Michael Hodges

All files can be found here
https://github.com/Michael-Hodges/EECE5644_Machine_Learning.git
or in the appendix

Problem 1

In many pattern classification problesm one has the option either to assign the pattern to one of c classes, or to reject it as being unrecognizable. If the cost for rejects is not too high, rejection may be a desirable action. Let

$$\lambda(\alpha_i|\omega_j) = \begin{cases} 0 & i = j \qquad i,j = 1,...,c \\ \lambda_r & i = c+1 \\ \lambda_s & otherwise \end{cases} \tag{1}$$

where λ_r is the loss incurred for choosing the $(c+1)th$ action, rejection, and λ_s is the loss incurred for making any substitution error. Show that the minimum risk is obtained if we decide ω_i if $P(\omega_i|\boldsymbol{x}) \geq P(\omega_j|\boldsymbol{x})$ for all j and if $P(\omega_i|\boldsymbol{x}) \geq 1 - \lambda_r/\lambda_s$, and reject otherwise.

1. for i = 1,...,c

$$\begin{aligned} R(\alpha_i|x) &= \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)P(\omega_j|x) \\ &= \lambda_s \sum_{j=1, j\neq i}^{C} P(\omega_j|x) \\ &= \lambda_s \left[1 - P(\omega_i|x)\right] \end{aligned}$$

   for i=c+1

$$R(\alpha_{c+1}|x) = \lambda_r$$

   We choose to decide ω_i if $R(\alpha_i|x) \leq R(\alpha_{c+1}|x)$ which is as follows:

$$\begin{aligned} R(\alpha_i|x) \leq R(\alpha_{c+1}|x) &=> \lambda_s\left[1 - P(\omega_i|x)\right] \leq \lambda_r \\ &= 1 - P(\omega_i|x) \leq \frac{\lambda_r}{\lambda_s} \\ &= P(\omega_i|x) \geq 1 - \frac{\lambda_r}{\lambda_s} \end{aligned}$$

2. What happens if $\lambda_r = 0$: In this case we will always reject since $P(\omega_i|x) \ngeq 1$
3. What happens if $\lambda_r > \lambda_s$:In this case we will never reject since $P(\omega_i|x \geq 0$

Problem 2 & 3

Write a function that generates a specified number of independent and identically distributed samples paired with the class labels that generated these samples. Specifically, the data distribution is a mixture of Gaussians with specified prior probabilities for each Gaussian class conditional pdf, as well as respective mean vectors and covariance matrices. Generate and visualize data in the form of scatter plots, with a color/marker based identification of the class label for each sample for each of the following cases (using Matlab syntax for 2 x 2 matrices):

1. Number of samples = 400; class means $[0,0]^T$ and $[3,3]^T$; class covariance matrices both set to I; equal class priors.

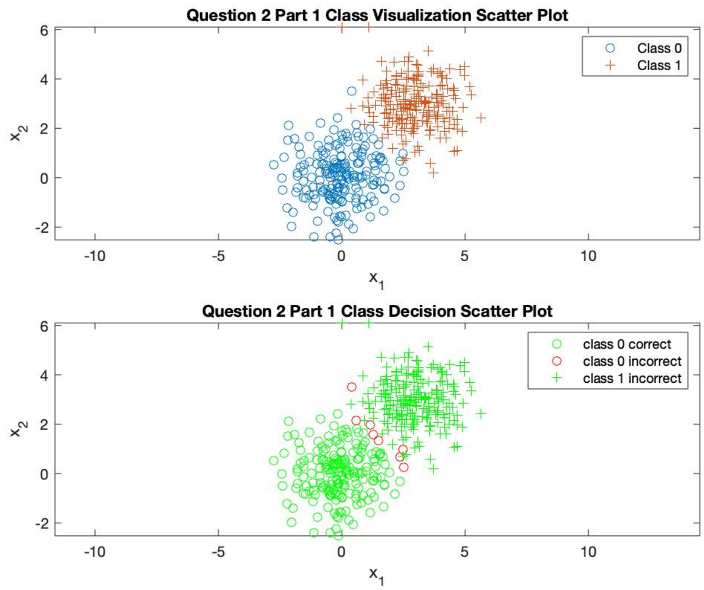

Figure 1: Total error: 8/400. P(error) = 2.00%

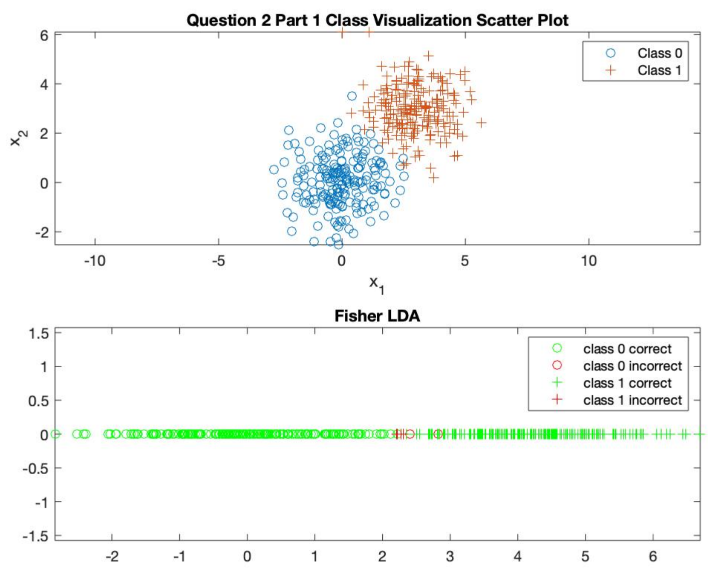

Figure 2: Total error: 7/400. P(error) = 1.75%

2. All parameters same as (1), but both covariance matrices changed to [3, 1; 1, 0.8].

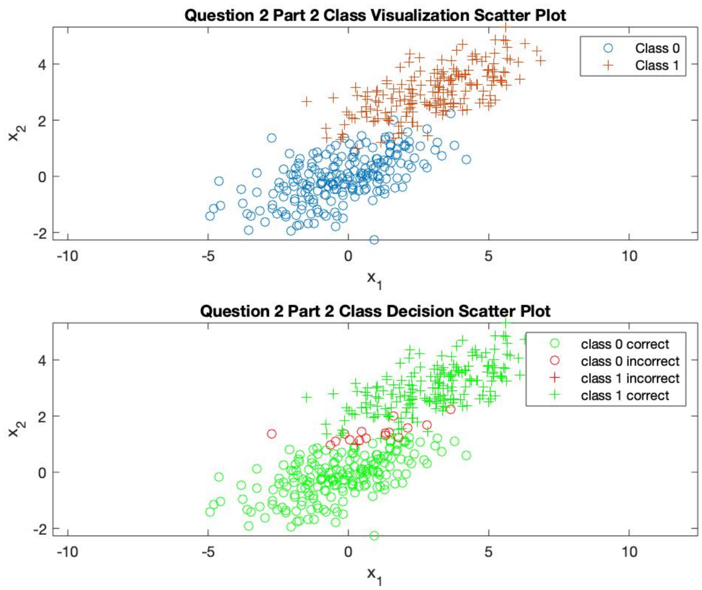

Figure 3: Total error: 17/400. P(error) = 4.25%

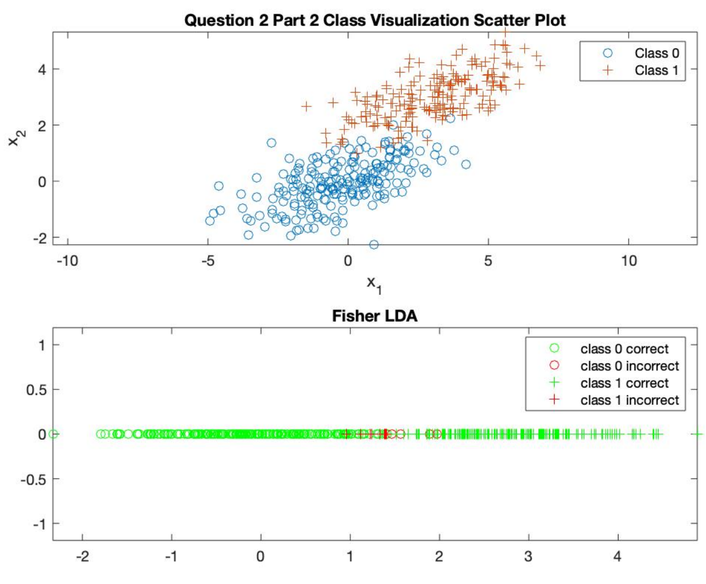

Figure 4: Total error: 13/400. P(error) = 3.25%

3. Number of samples = 400; class means $[0, 0]^T$ and $[2, 2]^T$; class covariance matrices $[2, 0.5; 0.5, 1]$ and $[2, -1.9; -1.9, 5]$; equal class priors.

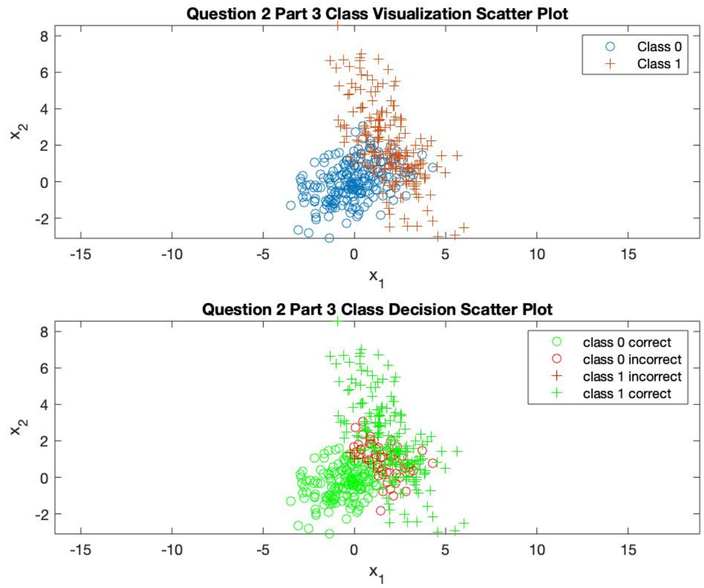

Figure 5: Total error: 65/400. P(error) = 16.25%

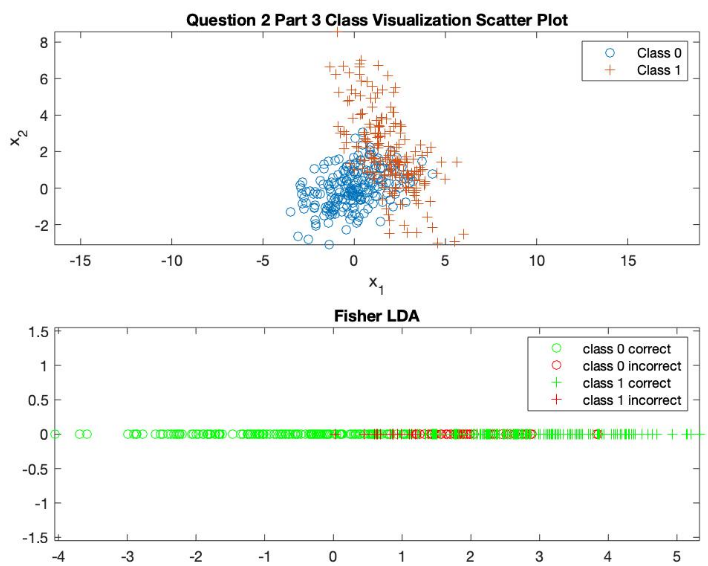

Figure 6: Total error: 66/400. P(error) = 16.50%

4. Same (1), but prior for class priors are 0.05 and 0.95.

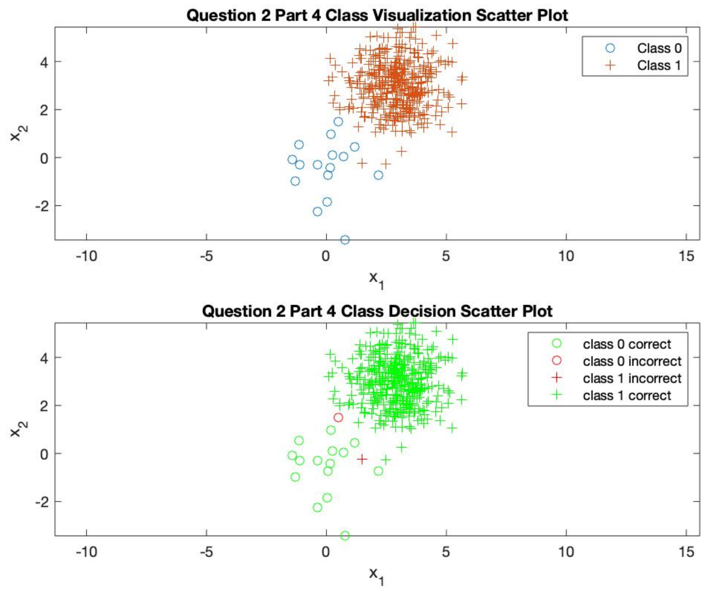

Figure 7: Total error: 2/400. P(error) = 0.50%

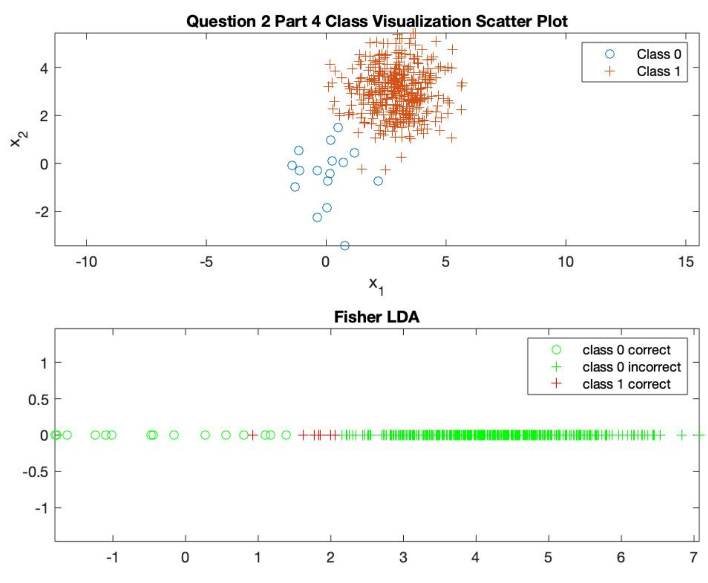

Figure 8: Total error: 8/400. P(error) = 2.00%

5. Same (2), but prior for class priors are 0.05 and 0.95.

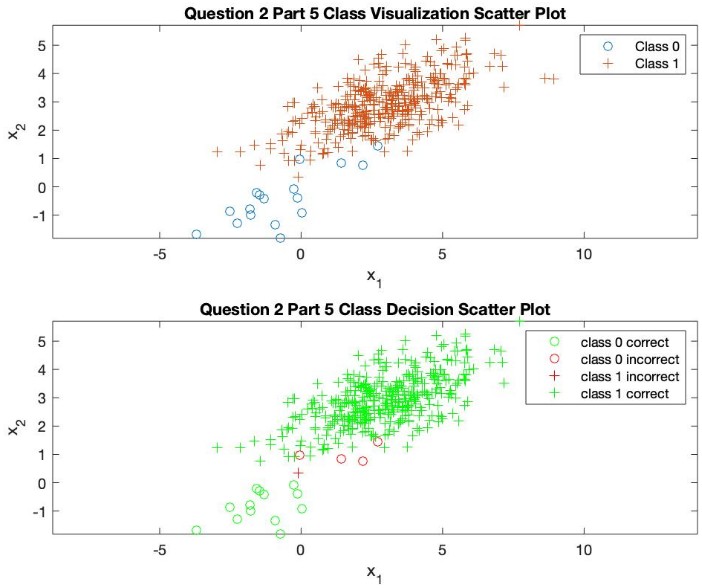

Figure 9: Total error: 6/400. P(error) = 1.50%

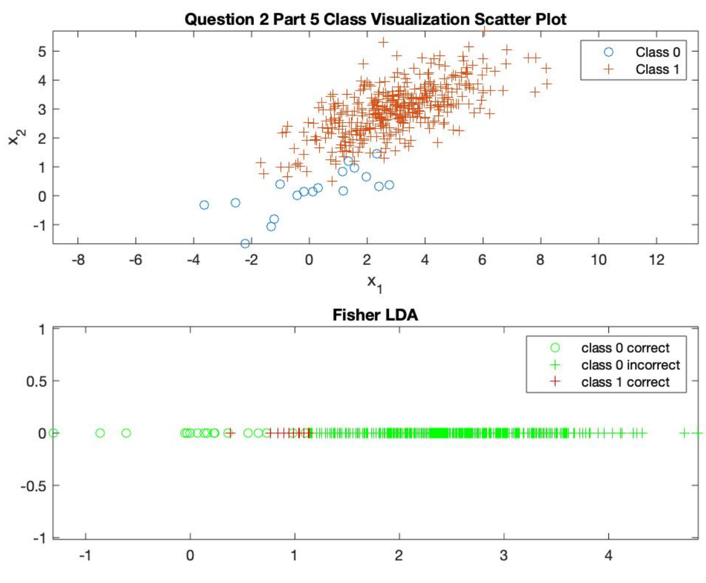

Figure 10: Total error: 15/400. P(error) = 3.75%

6. Same (3), but prior for class priors are 0.05 and 0.95.

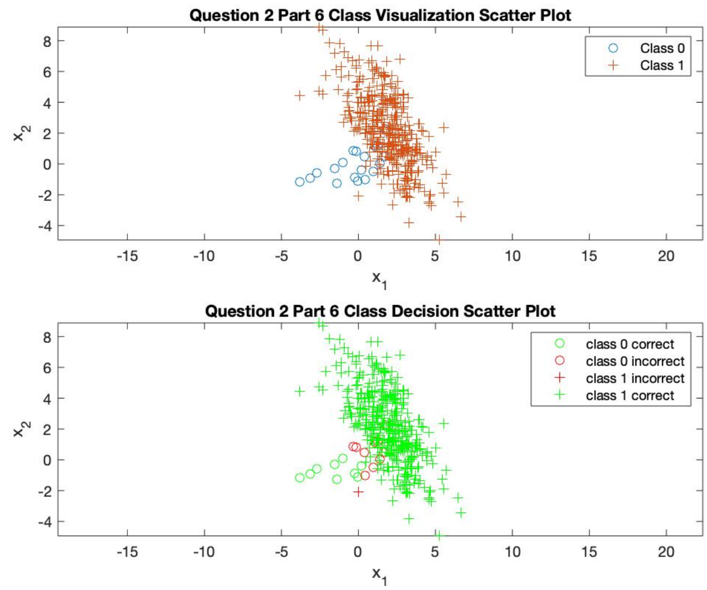

Figure 11: Total error: 11/400. P(error) = 2.75%

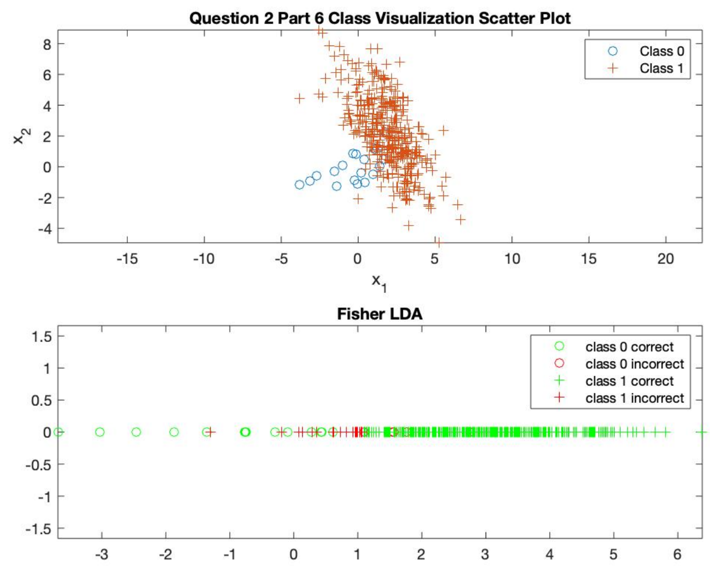

Figure 12: Total error: 25/400. P(error) = 6.25%

Make sure your plots include axis labels, titles, and data legends. Describe how your sampling procedure works, using zero-mean identity-covariance Gaussian sample generators.

The sampling procedure works as follows: We take random samples from a continuous uniform distribution of either 0 and 1. We set it to class 0 or by comparing the value generated to that of the prior. If it is greater than or equal to the prior for class 0 we set the value to one and less we set to 0. These 0's and 1's represent the multivariate gaussian we will generate the sample

Additionally, for each of these datasets, use the maximum-a-priori (MAP) classification rule (using full knowledge of the respective data pdfs) and produce inferred class labels for each data samples. In accompanying visualizations, demonstrate scatter plots of the data for each case along with their inferred (decision) labels. For each case, count the number of errors and estimate the probability of error based on these counts.

Problem 3

For the datasets you generated in Question 2, implement and apply the Fisher Linear Discrimi- nant Analysis classifier with the decision threshold for the linear discriminant score set to minimize the smallest probability error you can achieve on the specific data sets generated for each case. Vi- sualize the one-dimensional Fisher LDA discriminant scores and decision labels for each sample in separate plots for each case. Note: We will soon discuss the principle of cross-validation that dictates parameter selection and performance assessment must use independent datasets.

Please see plots in problem two plotted directly below the problem 2 plots per part.

Appendix

1. Problem 2 & 3

```
% MAP with 2 classes
clear all, close all,

% ----------------------------------------------------------------------------------------
%Part 1

n = 2; % number of feature dimensions
N = 400; % number of iid samples
mu(:,1) = [0;0]; mu(:,2) = [3;3];
Sigma(:,:,1) = [1 0;0 1]; Sigma(:,:,2) = [1 0;0 1];
p = [0.5,0.5]; % class priors for labels 0 and 1 respectively
label = rand(1,N) >= p(1);
Nc = [length(find(label==0)),length(find(label==1))]; % number of samples from each class
x = zeros(n,N); % save up space
% Draw samples from each class pdf
for l = 0:1
    x(:,label==l) = mvnrnd(mu(:,l+1),Sigma(:,:,l+1),Nc(l+1))';
end

pxw1 = mvnpdf(x',mu(:,1)',Sigma(:,:,1)); pxw2 = mvnpdf(x',mu(:,2)',Sigma(:,:,2));
pw1 = p(1); pw2 = (2);
px = pw1*pxw1 + pw2*pxw2;
pw1x = pw1*pxw1./px;
pw2x = pw2*pxw2./px;

decision = pw2x' > pw1x'; %choose whichever class is more likely based on the posterior after using bayes rule

ind00 = find(decision==0 & label==0); %p00 = length(ind00)/Nc(1); % probability of true negative
ind10 = find(decision==1 & label==0); %p10 = length(ind10)/Nc(1); % probability of false positive
ind01 = find(decision==0 & label==1); %p01 = length(ind01)/Nc(2); % probability of false negative
ind11 = find(decision==1 & label==1); %p11 = length(ind11)/Nc(2); % probability of true positive
disp('error MAP = ');
disp(length(ind10)+length(ind01));

figure(1),
```

```matlab
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 1 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),

subplot(2,1,2), % class 0 circle, class 1 +, correct green, incorrect red
plot(x(1,ind00),x(2,ind00),'og'); hold on,
plot(x(1,ind10),x(2,ind10),'or'); hold on,
plot(x(1,ind01),x(2,ind01),'+r'); hold on,
plot(x(1,ind11),x(2,ind11),'+g'); hold on,
legend('class 0 correct', 'class 0 incorrect', 'class 1 incorrect', 'class 1 correct')
title('Question 2 Part 1 Class Decision Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
axis equal,

mu0hat = mean(x(:,label==0),2); S0hat = cov(x(:,label==0)'); % estimated mean and covariance from the sampled data
mu1hat = mean(x(:,label==1),2); S1hat = cov(x(:,label==1)');

S_b = (mu0hat-mu1hat)*(mu0hat-mu1hat)'; %Scatter 1 wrt scatter 2
S_w = S0hat+S1hat;

[V,D] = eig(inv(S_w)*S_b);
[~,ind] = sort(diag(D),'descend');
w = V(:,ind(1)); % Fisher LDA projection vector

y = w'*x;
w = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*w; % ensures class1 falls on the + side of the axis
y = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*y; % flip y accordingly
mu0LDA = mean(y(find(label==0))); sig0LDA = (cov(y(find(label==0))));
mu1LDA = mean(y(find(label==1))); sig1LDA = (cov(y(find(label==1))));
decisionLDA = normpdf(y,mu1LDA,sig1LDA)>normpdf(y,mu0LDA,sig0LDA);

ind00 = find(label==0 & decisionLDA==0);
ind01 = find(label==0 & decisionLDA==1);
ind10 = find(label==1 & decisionLDA==0);
```

```
ind11 = find(label==1 & decisionLDA==1);

figure(7);
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 1 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
subplot(2,1,2), cla,
plot(y(ind00),zeros(1,length(y(ind00))),'og'), hold on,
plot(y(ind01),zeros(1,length(y(ind01))),'or');
plot(y(ind11),zeros(1,length(y(ind11))),'+g');
plot(y(ind10),zeros(1,length(y(ind10))),'+r');
legend('class 0 correct', 'class 0 incorrect', 'class 1 correct', 'class 1 incorrect')
title('Fisher LDA');
axis equal,
disp('error LDA =');
disp(length(ind01)+length(ind10))

%

%Part 2
n = 2; % number of feature dimensions
N = 400; % number of iid samples
mu(:,1) = [0;0]; mu(:,2) = [3;3];
Sigma(:,:,1) = [3 1;1 0.8]; Sigma(:,:,2) = [3 1;1 0.8];
p = [0.5,0.5]; % class priors for labels 0 and 1 respectively
label = rand(1,N) >= p(1);
Nc = [length(find(label==0)),length(find(label==1))]; % number of samples from each class
x = zeros(n,N); % save up space
% Draw samples from each class pdf
for l = 0:1
    x(:,label==l) = mvnrnd(mu(:,l+1),Sigma(:,:,l+1),Nc(l+1))';
end

pxw1 = mvnpdf(x',mu(:,1)',Sigma(:,:,1)); pxw2 = mvnpdf(x',mu(:,2)',Sigma(:,:,2));
pw1 = p(1); pw2 = (2);
px = pw1*pxw1 + pw2*pxw2;
pw1x = pw1*pxw1./px;
pw2x = pw2*pxw2./px;

```

```
decision = pw2x' > pw1x'; %choose whichever class is more likely based on the posterior after using bayes rule

ind00 = find(decision==0 & label==0);
ind10 = find(decision==1 & label==0);
ind01 = find(decision==0 & label==1);
ind11 = find(decision==1 & label==1);

disp('error MAP = ');
disp(length(ind10)+length(ind01))

figure(2),
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 2 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),

subplot(2,1,2), % class 0 circle, class 1 +, correct green, incorrect red
plot(x(1,ind00),x(2,ind00),'og'); hold on,
plot(x(1,ind10),x(2,ind10),'or'); hold on,
plot(x(1,ind01),x(2,ind01),'+r'); hold on,
plot(x(1,ind11),x(2,ind11),'+g'); hold on,
legend('class 0 correct', 'class 0 incorrect', 'class 1 incorrect', 'class 1 correct')
title('Question 2 Part 2 Class Decision Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
axis equal,


mu0hat = mean(x(:,label==0),2); S0hat = cov(x(:,label==0)'); % estimated mean and covariance from the sampled data
mu1hat = mean(x(:,label==1),2); S1hat = cov(x(:,label==1)');

S_b = (mu0hat-mu1hat)*(mu0hat-mu1hat)'; %Scatter 1 wrt scatter 2
S_w = S0hat+S1hat;

[V,D] = eig(inv(S_w)*S_b);
[~,ind] = sort(diag(D),'descend');
w = V(:,ind(1)); % Fisher LDA projection vector

y = w'*x;
w = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*w; %
```

```
    ensures class1 falls on the + side of the axis
y = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*y; %
    flip y accordingly
mu0LDA = mean(y(find(label==0))); sig0LDA = (cov(y(find(label
    ==0))));
mu1LDA = mean(y(find(label==1))); sig1LDA = (cov(y(find(label
    ==1))));
decisionLDA = normpdf(y,mu1LDA,sig1LDA)>normpdf(y,mu0LDA,
    sig0LDA);

ind00 = find(label==0 & decisionLDA==0);
ind01 = find(label==0 & decisionLDA==1);
ind10 = find(label==1 & decisionLDA==0);
ind11 = find(label==1 & decisionLDA==1);

figure(8);
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 2 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
subplot(2,1,2), cla,
plot(y(ind00),zeros(1,length(y(ind00))),'og'), hold on,
plot(y(ind01),zeros(1,length(y(ind01))),'or');
plot(y(ind11),zeros(1,length(y(ind11))),'+g');
plot(y(ind10),zeros(1,length(y(ind10))),'+r');
legend('class 0 correct', 'class 0 incorrect', 'class 1 correct
    ', 'class 1 incorrect')
title('Fisher LDA');
axis equal,
disp('error LDA =');
disp(length(ind01)+length(ind10))


%
    ------------------------------------------------------------------------

%Part 3
n = 2; % number of feature dimensions
N = 400; % number of iid samples
mu(:,1) = [0;0]; mu(:,2) = [2;2];
Sigma(:,:,1) = [2,0.5;0.5,1]; Sigma(:,:,2) = [2,-1.9;-1.9,5];
p = [0.5,0.5]; % class priors for labels 0 and 1 respectively
label = rand(1,N) >= p(1);
Nc = [length(find(label==0)),length(find(label==1))]; % number
```

```
    of samples from each class
x = zeros(n,N); % save up space
% Draw samples from each class pdf
for l = 0:1
    x(:,label==l) = mvnrnd(mu(:,l+1),Sigma(:,:,l+1),Nc(l+1))';
end

pxw1 = mvnpdf(x',mu(:,1)',Sigma(:,:,1)); pxw2 = mvnpdf(x',mu
    (:,2)',Sigma(:,:,2));
pw1 = p(1); pw2 = (2);
px = pw1*pxw1 + pw2*pxw2;
pw1x = pw1*pxw1./px;
pw2x = pw2*pxw2./px;

decision = pw2x' > pw1x'; %choose whichever class is more
    likely based on the posterior after using bayes rule

ind00 = find(decision==0 & label==0); %p00 = length(ind00)/Nc
    (1); % probability of true negative
ind10 = find(decision==1 & label==0); %p10 = length(ind10)/Nc
    (1); % probability of false positive
ind01 = find(decision==0 & label==1); %p01 = length(ind01)/Nc
    (2); % probability of false negative
ind11 = find(decision==1 & label==1); %p11 = length(ind11)/Nc
    (2); % probability of true positive

disp('error MAP = ');
disp(length(ind10)+length(ind01))

figure(3),
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 3 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),

subplot(2,1,2) % class 0 circle, class 1 +, correct green,
    incorrect red
plot(x(1,ind00),x(2,ind00),'og'); hold on,
plot(x(1,ind10),x(2,ind10),'or'); hold on,
plot(x(1,ind01),x(2,ind01),'+r'); hold on,
plot(x(1,ind11),x(2,ind11),'+g'); hold on,
legend('class 0 correct', 'class 0 incorrect', 'class 1
    incorrect', 'class 1 correct')
title('Question 2 Part 3 Class Decision Scatter Plot'),
```

```
xlabel('x_1'), ylabel('x_2'),
axis equal,

mu0hat = mean(x(:,label==0),2); S0hat = cov(x(:,label==0)'); % estimated mean and covariance from the sampled data
mu1hat = mean(x(:,label==1),2); S1hat = cov(x(:,label==1)');

S_b = (mu0hat-mu1hat)*(mu0hat-mu1hat)'; %Scatter 1 wrt scatter 2
S_w = S0hat+S1hat;

[V,D] = eig(inv(S_w)*S_b);
[~,ind] = sort(diag(D),'descend');
w = V(:,ind(1)); % Fisher LDA projection vector

y = w'*x;
w = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*w; % ensures class1 falls on the + side of the axis
y = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*y; % flip y accordingly
mu0LDA = mean(y(find(label==0))); sig0LDA = (cov(y(find(label==0))));
mu1LDA = mean(y(find(label==1))); sig1LDA = (cov(y(find(label==1))));
decisionLDA = normpdf(y,mu1LDA,sig1LDA)>normpdf(y,mu0LDA,sig0LDA);

ind00 = find(label==0 & decisionLDA==0);
ind01 = find(label==0 & decisionLDA==1);
ind10 = find(label==1 & decisionLDA==0);
ind11 = find(label==1 & decisionLDA==1);

figure(9);
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 3 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
subplot(2,1,2), cla,
plot(y(ind00),zeros(1,length(y(ind00))),'og'), hold on,
plot(y(ind01),zeros(1,length(y(ind01))),'or');
plot(y(ind11),zeros(1,length(y(ind11))),'+g');
plot(y(ind10),zeros(1,length(y(ind10))),'+r');
legend('class 0 correct', 'class 0 incorrect', 'class 1 correct', 'class 1 incorrect')
```

```
title('Fisher LDA');
axis equal,
disp('error LDA =');
disp(length(ind01)+length(ind10))
%
%Part 4

n = 2; % number of feature dimensions
N = 400; % number of iid samples
mu(:,1) = [0;0]; mu(:,2) = [3;3];
Sigma(:,:,1) = [1 0;0 1]; Sigma(:,:,2) = [1 0;0 1];
p = [0.05,0.95]; % class priors for labels 0 and 1 respectively
label = rand(1,N) >= p(1);
Nc = [length(find(label==0)),length(find(label==1))]; % number of samples from each class
x = zeros(n,N); % save up space
% Draw samples from each class pdf
for l = 0:1
    x(:,label==l) = mvnrnd(mu(:,l+1),Sigma(:,:,l+1),Nc(l+1))';
end

pxw1 = mvnpdf(x',mu(:,1)',Sigma(:,:,1)); pxw2 = mvnpdf(x',mu(:,2)',Sigma(:,:,2));
pw1 = p(1); pw2 = (2);
px = pw1*pxw1 + pw2*pxw2;
pw1x = pw1*pxw1./px;
pw2x = pw2*pxw2./px;

decision = pw2x' > pw1x'; %choose whichever class is more likely based on the posterior after using bayes rule

ind00 = find(decision==0 & label==0); %p00 = length(ind00)/Nc(1); % probability of true negative
ind10 = find(decision==1 & label==0); %p10 = length(ind10)/Nc(1); % probability of false positive
ind01 = find(decision==0 & label==1); %p01 = length(ind01)/Nc(2); % probability of false negative
ind11 = find(decision==1 & label==1); %p11 = length(ind11)/Nc(2); % probability of true positive

disp('error MAP = ');
disp(length(ind10)+length(ind01))

figure(4)
```

```
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 4 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),

subplot(2,1,2), % class 0 circle, class 1 +, correct green, incorrect red
plot(x(1,ind00),x(2,ind00),'og'); hold on,
plot(x(1,ind10),x(2,ind10),'or'); hold on,
plot(x(1,ind01),x(2,ind01),'+r'); hold on,
plot(x(1,ind11),x(2,ind11),'+g'); hold on,
legend('class 0 correct', 'class 0 incorrect', 'class 1 incorrect', 'class 1 correct')
title('Question 2 Part 4 Class Decision Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
axis equal,

mu0hat = mean(x(:,label==0),2); S0hat = cov(x(:,label==0)'); % estimated mean and covariance from the sampled data
mu1hat = mean(x(:,label==1),2); S1hat = cov(x(:,label==1)');

S_b = (mu0hat-mu1hat)*(mu0hat-mu1hat)'; %Scatter 1 wrt scatter 2
S_w = S0hat+S1hat;

[V,D] = eig(inv(S_w)*S_b);
[~,ind] = sort(diag(D),'descend');
w = V(:,ind(1)); % Fisher LDA projection vector

y = w'*x;
w = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*w; % ensures class1 falls on the + side of the axis
y = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*y; % flip y accordingly
mu0LDA = mean(y(find(label==0))); sig0LDA = (cov(y(find(label==0))));
mu1LDA = mean(y(find(label==1))); sig1LDA = (cov(y(find(label==1))));
decisionLDA = normpdf(y,mu1LDA,sig1LDA)>normpdf(y,mu0LDA,sig0LDA);

ind00 = find(label==0 & decisionLDA==0);
ind01 = find(label==0 & decisionLDA==1);
ind10 = find(label==1 & decisionLDA==0);
```

```
ind11 = find(label==1 & decisionLDA==1);

figure(10);
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 4 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
subplot(2,1,2), cla,
plot(y(ind00),zeros(1,length(y(ind00))),'og'), hold on,
plot(y(ind01),zeros(1,length(y(ind01))),'or');
plot(y(ind11),zeros(1,length(y(ind11))),'+g');
plot(y(ind10),zeros(1,length(y(ind10))),'+r');
legend('class 0 correct', 'class 0 incorrect', 'class 1 correct
    ', 'class 1 incorrect')
title('Fisher LDA');
axis equal,
disp('error LDA =');
disp(length(ind01)+length(ind10))
% ----------------------------------------------------------------------
%Part 5
n = 2; % number of feature dimensions
N = 400; % number of iid samples
mu(:,1) = [0;0]; mu(:,2) = [3;3];
Sigma(:,:,1) = [3 1;1 0.8]; Sigma(:,:,2) = [3 1;1 0.8];
p = [0.05,0.95]; % class priors for labels 0 and 1 respectively
label = rand(1,N) >= p(1);
Nc = [length(find(label==0)),length(find(label==1))]; % number
    of samples from each class
x = zeros(n,N); % save up space
% Draw samples from each class pdf
for l = 0:1
    x(:,label==l) = mvnrnd(mu(:,l+1),Sigma(:,:,l+1),Nc(l+1))';
end

pxw1 = mvnpdf(x',mu(:,1)',Sigma(:,:,1)); pxw2 = mvnpdf(x',mu
    (:,2)',Sigma(:,:,2));
pw1 = p(1); pw2 = (2);
px = pw1*pxw1 + pw2*pxw2;
pw1x = pw1*pxw1./px;
pw2x = pw2*pxw2./px;

decision = pw2x' > pw1x'; %choose whichever class is more
```

```
likely based on the posterior after using bayes rule

ind00 = find(decision==0 & label==0); %p00 = length(ind00)/Nc(1); % probability of true negative
ind10 = find(decision==1 & label==0); %p10 = length(ind10)/Nc(1); % probability of false positive
ind01 = find(decision==0 & label==1); %p01 = length(ind01)/Nc(2); % probability of false negative
ind11 = find(decision==1 & label==1); %p11 = length(ind11)/Nc(2); % probability of true positive

disp('error MAP = ');
disp(length(ind10)+length(ind01))

figure(5)
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 5 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),

subplot(2,1,2), % class 0 circle, class 1 +, correct green, incorrect red
plot(x(1,ind00),x(2,ind00),'og'); hold on,
plot(x(1,ind10),x(2,ind10),'or'); hold on,
plot(x(1,ind01),x(2,ind01),'+r'); hold on,
plot(x(1,ind11),x(2,ind11),'+g'); hold on,
legend('class 0 correct', 'class 0 incorrect', 'class 1 incorrect', 'class 1 correct')
title('Question 2 Part 5 Class Decision Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
axis equal,

mu0hat = mean(x(:,label==0),2); S0hat = cov(x(:,label==0)'); % estimated mean and covariance from the sampled data
mu1hat = mean(x(:,label==1),2); S1hat = cov(x(:,label==1)');

S_b = (mu0hat-mu1hat)*(mu0hat-mu1hat)'; %Scatter 1 wrt scatter 2
S_w = S0hat+S1hat;

[V,D] = eig(inv(S_w)*S_b);
[~,ind] = sort(diag(D),'descend');
w = V(:,ind(1)); % Fisher LDA projection vector

```

```
y = w'*x;
w = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*w; % ensures class1 falls on the + side of the axis
y = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*y; % flip y accordingly

mu0LDA = mean(y(find(label==0))); sig0LDA = (cov(y(find(label==0))));
mu1LDA = mean(y(find(label==1))); sig1LDA = (cov(y(find(label==1))));
decisionLDA = normpdf(y,mu1LDA,sig1LDA)>normpdf(y,mu0LDA,sig0LDA);

ind00 = find(label==0 & decisionLDA==0);
ind01 = find(label==0 & decisionLDA==1);
ind10 = find(label==1 & decisionLDA==0);
ind11 = find(label==1 & decisionLDA==1);

figure(11);
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 5 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
subplot(2,1,2), cla,
plot(y(ind00),zeros(1,length(y(ind00))),'og'), hold on,
plot(y(ind01),zeros(1,length(y(ind01))),'or');
plot(y(ind11),zeros(1,length(y(ind11))),'+g');
plot(y(ind10),zeros(1,length(y(ind10))),'+r');
legend('class 0 correct', 'class 0 incorrect', 'class 1 correct', 'class 1 incorrect')
title('Fisher LDA');
axis equal,
disp('error LDA =');
disp(length(ind01)+length(ind10))
%

%Part 6
n = 2; % number of feature dimensions
N = 400; % number of iid samples
mu(:,1) = [0;0]; mu(:,2) = [2;2];
Sigma(:,:,1) = [2,0.5;0.5,1]; Sigma(:,:,2) = [2,-1.9;-1.9,5];
p = [0.05,0.95]; % class priors for labels 0 and 1 respectively
label = rand(1,N) >= p(1);
```

```
Nc = [length(find(label==0)),length(find(label==1))]; % number
    of samples from each class
x = zeros(n,N); % save up space
% Draw samples from each class pdf
for l = 0:1
    x(:,label==l) = mvnrnd(mu(:,l+1),Sigma(:,:,l+1),Nc(l+1))';
end

pxw1 = mvnpdf(x',mu(:,1)',Sigma(:,:,1)); pxw2 = mvnpdf(x',mu
    (:,2)',Sigma(:,:,2));
pw1 = p(1); pw2 = (2);
px = pw1*pxw1 + pw2*pxw2;
pw1x = pw1*pxw1./px;
pw2x = pw2*pxw2./px;

decision = pw2x' > pw1x'; %choose whichever class is more
    likely based on the posterior after using bayes rule

ind00 = find(decision==0 & label==0); %p00 = length(ind00)/Nc
    (1); % probability of true negative
ind10 = find(decision==1 & label==0); %p10 = length(ind10)/Nc
    (1); % probability of false positive
ind01 = find(decision==0 & label==1); %p01 = length(ind01)/Nc
    (2); % probability of false negative
ind11 = find(decision==1 & label==1); %p11 = length(ind11)/Nc
    (2); % probability of true positive

disp('error MAP = ');
disp(length(ind10)+length(ind01))

figure(6)
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 6 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),

subplot(2,1,2), % class 0 circle, class 1 +, correct green,
    incorrect red
plot(x(1,ind00),x(2,ind00),'og'); hold on,
plot(x(1,ind10),x(2,ind10),'or'); hold on,
plot(x(1,ind01),x(2,ind01),'+r'); hold on,
plot(x(1,ind11),x(2,ind11),'+g'); hold on,
legend('class 0 correct', 'class 0 incorrect', 'class 1
    incorrect', 'class 1 correct')
```

```
title('Question 2 Part 6 Class Decision Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
axis equal,

mu0hat = mean(x(:,label==0),2); S0hat = cov(x(:,label==0)'); % estimated mean and covariance from the sampled data
mu1hat = mean(x(:,label==1),2); S1hat = cov(x(:,label==1)');

S_b = (mu0hat-mu1hat)*(mu0hat-mu1hat)'; %Scatter 1 wrt scatter 2
S_w = S0hat+S1hat;

[V,D] = eig(inv(S_w)*S_b);
[~,ind] = sort(diag(D),'descend');
w = V(:,ind(1)); % Fisher LDA projection vector

y = w'*x;
w = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*w; % ensures class1 falls on the + side of the axis
y = sign(mean(y(find(label==1)))-mean(y(find(label==0))))*y; % flip y accordingly
mu0LDA = mean(y(find(label==0))); sig0LDA = (cov(y(find(label==0))));
mu1LDA = mean(y(find(label==1))); sig1LDA = (cov(y(find(label==1))));
decisionLDA = normpdf(y,mu1LDA,sig1LDA)>normpdf(y,mu0LDA,sig0LDA);

ind00 = find(label==0 & decisionLDA==0);
ind01 = find(label==0 & decisionLDA==1);
ind10 = find(label==1 & decisionLDA==0);
ind11 = find(label==1 & decisionLDA==1);

figure(12);
subplot(2,1,1),
plot(x(1,label==0),x(2,label==0),'o'), hold on,
plot(x(1,label==1),x(2,label==1),'+'), axis equal,
legend('Class 0','Class 1'),
title('Question 2 Part 6 Class Visualization Scatter Plot'),
xlabel('x_1'), ylabel('x_2'),
subplot(2,1,2), cla,
plot(y(ind00),zeros(1,length(y(ind00))),'og'), hold on,
plot(y(ind01),zeros(1,length(y(ind01))),'or');
plot(y(ind11),zeros(1,length(y(ind11))),'+g');
plot(y(ind10),zeros(1,length(y(ind10))),'+r');
legend('class 0 correct', 'class 0 incorrect', 'class 1 correct
```

```
    ', 'class 1 incorrect')
title('Fisher LDA');
axis equal,
disp('error LDA =');
disp(length(ind01)+length(ind10))
```