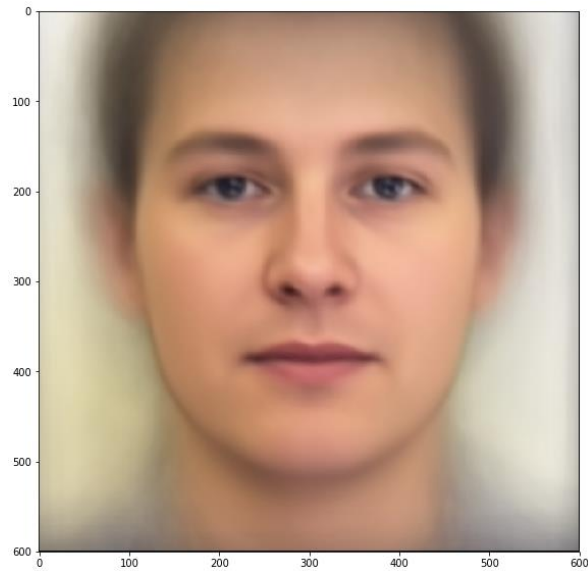


HW4

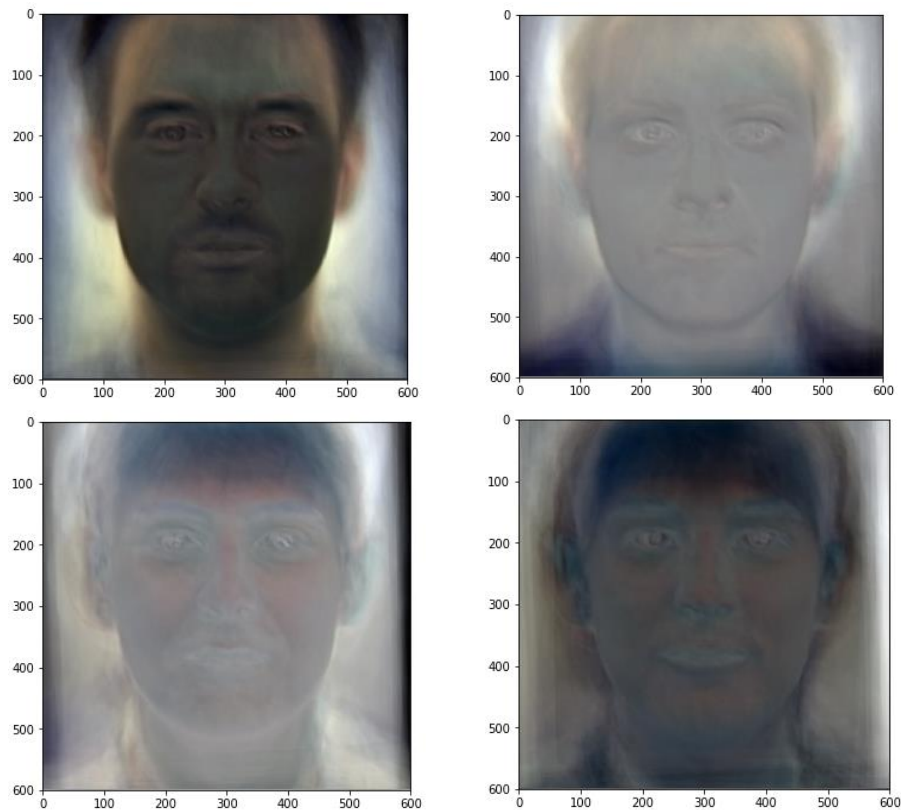
學號：B03902130 系級：資工四 姓名：楊書文

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。



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。



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。

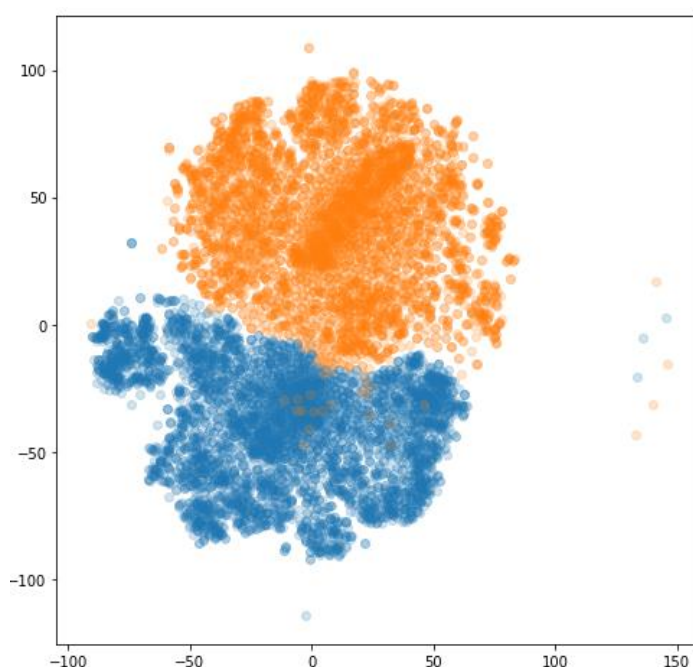
Significant	first	second	third	fourth
Ratio	4.1%	2.9%	2.4%	2.2%

B. Image clustering

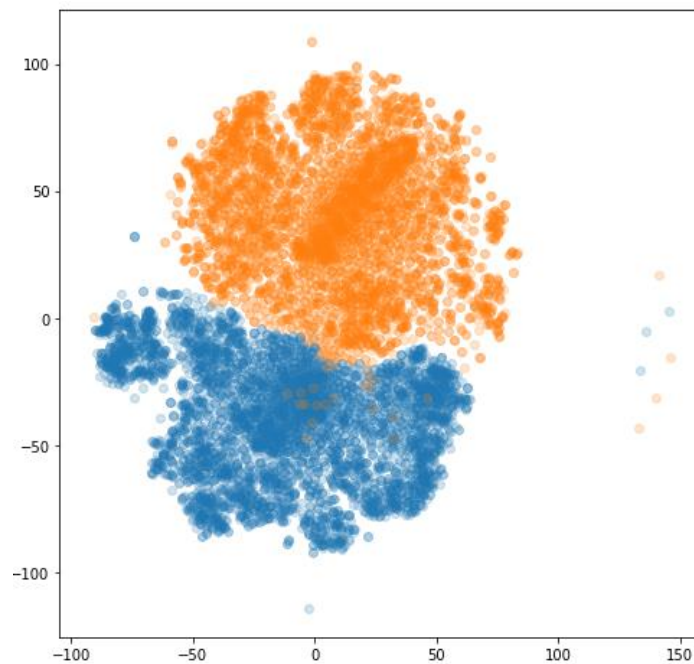
- B.1. (.5%) 請比較至少兩種不同的 **feature extraction** 及其結果。(不同的降維方法或不同的 **cluster** 方法都可以算是不同的方法)
- 我同樣都使用 PCA 降到 300 維，只是 **cluster** 的方法不太一樣。一種是直接用 Kmeans 分成兩個 **cluster**，試圖分成 **digit** 和 **fashion**。一種是先用 Kmeans 分成 20 個 **cluster**，其 **center faces** 印出來會看到有數字 0~9 也有各種衣服鞋子，然後再將想要分類的圖片拿去跟 20 個 **center faces** 比較 **similarity**，看最相近的 **center face** 是 **digit** 還是 **fashion** 來做分類。後者雖然比較複雜，但並沒有比較好。實際去看它分錯的圖片會有類似一件長得很像 "1" 的洋裝，或很像涼鞋的細線數字。也就是說那張圖片與最相近的 **digit center** 和最相近的 **fashion center** 都很像，以至於很難區分。而如果只是單純分成兩個 **cluster** 的話，可能因為 **center** 只有兩個，分別是數字的平均和衣服的平均，而那張很像 1 又很像洋裝的圖片與這兩個平均都很不像(至少形狀上不像)，這時就需要靠形狀以外比較細微的特徵來辨別(像數字線條的顏色變化較剛硬，而服飾就比較柔和)

METHOD	PUBLIC	PRIVATE
PCA, 2 cluster	0.99998	0.99998
PCA, 20 cluster	0.98988	0.98979

- 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。



在我的分類上看不太出 predicted label 和 ground true label 有什麼差異，因為經過檢查後發現我 predicted label 的確前 5000 都是 1，後 5000 都是 0，有完全正確的把兩個 dataset 分開。而從圖中看到有些橘色點和藍色點還分不太開，我覺得這是因為圖片中的確有一些與數字相近的衣服，或筆畫超粗容易被誤判為衣服的数字。因此藍色與橘色畫出來呈連續分布(沒有分很開)應該合理。而畫出來藍色橘色點交界處有 overlap 應該是因為我將 PCA 得到的 300 維直接用 TSNE 降到 2 維，而 TSNE 還無法將 300 維資訊完整的用 2 維表達，因此雖然用 300 維可以精準 label，但畫出來還是會有 overlap。

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 ensemble 的方法單純是將兩個訓練最好的 model 所輸出的 7 個 classes 機率兩兩平均起來，再從平均過的機率取 argmax。實驗

ensemble 的過程中發現，如果只是將同一個 model (完全相同的架構、訓練參數、data) 的不同 training 階段拿幾個來 ensemble，則 performance 幾乎沒變。而如果是拿兩個不同的 model (任一參數不同即可) 來 ensemble，則 performance 相當有機會暴升。這是因為不同 model 所擅長的領域不同，當 model1 對某張 test image 所輸出的七個機率都不高時，硬要把其最高的機率(可能只有 0.3) 當作答案並不合適，這時如果 model2 剛好相當有信心其為 class 2，(class 2 的機率可能高達 0.8 以上)，平均後的結果基本上就會是 class 2 機率最高。因此將兩個 model 輸出的機率平均起來就有互補效果，哪個 model 對哪張圖片較有信心，就會以他預測的結果為準。

MODEL	PUBLIC	PRIVATE
w/o ensemble	0.68180	0.67149
w ensemble	0.70799	0.70103