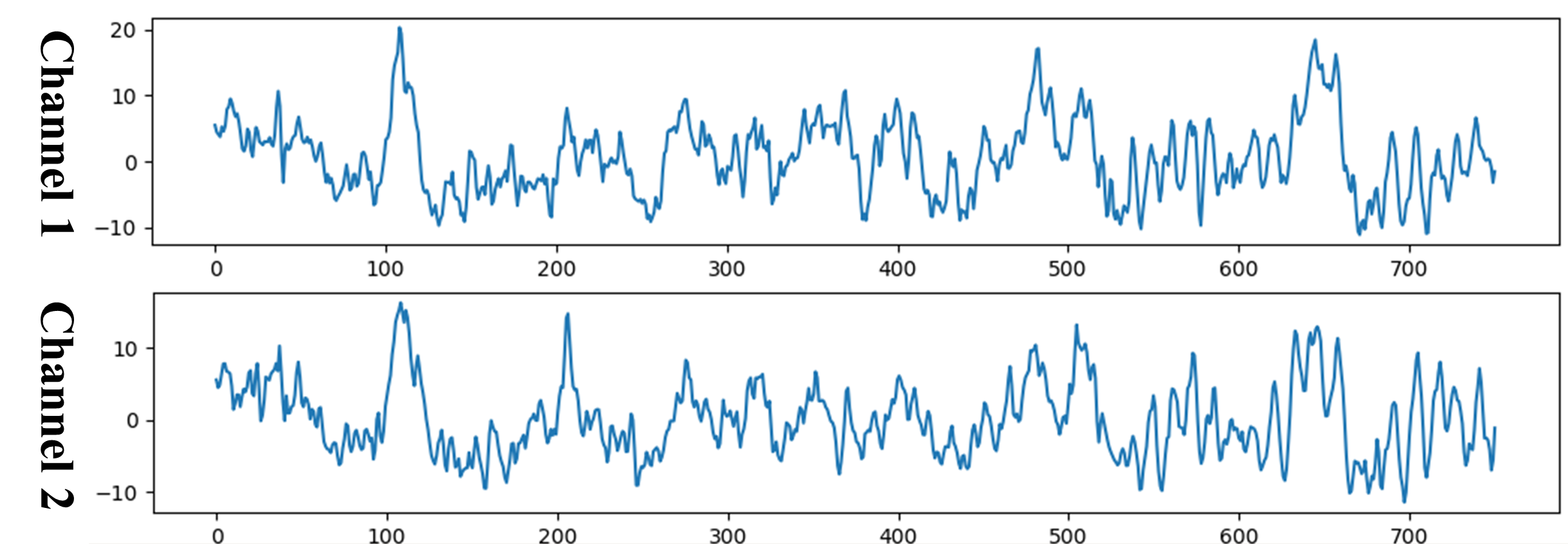
Lab2 : EEG classifications

311511043 李承翰

1. Introduction

在這次Lab，需要我們使用pytorch去實作EEGNet還有DeepConvNet兩種分類模型，用來分類BCI資料集，而這個資料集會有兩個channel，分別代表著左手與右手，而總共會分別有1080筆的訓練資料及測試資料

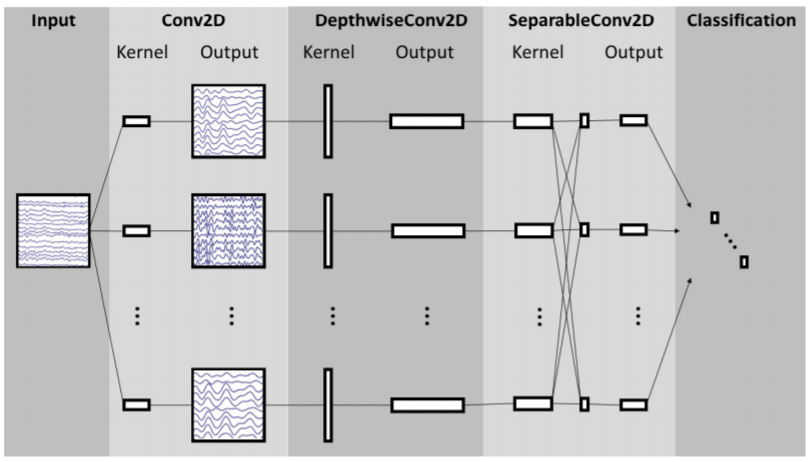


上圖為資料範例

此外，我們需要分別嘗試三種的activation function(ReLU, Leaky\_ReLU, ELU)，並且探討、比較實驗結果。

1. Experiment setup
2. The detail of models:

* EEGNet



EEGNet是一個小型的卷積神經網路，常用於腦電波類的資料，因為大幅降低了模型當中的參數數量，所以可以在有限的硬體當中還是有不錯的訓練速度。

而整個模型的架構如上圖所示，大致上可以分成三個convolutional layer，其設定如下:

一張含有 文字 的圖片

自動產生的描述

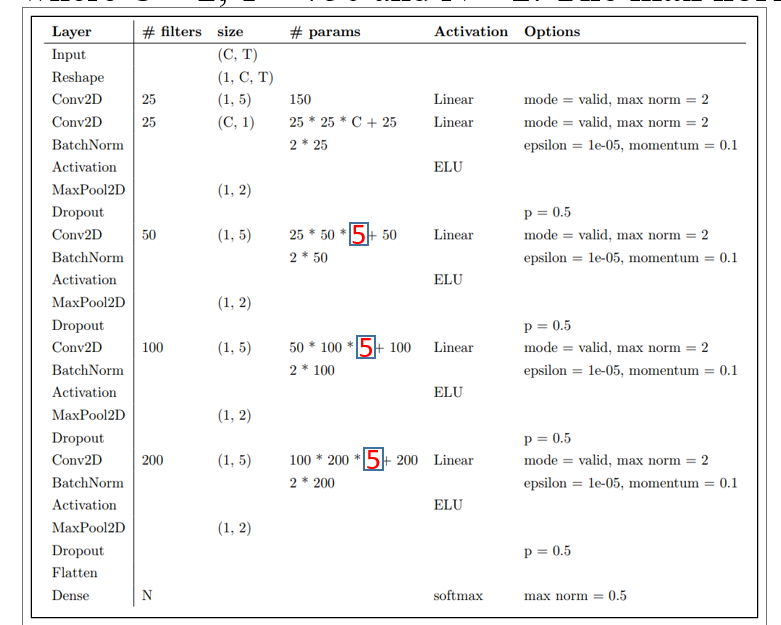
第一層的FistConv layer 負責的是將input 的資料以頻率的方式分成16個channel

第二層的 Depthwise layer 負責將第一層經過frequency filter的各個通道再經過spatial filter，而因為我們是直接將第一層的output分別當作第二層的input，讓使用上減少了參數數量，不過後果就是我們沒有辦法獲取不同頻率(通道)下同個空間的資料

而最後一層則是總結了第二層過來的資料，彌補了第二層的時候看不見不同頻率下相同空間的資料，將資料特徵都完整取出，交由下一層的linear classification來做判斷。

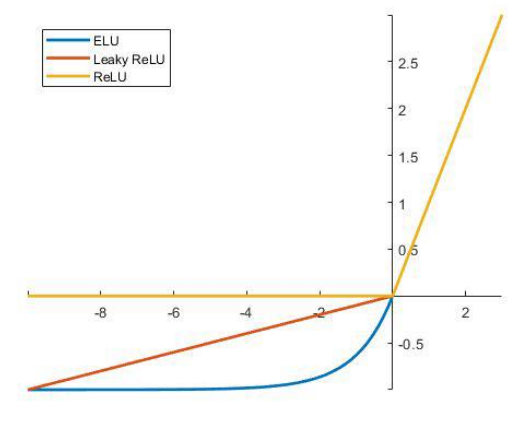
* DeepConvNet

DeepConvNet是傳統的深度學習架構，由一層捲積層，在後面再接上convolution、batch normalization、activation function、pooling、dropout，以這樣的順序重複4次，最後再用一個fully connected配合softmax進行分類，而每一層的參數設定如下圖: C = 2, T = 750, N = 2



1. Explain the activation function:

激勵函數的存在是為了讓神經網路模型可以判斷非線性的資料，如果沒有加上激勵函數，那麼整個神經網路的模型在判斷資料時就會出現困難。而這次我們分別要使用ReLU,Leaky\_ReLU,ELU三個激勵函數



* ReLU

f(x) = max(0,x), f’(x) =

ReLU函數是現在常用的激勵函數，因為他運算速度快，而且不會有梯度消失的問題存在。不過因為她在x<0的時候梯度為0，因此在某些情況下面有可能完全不激發，而這種問題被稱為dead ReLU問題。

* Leaky\_ReLU

f(x) = max(0,x) + min(0.001x,0) , f’(x) =

* ELU

1. Experimental results
2. Discussion

