Tools for Deep Learning
Part 1

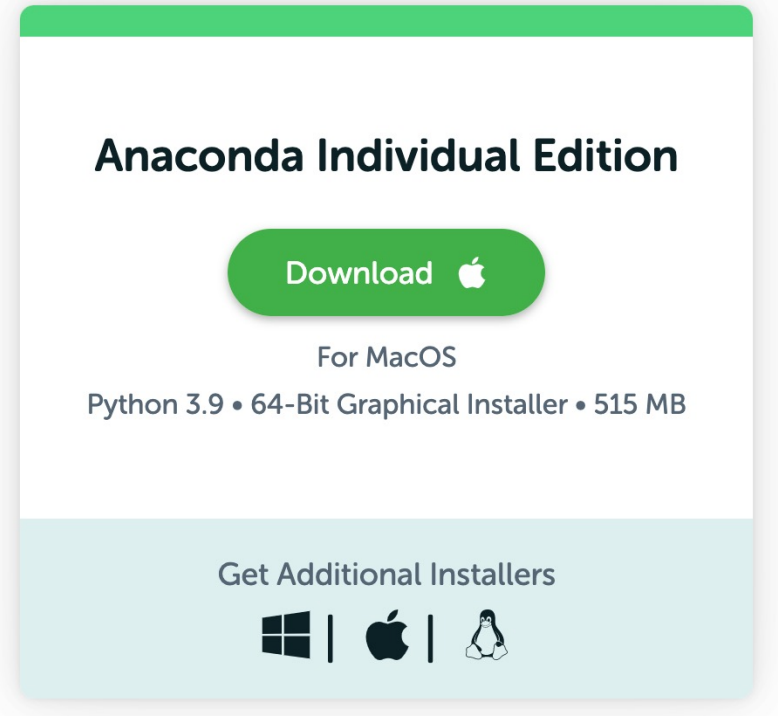
Rowel Atienza, PhD
University of the Philippines
github.com/roatienza
2022

Outline

- Environment, Code Editor, Python
- Tensor library – numpy and einops
- Timm
- Huggingface (HF)
- Gradio
- Streamlit
- PyTorch, HF Accelerator, GitHub
- Machines – Colab, DeepNote, Kaggle, SageMaker
- Other tools

Container Environment

Anaconda



Venv

Unix/macOS Windows

```
python3 -m pip install --user virtualenv
```

Container Environment

Anaconda

```
conda create --name dl_course
```

Venv

```
python3 -m venv dl_course
```

Container Environment

Anaconda

```
conda activate dl_course
```

Venv

```
source dl_course/bin/activate
```

Python package installer

`pip3 or pip`

Example:

```
pip3 install torch torchvision torchaudio
```

Anaconda – Machine Learning Toolkit

https://www.anaconda.com/

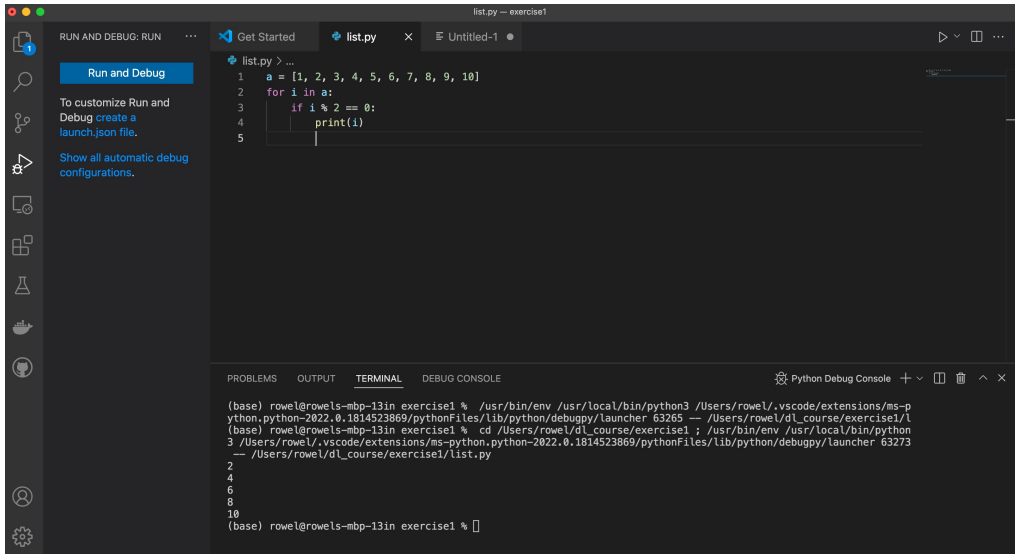
`venv` — lightweight virtual environment

```
pip3 install virtualenv
python3 -m venv dl_course
source dl_course/bin/activate
which python3
/Users/rowel/dl_course/bin/python3
which pip3
/Users/rowel/dl_course/bin/pip3
```

Code Editor

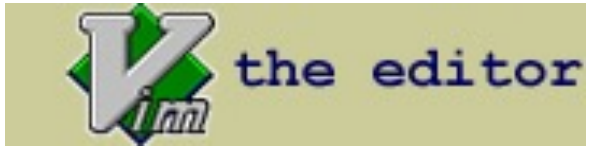
Text Editor / IDE

- Visual Studio Code



- Recommended for its features

- `vim`



- Recommended for its portability

Python

https://github.com/dabeaz-course/practical-python

Python

- Scripting interpreted language
- Exercise: activate python on your terminal
- Exercise: create a new python source file in vscode

Numbers

- No need to declare the data type but common data types are supported : Boolean to complex numbers
- Exercise:
  - Generate 10 integers. Store in a list. Print.
  - Print the min and max
  - Print in ascending order
- Supports data type cast like in C
- Exercise:
  - Generate 10 floats. Store in a list. Print.
  - Convert all floats to int. Print.

Strings

- Declared using single or double quotes

```
name = "deep learning is fun"
```

- Can be indexed

```
print(name[5:])
```

- Can be concatenated

```
print(name + "!")
```

- Supports string manipulation

```
print(name.replace("deep", "machine"))
```

- Search

```
print("learn" in name)
```

- String functions

```
print(name.upper())
```

None

- None is used as a placeholder for unsure or missing data type or value

```
email_address = None
```

List

- A **list** is a data structure that is a mutable, or changeable, ordered sequence of elements
- Collection of values possibly of different data types

```
x = [1, "fox", 3.4, [8, 16]]
```

- Indexed

```
print(x[1])
```

- Concatenate

```
y = [1, 2, 3, 4, 5]
z = [1, 4, 9, 16, 25, 36]
y + z
```

List - Slicing

`y[start:end:interval]`

```
y[0:4:2]
y[::3]
y[::-1]
```

Loops

- for

```
>>> x = [1, "fox", 3.4, [8, 16]]
>>>
>>> for i in x:
...     print(i)
...
1
fox
3.4
[8, 16]
```

- while

```
>>> i = 0
>>> while i < len(x):
...     print(x[i])
...     i += 1
...
1
fox
3.4
[8, 16]
```

Function

- We use the `def` keyword to define a function
- A function has 0 or more input. Same with output.
- Example: given a list of integers, get all even integers, store in a new list and print

```
y = [8, 1, 4, 2, 0, 7, 5, 6, 3]
def filter_even(x):
    result = []
    for i in x:
        if i % 2 == 0:
            result.append(i)
    return result

print(filter_even(y))
```

Object Oriented

- Class and inheritance
- Methods and properties

```
class Person:
    def __init__(self, name, age):
        self.name = name
        self.age = age

    def __str__(self):
        return f"{self.name} is {self.age} years old."


x = Person("John", 30)
print(x)
```

Object Oriented - PyTorch

- Our deep learning models will be build using OO techniques

```
import torch

class GNet(torch.nn.Module):
    def __init__(self, mean=0., std=1.):
        super(GNet, self).__init__()
        self.mean = torch.Tensor([mean])
        self.std = torch.Tensor([std])

    def forward(self, x):
        return x*torch.normal(mean=self.mean, std=self.std)

x = GNet()
print(x(3))
```

Reference

- Practical python https://github.com/dabeaz-course/practical-python
- https://www.anaconda.com/
- https://pytorch.org/
- https://www.python.org/

End