**ML2016 HW3 report** 姓名: 徐有慶 學號: R05922162

1. **Supervised learning:**

Step1. 讀取label, unlabeled, test data

Step2. 建構CNN

Convolution

25, 3, 3

MaxPooling

2, 2

MaxPooling

2, 2

Convolution

64, 3, 3

Neural Network

15 layers

100 neurons each layer

Activation: ReLu

其中，Neural network的input layer及output layer的前一層hidden layer皆加上dropout = 0.3，loss function使用categorical cross entropy, optimizer使用adam，batch size = 128, number of epochs = 50。在此model下，利用label data進行training。

1. **Semi-supervised learning(1):**

整體CNN架構同supervised learning，先利用label data訓練出cnn model，再利用其進行self-training。

Step1. 經由label data先訓練出一個cnn model(C1)

Step2. 建立一label\_flag array，紀錄unlabeled data是否已經被label，若label\_flag[i] = 0則表示第i筆unlabeled data尚未被label, 反之則已被label。

Step3. 利用C1預測所有unlabeled data，檢查所有label\_flag為0的data，若預測其屬於class n且機率大於0.9(confidence value)，則該unlabeled data的label設為class n，並將其加入到label data當中，且label\_flag設為1，

Step4. 利用新的label data對C1再進行training，number of epoch = 15， batch size = 128

Step5. 重複Step3~4，直到所有unlabeled data都被label或是重複5次

1. **Semi-supervised learning(2):**

利用auto encoder做clustering，整體CNN架構同supervised learning

Step1. 建立Neural network拿來train auto encoder其層數為

Input layer -> 512 neurons -> 256 neurons -> 128 neurons (bottleneck layer) -> 256 neurons -> 512 neurons -> Output layer。activation = ReLu, optimizer = adam, loss function = mse

Step2. 將每筆label, unlabeled data經過encoder壓縮成128維的features，5000筆label data分成10個clustering，並分別算出每個clustering的centroid。

Step3. 計算unlabeled data與10個centroid的euclidean distace，並將其加入距離最近的clustering，同時更新10個clustering的centroid

Step4. 重複Step3，直到所有unlabeled data都有label

Step5. 利用原先label data及被label過的unlabeled data去訓練CNN

1. **Compare and analyze your results:**

Kaggle score

Supervise learning:

Semi-supervise learning (self-training):

Semi-supervise learning (auto encoder):

Self-training的部分設置了一個confidence value，超過才當作是可信任的預測結果，而使用了self-training的semi-supervise learning也明顯比supervise learning的結果還要佳。Auto encoder的部分，由於沒有加上noisy，只是單純地做deep auto encoder，且分群使用euclidean distance去做分群，可能較容易有分錯群的問題，而導致最後訓練出的CNN accurate很高，但predict的結果卻不是很好。

Unlabeled data的部分，並沒有將test data一起用上，因為加上後會有out of memory的問題