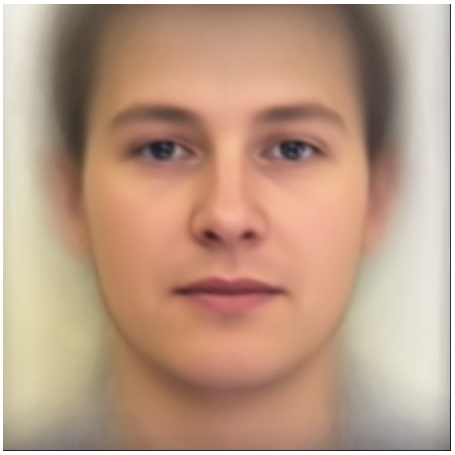
ML hw4 Report

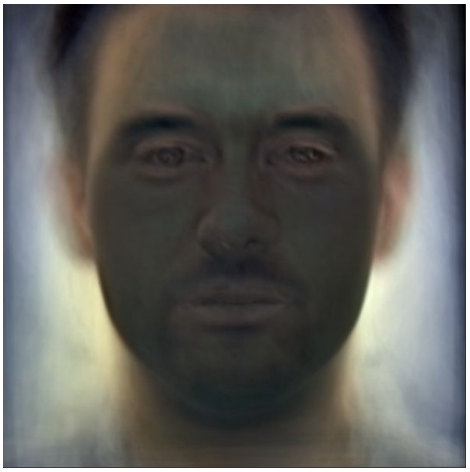
學號：生機二 系級：b05611033 姓名：杜杰翰

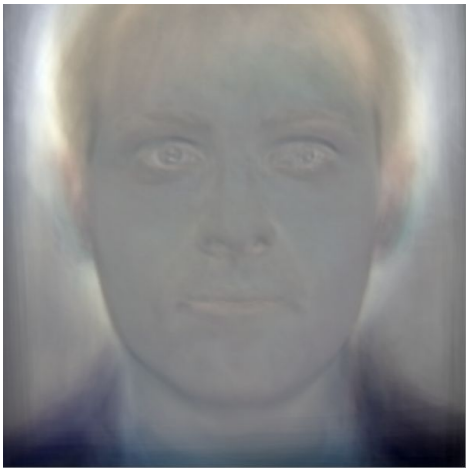
A. PCA of colored faces

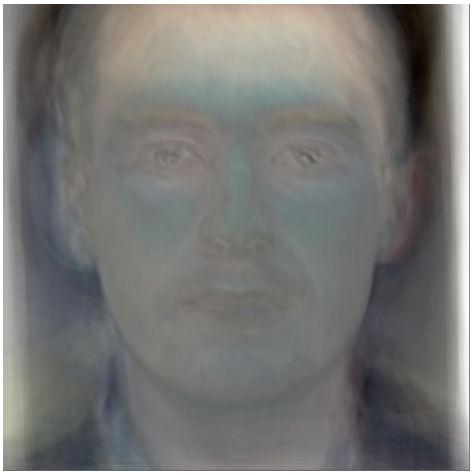
A.1. (.5%) 請畫出所有臉的平均。

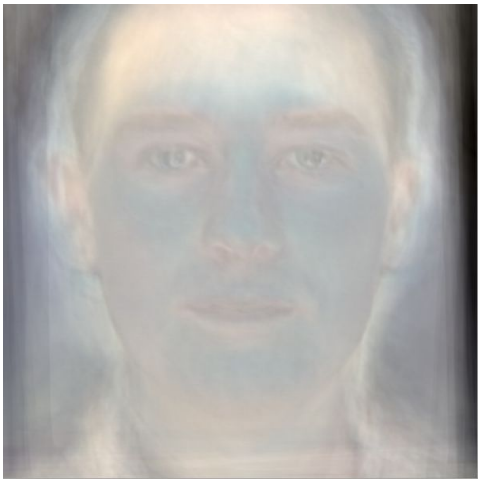


A.2. (.5%) 請畫出前四個 Eigenfaces，也就是對應到前四大Eigenvalues 的 Eigenvectors。

1) 

2) 

3) 

4) 

A.3. (.5%) 請從數據集中挑出任意四個圖片，並用前四大 Eigenfaces 進行 reconstruction，並畫出結果。

我用random隨機挑選4個圖片進行重構，分別是64, 256, 314, 380(從0起算)

A.4. (.5%) 請寫出前四大 Eigenfaces 各自所佔的比重，請用百分比表示並四捨五入到小數點後一位。

First: 4.1%　Second: 2.9%　Third: 2.4%　Forth: 2.2%

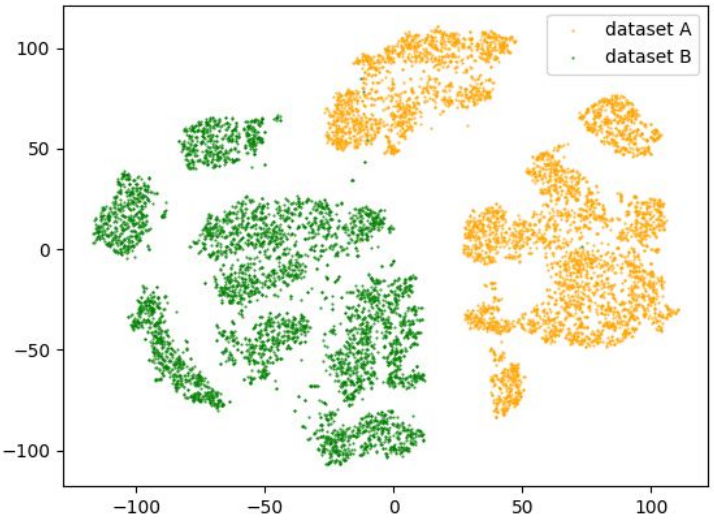
B. Image clustering

B.1. (.5%) 請比較至少兩種不同的 feature extraction 及其結果。(不同的降維方法或不同的 cluster 方法都可以算是不同的方法)

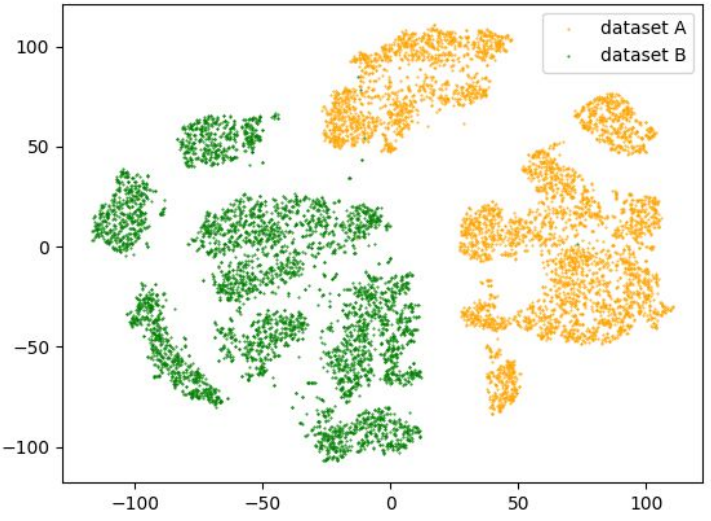
1. 我用pca降到256維，並把whiten=True，再使用kmeans之後在kaggle上得到的成績是1。
2. 我用autoencoder之後用kmeans做cluster，在kaggle上得到的成績是0.96943。autoencoder結構如下：

```
inputs = Input(shape=(784,))
en = Dense(128, activation='relu')(inputs)
en = Dense(64, activation='relu')(en)
en = Dense(32, activation='relu')(en)

den = Dense(64, activation='relu')(en)
den = Dense(128, activation='relu')(den)
den = Dense(784, activation='relu')(den)

autoencoder = Model(inputs=inputs, outputs=en)
decoder = Model(inputs=inputs, outputs=den)
```

B.2. (.5%) 預測 visualization.npy 中的 label，在二維平面上視覺化 label 的分佈。



B.3. (.5%) visualization.npy 中前 5000 個 images 跟後 5000 個 images 來自不同 dataset。請根據這個資訊，在二維平面上視覺化 label 的分佈，接著比較和自己預測的 label 之間有何不同。



從這兩張圖中，可以看到他們長一樣。我想是由於visualization data是從images.npy中抽出來的，因此我的model在images.npy中做到1的正確率，則在visualization中也會達到1的正確率。

C. Ensemble learning

C.1. (1.5%) 請在hw1/hw2/hw3的task上擇一實作ensemble learning，請比較其與未使用ensemble method的模型在 public/private score 的表現並詳細說明你實作的方法。（所有跟ensemble learning有關的方法都可以，不需要像hw3的要求硬塞到同一個model中）

我在hw3使用了7個model做ensemble，以下是各model在kaggle的成績：

	Private score	Public score
gen08.csv	0.64586	0.65199
gen09.csv	0.65143	0.65672
gen12.csv	0.64864	0.65812
func01.csv	0.66787	0.66202
func02.csv	0.65171	0.65812
res02.csv	0.63081	0.64251
res03.csv	0.64196	0.64502

我是使用keras進行hw3，而keras進行ensemble時必須要有同樣的Input。首先，我將各model重建成function，之後建立7個擁有同樣Input的model，再把之前train好的7個model各自的weight讀進新建的model中。之後我創造一個新的layer average將7個model的output組合起來並平均，然後新建一個

```
ensemble = Model(inputs=inputs, outputs=average)
```

於是便得到了一個ensemble過後的model。而由它predict出來的ensemble.csv在kaggle上的表現為：

Private score: 0.70381　　Public score: 0.70911

明顯高於沒有ensemble的結果。