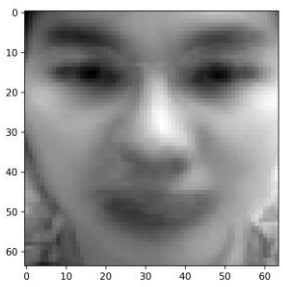
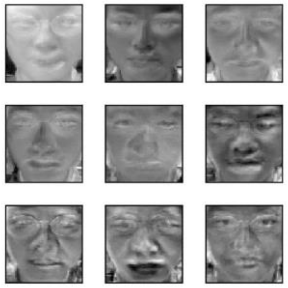
學號：B03902090 系級： 資工三 姓名：邵楚荏

1.1. Dataset 中前 10 個人的前 10 張照片的平均臉和 PCA 得到的前 9 個 eigenfaces:

答：(左圖平均臉，右圖為 3x3 格狀 eigenfaces, 順序為 左到右再上到下)

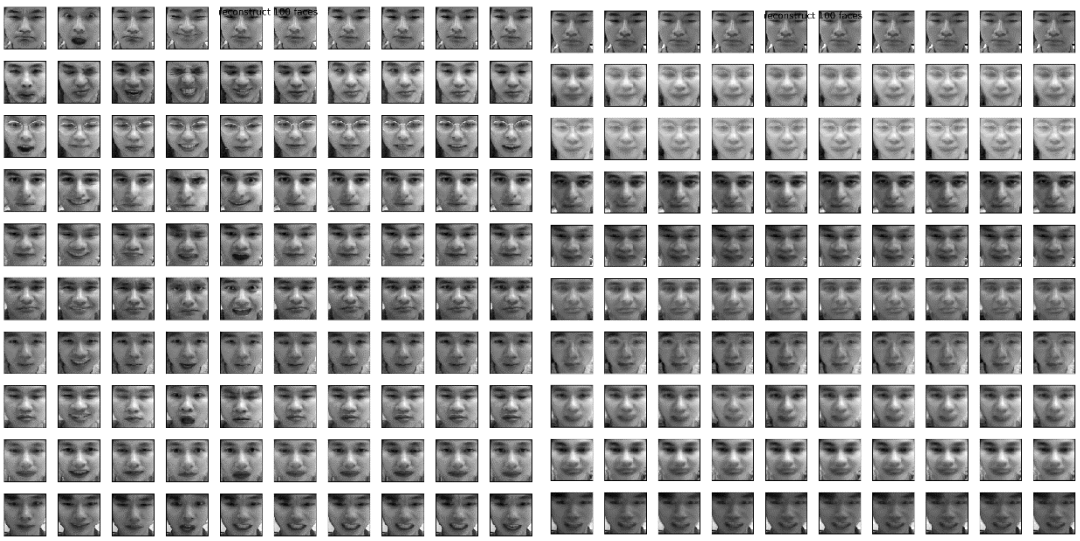




Average face **top 9 eigenfaces**

1.2. Dataset 中前 10 個人的前 10 張照片的原始圖片和 reconstruct 圖 (用前 5 個 eigenfaces):

答：(左右各為 10x10 格狀的圖, 順序一樣是左到右再上到下)



Origin faces **Reconstructed faces**

1.3. Dataset 中前 10 個人的前 10 張照片投影到 top k eigenfaces 時就可以達到 < 1% 的 reconstruction error.

答：(回答 k 是多少)

```
56 :  0.011019974866363266
57 :  0.010653702915789126
58 :  0.010299886209991293
59 :  0.009968231970979577
answer: 59
```

Answer: k = 59 以上的時候，reconstruction error 會小於 1%!

2.1. 使用 word2vec toolkit 的各個參數的值與其意義:

答：

```
word2vec.word2vec(
    train='text_data/all.txt',
    output='model/model.bin',
    size = 100,
    min_count = 6,
    window = 5,
    verbose=True)
```

cbow 這個參數我使用的是 default 的值(也就是 skip-gram)，skip-gram 大致上的結構分為三層:Input, projection, output，其中 input 和 projection layer 都是在 window 內最中間的那個詞，在 projection layer -> output layer 的過程中，利用建造好的 Huffman tree 去不斷的更新子節點中的向量，最後 output 出 window 中除了中間詞以外的周圍向量。

Window 這個參數的意思就是說一次要取幾個詞當作一組去 predict。

Min_count 這個參數的意思是如果在 corpus 中某個詞出現的次數小於 min_count，那麼這個詞就會被丟棄。

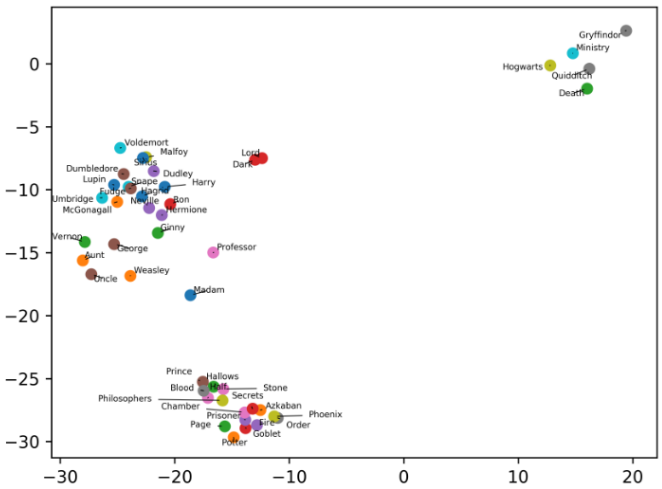
Hs 這個參數代表了 hierarchical softmax，它不同於一般的 softmax 的地方是他可以利用 binary tree 去加快他 softmax 的速度，將所有的 word vector 都存於葉子節點的話，算出 softmax 就不需要走到其他 word vector 的節點上，可以大量的減少計算的時間。

Size 這個參數代表了 output 出來的 word embedding 每個詞的向量的維度。

Sample 這個參數就是 subsampling，在 corpus 中越高頻的不一定越是重要(因為有可能是 stop words)，所以 Sample 的意思就是設一個 threshold 去 random 濾掉這些詞。

2.2. 將 word2vec 的結果投影到 2 維的圖:

答：(圖)



2.3. 從上題視覺化的圖中觀察到了什麼？

答：

從上圖可以發現將 word2vec 投影到 2 維上，詞彙主要被分為三群，經過查了一下，發現圖下方的那一群，主要是”阿茲卡班”、”秘密”、”王子”、”石頭”、”血”，可以輕易地得知這些詞彙都出現在哈利波特七集的標題當中，所以我想這群的相關性蠻高的!左上角那群主要是哈利波特當中，比較常見角色的名字，例如:”哈利”、”威斯利”，之類的人名。右上角則是”葛來分多”、”魁地奇”等詞彙。

3.1. 請詳加解釋你估計原始維度的原理、合理性，這方法的通用性如何？

答：

做完第一題 eigenface 的題目之後，我突然想到同樣的方法也可以套用在這題，首先我利用助教的 gen.py 這份檔案生出 3000 個 set，每個 set 的 intrinsic dimension 都是從 1~60 中隨機取一個整數值，每個 set 中又各有 6000 筆 datapoints，然後將每一個 set 去做 svd，找出這個 set 的 singular value 值，因為在訓練資料裡實際上是知道他的維度的，所以假設這個 set 的 intrinsic dimension 是 40，那麼我將 s(svd 後的 singular value)中第 41 大的值當作 threshold，然後我想要輸入一筆 set 的 s vector，我能輸出這一個 set 預測的 threshold。

於是我用了 sklearn 的 linear_regression 去訓練我的模型，總共 3000 個 set。訓練完之後，在 testing set 上，每一個 set 先去做 svd 分解之後將 singular value 丟進 model 裡可以 predict 出這個 set 的 threshold 值，之後再去看在這個 s 有多少值是大於 threshold 的，當作所預測的 intrinsic dimension。

但這個方法必須是用在當每一個 set 都是同樣的 datapoints 的時候，而且必須知道原本 data 的維度(像這題就是 100 維)，在已知這些資訊的情況下，這個方法的效果才會比較好。

3.2. 將你的方法做在 `hand rotation sequence datatset` 上得到什麼結果？合理嗎？請討論之。

答：

Dataset 中總共 481 張圖片，每一張圖片都是 480*512 個 pixel，我把它整個變成(481 , 480*512)的維度，相對原本的題目來說，就是有 481 個點，每個點的維度都是(480*512)這麼多，經過我的 model 之後，所預測出來的 intrinsic dimension 是 6，在網路上查到的結果幾乎都是 3~4，我想這是因為我的 model 原本都是取 6000 個 datapoint 去做分析，但在這題裡面只有 481 個 datapoint，而且每個 datapoint 的 input dimension 有點大，導致我做出來的答案沒有那麼精準。