

## Keras Installation (Linux,Mac)

1. 打開終端機(terminal) 安裝 python 基礎開發包，輸入：

```
sudo apt update  
sudo apt upgrade  
sudo apt install -y python-dev python-pip python-nose gcc g++ git
```

2. 安裝運算加速庫

```
sudo apt install -y libopenblas-dev liblapack-dev libatlas-base-dev
```

3. 安裝 numpy, matplotlib, scikit-learn, theano, Keras

```
sudo pip install numpy scipy matplotlib scikit-learn scikit-image  
sudo pip install theano  
sudo pip install keras
```

4. 安裝完成後，在 terminal 上輸入 python 並且載入安裝包：

```
import theano  
import keras  
// 若沒有任何問題出現，即代表安裝完成
```

5. 可下載這份範例程式並執行

```
https://github.com/fchollet/keras/blob/master/examples/mnist\_mlp.py  
python mnist_mlp.py
```

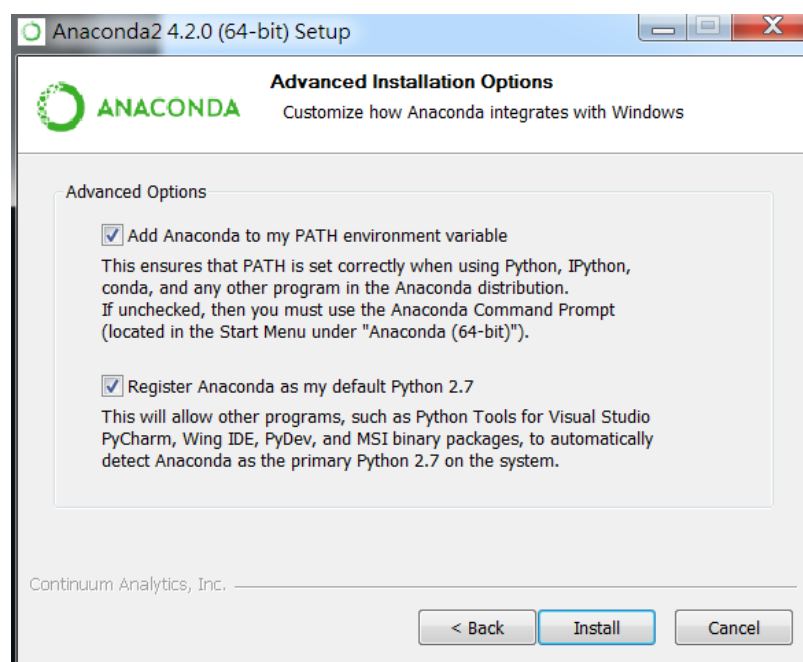
=====

## Keras Installation (Windows)

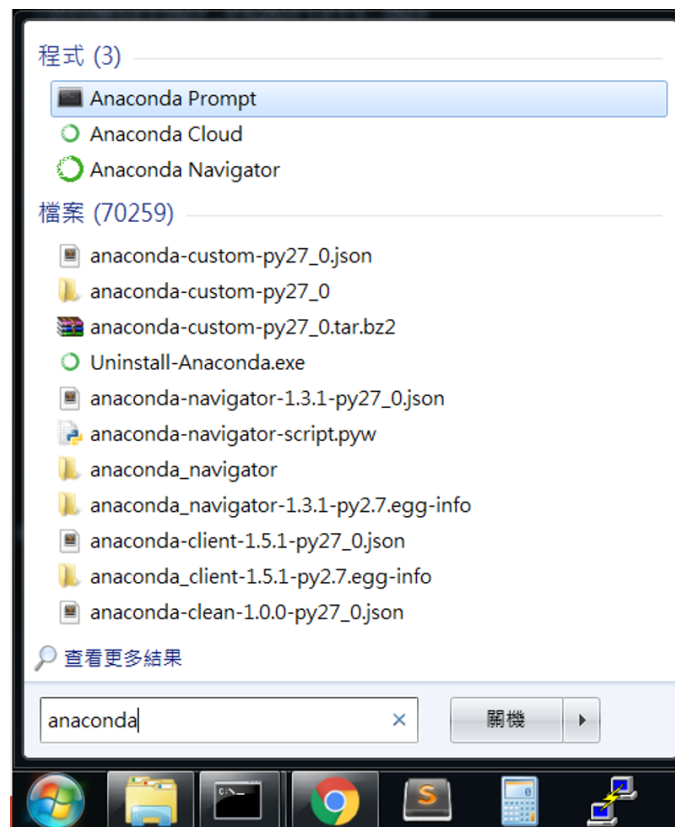
0. 即使電腦已有安裝 python，但仍建議各位安裝 anaconda，並使用 anaconda prompt 安裝套件與進行課程。

1. 下載 [Anaconda](#)，請選用安裝 Python 2.7 版本。Anaconda 本身就有 Python，事前無需再安裝 Python。

2. 安裝 Anaconda，勾選Add anaconda to my PATH environment variable。

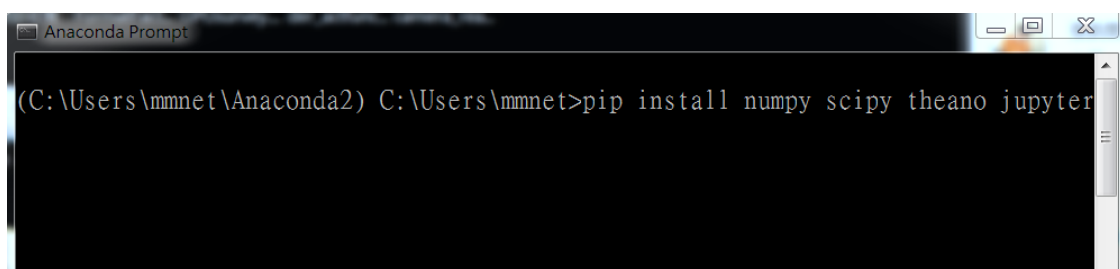


3. 添加環境變數 PYTHONPATH :  
`C:\Anaconda2\Lib\site-packages\theano;`  
請使用自己安裝 Anaconda 的路徑，在此範例為 `C:\Anaconda2\`。
4. 打開 cmd 並輸入：  
`conda install mingw libpython`
5. 在 cmd 的主目錄下，新增一份 `.theanorc.txt`，裡面請輸入  
`[global]`  
`openmp=False`  
`[blas]`  
`ldflags=`  
`[gcc]`  
`cxxflags = -I C:\Anaconda2\MinGW`
6. 添加環境變數 PATH  
`C:\Anaconda2\MinGW\bin;C:\Anaconda2\MinGW\x86_64-w64-mingw32\lib`
7. 搜尋 anaconda prompt 並開啓



輸入：

`pip install numpy scipy theano jupyter`



8. 進入 python 並輸入

```
import theano
```

若沒有錯誤就是 theano 安裝成功了！

9. 繼續在 anaconda prompt 中，輸入：

```
pip install keras
```

10. 進入 python 並輸入：

```
import keras
```

若沒有錯誤就是 keras 安裝成功了！

11. 可下載這份範例程式並執行

[https://github.com/fchollet/keras/blob/master/examples/mnist\\_mlp.py](https://github.com/fchollet/keras/blob/master/examples/mnist_mlp.py)  
`python mnist_mlp.py`

## # 常見問題

Q1: 如何修改 keras backend 使用 theano 或 tensorflow?

A1: 用任一文字編輯器開啓 keras.json

[Linux or Mac] 在 ~/.keras/keras.json

[Windows] 使用者主目錄下，如 C:\Users\mmnet\

將 backend 改成 theano 或 tensorflow 即可

```
{  
  "image_dim_ordering": "tf",  
  "epsilon": 1e-07,  
  "floatx": "float32",  
  "backend": "theano"  
}
```

```
{  
  "image_dim_ordering": "tf",  
  "epsilon": 1e-07,  
  "floatx": "float32",  
  "backend": "tensorflow"  
}
```

請確定 “image\_dim\_ordering” 為 “tf” ！