Pete Mollica

Machine Learning Homework #1

*Please refer to the Appendix for the source code.

Part 1:

The first part of the homework asks us to generate two Gaussian random variables from 10,000 data points. Where one contains a mean of 1 and variance of 1, which can be referred to as GRV1, and the other containing a mean of -1 and a variance of 1, which can be referred to as GRV2. After that we are asked to compute the error rate using an optimal maximum likelihood decoder, which will set the threshold to zero. The next part asked to compute the error rate when the threshold is changed to 0.5 and -0.5. In figure 1 and figure 2 we will be able to see the Matlab generated PDF's for GRV1 and GRV2.

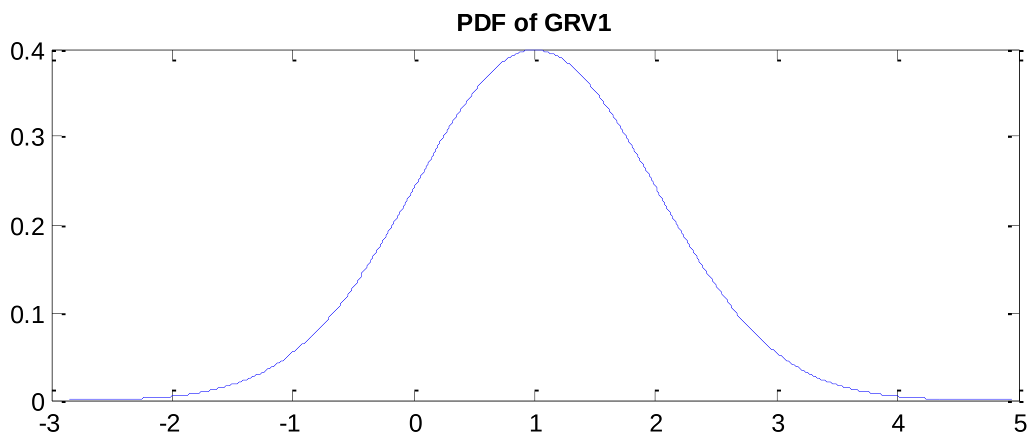

Figure 1: PDF of GRV1

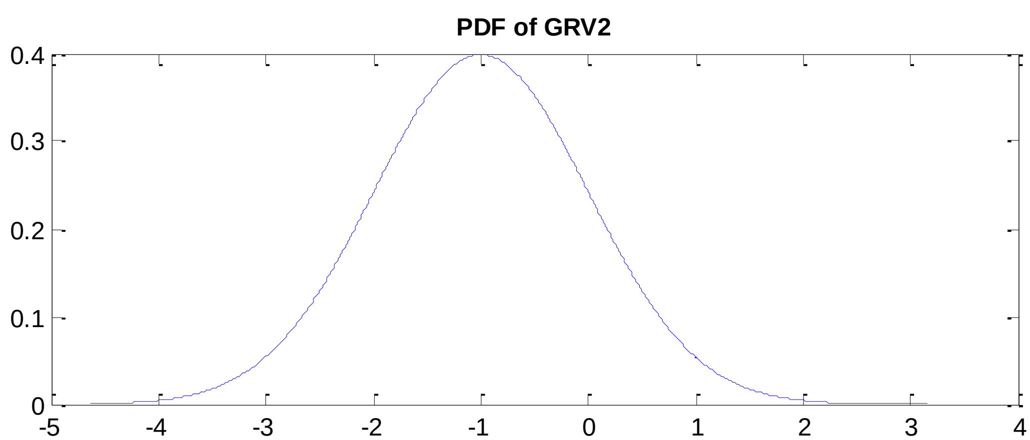

Figure 2: PDF of GRV2

Figure 3 will show the PDF's of both the Gaussian random variables overlapping, which will provide a better visualization of the optimal threshold as well as the areas of error.

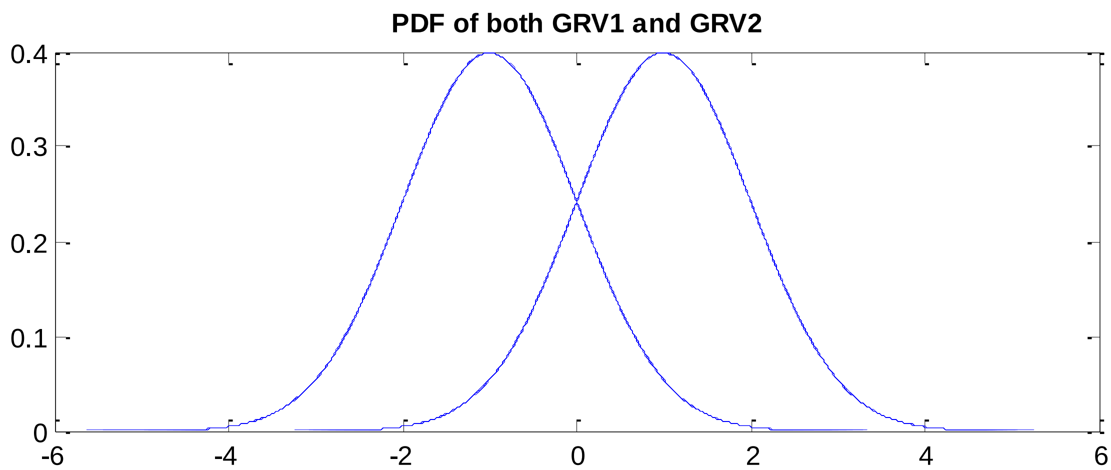

Figure 3: Overlapping PDF's of GRV1 and GRV2

From looking at figure 3 we can see that the both the PDF's intersect at 0, which is the reason why that is our optimal likelihood threshold. This threshold should also give us an equal error for an incorrect decision of either GRV1 or GRV2, which can be observed below in figure 4. Figure 4 will also contain all of the probability errors for each of the thresholds. Each of the probabilities where calculated by adding all of the data points up that crossed the threshold value and then divided that number by the total number of data points.

Threshold	P(Incorrectly Deciding GRV1)	P(Incorrectly Deciding GRV2)
-0.5	0.3112	0.0674
0	0.1628	0.1576
0.5	0.0708	0.3089

Figure 4: Observed Probability Errors for Varying Thresholds

<u>Part 2:</u>

Part 2 of the homework required us to change the variance of GRV1 and plot the error rate across the. Figure 5 will show the PDFs of GRV1 as the variance varies from .1 to 2 in increments of .1. The error rate graph is representation of the error for when the threshold is set to zero and an incorrect decision of GRV2 is picked given the correct decision is GRV1.

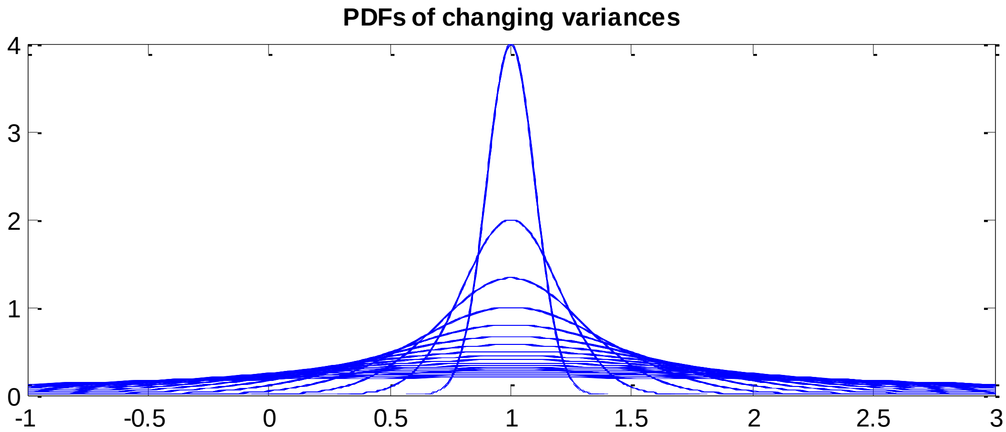

Figure 5: Changing Variances for GRV1

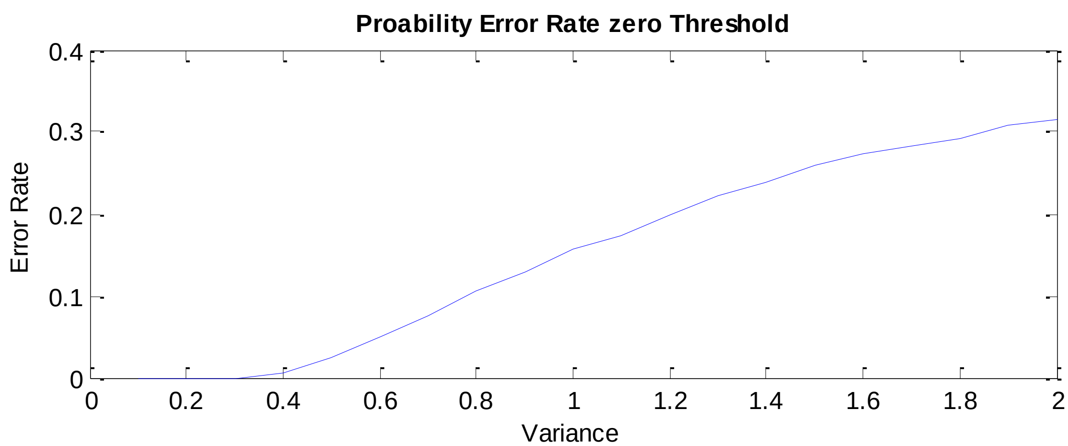

Figure 6: The error rate plot as the variance changed.

Part 3:

The third part of this homework assignment required us to generate a 2D GRV with a mean of [-1,1] and an identity covariance matrix, which can be referred to as GRV2D1. Generate a second GRV with a mean of [1,1] and an identity covariance matrix, which can be referred to as GRV2D2. Compute the error rate for the optimal maximum likelihood decoder. Figure 7 below shows the scatter plot of the of both 2-D Gaussian random variables. From the scatter plot we can infer that the maxim likelihood threshold will be at zero along the x-axis. The y-axis does not have an effect on the decision because we are using an identity covariance matrix.

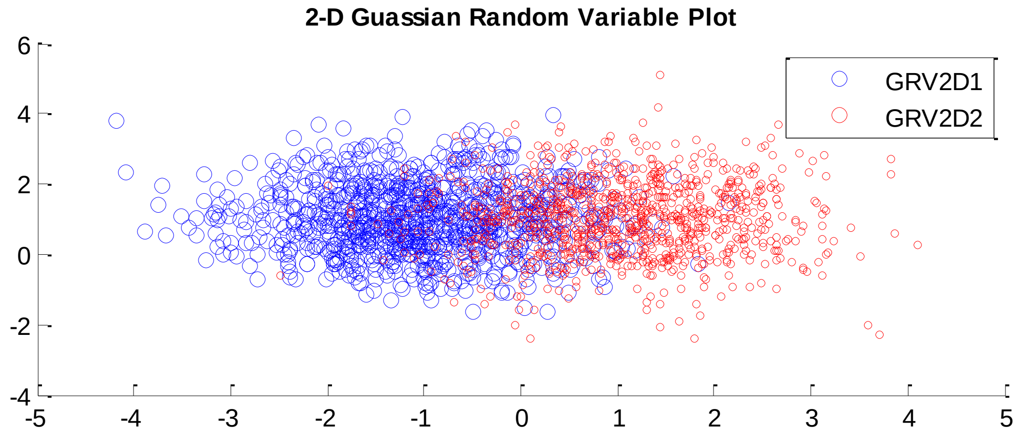

Figure 7: Scatter plot of two different 2-D Gaussian Random Data Distributions

The error rates were also conformation that setting zero as the threshold was the optimal likelihood threshold. This was because the error of incorrectly choosing GRV2D1 was approximately the same as incorrectly choosing GRV2D2. The probability error that was simulated was approximately 0.085.

Part 4:

The fourth part of this homework assignment required us to generate a plot of the support region for the first GRV in part 3 as you vary the covariance. In order to compare the results of the support regions I decided to use the same covariance matrices that were used in the lecture 2's power point slides. Which are the following :

Matrix 1=[1, 0 ; 0, 1]   Matrix 2=[5, 0 ; 0 ,2]   Matrix 3=[1, 0.5 ; 0.5, 1]   Matrix 4=[5, 0.5 ; 0.5, 2]

The PDF's and the support regions for each matrix can be found in the figures below.

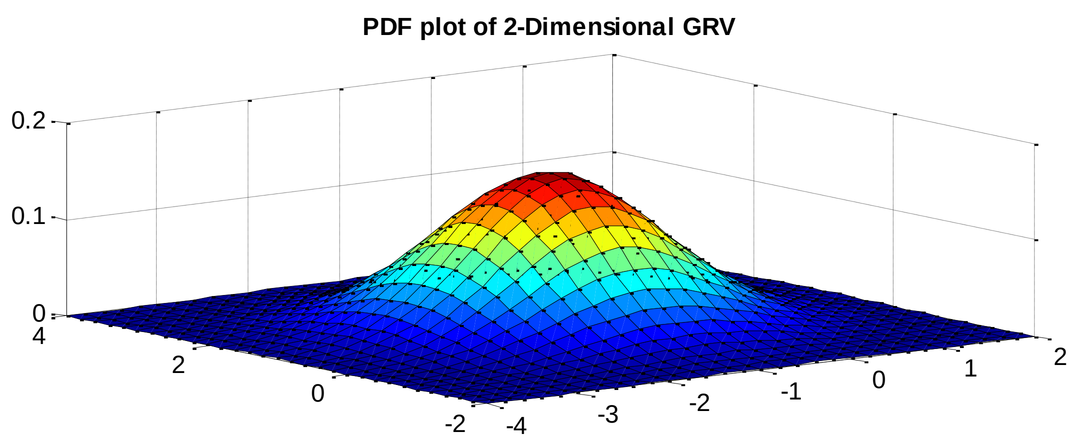

Figure 8: PDF plot GRV2D1 and Covariance Matrix 1

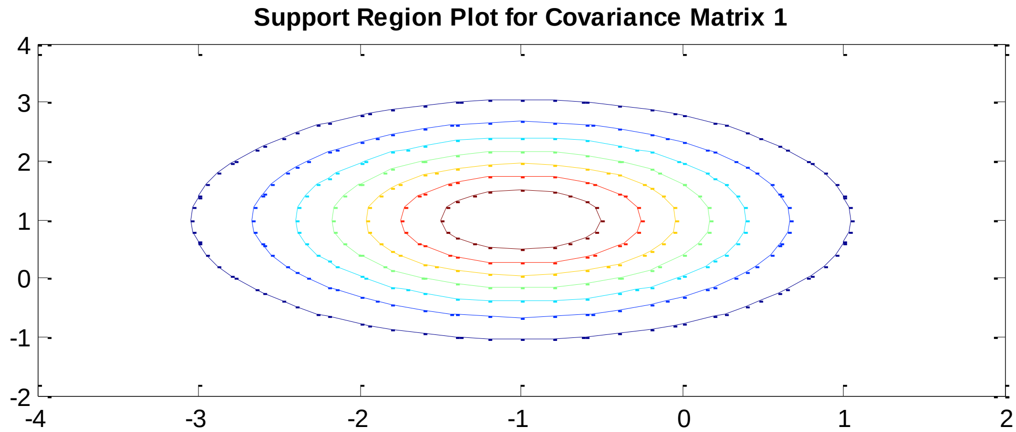

Figure 9: Support Region Plot for GRV2D1 and Covariance Matrix 1

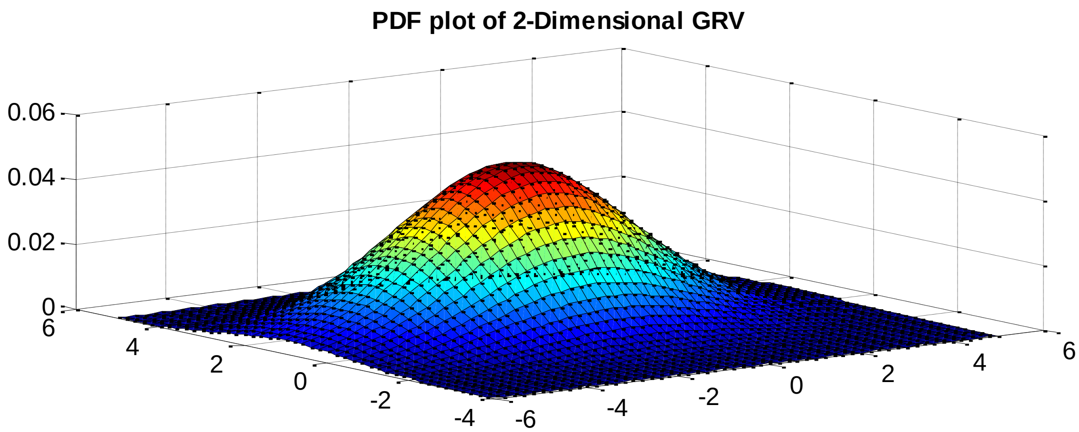

Figure 10: PDF plot GRV2D1 and Covariance Matrix 2

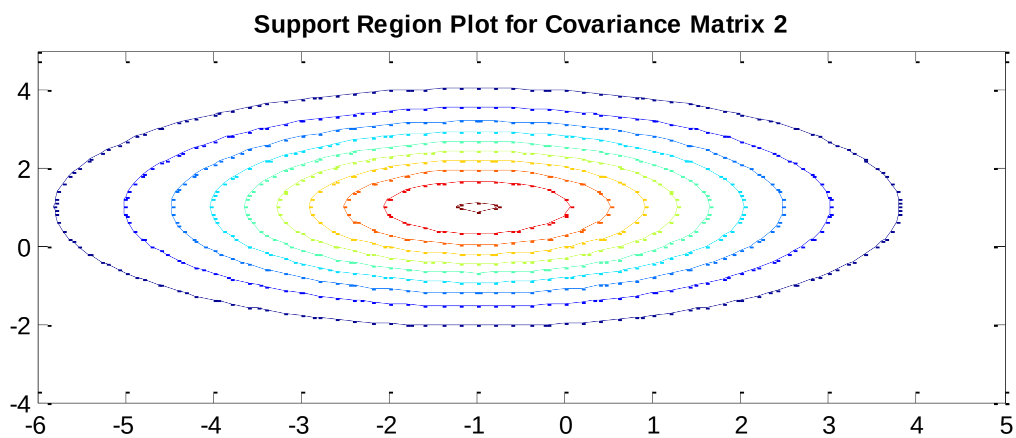

Figure 11: Support Region Plot for GRV2D1 and Covariance Matrix 2

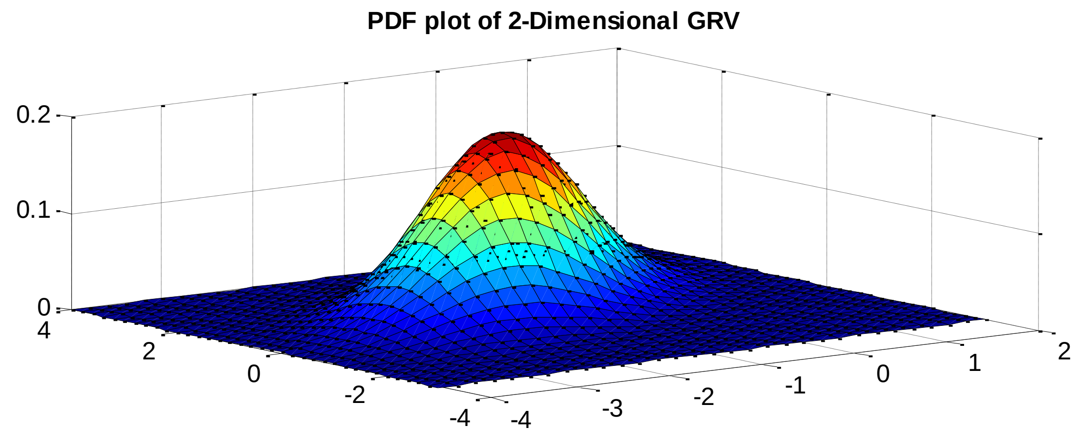

Figure 10: PDF plot GRV2D1 and Covariance Matrix 3

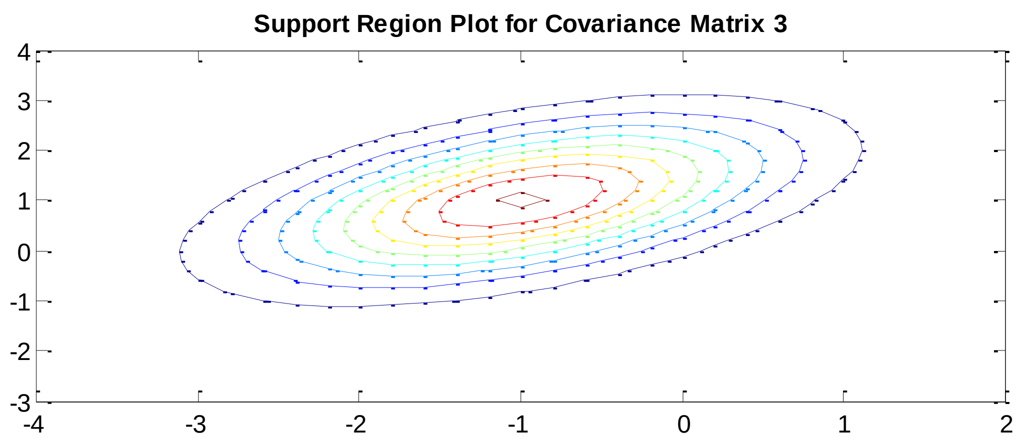

Figure 11: Support Region Plot for GRV2D1 and Covariance Matrix 3

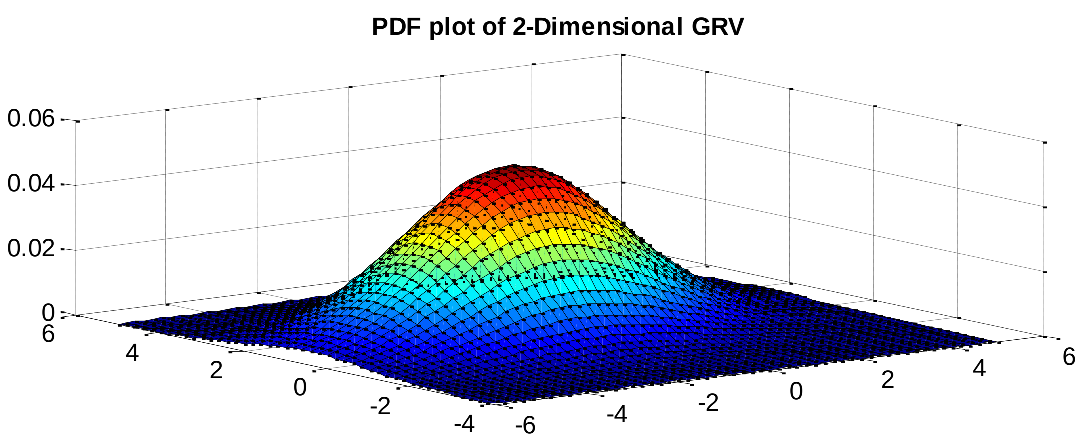

Figure 12: PDF plot GRV2D1 and Covariance Matrix 4

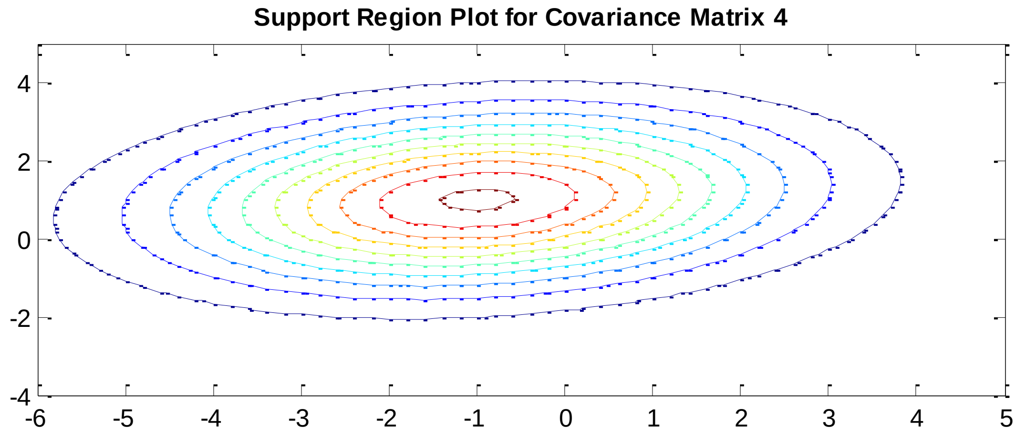

Figure 13: Support Region Plot for GRV2D1 and Covariance Matrix 4

We can compare these figures to the ones in lecture 2's power points and observe that the support regions are approximately identical. The only difference is the axis which is expected since they represent the mean of the Gaussian random variables, which is different to our problem set.

<u>Appendix:</u>

```
%% Machine Learning Homework assignment 1
% Pete Mollica:
% Generating the Gaussian random vectors and ploting their PDF's

% randn() function generates normally distributed random numbers
GRV1 = 1 + randn([10000,1]);     %mean of 1 variance of 1
GRV2 = -1 + randn([10000,1]);    %mean of -1 variance of 1
MAX1 = max(GRV1);
MAX2 = max(GRV2);
MIN1 = min(GRV1);
MIN2 = min(GRV2);
STEP1 = (MAX1 - MIN1)/1000;
STEP2 = (MAX2 - MIN2)/1000;
p1 = normpdf(MIN1:STEP1:MAX1,1,1);
p2 = normpdf(MIN2:STEP2:MAX2,-1,1);
figure(1)
plot(MIN1:STEP1:MAX1, p1);        %plotting the pdf of the GRV1
title('PDF of GRV1','FontWeight','bold');
figure(2);
plot(GRV1);                       %plotting the GRV1 data
figure(3)
hist(GRV1)                        %Plotting the Histogram of GRV1

figure(4)
plot(MIN2:STEP2:MAX2, p2);        %plotting the pdf of GRV2
title('PDF of GRV2','FontWeight','bold' );
figure(5);
plot(GRV2);                       %plotting the GRV2 data
figure(6)
```

```matlab
hist(GRV2)                          %Plotting the Histogram of GRV2
figure(7)
plot(MIN1:STEP1:MAX1, p1);
hold on;             %Plotting both of the PDF's to see the overlapping error
plot(MIN2:STEP2:MAX2, p2);
title('PDF of both GRV1 and GRV2','FontWeight','bold');


%% Part 1 Generating an optimal maximum likelihood decoder with
%  the following thresholds: 0, 0.5, -0.5

GRV1 = 1 + randn([10000,1]);
GRV2 = -1 + randn([10000,1]);
%initializing error counts
eCount1GRV1=0;      % counts for the number of errors for the first
eCount1GRV2=0;      % threshold at 0

eCount2GRV1=0;      % counts for the number of errors for the first
eCount2GRV2=0;      % threshold at 0.5

eCount3GRV1=0;      % counts for the number of errors for the first
eCount3GRV2=0;      % threshold at -0.5
for i=1:10000
    % counting all the points that are across the decision threshold at 0
    % for both Random variables (optimal threshold)
    if (GRV1(i) < 0)
        eCount1GRV1=eCount1GRV1+1;
    end
    if (GRV2(i) > 0 )
        eCount1GRV2=eCount1GRV2+1;
    end

    % counting all the points that are across the decision threshold at 0.5
    % for both Random variables
    if (GRV1(i) < 0.5)
        eCount2GRV1=eCount2GRV1+1;
    end
    if (GRV2(i) > 0.5)
        eCount2GRV2=eCount2GRV2+1;
    end

    % counting all the points that are across the decision threshold at -0.5
    % for both Random variables
    if (GRV1(i) < -0.5)
        eCount3GRV1=eCount3GRV1+1;
    end
    if (GRV2(i) > -0.5)
        eCount3GRV2=eCount3GRV2+1;
    end
end
% Calculated Probabilities
% probability errors for threshold set at 0
pError1GRV1= eCount1GRV1/10000      % Incorrectly deciding GRV2
pError1GRV2= eCount1GRV2/10000      % Incorrectly deciding GRV1
% probability errors for threshold set at 0.5
```

```
pError2GRV1= eCount2GRV1/10000      % Incorrectly deciding GRV2
pError2GRV2= eCount2GRV2/10000      % Incorrectly deciding GRV1
% probability errors for threshold set at -0.5
pError3GRV1= eCount3GRV1/10000      % Incorrectly deciding GRV2
pError3GRV2= eCount3GRV2/10000      % Incorrectly deciding GRV1

%% Part 2 Plotting the error rate across changing variances
% will calculate the error rate for each variance for when the threshold is
% set to 0

grv2 = -1 + randn([1000,1]);  % mean of -1 variance of 1
grv1data = randn([1000,1]);   % grv1 data that will be varied
grv1vector = zeros(1000,20);  % vector holding all of the random varying
                              % grv1 data, with changing variances.

eVectorGRV1=zeros(20,1);                %Error vectors for changing variances
eVectorGRV2=zeros(20,1);
for i = 1:1:20                % Increasing the variance by increments of .1
   grv1vector(:,i) = 1 + (i/10)*grv1data;
   figure(8)
   PDF1(grv1vector(:,i),1,i/10); %created PDF1 function to plot the pdf of
                                 %a given data set with a specific mean and
                                 %variance
   hold on;                      %plotted each PDF for different variance
   xlim([-1,3]);                 %Largest peak is the smallest variance(.1)

   %counting all of the error decissions for the changing variances
   %by using zero as the threshold
   for j=1:1000
        if (grv1vector(j,i) < 0)
            eVectorGRV1(i)=eVectorGRV1(i)+1;
        end
   end
end

pErrorGRV1 = eVectorGRV1/1000;
figure(9)
plot(0.1:0.1:2,pErrorGRV1);
title(['Proability Error Rate zero Threshold']);
xlabel('Variance');
ylabel('Error Rate');

%% Part 3 Finding the error rate of a 2-D GRV

figure(10)
clf;
% initalizing Multivariant Normal Random variables
GRV2D1 = mvnrnd([-1 1], [1 0; 0 1], 1000); % Mean = -1 and 1 variance matrix
is [1 0; 0 1]
GRV2D2 = mvnrnd([1 1], [1 0; 0 1], 1000);  % Mean = 1 and 1 variance matrix
is [1 0; 0 1]
% Plotting the Multivariant Normal distributions for both 2-D variables
scatter(GRV2D1(:,1),GRV2D1(:,2), 'blue');
hold on;
scatter(GRV2D2(:,1),GRV2D2(:,2),10,'red');
```

```
title('2-D Guassian Random Variable Plot');
legend('GRV2D1','GRV2D2');
%Initializing error counts
eCount2DGRV1=0;
eCount2DGRV2=0;

%Because the means for the second dimension of both variables are the same
%the optimal maximum likelihood will be a vertical decision region at zero.
%The decsion is between -1 and 1 (the means of the first dimension of the
%2-D GRV's and vertical because the covariance matrix is an identity matrix
%for both.
for i = 1:1000
    if (GRV2D1(i,1) < 0)
        eCount2DGRV1=eCount2DGRV1+1;
    end
    if (GRV2D2(i,1) > 0 )
        eCount2DGRV2=eCount2DGRV2+1;
    end
end
% Calculated Probabilities
% probability errors for optimal threshold
pError1GRV1= eCount2DGRV1/10000      % Incorrectly deciding GRV2D2
pError1GRV2= eCount2DGRV2/10000      % Incorrectly deciding GRV2D1

%as we can see the error probabilities are approximately identical, which
%tells us that the decision region was correctly chose for optimal maximum
%likelihood

%% Part 4 Generating support regions for varying covariance matrix for GRV2D1

%Initalizing covariance matrix
A = [1 0;0 1];      %Covariance matrix 1
B = [5 0;0 2];      %Covariance matrix 2
C = [1 0.5;0.5 1];  %Covariance matrix 3
D = [5 0.5;0.5 2];  %Cocariance matrix 4
mean = [-1 1];         %Mean matrix

%Plotting the PDF and the Support Region for Covariance Matrix 1
x1 = -4:.2:2; x2 = -2:.2:4;
[X1,X2] = meshgrid(x1,x2);
GRV2D3 = mvnpdf([X1(:) X2(:)],mean,A);
GRV2D3 = reshape(GRV2D3,length(x2),length(x1)); %Reshaping to fit the Surf
Plot
figure(11)
surf(x1,x2,GRV2D3);         %Plotting the PDF in 3-Dimensions
title('PDF plot of 2-Dimensional GRV');
figure(12)
contour(x1,x2,GRV2D3);      %Plotting the support region
title('Support Region Plot for Covariance Matrix 1');

%Plotting the PDF and the Support Region for Covariance Matrix 2
x1 = -6:.2:5; x2 = -4:.2:5;
[X1,X2] = meshgrid(x1,x2);
GRV2D4 = mvnpdf([X1(:) X2(:)],mean,B);
GRV2D4 = reshape(GRV2D4,length(x2),length(x1)); %Reshaping to fit the Surf
Plot
```

```
figure(13)
surf(x1,x2,GRV2D4);          %Plotting the PDF in 3-Dimensions
title('PDF plot of 2-Dimensional GRV');
figure(14)
contour(x1,x2,GRV2D4);      %Plotting the support region
title('Support Region Plot for Covariance Matrix 2');

%Plotting the PDF and the Support Region for Covariance Matrix 3
x1 = -4:.2:2; x2 = -3:.2:4;
[X1,X2] = meshgrid(x1,x2);
GRV2D5 = mvnpdf([X1(:) X2(:)],mean,C);
GRV2D5 = reshape(GRV2D5,length(x2),length(x1)); %Reshaping to fit the Surf
Plot
figure(15)
surf(x1,x2,GRV2D5);          %Plotting the PDF in 3-Dimensions
title('PDF plot of 2-Dimensional GRV');
figure(16)
contour(x1,x2,GRV2D5);      %Plotting the support region
title('Support Region Plot for Covariance Matrix 3');

%Plotting the PDF and the Support Region for Covariance Matrix 4
x1 = -6:.2:5; x2 = -4:.2:5;
[X1,X2] = meshgrid(x1,x2);
GRV2D6 = mvnpdf([X1(:) X2(:)],mean,D);
GRV2D6 = reshape(GRV2D6,length(x2),length(x1)); %Reshaping to fit the Surf
Plot
figure(17)
surf(x1,x2,GRV2D6);          %Plotting the PDF in 3-Dimensions
title('PDF plot of 2-Dimensional GRV');
figure(18)
contour(x1,x2,GRV2D6);      %Plotting the support region
title('Support Region Plot for Covariance Matrix 4');
```