ML2016 HW#3 Report

電機三 徐彥旻 b03901027

1. Supervised learning

使用 keras.preprocessing.image.ImageDataGenerator 從原本的訓練資料中產生更多的訓練資料（註解：使用的參數為 rotation range = 15, width shift range = height shift range = 0.1, horizontal flip = True，其他參數皆 default），使得在 kaggles 上的分數由未使用圖像資料生產的 0.48 上升至 0.61 （訓練至收斂，前者為 50 epochs，後者大約 200 epochs），表現有顯著的提升。validation 的表現請見圖一。

2. Semi-supervised learning(1)

使用由 Supervised learning 當中訓練好的模型，再加入 unlabel 做訓練，這邊使用的是作業說明投影片所提供的 “add a few most confident (x, f(x))” 之方法，加上與原本的 label 資料交替訓練，表現提升至 0.68 (kaggles public score) ；validation 的表現請見圖二。

（交替訓練過程的詳細說明至於第四點當中）

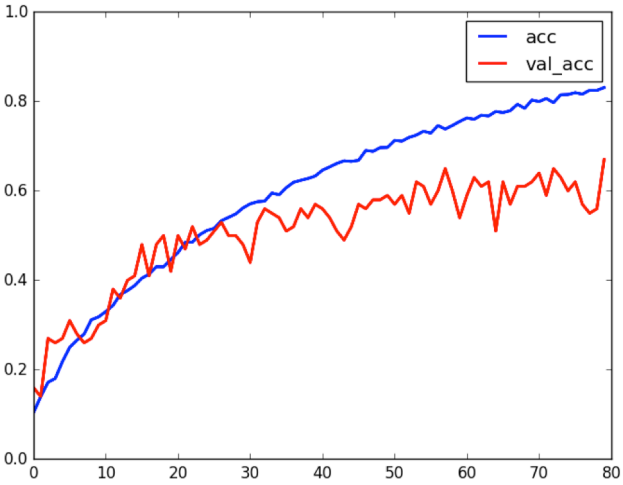
3. Semi-supervised learning(2)

使用 autoencoder 做 clustering ，訓練的資料為 label + unlabel + test 共六萬筆資料，得到 encoder 之後再將 label 資料轉換為 128 維的資料放進 DNN 做訓練，表現為 0.35 (kaggles public score)

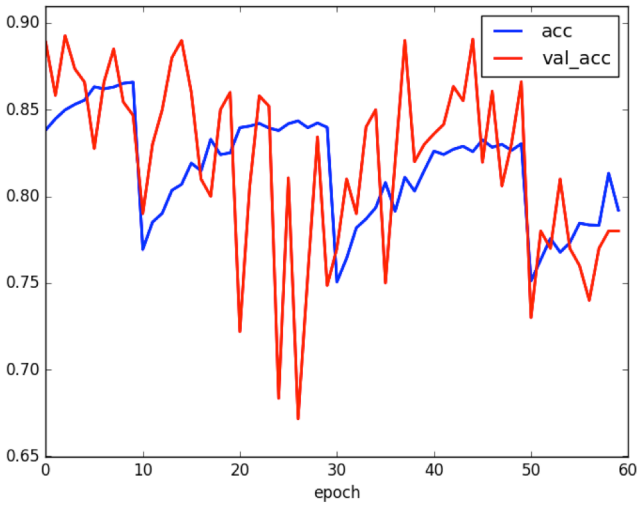
4. Result comparison and analysis

由圖二可以看出 semi-supervised learning(1) 在第二階段的訓練當中，acc 與 val_acc 皆呈週期波動（ acc 較為明顯），推測原因為訓練方式的設計是「用信任的資料訓練十次，在用一定正確的資料訓練十次，再重新預測未信任的資料，若特定種類的機率大於一定門檻，則加入信任的資料集當中，如此重複三次。」。不過值得一提的是，整體的正確率有輕微下降的趨勢，但是總體而言，semi-supervised learning(1) 都是比 supervised learning 表現要來得好的，也是本次作業當中最成功的部分。

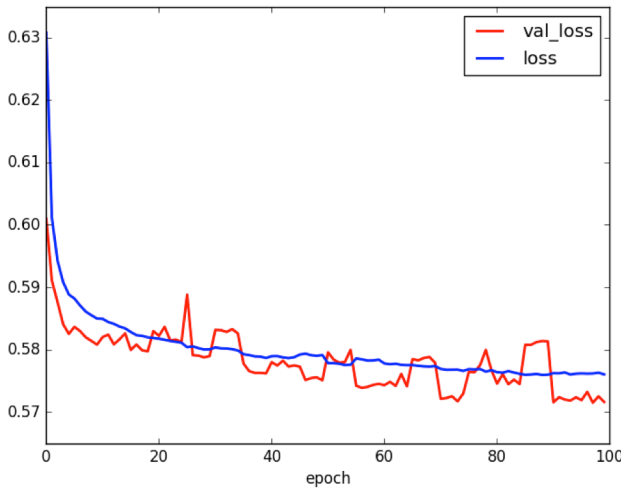
至於 autoencoder 表現不佳的原因，在將有 label 的資料通過訓練所得的 encoder 之後，使用 TSNE 降維保留計距離特性做圖之後，可以看出並沒有將各個類別明顯區分開來，所以後續接上 DNN 後無法訓練出表現好的分類模型。



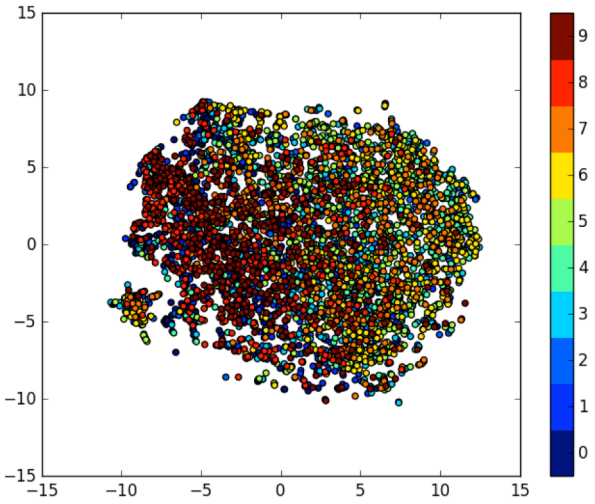
圖一 supervised learning 的 acc 變化



圖二 semi-supervised(1) 的 acc 變化



圖三 自動編碼器在訓練中的 loss 變化



圖四 編碼結果的 TSNE 呈現