Michael Chan

Accuracies for N=30 and C=10

```
result =

    1.0000    0.9417    0.5750    0.2357    0.1316


result15 =

    1.0000    0.9083    0.2917    0.6571    1.0000


resultF =

    1.0000    0.9833    0.5750    0.1929    0.1368


resultF15 =

    1.0000    0.8667    0.4667    0.5000    1.0000
```

	Set 1 Accuracy	Set 2 Accuracy	Set 3 Accuracy	Set 4 Accuracy	Set 5 Accuracy
Eigenfaces trained on subset 1 N=30	1.00	0.9417	0.5750	0.2357	0.1316
Eigenfaces trained on subset 1 and 5 N=30	1.00	0.9083	0.2917	0.6571	1.00
Fisherfaces trained on subset 1 C=10	1.00	0.9833	0.5750	0.1929	0.1368
Fisherfaces trained on subset 1 and 5 C=10	1.00	0.8667	0.4667	0.50	1.00

Error Rates for N=30 and C=10

```
result =

         0    0.0583    0.4250    0.7643    0.8684


result15 =

         0    0.0917    0.7083    0.3429         0


resultF =

         0    0.0167    0.4250    0.8071    0.8632


resultF15 =

         0    0.1333    0.5333    0.5000         0
```

	Set 1 Error	Set 2 Error	Set 3 Error	Set 4 Error	Set 5 Error
Eigenfaces trained on subset 1 N = 30	0	0.0583	0.4250	0.7643	0.8684
Eigenfaces trained on subset 1 and 5 N = 30	0	0.0917	0.7083	0..8071	0
Fisherfaces trained on subset 1 C= 10	0	0.0167	0.4250	0.8071	0.1368
Fisherfaces trained on subset 1 and 5 C = 10	0	0.1333	0.5333	0.500	0

Accuracies for N=9 and C=31

```
result =

    1.0000    0.9167    0.4667    0.1929    0.1211


result15 =

    1.0000    0.5750    0.1917    0.3357    1.0000


resultF =

    1.0000    0.9833    0.5750    0.1929    0.1368


resultF15 =

    1.0000    0.8667    0.4667    0.5000    1.0000
```

	Set 1 Accuracy	Set 2 Accuracy	Set 3 Accuracy	Set 4 Accuracy	Set 5 Accuracy
Eigenfaces trained on subset 1 N = 9	1.0	0.9167	0.4667	0.1929	0.1211
Eigenfaces trained on subset 1 and 5 N = 9	1.0	0.5750	0.1917	0.3357	1.0
Fisherfaces trained on subset 1 C= 31	1.0	0.9833	0.5750	0.1929	0.1368
Fisherfaces trained on subset 1 and 5 C = 31	1.0	0.8667	0.4667	0.5	1.0

Error Rate for N=9 and C=31

```
result =

         0    0.0833    0.5333    0.8071    0.8789


result15 =

         0    0.4250    0.8083    0.6643         0


resultF =

         0    0.0167    0.4250    0.8071    0.8632


resultF15 =

         0    0.1333    0.5333    0.5000         0
```

	Set 1 Error	Set 2 Error	Set 3 Error	Set 4 Error	Set 5 Error
Eigenfaces trained on subset 1 N = 9	0	0.0833	0.5333	0.8071	0.8789
Eigenfaces trained on subset 1 and 5 N = 9	0	0.4250	0.8083	0.6643	0
Fisherfaces trained on subset 1 C= 31	0	0.0167	0.4250	0.8071	0.8632
Fisherfaces trained on subset 1 and 5 C = 31	0	0.1333	0.5333	0.500	0

Eigenface Algorithm

1) Find mean face

$$uj = \frac{1}{n} * \sum(subset)$$

2) Subtract mean from each column of subset

$$x = subset(i) - uj$$

3) Now that data is preprocessed, solve for eigenvectors of the preprocessed data.

$$[U, S, V] = svd(x)$$

4) Now that we have the largest eigenvectors of the data, reduce it to k eigenvectors.

$$ureduce = U(:, 1{:}k)$$

5) We now have our the top eigenfaces for the subset, to project it onto our eigenspace w, we use the equation

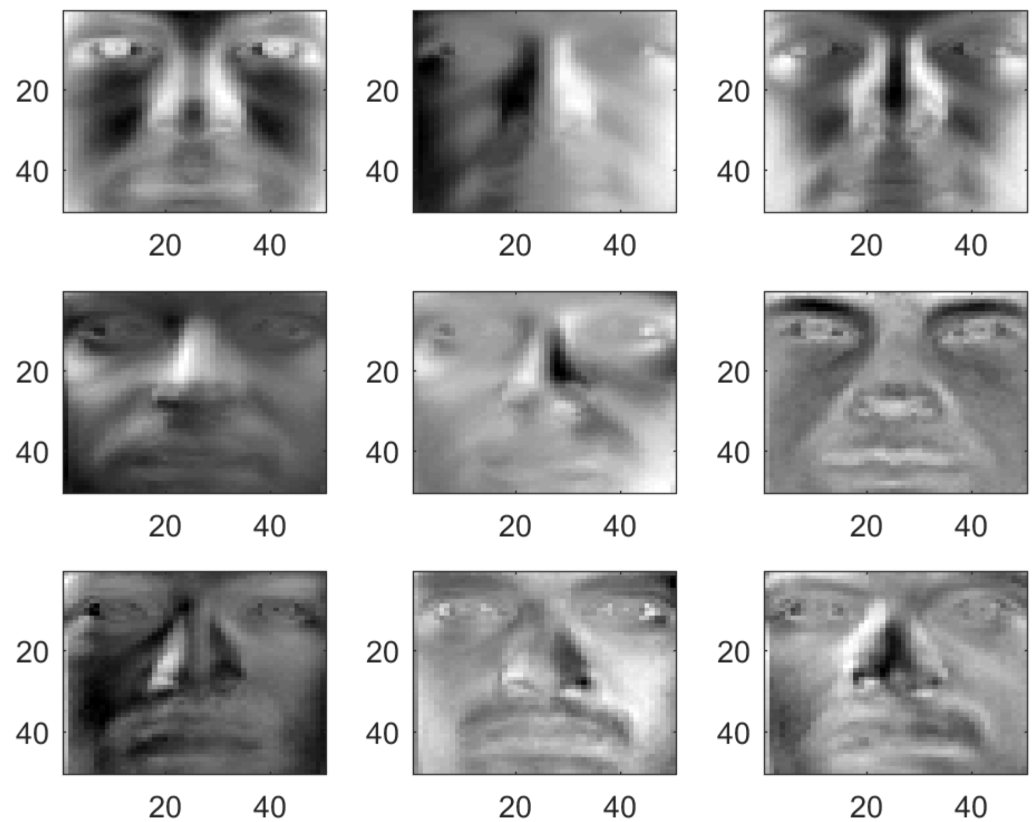
$$w_{train} = ureduce' * x$$

6) We can then apply the ureduce to any new subset x to test on

$$w_{test} = ureduce' * x_{test}$$

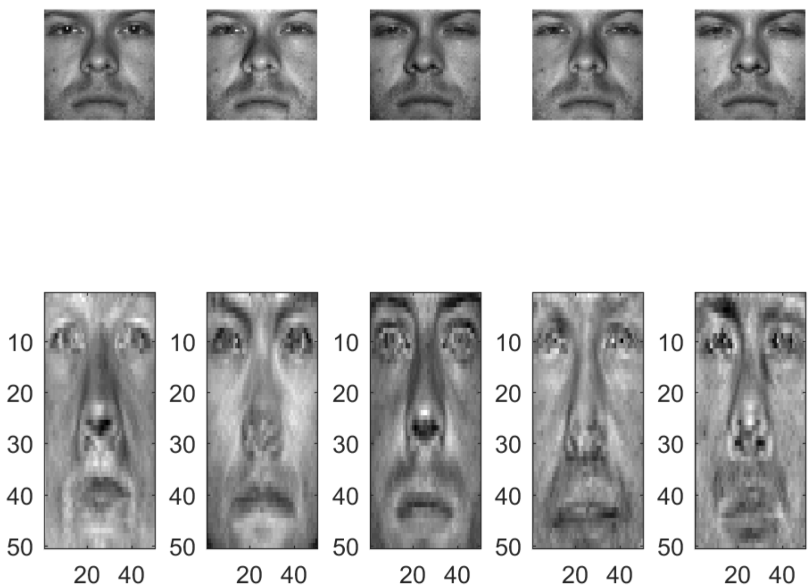
7) Then we compare the training eigenspace and the test eigenspace against each other using nearest neighbors algorithm.

Top 9 Eigenvectors after training on subset 1

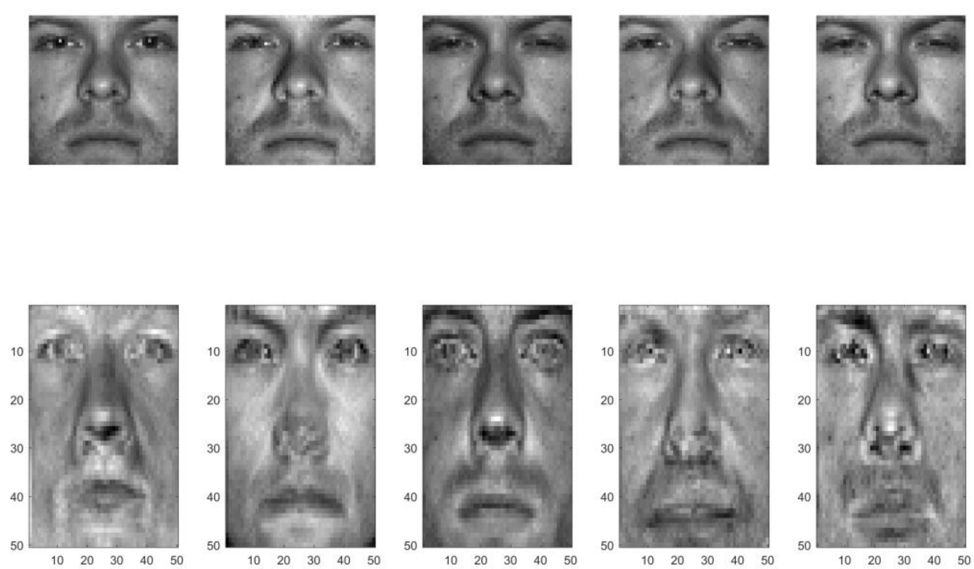


Reconstructed Eigenfaces

D = 9



D = 30



Fisherfaces Algorithm

1) Do the eigenface algorithm as shown above. We ended with our eigenspace w.

$$w_{train} = ureduce' * x$$

2) FLD requires us to calculate the different scatters of the data. First, we calculate the scatter of each individual class.

$$S_i = \sum(x_k - u_i) * (x_k - u_i)'$$

Where u_i = the local mean of the class i and x_k is the subset for class i.

3) Now that we have the scatter of each individual class, we can then calculate for the within class scatter using the equation

$$S_w = \sum S_i$$

4) In addition to this, we also need to calculate our between class scatter, which requires the mean of the whole dataset.

$$S_b = \sum N_i(u_i - u)(u_i - u)'$$

5) Now that we have Sw and Sb, we then solve the generalize eigenvector problem associated through the use of Sw and Sb.

$$[V, D, W_{fld}] = eig(Sb, Sw)$$

6) Now that we have the FLD of the data, we can now compute Wopt by applying our FLD to our PCA.

$$W'_{opt} = W_{fld}' * ureduce$$

Extra:

Accuracies when cutting out the top 3 eigenvectors for Eigenfaces

```
result =

    1.0000    1.0000    0.7667    0.2929    0.1211


result15 =

    1.0000    1.0000    0.9333    0.8929    1.0000
```