**Homework 1 – part C**

環境建置：

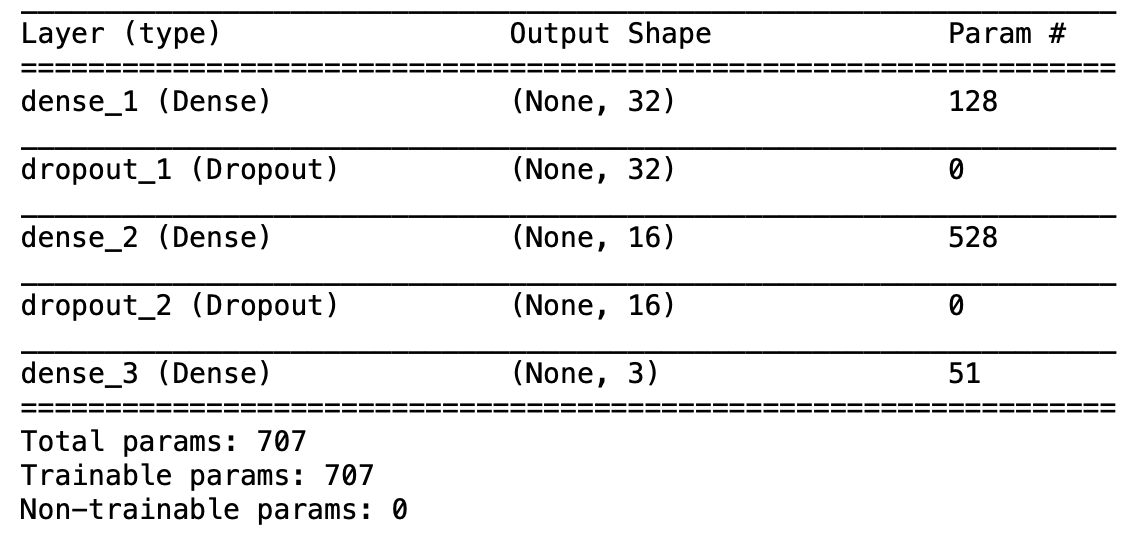
使用jupyter notebook 和 Visual Studio Code。

組員：

B10615056 黃暉翔 B10615045 陳尚富 B10615046 柯元豪

1. .

這邊使用hw1-partB收集到的資料，來對其使用deep learning method（multiple hidden layer）。Model summary:



Input layer: Activation function使用relu，input\_dim為3因為只有三個特徵。

Hidden layer: 使用了兩層的dropout和一層dense layer使用relu作為activation function。

Output layer: dense layer，因為是分類所以選用softmax作為輸出層的activation function。

**方法流程：**

先將收集到分散的各種類csv結合成一個csv file。但是由於label為字串，若是直接轉為數字對於Ann或是deep learning而言，會更像是二分類法或是regression，所以要先將label轉為category（類似one hot encoding）。

  再來應用sliding window將所有的data取平均值，window size這邊定義為100。然後再將一些nan的資料砍掉，前處理就完成了。再來只需要套用上面介紹的deep learning model就完成了。

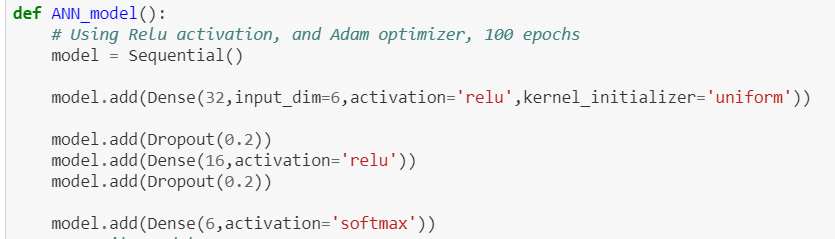
1. .

由於是將(A)的實作方式，套用在kaggle上的資料，因此只在底下列出不同處。

* Attribute不只有加速度的X,Y,Z，而act也不只三種，一樣不需要的drop掉



* 在ANN的部分dim也要做對應的更改，因為特徵變成六種



其餘步驟與實作方法皆與(A)相同

1. .

先將subject.csv 和 device motion data 結合做出time series的資料，依照每個人的資料做出24個csv檔，讀取前20個當training data，用rolling函數做sliding windows後使用ann model訓練。

讀取後面4個一樣做sliding windows 當testing data 後驗證模型準確度。

Epoch 46/50

1176364/1176364 [==============================] - 2s 2us/step - loss: 0.7627 - accuracy: 0.7084

Epoch 47/50

1176364/1176364 [==============================] - 2s 2us/step - loss: 0.7603 - accuracy: 0.7096

Epoch 48/50

1176364/1176364 [==============================] - 2s 2us/step - loss: 0.7593 - accuracy: 0.7098

Epoch 49/50

1176364/1176364 [==============================] - 2s 2us/step - loss: 0.7564 - accuracy: 0.7112

Epoch 50/50

1176364/1176364 [==============================] - 2s 2us/step - loss: 0.7544 - accuracy: 0.7121

229325/229325 [==============================] - 0s 1us/step

training 準確率: 0.640872124713834