/scipy_intro.asp
- https://www.tutorialspoint.com/keras/keras_introduction.htm
- <https://realpython.com/python-matplotlib-guide/>