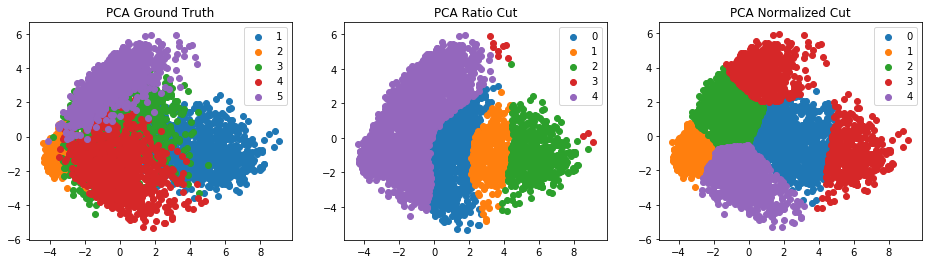
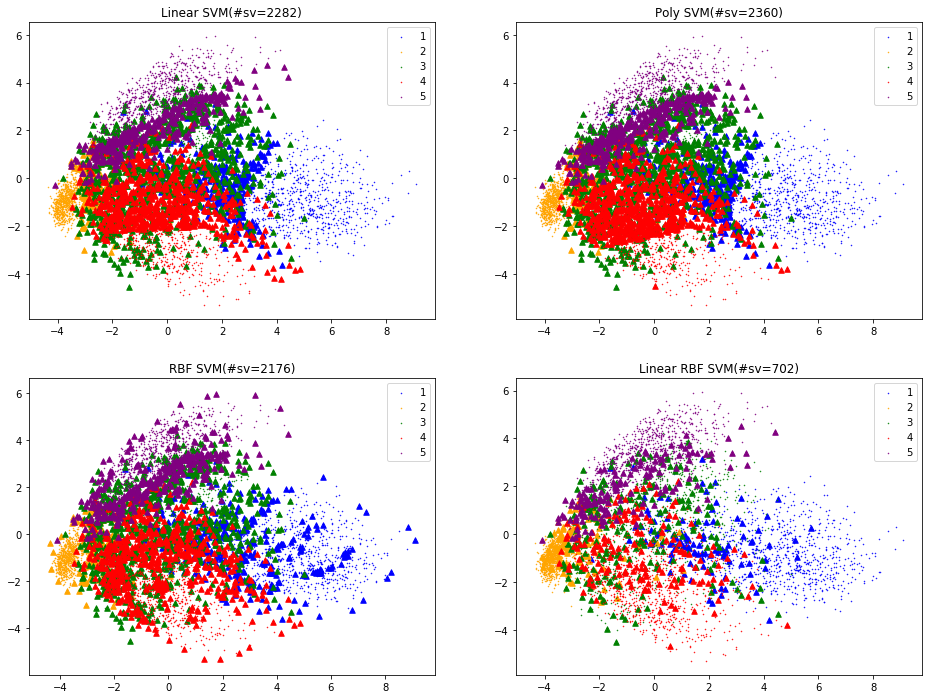
**Homework# 7**

**0756079 陳冠聞**

1. **PCA projection and clustering**

****

可以發現 normalized cut 不同 cluster 間的個數較為平均。

1. **PCA projection and SVM classification**

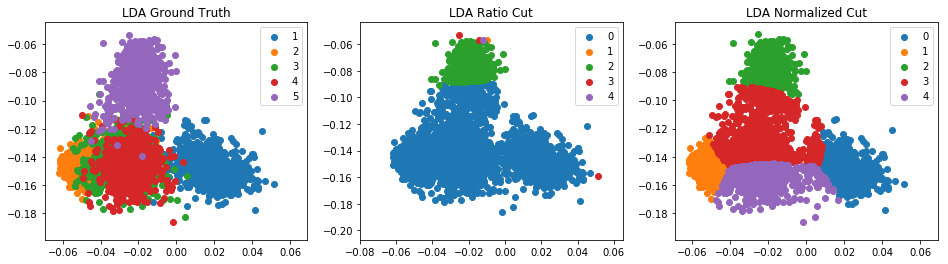
可以發現使用 Linear + RBF kernel 所用的 support vector 最少

(support vector 越少代表 overfitting 的風險越小)

此處 kernel 參數採用 default

( C=0, degree=3 for poly, coef0=0.0, gamma = 1/784 )

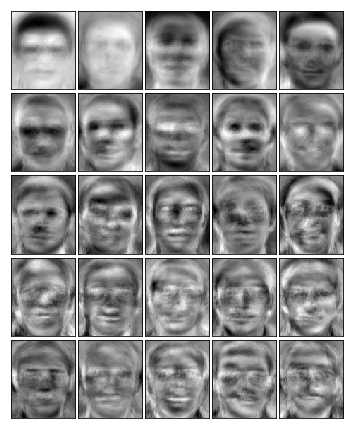
1. **LDA projection**

****

同上，可以發現 normalized cut 不同 cluster 間的個數較為平均。

Top 25 EigenFaces

1. **EigenFace**

****

Reconstruction

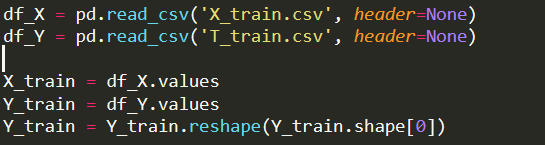
****

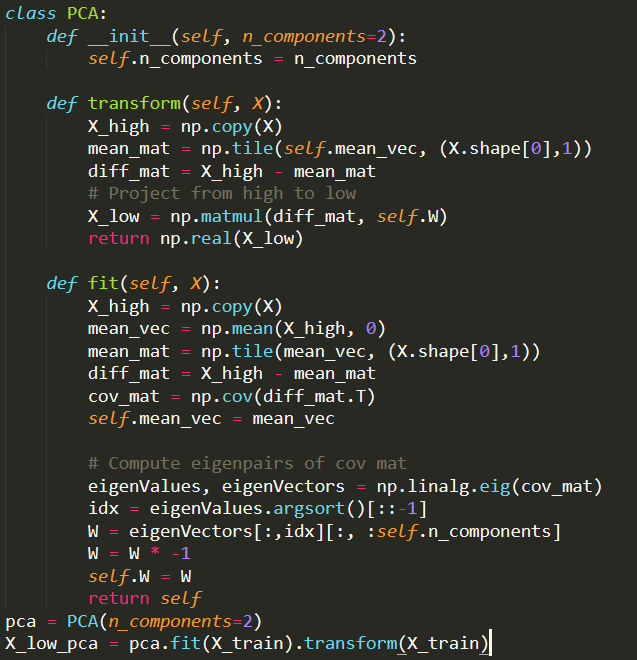
****

可以看到使用越多的 eigenFace 來重建，會越像本來的臉。

**Code Detail**

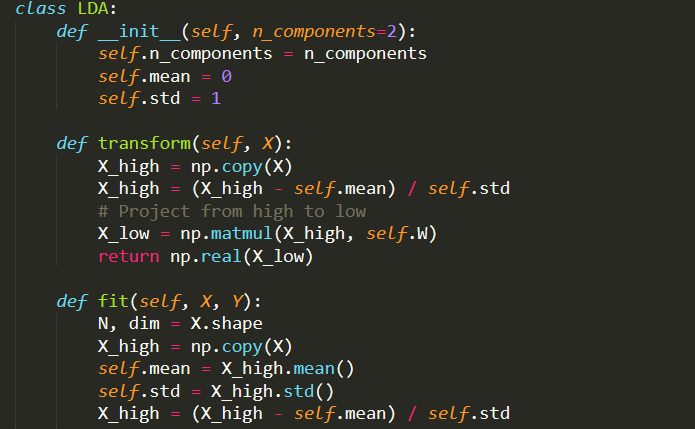
1. Read Data

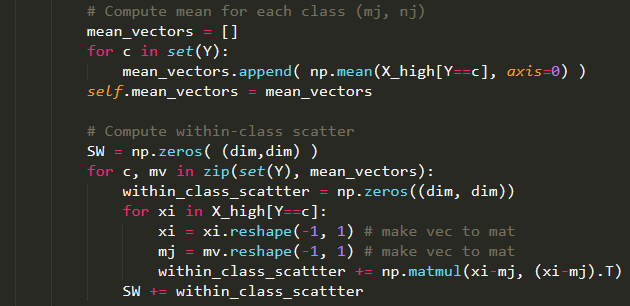


1. Use PCA to project all your data X\_train.csv onto 2D space

Compute first-k eigenvector of covariance matrix

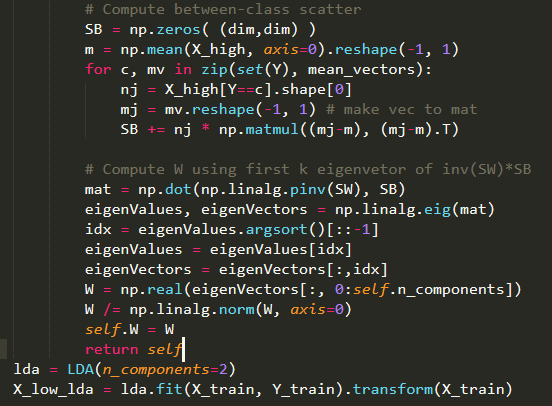
Compute covariance matrix

1. Use LDA to project all your data X\_train.csv onto 2D space



Compute Within-class

scatter

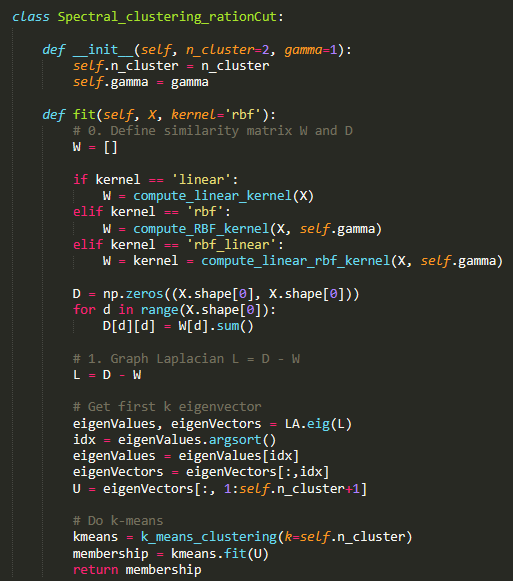


Compute Within-class scatter

Compute first-k largest

eigvector of inv(SW) \* SB

1. RatioCut



1. Normalized Cut



1. SVM

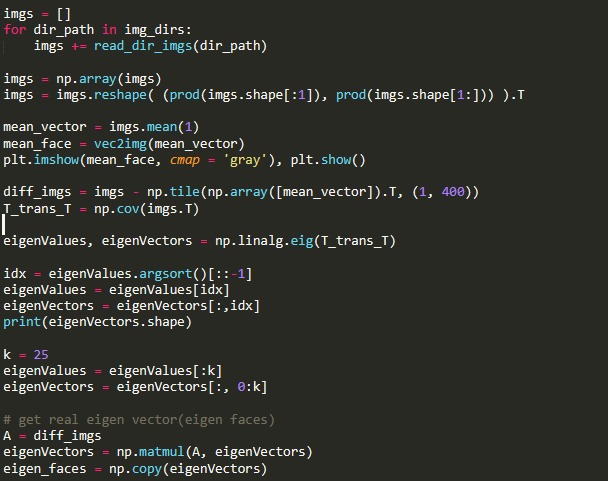
Linear SVM

Poly SVM

RBF SVM

Linear + RBF SVM

1. EigenFaces



Get 25 largest eigenvector

And project mean face to eigenspace to get 25 eigen faces

Compute Mean Face

And covariance matrix

Read Images

For more detail, please check EigenFace.py and HW7.py