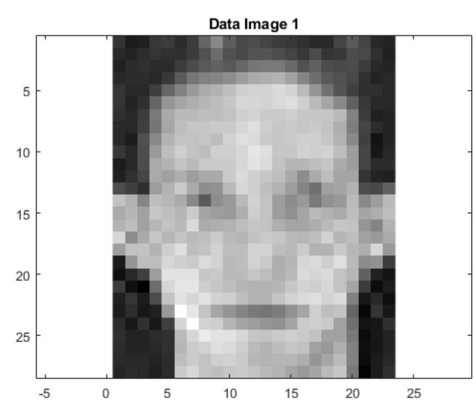
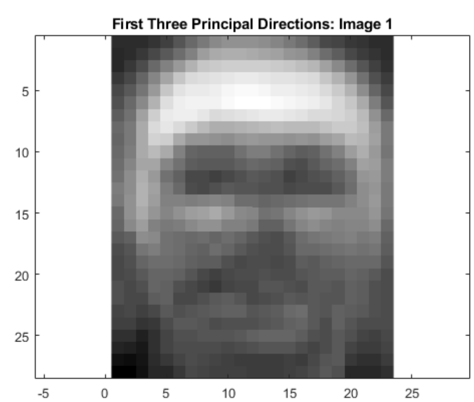
MATLAB (Problems 1-2)

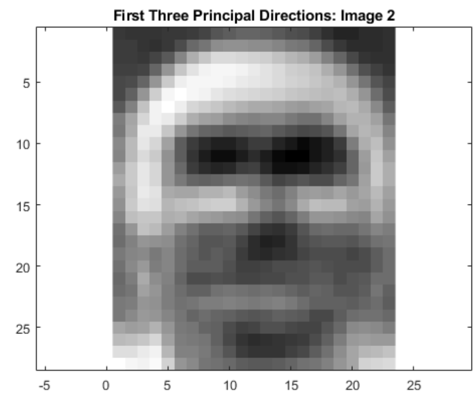
Output:

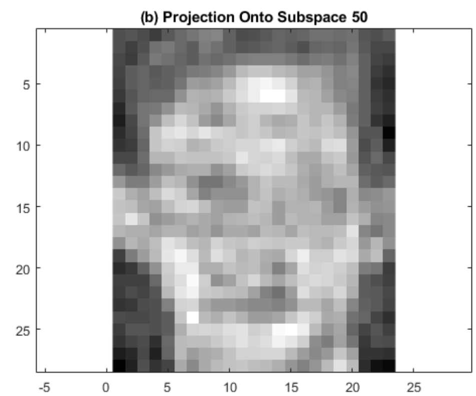
```
-------------------------------------------------------------
Oscar Martinez          Homework 4: Problems 1-2          STA 5106
-------------------------------------------------------------
-----Problem 1-----
m =

200


n =

644

---Part (a)---
---Part (b)---
-----Problem 2
--------Part (a)---
---Part (b)---

U =

0.0144
0.0195
0.0169

---Part (c)---
```

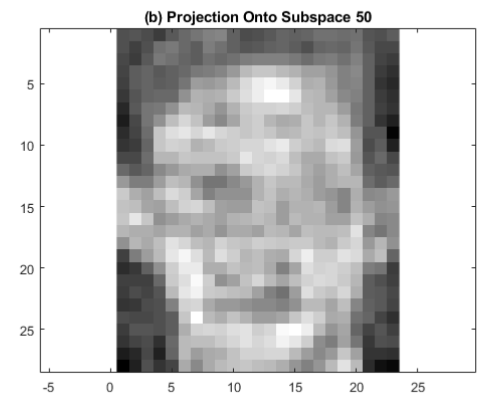
Figures:

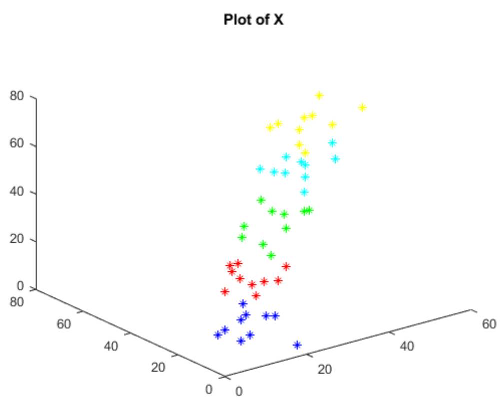


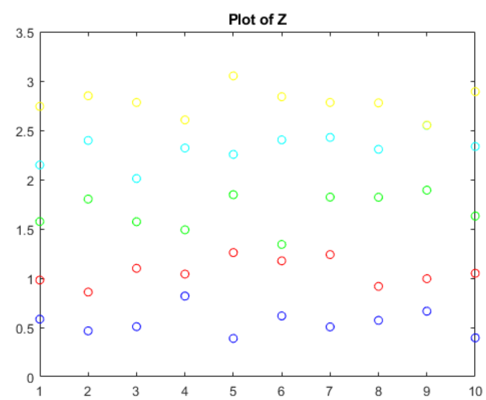












Code:

```
clc
clear

%Diary
dfile ='MATLAB_Output_OM.txt';
if exist(dfile, 'file') ; delete(dfile); end
diary(dfile)
```

```
diary on
diary MATLAB_Output_OM.txt

%Introduction
fprintf('-----------------------------------------------------------------------------\n');
fprintf('Oscar Martinez \t Homework 4: Problems 1-2 \t STA 5106\n');
fprintf('-----------------------------------------------------------------------------\n');

%-------Problem 1:-------
fprintf('-------Problem 1-------\n');

%Load the data
load hw4_1_data.mat

%size
[m,n] = size(X)

%Part a
fprintf('----Part (a)----\n');

for i = 1:m
I = reshape(X(i,:),28,23); %Reshape column i of X into a 28x23 matrix
imagesc(I); %Takes the reshapen column/matrix and converts it to an image
colormap(gray); %The image's color scheme is grayscale
axis equal; %Equal Margins
title(['Data Image ', num2str(i)]) %Title
pause; %Wait until user presses a key to move on to the next image
end;

%PCA
C = cov(X); %Variance Covariance
[U,S,V]=svd(C); %SVD decomposition w/ U = V being our matrix of interest
U1=U(:,1:3); %U1 is the first 3 columns of U ie 3 principal directions
%Z=X*U1; %Z is our typical transformation but is unused

%1st three components
for i = 1:3
I = reshape(U1(:,i),28,23); %Reshape column i of U1 into a 28x23 matrix
imagesc(I); %Takes the reshapen column/matrix and converts it to an image
colormap(gray); %The image's color scheme is grayscale
axis equal; %Equal Margins
title(['First Three Principal Directions: Image ', num2str(i)]) %Title
pause; %Wait until user presses a key to move on to the next image
```

```
end

%Part b
fprintf('---Part (b)---\n');

%Show the projection of the first image
i=0; %start counter at 0
for d = [50 100] %Doing this for d=50 and d=100
i = i+1; %Update the counter by 1
Z(:,i) = U(:,1:d)*(X(1,:)*U(:,1:d))'; %Create a new matrix Z, whose ith column corresponds to the ith value of d as per the prompt
end

d = [50 100];
for i = 1:size(Z, 2) %For i from 1 to the # of columns in Z (here 2)
I = reshape(Z(:,i),28,23); %Reshape column i of Z into a 28x23 matrix
imagesc(I); %Takes the reshapen column/matrix and converts it to an image
colormap(gray); %The image's color scheme is grayscale
axis equal; %Equal Margins
title(['(b) Projection Onto Subspace ', num2str(d(1,i))]) %Title
pause %Wait until user presses a key to move on to the next image
end

%-----Problem 1:-----
clear
fprintf('-----Problem 2-----\n');

%Load the data
load hw4_2_data.mat

%Relevant Details
% m = 5 classes
%k = 10 observations
%n = 3 size of vector per observation
%d = 1

%Part a
fprintf('---Part (a)---\n');

%Plot X
[n, m, k] = size(X);
figure(1);
color = 'brgcy';
for j = 1:m
for i = 1:k
```

```
plot3(X(1,j,i), X(2,j,i), X(3,j,i), [color(j) '*']);
%grid on;
title('Plot of X');
hold on;
end
end

%Part b
fprintf('——Part (b)——\n');

%Class Means
for i = 1:n
for j = 1:m
mu(i,j) = mean(X(i,j,:));
end
end

%Total Mean
MU = mean(mu, 2); %column vector w/ mean of each row of mu

%Between-class Scatter Matrix
S_B = zeros(n, n);
for j = 1:m
S_B = S_B + (mu(:,j) - MU)*(mu(:,j) - MU)';
end

% compute within-class scatter matrix
S_W = zeros(n, n);
for j = 1:m
for i = 1:k
S_W = S_W + (X(:,j,i) - mu(:,j))*(X(:,j,i) - mu(:,j))';
end
end

% compute U
[U, D] = eig(S_B, S_W);
U = U(:,n)  % take the last component (w.r.t. largest generalized
    eigenvalue)

%Part c
fprintf('——Part (c)——\n');

% projection to Z
for j = 1:m
for i = 1:k
```

```
Z(i,j) = U'*X(:,j,i);
end
end

% plot Z
figure(2);
color = 'brgcy';
for j = 1:m
plot(Z(:,j), [color(j) 'o']);
title('Plot of Z');
hold on;
end

%closing output
diary off
```

Problem 3

In the LLE framework, we minimize the error for a point $\vec{X}$ with K neighbors $\vec{\eta}_j$

$$\left|\vec{X} - \sum_{j=1}^{K} W_j \vec{\eta}_j\right|^2 = \left|\sum_{j=1}^{K} W_j \left(\vec{X} - \vec{\eta}_j\right)\right|^2 = \sum_{jk} W_j W_k G_{jk}$$

Where the Gram matrix

$$G_{jk} = \left(\vec{X} - \vec{\eta}_j\right) \cdot \left(\vec{X} - \vec{\eta}_k\right)$$

Prove that the optimal reconstruction weights are

$$W_j = \frac{\sum_k [G^{-1}]_{jk}}{\sum_{lm} [G^{-1}]_{lm}}$$

Proof. First, observe that:

$$
\begin{aligned}
\left|\underbrace{\vec{X}}_{D\times 1} - \sum_{j=1}^{K} \underbrace{W_j}_{1\times 1} \underbrace{\vec{\eta}_j}_{D\times 1}\right|^2 &= \left|\sum_{j=1}^{K} W_j \vec{X} - \sum_{j=1}^{K} W_j \vec{\eta}_j\right|^2 \\
&= \left|\sum_{j=1}^{K} W_j \left(\vec{X} - \vec{\eta}_j\right)\right|^2 \\
&= \left(\sum_{j=1}^{K} W_j \left(\vec{X} - \vec{\eta}_j\right)\right)^T \left(\sum_{j=1}^{K} W_j \left(\vec{X} - \vec{\eta}_j\right)\right) \\
&= \left(\sum_{j=1}^{K} W_j \vec{\epsilon}_j\right)^T \left(\sum_{j=1}^{K} W_j \vec{\epsilon}_j\right), \quad \underbrace{\vec{\epsilon}_j}_{D\times 1} = \left(\vec{X} - \vec{\eta}_j\right) \\
&= \left(W_1 \vec{\epsilon_1^T} + W_2 \vec{\epsilon_2^T} + \cdots + W_K \vec{\epsilon_K^T}\right)^T \left(W_1 \vec{\epsilon}_1 + W_2 \vec{\epsilon}_2 + \cdots + W_K \vec{\epsilon}_K\right) \\
&= \left(\underbrace{\begin{bmatrix} W_1 & W_2 & \dots & W_K \end{bmatrix}}_{1\times K} \underbrace{\begin{bmatrix} \vec{\epsilon_1^T} \\ \vec{\epsilon_2^T} \\ \vdots \\ \vec{\epsilon_K^T} \end{bmatrix}}_{K\times D}\right) \left(\underbrace{\begin{bmatrix} \vec{\epsilon}_1 & \vec{\epsilon}_2 & \dots & \vec{\epsilon}_K \end{bmatrix}}_{D\times K} \underbrace{\begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_K \end{bmatrix}}_{K\times 1}\right) \\
&= \underbrace{\vec{W}^T}_{1\times K} \underbrace{G}_{K\times K} \underbrace{\vec{W}}_{K\times 1}, \quad G = \begin{bmatrix} g_{1,1} & g_{1,2} & \cdots & g_{1,K} \\ g_{2,1} & g_{2,2} & \cdots & g_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ g_{K,1} & g_{K,2} & \cdots & g_{K,K} \end{bmatrix}, \quad \underbrace{g_{i,j}}_{1\times 1} = \vec{\epsilon}_i^T \vec{\epsilon}_j
\end{aligned}
$$

Now, we construct and solve our Lagrangian:

$$
\begin{aligned}
\mathcal{L} &= \vec{W}^T G \vec{W} + \lambda(1 - \vec{W}^T \mathbb{1}) \\
\frac{\partial \mathcal{L}}{\partial \vec{W}^T} &= 2G\vec{W} - \lambda \mathbb{1} = 0 \\
&\Rightarrow 2G\vec{W} = \lambda \mathbb{1} \\
&\Leftrightarrow \vec{W} = \frac{1}{2} G^{-1} \lambda \mathbb{1} \\
&\Rightarrow \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_K \end{bmatrix} = \frac{1}{2} \begin{bmatrix} g_{1,1}^{-1} & g_{1,2}^{-1} & \cdots & g_{1,K}^{-1} \\ g_{2,1}^{-1} & g_{2,2}^{-1} & \cdots & g_{2,K}^{-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{K,1}^{-1} & g_{K,2}^{-1} & \cdots & g_{K,K}^{-1} \end{bmatrix} \begin{bmatrix} \lambda \\ \lambda \\ \vdots \\ \lambda \end{bmatrix} \\
&\Leftrightarrow \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_K \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \lambda(g_{1,1}^{-1} + g_{1,2}^{-1} + \cdots + g_{1,K}^{-1}) \\ \lambda(g_{2,1}^{-1} + g_{2,2}^{-1} + \cdots + g_{2,K}^{-1}) \\ \vdots \\ \lambda(g_{K,1}^{-1} + g_{K,2}^{-1} + \cdots + g_{K,K}^{-1}) \end{bmatrix} \\
&\Leftrightarrow W_j = \frac{1}{2}\lambda(g_{j,1}^{-1} + g_{j,2}^{-1} + \cdots + g_{j,K}^{-1}) = \frac{1}{2}\lambda \sum_{k=1}^{K} [G^{-1}]_{jk} \\
&\Rightarrow \mathbb{1}^T W_j = \frac{1}{2} \lambda \mathbb{1}^T G^{-1} \mathbb{1} \\
&\Rightarrow \lambda = \frac{2}{\mathbb{1}^T G^{-1} \mathbb{1}} \\
&\Rightarrow W_j = \frac{1}{2} \left(\frac{2}{\mathbb{1}^T G^{-1} \mathbb{1}} \right) G^{-1} \mathbb{1} = \frac{\sum_k [G^{-1}]_{jk}}{\sum_{lm} [G^{-1}]_{lm}}
\end{aligned}
$$

$\blacksquare$