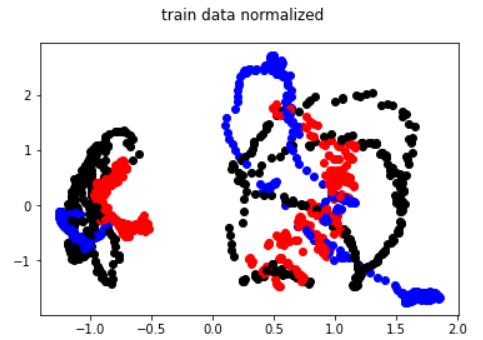
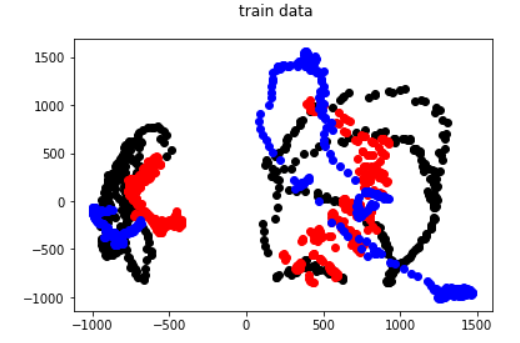
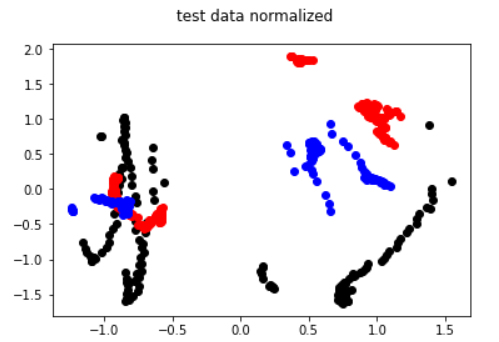
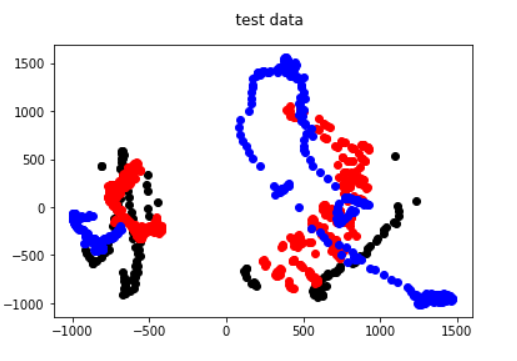
設計方法

一開始先將圖片讀進來後調整形狀成一維向量，之後利用PCA抽取特徵為兩個主要特徵，並加入一個值為1的向量做bias。之後再將所有資料進行標準化，然後利用SGD算法進行訓練，初始化參數使用xavier，激活函數皆使用sigmod，最後輸出函數使用softmax。

數據討論

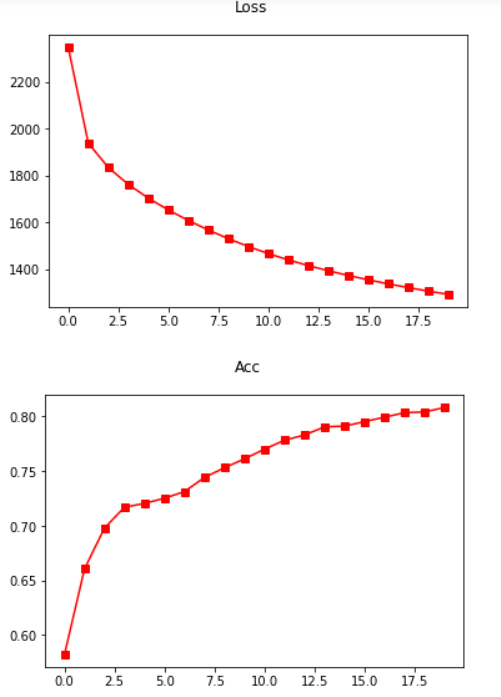
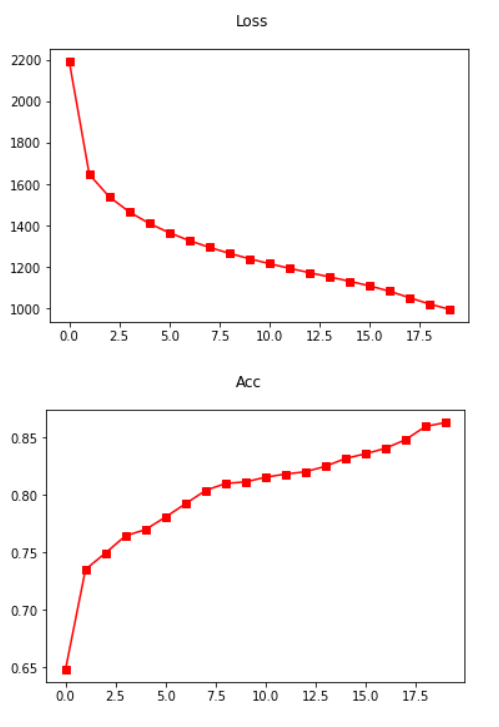




上圖是將抽取過後的特徵進行繪圖的結果，同時有進行標準化和未標準化的結果，測試資料的標準化使用的平均值和標準差是從訓練資料取得的。可以觀測到三類圖片大致可以分為兩群，而且重複性極高，但進行標準化後可發現測試資料的重複性變低，也可以從模型訓練結果觀測到成功率提升。

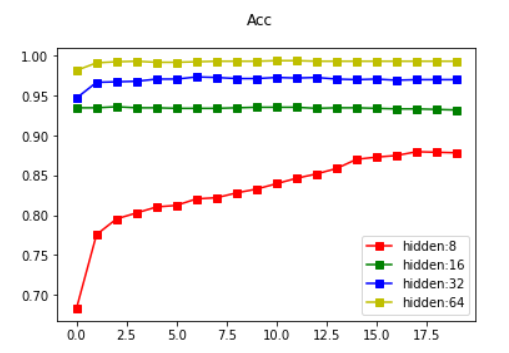
Two-layer vs three-layer

下面左圖是three-layer的訓練資料預測結果，右圖是two-layer的訓練資料預測結果，都是訓練epoch:10、batch\_size:1、learning\_rate=0.01，而最終three-layer在測試資料的準確度為60%，而two-layer則是56%，而如果將epoch調高到50在two-layer可以得到73%的準確度。從中可發現three-layer在相同的狀況訓練結果比two-layer好，但花費的時間卻超過一倍，除此之外一開始的參數初始化也是非常重要的，常常會有一樣的model跑兩次在測試結果的正確率差超過20%。

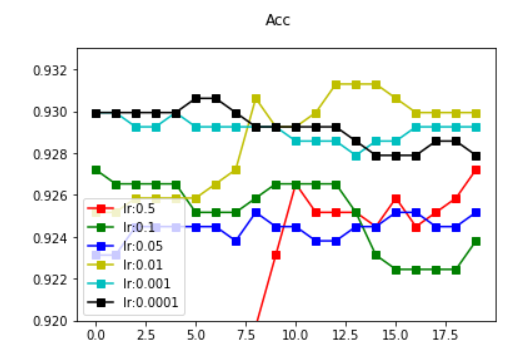


各項參數調整

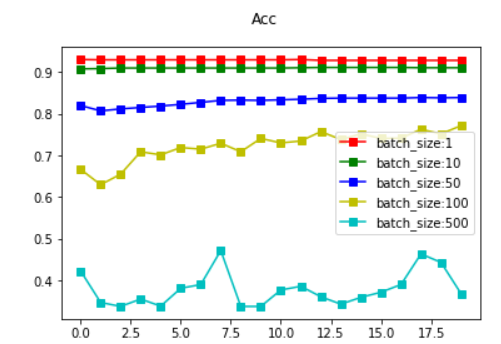
接下來的測試已two-layer為基準，各項參數分別為hidden layer:32、batch\_size:1、learning rate=0.1。



上圖為two-layer模型在不同hidden layer數量下的訓練結果，而最終測試資料8、16、32、64層分別是45.5%、59.6%、60.2%、61.6%，可以發現層數越深結果越好的趨勢。



上圖為模型在各種learning rate的訓練結果，從測試資料的結果來看0.001和0.0001都可以到達62.4而大於0.01後雖然降低到61或60但還是差不多的，因此判斷learning rate大概設定為0.001是比較合適的。



上圖為模型在各種batch\_size的訓練結果，可以發現batch\_size越大訓練成果會越糟糕，而在測試資料的結果來看batch\_size在1和10都可以獲得63%的準確度，但當變成500時卻下降到33%，跟沒有訓練一樣。