AIML Report

Principal Component Analysis

Ramnath Pillai

Fall, 2015

Homework 4

Q1.a Refer code **Q1.m**

Following are the steps involved in generating the principal components of the set.

We have the initial dataset X. First we find the mean of the data and subtract it from the original dataset and get the following:

D=X-meanX

Then we find the covariance matrix using:

S=D x D^T

After obtaining the S matrix, we find the eigenvectors and eigenvalues of this matrix. The eigenvectors are normalized (optional) and then the eigenvectors are arranged in the increasing order of the eigenvalues. The eigenvectors corresponding to the higher eigenvalues show more scatter and therefore is more important compared to the other eigenvectors. e1 and e2 are the eigenvectors. Following is a plot of the data and its principal components (shown in red)

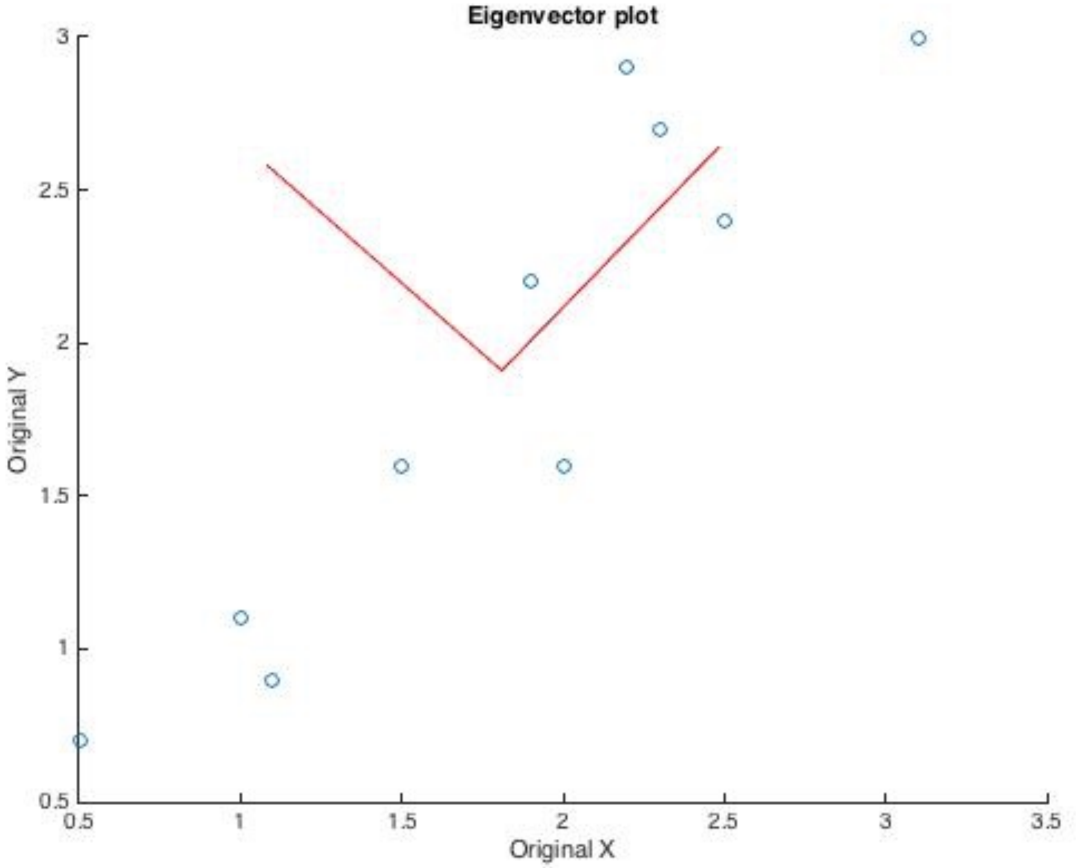
We can now determine the coefficients in the new space using the equation

$ak=E^T$ x D

The eigenvectors for this problem obtained are:

e1= [-0.7352   0.6779]

e2= [0.6779   0.7352]



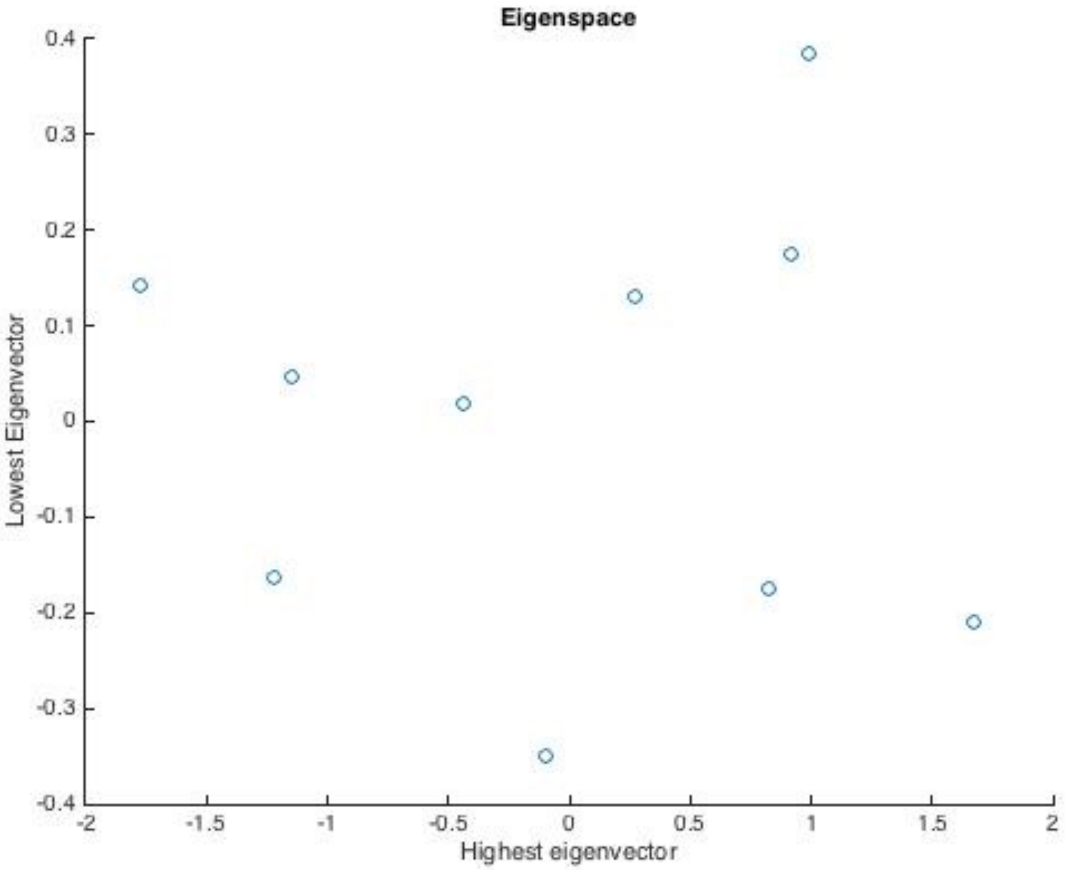
Q1.b These are the new coefficients for the new eigenspace. The coordinates in this new reference frame is shown below:

Computed values for a1 and a2 are:

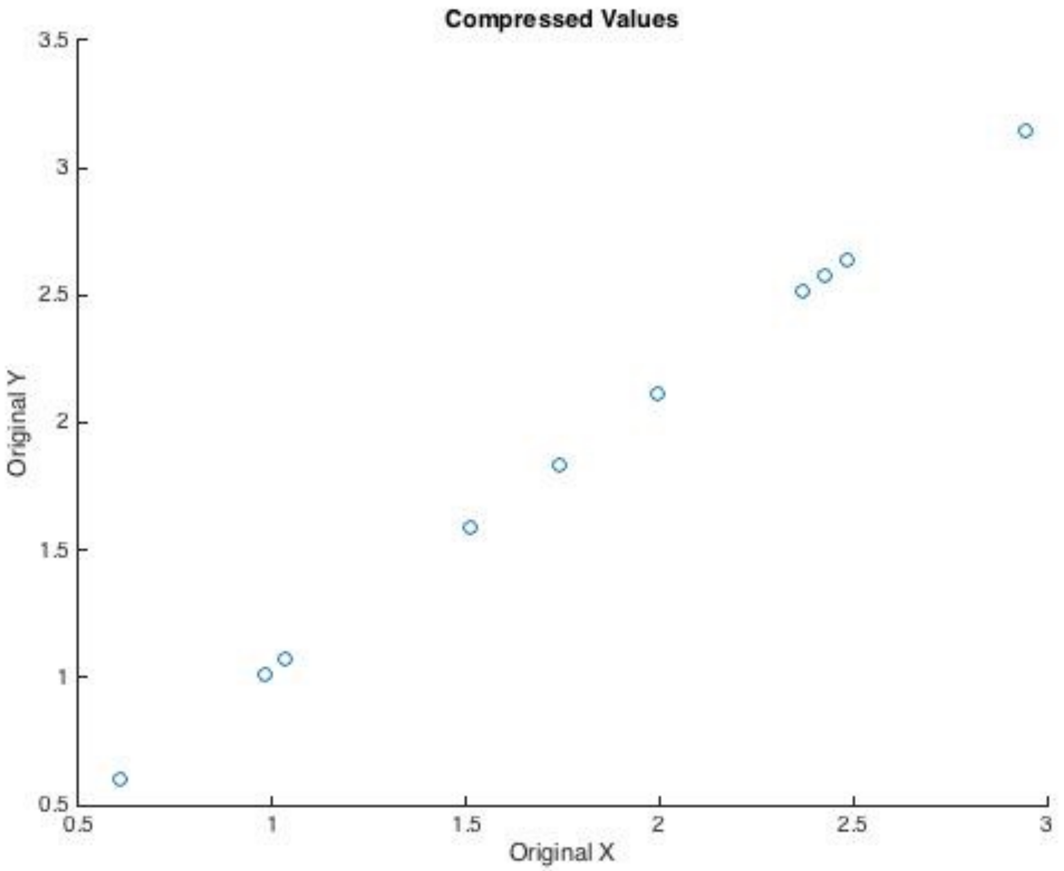
```
ak =


   -0.1751    0.8280
    0.1429   -1.7776
    0.3844    0.9922
    0.1304    0.2742
   -0.2095    1.6758
    0.1753    0.9129
   -0.3498   -0.0991
    0.0464   -1.1446
    0.0178   -0.4380
   -0.1627   -1.2238
```

These points are equivalent to the figure 1 rotated clockwise by a suitable angle.



Q1.c We can now delete the lowest eigenvector and represent the entire data in a one dimensional form as shown in the figure below. The range of data is 3.4534 between the points (0.605,0.6032) and (2.946,3.142) in the original XY space.


Compressed Values
Original Y
Original X
3.5
3
2.5
2
1.5
1
0.5
0.5
1
1.5
2
2.5
3

Q2. Refer code Q2.m: In the following question, we have the dimensionality greater than the sample size. This implies that if we use the traditional approach several of the eigenvalues will be zero due to rank deficiency. The first few steps of the derivation are similar to Q1.

We have the initial dataset X. First we find the mean of the data and subtract it from the original dataset and get the following:

D=X-meanX

Now, we find

S= D^TxD which is a 4x4 matrix.

Let vi be the eigenvectors of the matrix S.

Thus, the eigenvectors of the original matrix D D^T is given by

A x vi

Now we have ui = A x vi.

The original data can be reconstructed using a weighted sum of these ui as shown in the Q1. The reconstruction is equal to:

x=mean + a1 v1 + a2 v2 + …. + an vn

In order to reduce dimensionality we choose K prominent eigenvectors to reconstruct the data and this x is given by

x=mean + a1 v1 + a2 v2 + …. + aK vK

Q2.a The inner product matrix computed in the following question is given as:

S =

```
   69.8750  -18.8750  -26.3750  -24.6250
  -18.8750  121.3750  -53.1250  -49.3750
  -26.3750  -53.1250   98.3750  -18.8750
  -24.6250  -49.3750  -18.8750   92.8750
```

The value of the minimum mean squared error is given by the sum of the eigenvalues of the orthogonal subspace to the PCA subspace. Using this the minimum squared error representation of the data in 3D space is given as:

The transpose of the X reconstructed is shown for legibility.

```
reconstX_3’=

   -2.0000    1.0000    1.0000    3.0000
    1.0000    2.0000   -3.0000   -1.0000
    2.0000   -4.0000    2.0000   -0.0000
   -3.0000    2.0000    1.0000    2.0000
    4.0000   -4.0000   -0.0000    2.0000
    1.0000    2.0000   -3.0000   -5.0000
    0.0000    5.0000   -5.0000   -4.0000
    3.0000    2.0000   -1.0000   -1.0000
    0.0000    2.0000    3.0000    2.0000
    2.0000    1.0000    3.0000   -1.0000
    1.0000   -3.0000   -2.0000    3.0000
    1.0000   -0.0000   -3.0000    4.0000
    2.0000   -0.0000   -2.0000    4.0000
    3.0000    1.0000   -1.0000    2.0000
   -2.0000   -2.0000    1.0000    1.0000
   -3.0000    1.0000    0.0000    2.0000
    2.0000    1.0000    5.0000   -2.0000
    1.0000   -3.0000    4.0000    1.0000
   -0.0000   -2.0000    2.0000   -1.0000
```

The rms error is equal to $\sum(X-Xreconst)^2$ for each data point and is given to be:

```
1.0e-14 *
    0.0894
    0.0645
    0.1199
    0.0651
```

for each data point.

Q2.b. The weights required to generate a 3D representation of the data is given by:

```
   -3.1769   -1.3127    0.9665
    1.1239   -1.2603   -3.4494
    2.6734    1.0271    3.9747
   -4.0920   -0.4850   -0.1414
    4.2499   -1.8743    3.6640
    2.3027    1.6126   -4.9847
    0.5445   -0.2422   -7.8514
    2.4080    0.0397   -2.6363
   -1.9455    0.7737    0.6054
    1.0110    2.7777    0.1118
    1.4613   -3.7830    2.5106
    0.3451   -4.9878    0.0504
    0.9871   -4.3378    0.4570
    1.8712   -2.1832   -0.6944
   -1.5455   -0.0809    2.5700
   -3.5242   -1.2445    0.1761
    0.8648    4.8784    0.6737
    0.7279    1.8008    4.5801
    0.5484    1.9642    2.1427
```

Q2.c In this case, the mean squared error comes out to be:

```
1.0e-14 *
    0.0894
    0.0645
    0.1199
    0.0651
```

for each data point.

Q2.d The new error in the data is found to be increased and is equal to

```
    1.9044
    0.7669
    0.4868
    0.6507
```

for each data sample

Q2. e,f The euclidean distance of the given dataset to the 3D space is given by

```
dist_3D =


   12.9228    5.6569   16.7929   15.7797
```

and to the original 4D space is given by

```
dist_4D =


   12.9228    5.6569   16.7929   15.7797
```

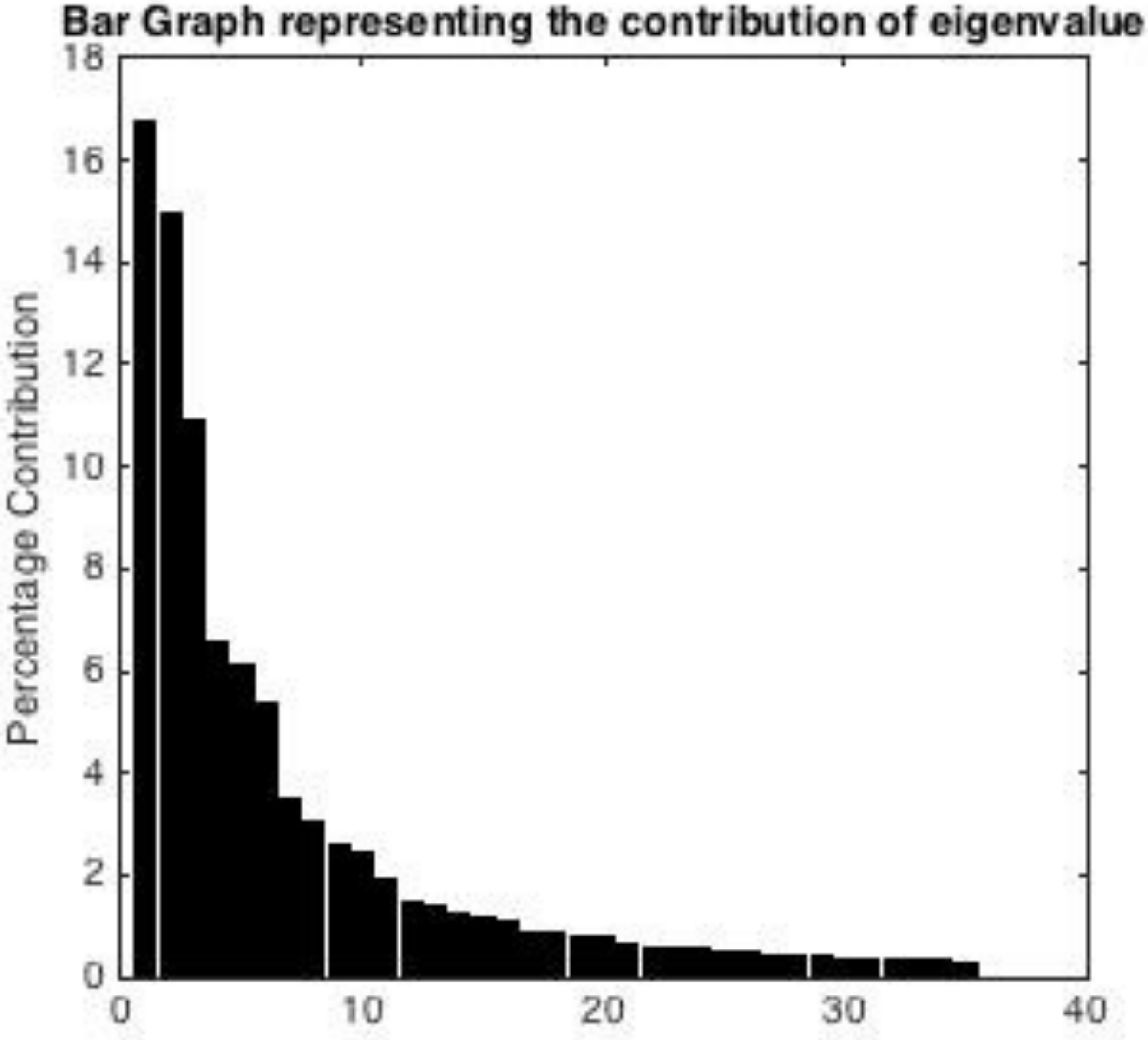
It is most similar to the SECOND SAMPLE.

Both of them come out to be the same because the rms error in dimensionality reduction is extremely less and can be neglected. This makes intuitive sense.

Q3. a. Refer Code Q3.m The new number of dimensions is 35. Following is the visualization of the first 5 eigenfaces for training samples:

Shown below is the bar graph showing the prominence of the eigenvectors from the one corresponding to the highest eigenvalue to that of the lowest eigenvalue after allowing 90% variation in the PCA.

Sum of the percentages =90%



Q3.b

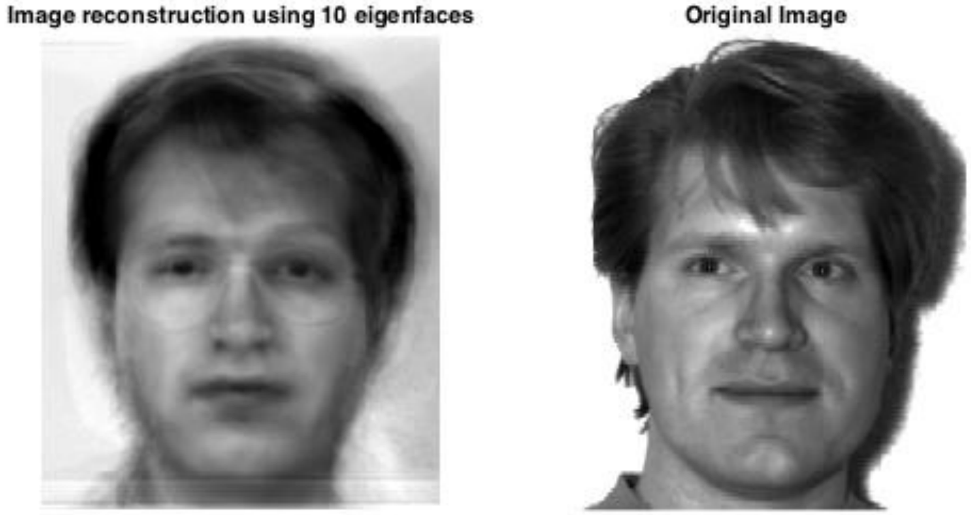


Image reconstruction using 10 eigenfaces

Original Image

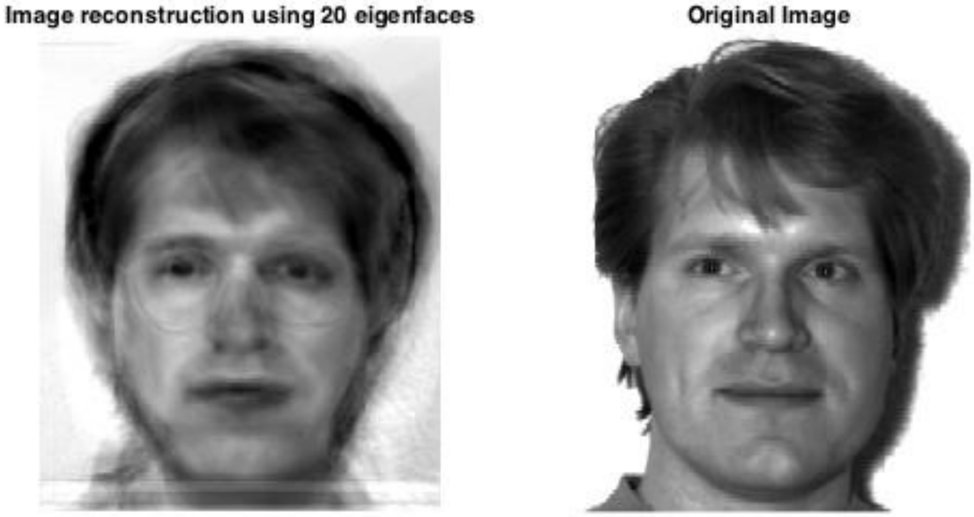


Image reconstruction using 20 eigenfaces

Original Image

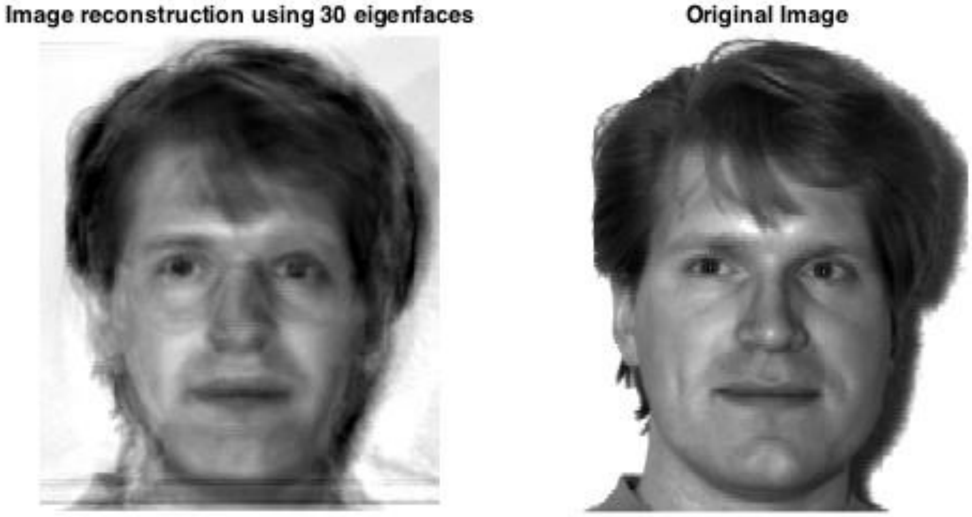
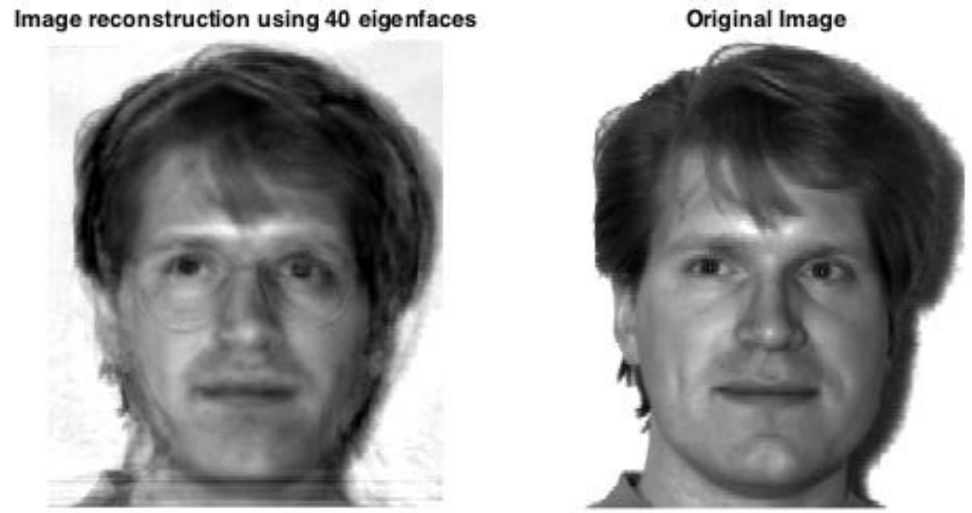
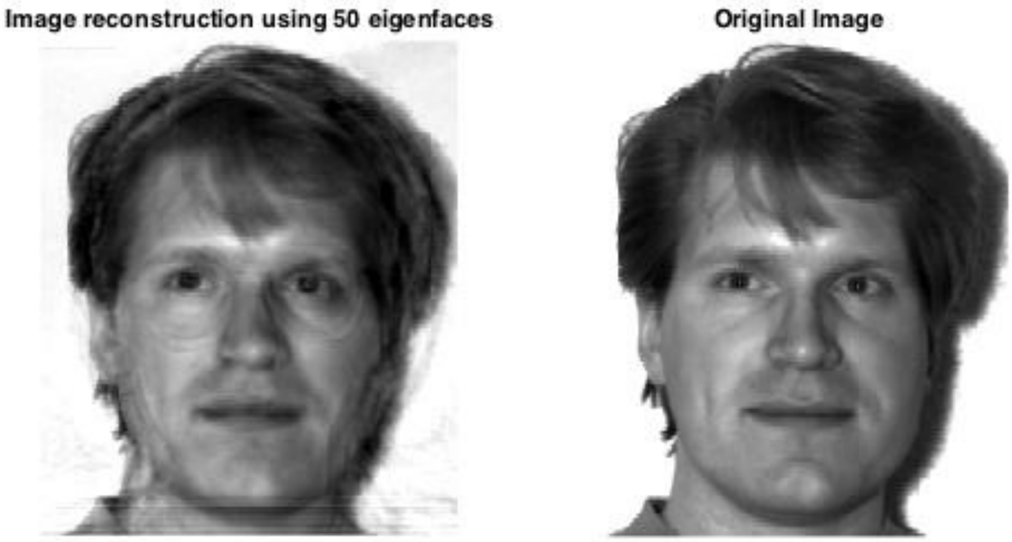


Image reconstruction using 30 eigenfaces

Original Image





Q3.c Refer **Q3_b.m** The accuracy that is obtained from 30 test data using the nearest neighbor algorithm is 24/30 on the reduced space.

Q3.d. The accuracy that is obtained from 30 test data using the nearest neighbor algorithm is 25/30 (83.3%) on the complete space. It becomes 24/30 using the reconstructed space of PCA. We would have expected a higher accuracy in this case as the dimensionality is higher and the level of detail of the images is higher. And it is exactly as the intuition suggests. This can be reasoned out by the fact that the PCA reduces the accuracy by approximating the function along the principal directions by neglecting some of the small variations.Since we have considered a 90% variation in the reduced space, the accuracy changes only very slightly as compared to a case where we reduce the allowed variation to around 75%. As a given face is approximated as the weighted sum of eigenfaces, the ambiguity in the face increases when we reduce the allowed variation.