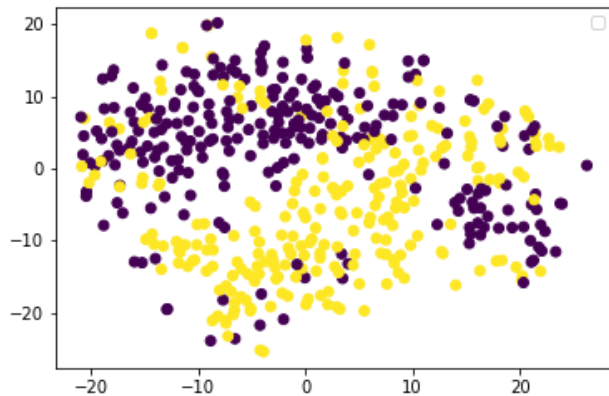


1. (3%) 請至少使用兩種方法 (autoencoder 架構、optimizer、data preprocessing、後續降維方法、clustering 算法等等) 來改進 baseline code 的 accuracy。
  - a. 分別記錄改進前、後的 test accuracy 為多少。
  - b. 分別使用改進前、後的方法，將 **val data** 的降維結果 (embedding) 與他們對應的 label 畫出來。
  - c. 盡量詳細說明你做了哪些改進。

Baseline model:

test accuracy: 0.716

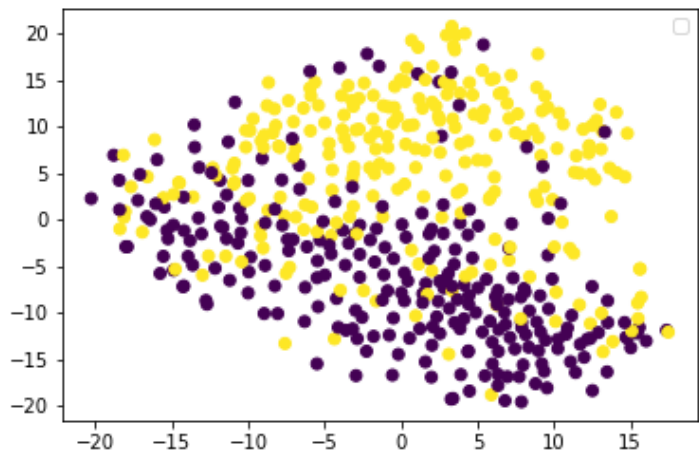


方法1:

不通過autoencoder，3072維直接 PCA到200維，再TSNE到2維，再經過原本的clustering。

可以發現結果意外的分的還不錯，代表問題本身可能是linear可解的，不需要複雜的model。

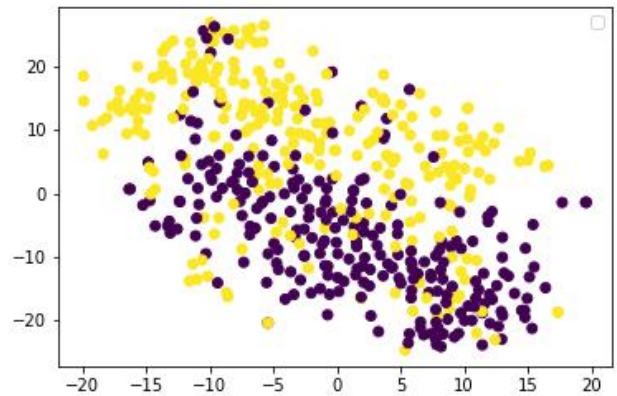
test accuracy: 0.76



方法2:

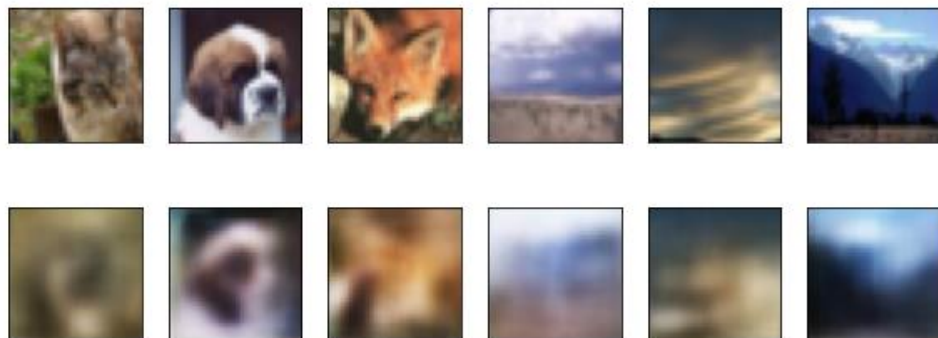
Autoencoder增加一層conv2d(256,32)和maxpool(2)，輸出128維，再PCA到50維，再TSNE到2維，再經過原本的clustering。

想法: 原本上課說中間那層應該要是Bottleneck layer，所以中間層該要夠小，不然沒有降維的效果，所以加了一層conv2d(256,32)。



test accuracy: 0.7

2. (1%) 使用你 test accuracy 最高的 autoencoder，從 trainX 中，取出 index 1, 2, 3, 6, 7, 9 這 6 張圖片
  - a. 畫出他們的原圖以及 reconstruct 之後的圖片。



reconstruct的圖片還算可以看出原樣。

3. (2%) 在 autoencoder 的訓練過程中，至少挑選 10 個 checkpoints
  - a. 請用 model 的 train reconstruction error (用所有的 trainX 計算 MSE) 和 **val accuracy** 對那些 checkpoints 作圖。

橫軸單位: 10 epochs

b. 簡單說明你觀察到的現象。

隨著epochs 增加MSE越來越小(gradient descent的結果)，但是accuracy並沒有明顯趨勢。所以autoencoder其實沒有很好的抽取特徵，而是靠後續的PCA和TSNE來達成預測。可以跟第一題的方法1可以互相印證。

