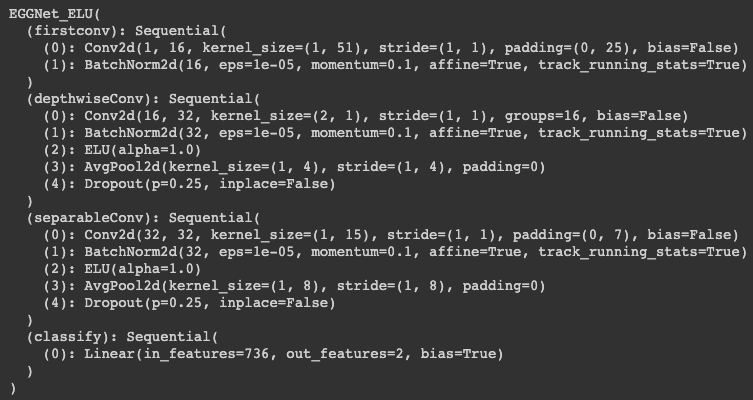
Introduction:

這次作業要對 EEG (Electroencephalography)圖進行二元分類，腦波圖（EEG）是一種腦電波的紀錄方法，放置電極於頭皮記錄腦神經元的電壓變化。而相同動作或想法，可能會有相似的pattern，因此如果可以依據腦波圖來分析受試者可能的相應動作，便可以依此做出應對與回饋。此作業便是要利用CNN網路，分析其中的pattern進而再做出分類。

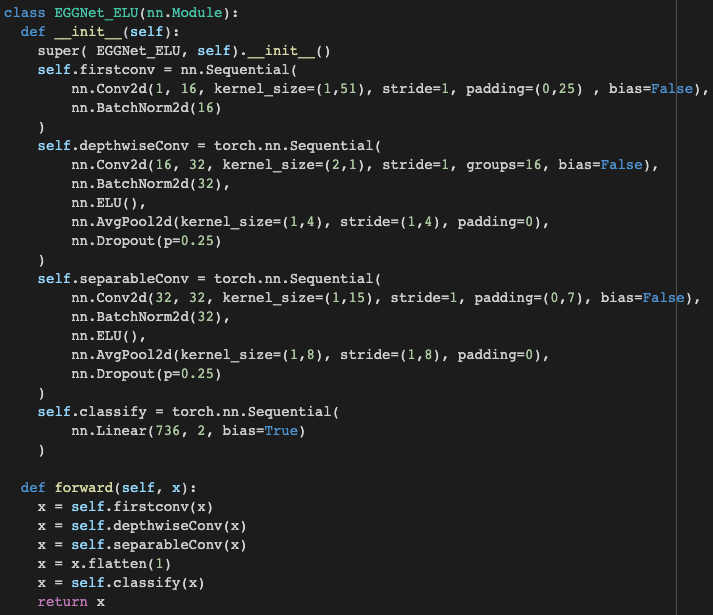
而且CNN (convolutional neural network、卷積神經網路)，在深度學習中，擅長於處理圖片的資訊，層與層之間會有filter提取相關特徵，filter會是一個矩形，通常會是正方形，但也可以根據你對問題的了解（也就是所謂的domain knowledge），自行定義認為有助於的提取特徵的形狀。

而filter向右或向下移動的步伐稱為stride，而全部的filter掃出來的結果即為下一層的layer。此外如果filter超出layer的長與寬可以利用padding決定是否要填補。除了有一般的卷積層其中可以還可以安插batch normalization 將參數調整以致於易於訓練。也可以加入激活函數，或是pool層對資料做處理。重複疊加幾層，最後再連結全連接層，轉乘所需要的class數量，就可以利用CNN + fully connected 處理圖片分類的問題。

Experiment set up:

EEGNet:

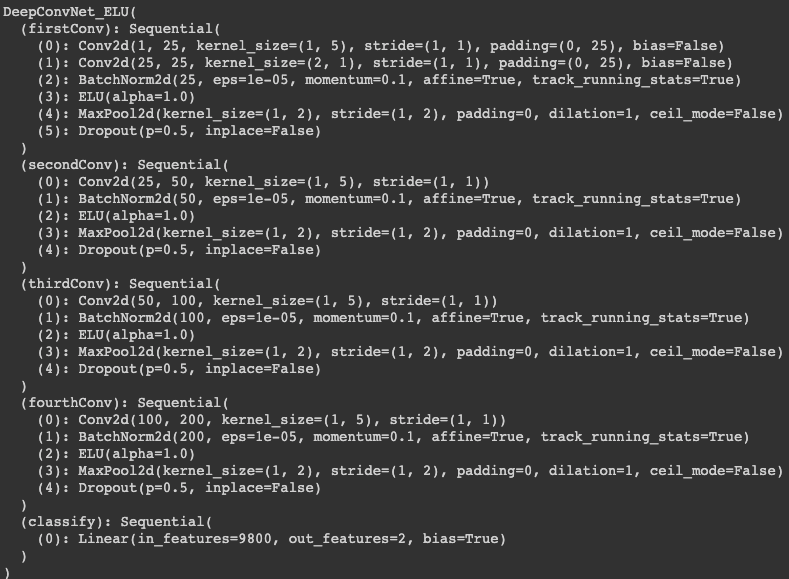
圖一、EEG模型架構．



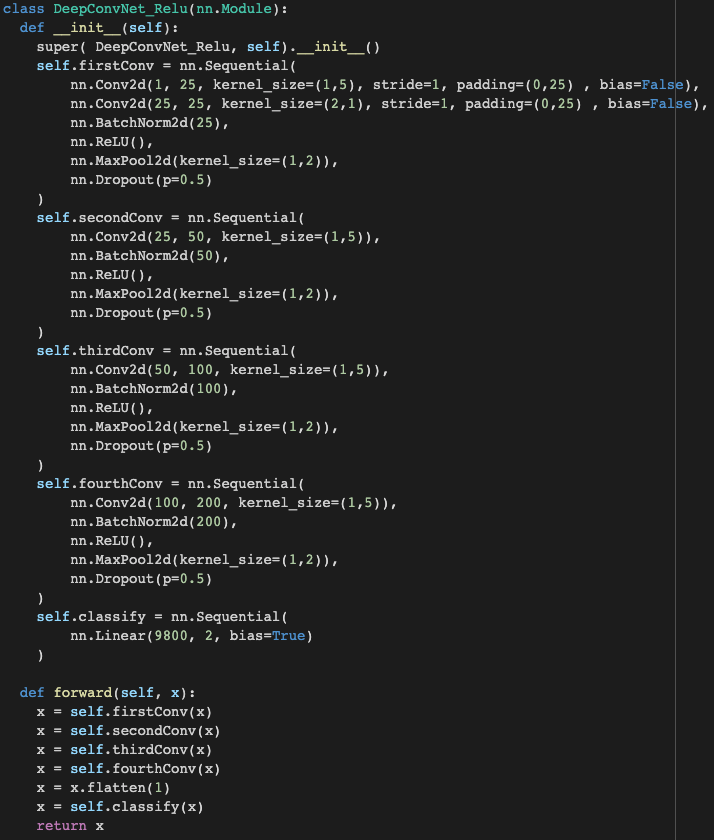
圖二、EEG實作細節。

EEG網路主要可以看成四層，第一層只有簡單的卷積加上batchnorm，第二、三層則是多加上激活函數、pooling、Dropout，最後則接上linear fully connected。最後定義整個網路的forward，整個網路就算是定義好了。在這邊直得注意的是，kernel size也就是所謂的filter不是正方形的而是 1 by 51或是 2 by 1，這種設計即是對問題也就是EEG腦波圖有更深的了解，這樣的觀察認為有助於模型能夠提取有用的特徵。

DeepConvNet:



圖三、DeepConvNet模型架構。



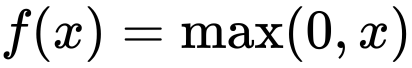
圖四、DeepConvNet實作細節。

而在DeepConvNet中，前四層基本上架構相同，而且也沒有什麼特別的地方，沒有domain knowledge 的觀察，看起來就是利用Deep CNN單純的解決問題。

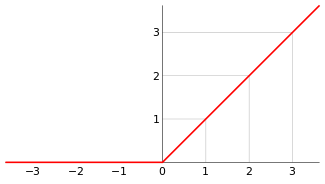
Explain the activation function ( ReLU , Leaky ReLU , ELU ) :

ReLU :

ReLU作為DL中最基本的也做常見的激活函數其公式及圖如下分別為圖五及圖六。



圖五、ReLU公式。



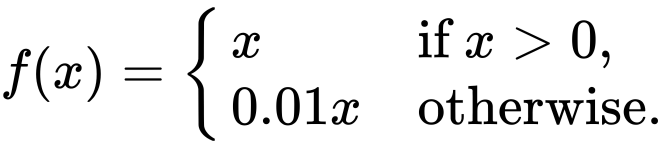
圖六。

在輸入大於零時，及輸出完整的輸出。小於零時，輸出為零。這樣最大的好處是其斜率為１很好計算。其函數相較sigmoid也更符合生物神經網路的表現，有一定的生物學原理。而通常也有較好的表現。而且也有較好的gradient back propagation表現，較少發生vanishing gradient （梯度消失）的情況。

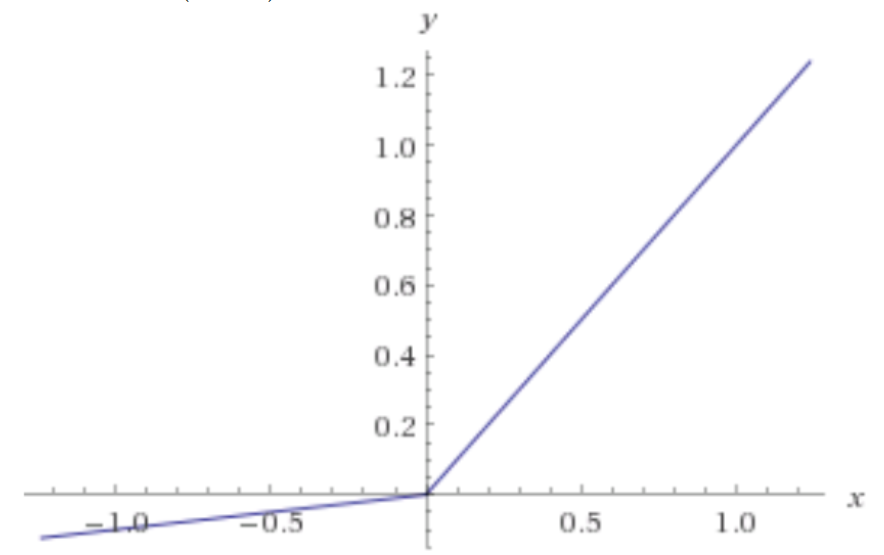
而ReLU也有眾多的變形，其主要變化大多在於零之前的，試圖解決當輸入小於零時，輸出為零梯度為零，不易於反向更新網路參數。例如Leaky ReLU、Parametric ReLU、ELU 。

Leaky ReLU :

Leaky ReLU作為ReLU的變形，其公式及圖分別為圖七、圖八。



圖七，Leaky ReLU公式。



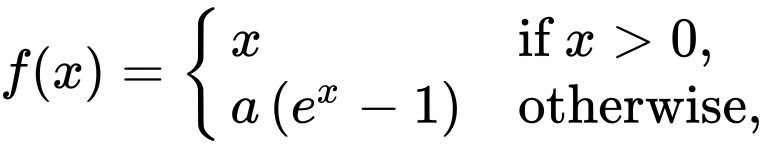
圖八。

Leaky ReLU作為ReLU的改良，將小於零時輸出的定義，改為輸出相對於輸入較小的比例，這樣小小的修改就可以對左側的情況也就是小於零時，有較小的gradient而不為零。對於神經網路節點來說，此節點不再是沒有反應，而是也有少許的輸出。

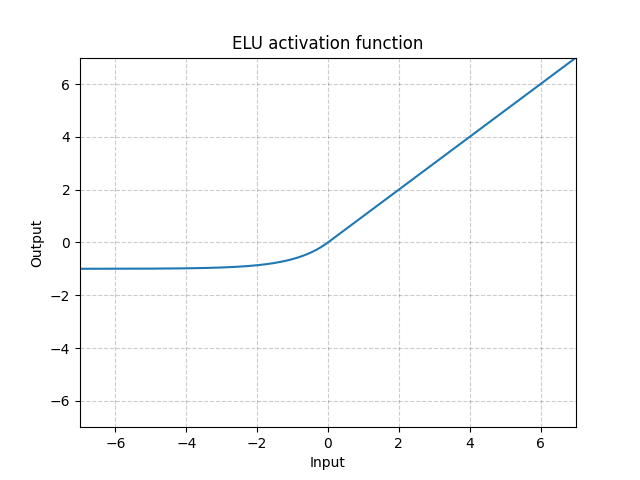
常用於可能會遭遇到sparse gradients的模型上，例如生成對抗網路 (Generative Adversarial Networks, GAN)。

ELU:

ELU作為ReLU的變形，其公式及圖分別為圖九、圖十。



圖九、ELU公式。



圖十。

Exponential linear units (ELU)，一樣作為ReLU的改型，也是對左側，也就是小於零時作修改。而其頃向將輸出收斂於零來產生更好的結果。而ELU有以下幾點優點

1. 當輸出為負時，輸出較平滑，而不是ReLU那麼尖銳。
2. 強於ReLU
3. 相對於ReLU，ELU可以產生負值。

而ELdU也已被證明相較於ReLU來說ELU對分類問題（classification）有更好的表現。

Experimental results:

The highest testing accuracy:

DeepConvNet\_Relu: test\_set acc = 0.86175

epochs=300 optimizer=Adam lr=1e-3 batch\_size=1024

DeepConvNet\_LeakyReLU: test\_set acc = 0.87116

epochs=300 optimizer=Adam lr=1e-3 batch\_size=1024

DeepConvNet\_ELU: test\_set acc = 0.84535

epochs=300 optimizer=Adam lr=1e-3 batch\_size=1024

EGGNet\_ReLU: test\_set acc = 0.89265

epochs=300 optimizer=Adam lr=1e-3 batch\_size=1024

EGGNet\_LeakyReLU: test\_set acc = 0.87626

epochs=300 optimizer=Adam lr=1e-3 batch\_size=1024

EGGNet\_ELU: test\_set acc = 0.87395

epochs=300 optimizer=Adam lr=1e-3 batch\_size=1024

上述的結果皆為無數次嘗試的結果，其中有嘗試不同的optimizer分別為Adam、SGDM、RSMprop，表現如同上述順序。

發現不同的batch\_size所造成的影響：

Batch較小時，更新較積極，且一整個data set中一次可以更新的次數較多。

Batch較大時，更新較穩重，且一整個data set 中一次可以更新的次數較少。

最後選擇較大的Batch作為訓練策略。

此問題在眾多嘗試中，任然無法將test\_acc拉到接近95%，因為我認為一方面為對此問題的了解不夠透徹，沒有關於EEG的domain knowledge 。

給予用於訓練的資料也偏少再者因其EEG的資料特性也不適合做data augmentation。倘若給予更多的資料甚至是沒有labels的也可以利用semi supervised learning對沒有labels的資料給予pseudo labels來增加訓練資料。

Comparison figures:



橘色為DeepConvNet\_Relu、

深藍為DeepConvNet\_LeakyReLU、

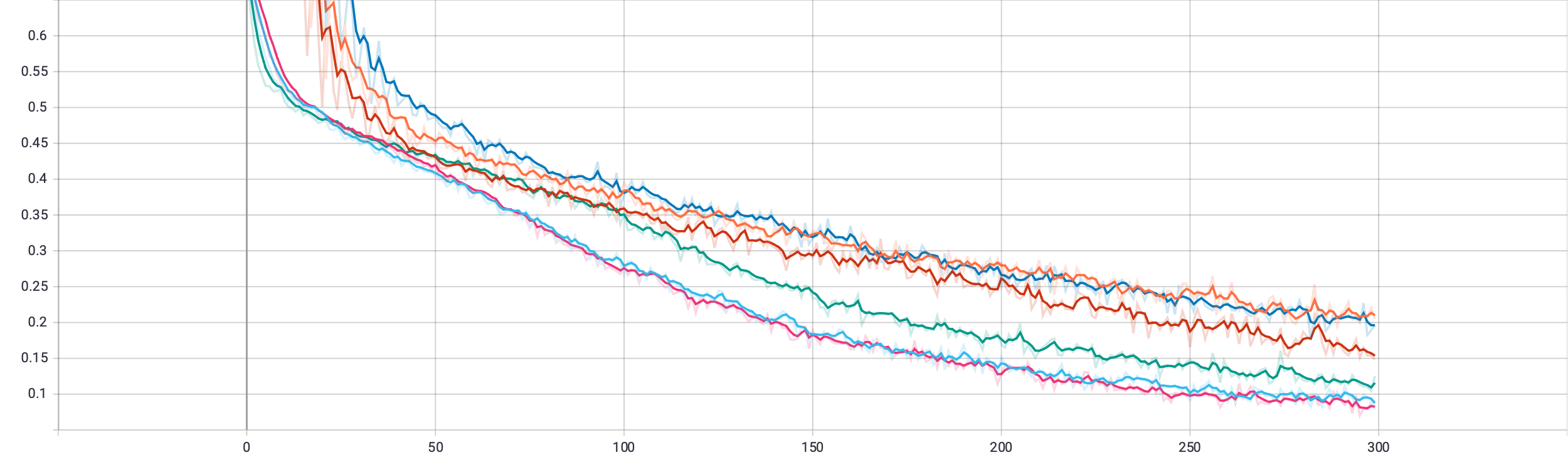
紅色為DeepConvNet\_ELU、

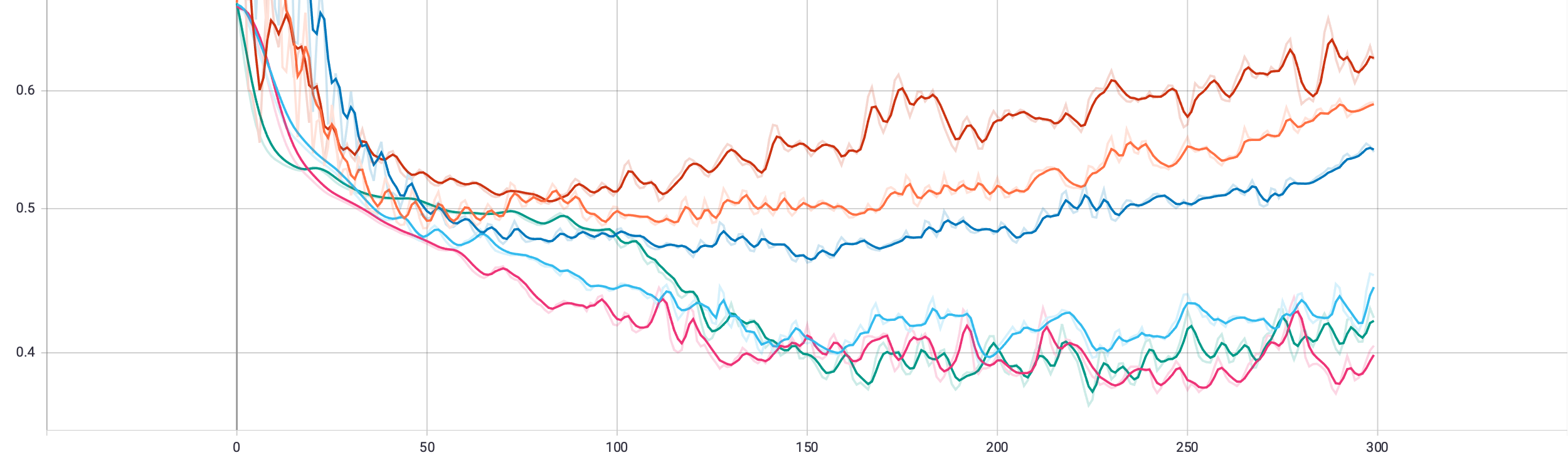
淡藍為EGGNet\_ReLU、

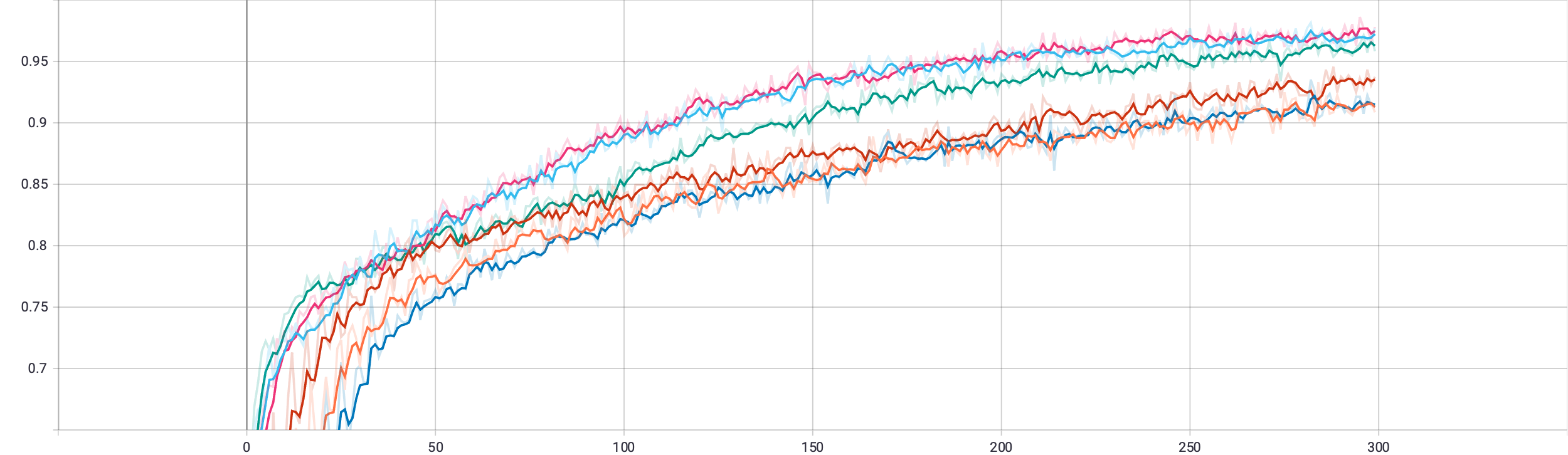
粉紅為EGGNet\_LeakyReLU、

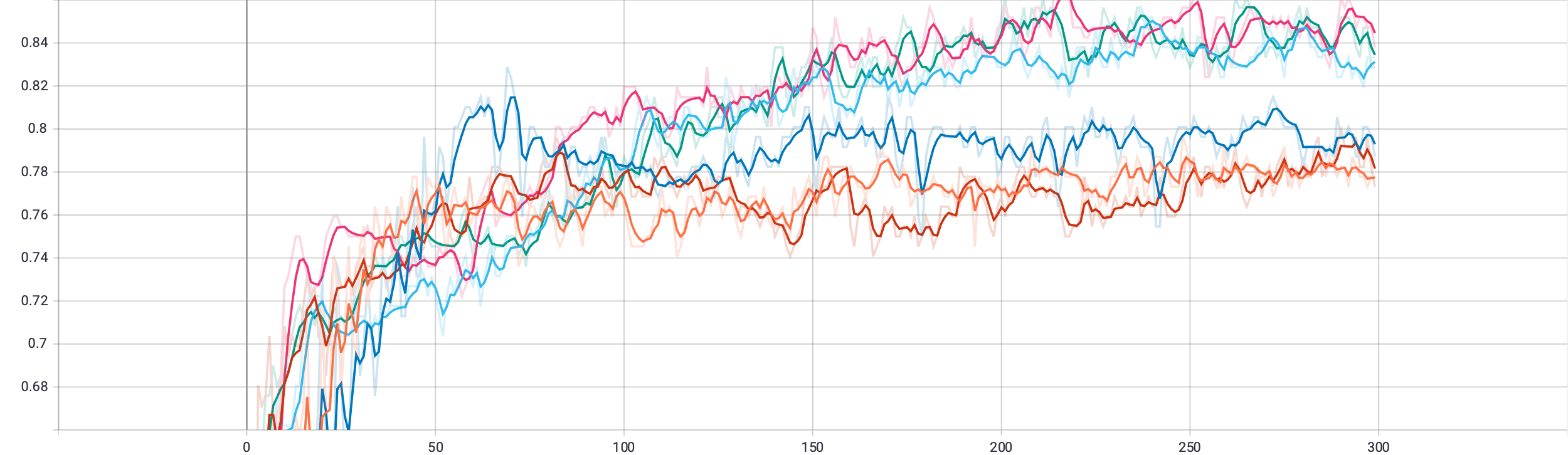
墨綠為EGGNet\_ELU。

圖十一、Tensorboard數據資料圖。



圖十二、loss\_train。圖十三、loss\_vaild。



圖十四、accuracy\_train。

圖十五、accuracy\_valid。

由上述圖可以得知EEGNet的表現較DeepConvNet好，不管是在loss或是accuracy中都是。而且在DeepConvNet中越訓練loss\_valid不降反升。猜測可能是overfitting，以至於在Valid中表現不好。EEGNet則是無法有效的收斂，loss震盪幅度巨大。

Discussion:

Anything you want to share:

發現tensorflow中的資料分析工具中的tensorboard竟然可以用在pytorch上面，而且相當簡單不複雜。

Pytorch相較於tensorflow真的相對簡單很多，學習成本低，Pytorch的文件也清楚明瞭。

在定義model的forward時，如果途中flow有錯，pytorch不會提出警告，而且也可以進行訓練，但內部可能真的沒有flow，但進行loss BP時參數不會update，模型表現很差，訓練不起來。

Colab白嫖搶不到GPU，所以我買了Colab Pro 10.88鎂。

可以將Google Drive mount在colab的VM上，之後所有資料都可以從Google Drive上存取。

Colab上可以直接將tensorboard顯示出來，訓練Model時可以即時的觀看loss\_train、loss\_valid的表現。

台大李宏毅的課程2021 Machine Learning 和其Homeworks對我的幫助很大，學到很多東西。