# Project Report

Please refer to the source code within the Jupyter Notebook for more information. Float figures, where applicable, are reported to 3 decimal places.

## A. Introduction

This report is for the EE5907 CA2 task on Face Recognition.

## B. Dataset

**Data Selection**

The project utilised the CMU PIE dataset plus personal selfies. Out of the 68 different subjects, 25 were chosen to be included in this study.

Selected subjects: [2, 8, 13, 14, 15, 17, 19, 20, 23, 24, 27, 31, 32, 33, 34, 40, 42, 44, 46, 48, 50, 52, 59, 63, 65]

Since each subject has 170 images, plus 10 selfies were included, a final total of 4260 images are utilised in this project.

Number of selected PIE images: 4250

Number of selected selfies: 10

Number of selected images: 4260

**Train Test Split**

The CMU PIE images and selfies were each split into 70% training and 30% testing groups. The training and testing samples remain fixed throughout the project.

Number||Proportion of train PIE images: 2975 || 0.7

Number||Proportion of test PIE images: 1275 || 0.3

Number||Proportion of train selfies: 7 || 0.7

Number||Proportion of test selfies: 3 || 0.3

## 01. Principal Component Analysis (PCA)

**Pre-processing**

Each raw face image was converted from 32 x 32 pixels to a 1024-dimensional vector. I randomly sample 500 images from the CMU PIE training set and added the 7 training selfies. This resulted in a working matrix of the following shape.

Vectorised and loaded data: (507, 1024)

**PCA 2D & 3D**

PCA package from sklearn.decomposition was utilised to reduce the dimensionality of vectorized images to 2 and 3. The projected data vectors in 2D and 3D plots are shown below respectively in Figures 1 and 2, where the blue points reflect ‘PIE’ images while orange reflects the ‘selfies’. Figures 3 and 4 correspond exactly to 1 and 2, but all the individual subjects belonging to PIE are now highlighted by a different colour each. There is a clear clustering effect of selfies in all four figures, but it is a little harder to discern the clustering patterns for PIE subjects.

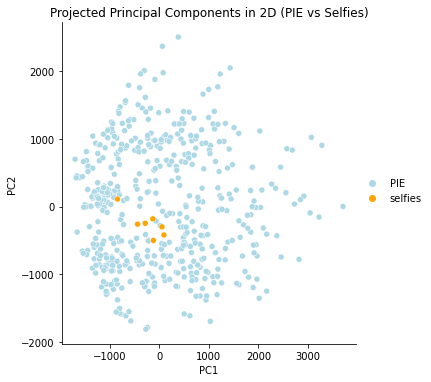
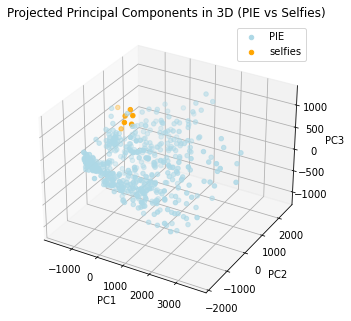
2-PCs explained variance ratio: [0.416 0.263]

2-PCs singular values: [24469.939 19480.850]

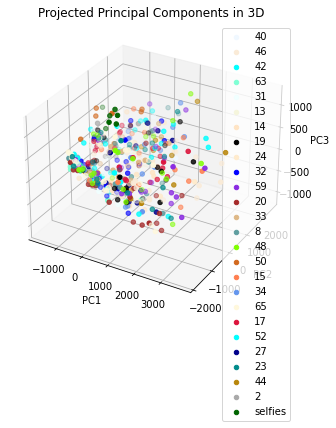
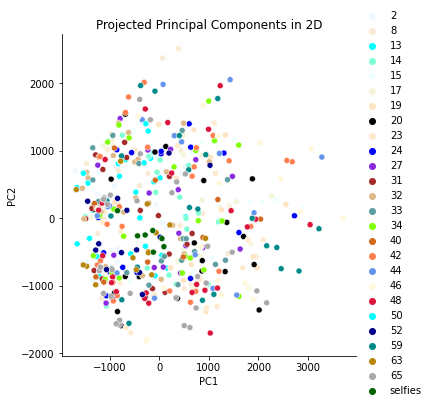
3-PCs explained variance ratio: [0.416 0.263 0.073]

3-PCs singular values: [24469.939 19480.850 10266.821]

**Figure 1 Figure 2**

**Figure 3 Figure 4**



**Eigenfaces**

The following is the visualisation of the 3 eigenfaces used for dimensionality reduction:



**PCA Classification**

I repeated PCA to reduce the dimensionality of face images to number of principal components n\_pc=40, 80, 200. For each n\_pc value, 1-NN (nearest neighbour) classification using KNeighborsClassifier package from sklearn.neighbors with k=1 was performed. The classification accuracies are as follows:

Accuracy of 1-NN with 40-components PCA for ALL test set: 93.114%

Accuracy of 1-NN with 40-components PCA for PIE test set: 93.171%

Accuracy of 1-NN with 40-components PCA for selfies test set: 66.667%

Accuracy of 1-NN with 80-components PCA for ALL test set: 95.149%

Accuracy of 1-NN with 80-components PCA for PIE test set: 95.212%

Accuracy of 1-NN with 80-components PCA for selfies test set: 66.667%

Accuracy of 1-NN with 200-components PCA for ALL test set: 96.088%

Accuracy of 1-NN with 200-components PCA for PIE test set: 96.154%

Accuracy of 1-NN with 200-components PCA for selfies test set: 66.667%

Based on the above results, the greater the number of PCA components, the better the classification accuracy on PIE images but the changes are minimal. For selfies, the accuracy is the same throughout at (2/3)%.

## 02. Linear Discriminant Analysis (LDA)

**LDA 2D & 3D plots**

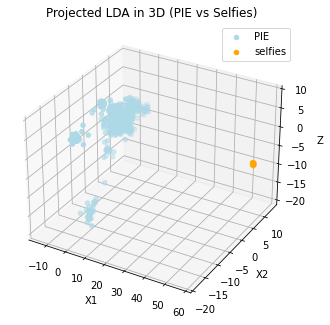
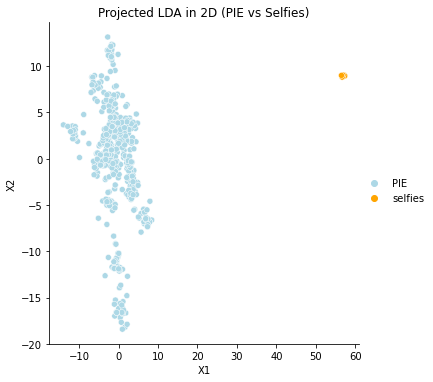
LinearDiscriminantAnalysis package from sklearn. discriminant\_analysis was utilised to reduce the dimensionality of vectorized images to 2, 3 and 9. The projected data vectors in 2D and 3D plots are shown below respectively in Figures 5 and 6, where the blue points reflect ‘PIE’ images while orange reflects the ‘selfies’. Figures 7 and 8 correspond exactly to 5 and 6, but all the individual subjects belonging to PIE are now highlighted by a different colour each. There is a very clear clustering effect of selfies in all four figures, as shown by how disjoint the selfies’ points are from the main group across axis X1. Figure 7 also show some clustering patterns for each PIE subject, especially for subjects 48, 65, 20 and 17 that are spread out across axis X2 and 24 that is slightly further to the left on axis X1.

2-components LDA explained variance ratio: [0.144 0.094]

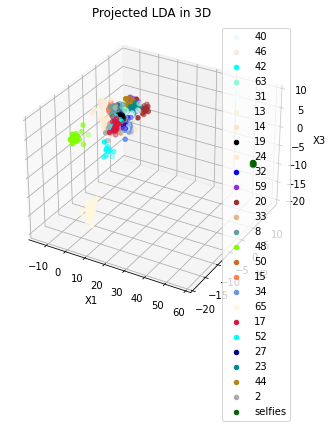
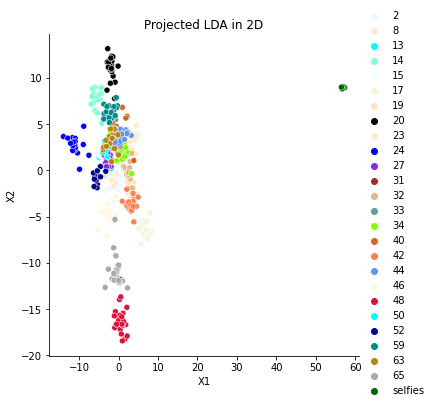
3-components LDA explained variance ratio: [0.144 0.094 0.074]

9-components LDA explained variance ratio: [0.144 0.094 0.074 0.068 0.063 0.055 0.053 0.046 0.041]

**Figure 5 Figure 6**



**Figure 7 Figure 8**



**LDA Classification**

I repeated PCA to reduce the dimensionality of face images to number of principal components n\_pc=2,3,9. For each n\_pc value, 1-NN (nearest neighbour) classification using KNeighborsClassifier package from sklearn.neighbors with k=1 was performed. The classification accuracies are as follows:

Accuracy of 1-NN with 2-components LDA for ALL test set: 23.944%

Accuracy of 1-NN with 2-components LDA for PIE test set: 23.94%

Accuracy of 1-NN with 2-components LDA for selfies test set: 0.0%

Accuracy of 1-NN with 3-components LDA for ALL test set: 41.706%

Accuracy of 1-NN with 3-components LDA for PIE test set: 41.758%

Accuracy of 1-NN with 3-components LDA for selfies test set: 0.0%

Accuracy of 1-NN with 9-components LDA for ALL test set: 90.219%

Accuracy of 1-NN with 9-components LDA for PIE test set: 90.424%

Accuracy of 1-NN with 9-components LDA for selfies test set: 0.0%

Based on the above results, the greater the number of LDA components, the better the classification accuracy on PIE images, with rather drastic increases in accuracy (2-components to 3 warrants a doubling in accuracy). For selfies, the accuracy is the same throughout at 0%, which is rather surprising given how well clustered selfies seems in Figures 5-8. I think perhaps the few number of train selfies compared to PIE subjects’ images (7 vs 119) led to under-sampling bias, and the classifier is unable to properly discern selfies. As an exploratory exercise, I retrained the LDA x 1NN classifier with the 507-subset training data, and indeed, accuracy for selfies test set is high:

Accuracy of 1-NN with 2-components LDA for selfies test set: 100.0%

Accuracy of 1-NN with 3-components LDA for selfies test set: 66.667%

Accuracy of 1-NN with 9-components LDA for selfies test set: 100.0%

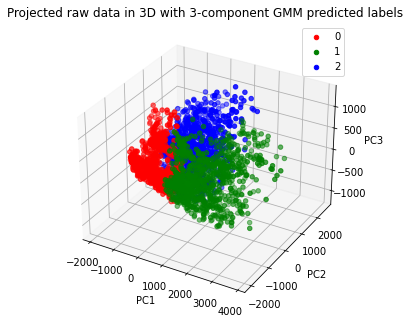
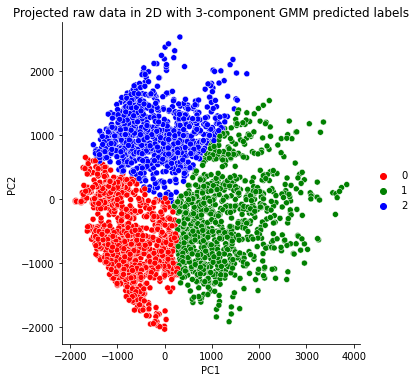
This makes sense since 507-subset training data holds (7/507)% of selfies compared to full training data that only holds (7/2982)%, making it hard for any classifier to probabilistically predict selfie as a class label.

## 03. Gaussian Mixture Model (GMM)

**Using raw face images**

Using vectorised raw face images (i.e. data of shape (2982, 1024)) as inputs to train a 3-Gaussian components GMM model, we perform clustering by assigning each data point by one of the three possible labels. Figures 9 and 10 plots the GMM clustering results on 2D and 3D spaces (based on PCA components from the earlier section). Due to the large number of PCs, I am only visualising scatterplots based on the first 2 and 3 components. The predicted labels are highlighted by 0=red, 1=green and 2=blue.

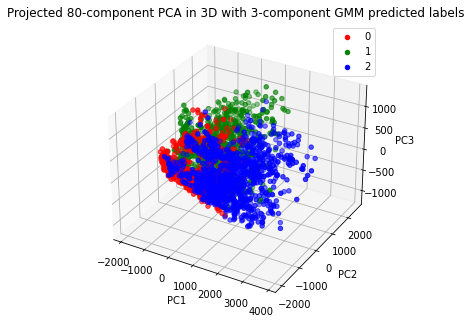
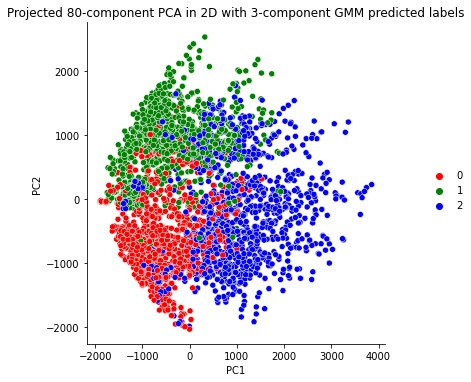
**Figure 9 Figure 10**



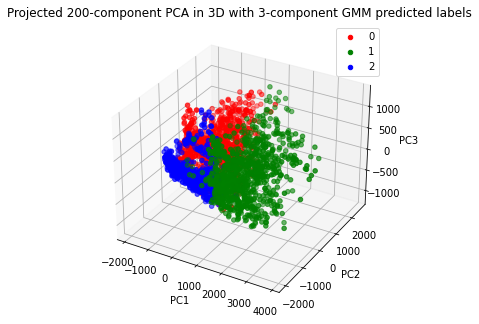
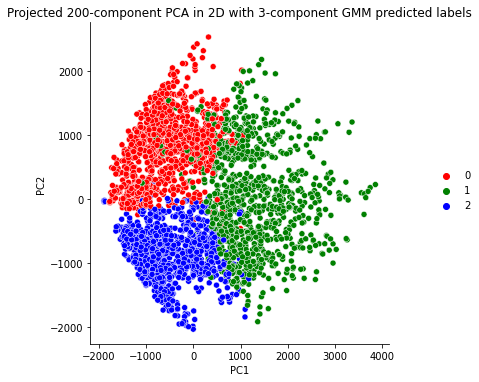
**Using PCA pre-processed images**

I used face vectors after PCA pre-processing at n\_pc=80 and 200. These serve as training data which was fed into a 3-component GMM model. To observe the clustering by GMM, Figures 11-14 plots PCA transformed data in 2D and 3D spaces, and predicted labels are again highlighted by 0=red, 1=green and 2=blue.

**Figure 11 Figure 12**

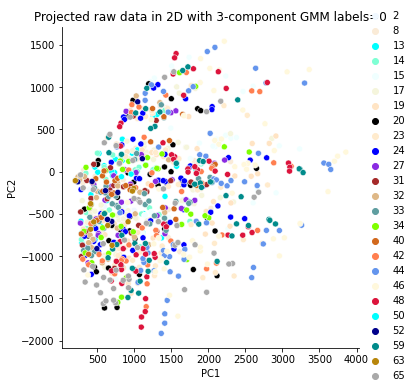
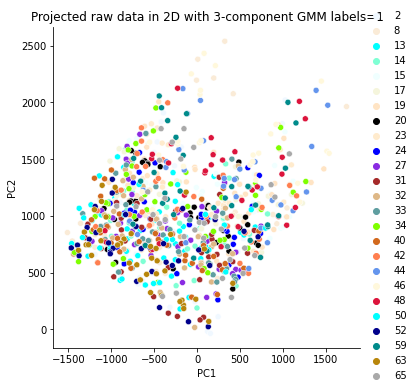
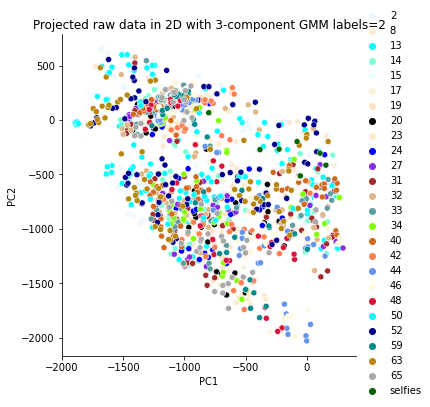


**Figure 13 Figure 14**



For all scatterplots in Figures 9-14, we observe three clusters per plot with differing mean, but some overlap of points across cluster boundaries. Usage of raw data inputs seems to generate the cleanest clusters in Figures 9 and 10 compared to the 80 and 200 component set-ups. This might be because GMM learnt latent characteristics that reflect PC1 and PC2. Intuitively, the input data is also largest with raw data (utilises all 1024 features), thus allowing for better expectation maximisation. On a similar note, we observe slightly better clustering (fewer overlapping points across clusters) for n\_pc=200 compared to 80 plots. The PCA+GMM set-up that utilises dimensionality reduction via PCA does not help uncover clustering characteristics (Scrucca, 2015).

To dive deeper, the images assigned to the same GMM label based on the raw data set-up are plotted against PC1 and PC2. One notable point is selfies only appear for GMM label=2. For the PIE subjects, the plots are rather inconclusive and hard to observe any clear clustering of the actual subjects’ labels. For this reason, I do not include the 80-component and 200-component plots, but they are available in the Jupyter notebook if interested.

## 04. Support Vector Machine (SVM)



References

Scrucca, Luca. “Dimension Reduction for Model-Based Clustering.” Statistics and Computing, vol. 20, no. 4, 1 July 2009, pp. 471–484, 10.1007/s11222-009-9138-7. Accessed 12 Nov. 2019.

Appendix

x