SMAI Assignment 2

Aditya Yadavalli

November 3, 2019

Abstract

This is a short technical report. It summarises my findings after coding up the problems given.

1 Problem 1

1.1 How many eigen faces are needed to satisfactorily reconstruct the image

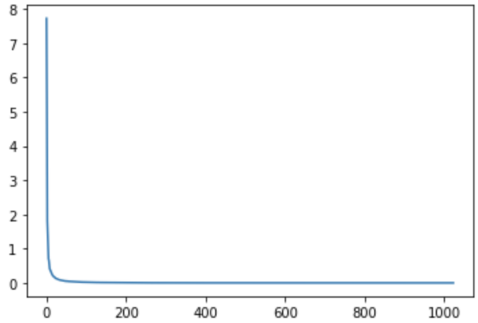
I have found the least number of eigen faces where the ratio of selected eigen faces divided the total number of eigen faces is equal to or greater than 0.95

$$x = \frac{\lambda_{selected}}{\lambda_{total}} > 0.95$$

where x is the number of number of values selected

1.1.1 For Dataset IMFDB

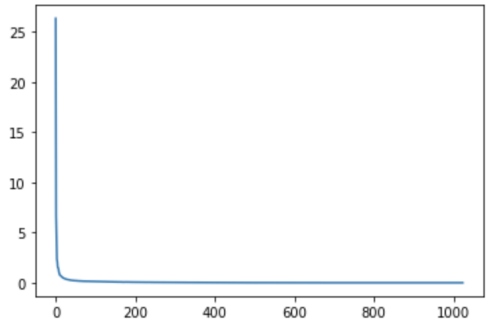
I found the value discussed above to be 108 out of 1024. The eigen faces contain more or less 95% of the information in the picture



if you look at the plots, you will notice that it drastically drops after the numbers mentioned above

1.1.2 For Dataset IIIT-CFW

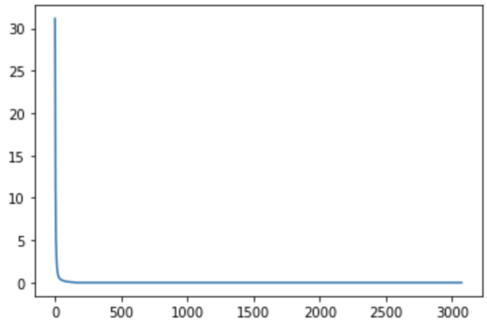
I found the value discussed above to be 309 out of 1024. The eigen faces contain more or less 95% of the information in the picture



if you look at the plots, you will notice that it drastically drops after the numbers mentioned above

1.1.3 For Dataset Yale Face Database

I found the value discussed above to be 62 out of 1024. The eigen faces contain more or less 95% of the information in the picture



if you look at the plots, you will notice that it drastically drops after the numbers mentioned above

1.2 Reconstruct Images

One can reconstruct images by taking dot product of the selected eigenvectors to compress and the data.

1.2.1 For Dataset IMFDB

The error (metric used is similar to euclidean distance) is 0.04

1.2.2 For Dataset IIIT-CFW

The error (metric used is similar to euclidean distance) is 0.06

1.2.3 For Dataset Yale Face Database

The error (metric used is similar to euclidean distance) is 0.14

1.3 Which person is difficult to represent with fewer eigen vectors

I have classified each person according to his/her class and did the same procedure as before.

1.3.1 For Dataset IMFDB

Madhuri(class 0) is difficult to represent using fewer eigen faces (35) while Katrina(class5) is the easiest to represent (27)

1.3.2 For Dataset IIIT-CFW

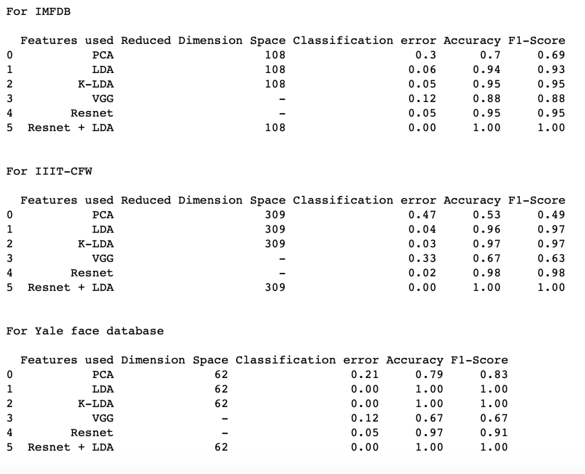
Man Mohan Singh (class 5) is difficult to represent using fewer eigen faces (79)

1.3.3 For Dataset Yale Face Database

Class 7 is slightly harder to represent with fewer eigen faces.

2 Problem 2

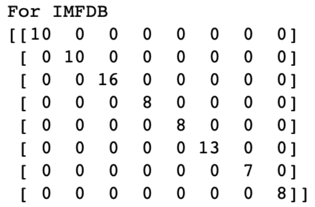
I have selected 6 combinations of features and classifiers. In fact, I used only MLP as my classifier and fiddled with different combinations of features provided to me.



```
For IMFDB

  Features used Reduced Dimension Space Classification error Accuracy F1-Score
0           PCA                     108                  0.3      0.7     0.69
1           LDA                     108                 0.06     0.94     0.93
2         K-LDA                     108                 0.05     0.95     0.95
3           VGG                       -                 0.12     0.88     0.88
4        Resnet                       -                 0.05     0.95     0.95
5  Resnet + LDA                     108                 0.00     1.00     1.00


For IIIT-CFW

  Features used Reduced Dimension Space Classification error Accuracy F1-Score
0           PCA                     309                 0.47     0.53     0.49
1           LDA                     309                 0.04     0.96     0.97
2         K-LDA                     309                 0.03     0.97     0.97
3           VGG                       -                 0.33     0.67     0.63
4        Resnet                       -                 0.02     0.98     0.98
5  Resnet + LDA                     309                 0.00     1.00     1.00


For Yale face database

  Features used Dimension Space Classification error Accuracy F1-Score
0           PCA             62                 0.21     0.79     0.83
1           LDA             62                 0.00     1.00     1.00
2         K-LDA             62                 0.00     1.00     1.00
3           VGG              -                 0.12     0.67     0.67
4        Resnet              -                 0.05     0.97     0.91
5  Resnet + LDA             62                 0.00     1.00     1.00
```

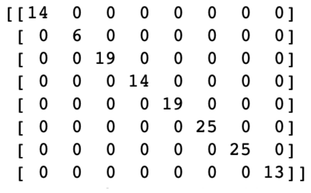
This table is updated in the notebook but not here. Classification error is wrong here. Please look at the notebook for right information.

2.0.1 Confusion matrix for IMFDB



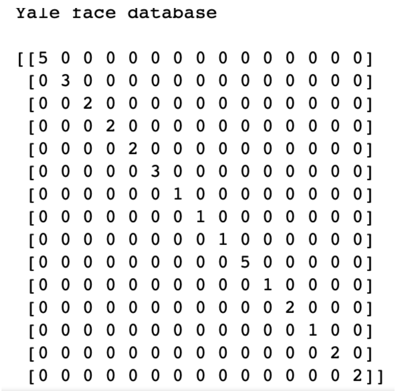
```
For IMFDB
[[10  0  0  0  0  0  0  0]
 [ 0 10  0  0  0  0  0  0]
 [ 0  0 16  0  0  0  0  0]
 [ 0  0  0  8  0  0  0  0]
 [ 0  0  0  0  8  0  0  0]
 [ 0  0  0  0  0 13  0  0]
 [ 0  0  0  0  0  0  7  0]
 [ 0  0  0  0  0  0  0  8]]
```

2.0.2 Confusion matrix for IIIT-CFW



```
[[14  0  0  0  0  0  0  0]
 [ 0  6  0  0  0  0  0  0]
 [ 0  0 19  0  0  0  0  0]
 [ 0  0  0 14  0  0  0  0]
 [ 0  0  0  0 19  0  0  0]
 [ 0  0  0  0  0 25  0  0]
 [ 0  0  0  0  0  0 25  0]
 [ 0  0  0  0  0  0  0 13]]
```

2.0.3 Confusion matrix for Yale Face Database



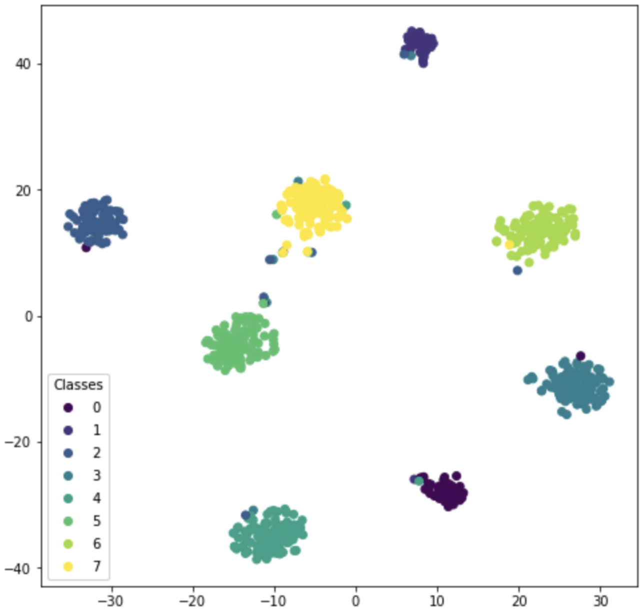
```
Yale face database

[[5 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 3 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 2 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 2 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 2 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 3 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 1 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 1 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 5 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 2 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 1 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 2 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 0 2]]
```

3 Problem 3

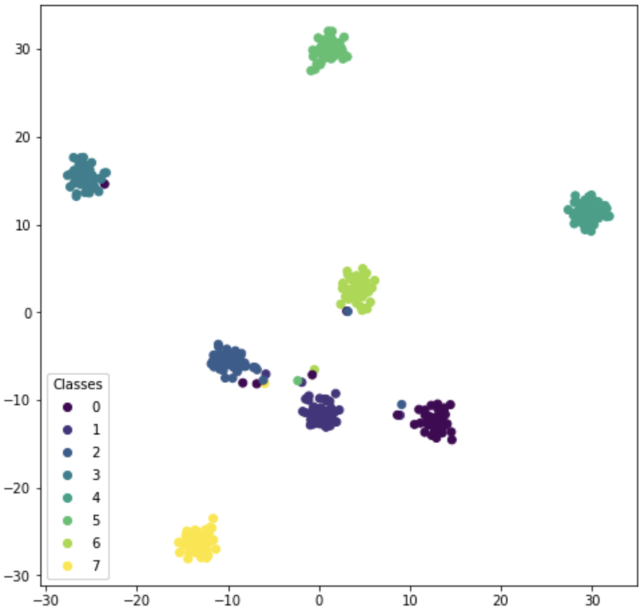
3.1 t-SNE

I have used several combinations of features to represent the features and found out that the best is either LDA or κ LDA. Even Resnet performs equally well but I chose to show using LDA here.

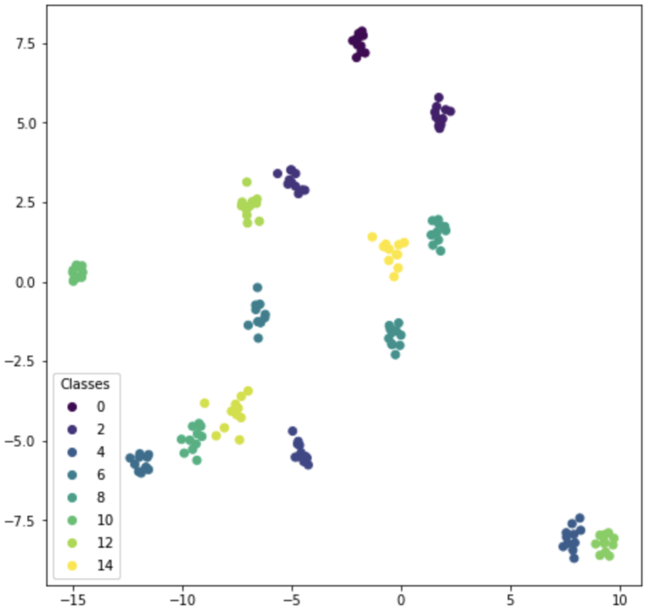
3.1.1 For IIIT-CFW



3.1.2 For IMFDB



3.1.3 For Yale Face Database



If you look at the above plots, you will notice that images belonging to the same class are coming together.

4 Problem 4

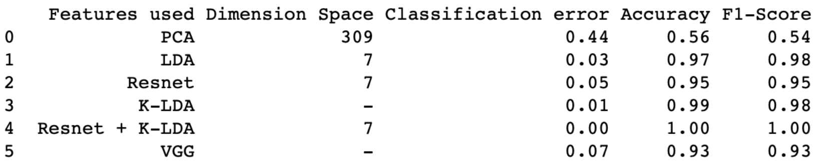
4.1 Face Verification using KNN

I have extracted some features and applied KNN on that. After testing a few combinations, I figured out that Resnet ,LDA or κ-LDA perform well. Resnet + LDA or Resnet + κ-LDA performs the best.

4.1.1 IIIT-CFW

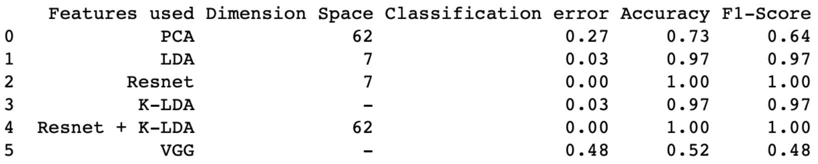
```
    Features used Dimension Space Classification error Accuracy F1-Score
0             PCA             108                 0.68     0.32     0.30
1             LDA               7                 0.06     0.94     0.93
2          Resnet               7                 0.01     0.99     0.98
3           K-LDA               -                 0.04     0.96     0.94
4  Resnet + K-LDA               7                 0.00     1.00     1.00
5             VGG               -                 0.33     0.67     0.59
```

4.1.2 IMFDB



```
     Features used Dimension Space Classification error Accuracy F1-Score
0              PCA             309                 0.44     0.56     0.54
1              LDA               7                 0.03     0.97     0.98
2           Resnet               7                 0.05     0.95     0.95
3            K-LDA               -                 0.01     0.99     0.98
4  Resnet + K-LDA               7                 0.00     1.00     1.00
5              VGG               -                 0.07     0.93     0.93
```

4.1.3 Yale Face Database

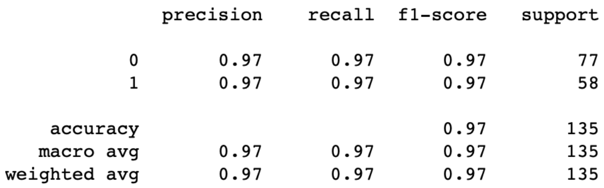


```
     Features used Dimension Space Classification error Accuracy F1-Score
0              PCA              62                 0.27     0.73     0.64
1              LDA               7                 0.03     0.97     0.97
2           Resnet               7                 0.00     1.00     1.00
3            K-LDA               -                 0.03     0.97     0.97
4  Resnet + K-LDA              62                 0.00     1.00     1.00
5              VGG               -                 0.48     0.52     0.48
```

One will notice that Resnet+κ-LDA gives almost 100% accuracy in all datasets

5 Application Problem

I have chosen Politicians vs Actors problem out of the 6 choices given. In my architecture, I have one hidden layer in my MLP. Took 20 neurons per layer. Anything more will be overparameterising. Anything less does not give as good results as this. After careful analysis, I figured out that Resnet performs well. The below image shows the results.



```
              precision    recall  f1-score   support

           0       0.97      0.97      0.97        77
           1       0.97      0.97      0.97        58

    accuracy                           0.97       135
   macro avg       0.97      0.97      0.97       135
weighted avg       0.97      0.97      0.97       135
```