CARNEGIE MELLON UNIVERSITY
APPLIED STOCHASTIC PROCESSES
(COURSE 18-751)
HOMEWORK 8

Daniel Marew

October 23, 2017

I collaborated with :

Nebyou Yismaw
Daniel Nkemelu
Agatha Niwomugizi

Q.1

(a)

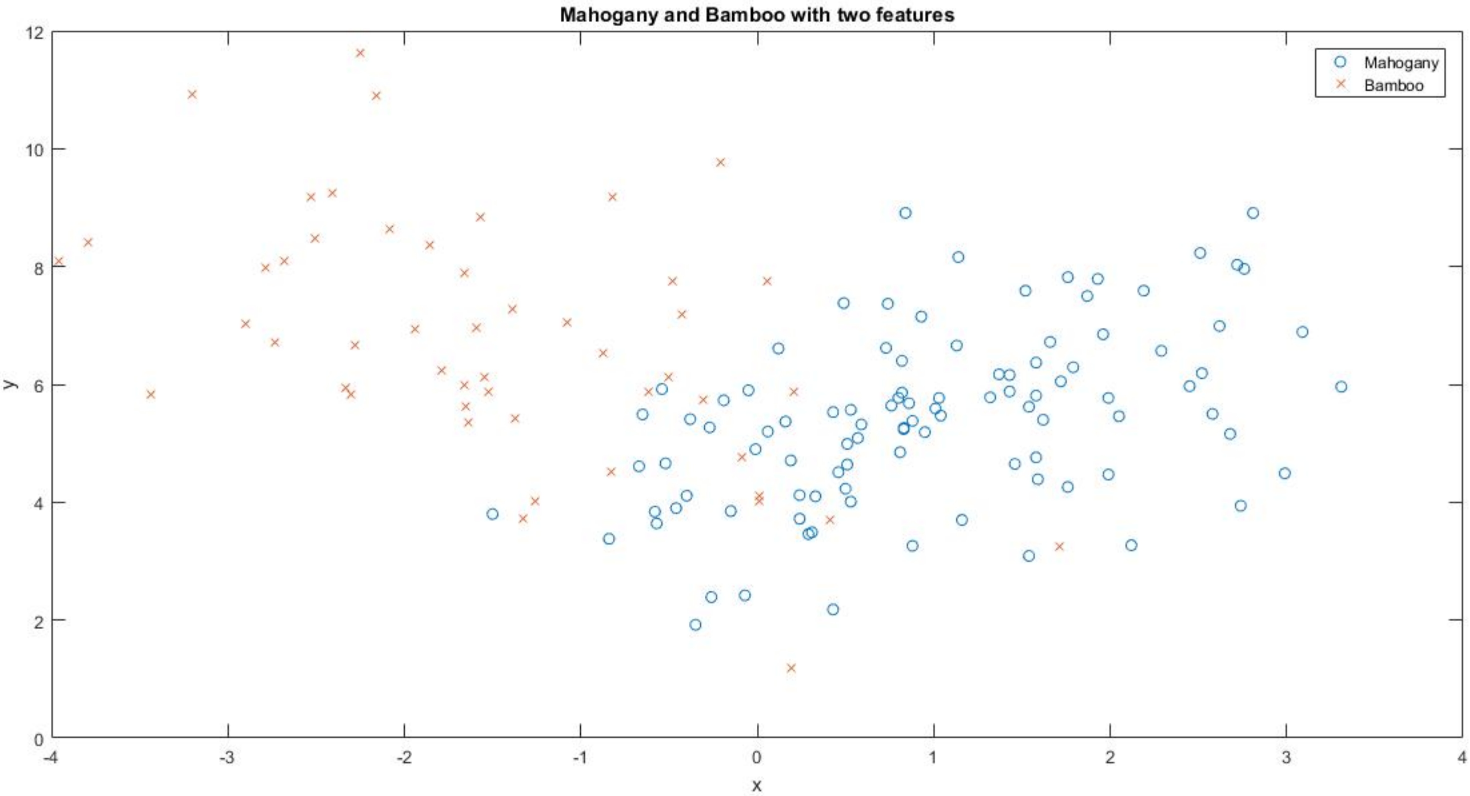

Figure 1: Mahogany and Bamboo with two features

(b)

$$P[M] = \frac{\#NumberofMahoganyInSample}{\#TotalSampleSize} = \frac{100}{150} = \frac{2}{3}$$

$$P[B] = \frac{\#NumberofBambooInSample}{\#TotalSampleSize} = \frac{50}{150} = \frac{1}{3}$$

$$\mu_{M_x} = \frac{1}{M}\sum_x M_x = 1.0044$$

$$\mu_{M_y} = \frac{1}{M}\sum_y M_y = 5.4160$$

$$\mu_M = [1.0044, 5.4160]^T$$

$$\mu_{B_x} = \frac{1}{B}\sum_x B_x = -1.4752$$

$$\mu_{B_y} = \frac{1}{B}\sum_y B_y = 6.7724$$

$$\mu_B = [-1.4752, 6.7724]^T$$

$$cov[X,Y]_M = K_M = \begin{bmatrix} 1.0852 & 0.7399 \\ 0.7399 & 2.1879 \end{bmatrix}$$

$$cov[X,Y]_B = K_B = \begin{bmatrix} 1.3980 & -1.3145 \\ -1.3145 & 4.4088 \end{bmatrix}$$

(c) Linear Estimators

$$\hat{Y}(x) = \mu_Y + \rho\frac{\sigma_Y}{\sigma_X}(x - \mu_X)$$

the slope is given by

$$slope = \rho\frac{\sigma_Y}{\sigma_X}$$

and the Intercept

$$Intercept = \mu_Y - \rho\frac{\sigma_Y}{\sigma_X}\mu_X$$

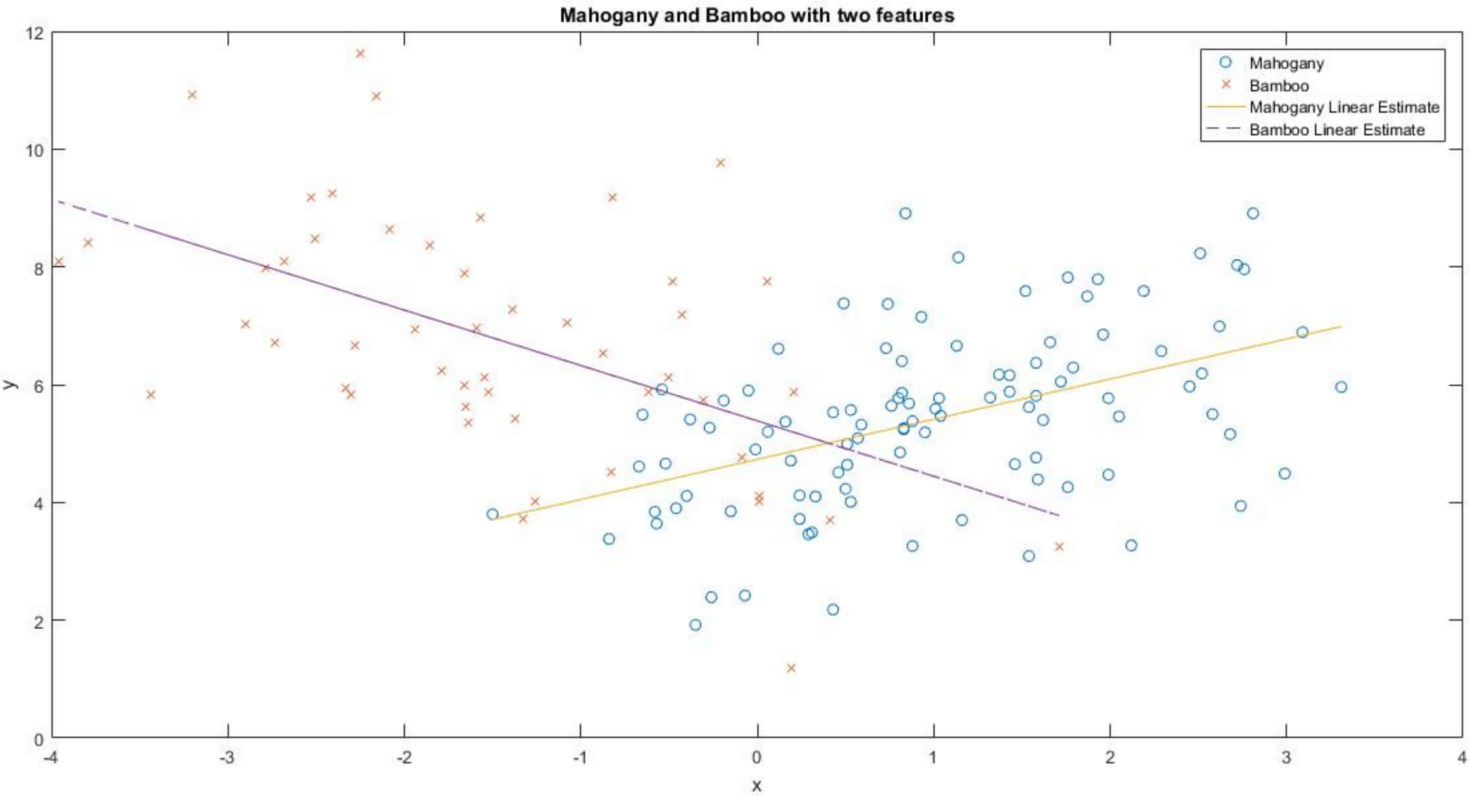

Figure 2: Mahogany and Bamboo with two features

Ans.
Using the equations above we get
Mahogany Linear Estimator Slope = 0.68
Mahogany Linear Estimato Intercept=4.73
Bamboo Linear Estimator Slope=−0.94
Bamboo Linear Estimator Intercept= 5.39

(d)

$$a = K_W^{-1}(\mu_M - \mu_B)$$

$$K_W^{-1} = \begin{bmatrix} 0.4110 & 0.0358 \\ 0.0358 & 0.1547 \end{bmatrix}$$

$$\mu_M - \mu_B = \begin{bmatrix} 2.4796 \\ -1.3564 \end{bmatrix}$$

$$a = \begin{bmatrix} 0.4110 & 0.0358 \\ 0.0358 & 0.1547 \end{bmatrix} * \begin{bmatrix} 2.4796 \\ -1.3564 \end{bmatrix}$$

$$a = \begin{bmatrix} 0.9705 \\ -0.1211 \end{bmatrix}$$

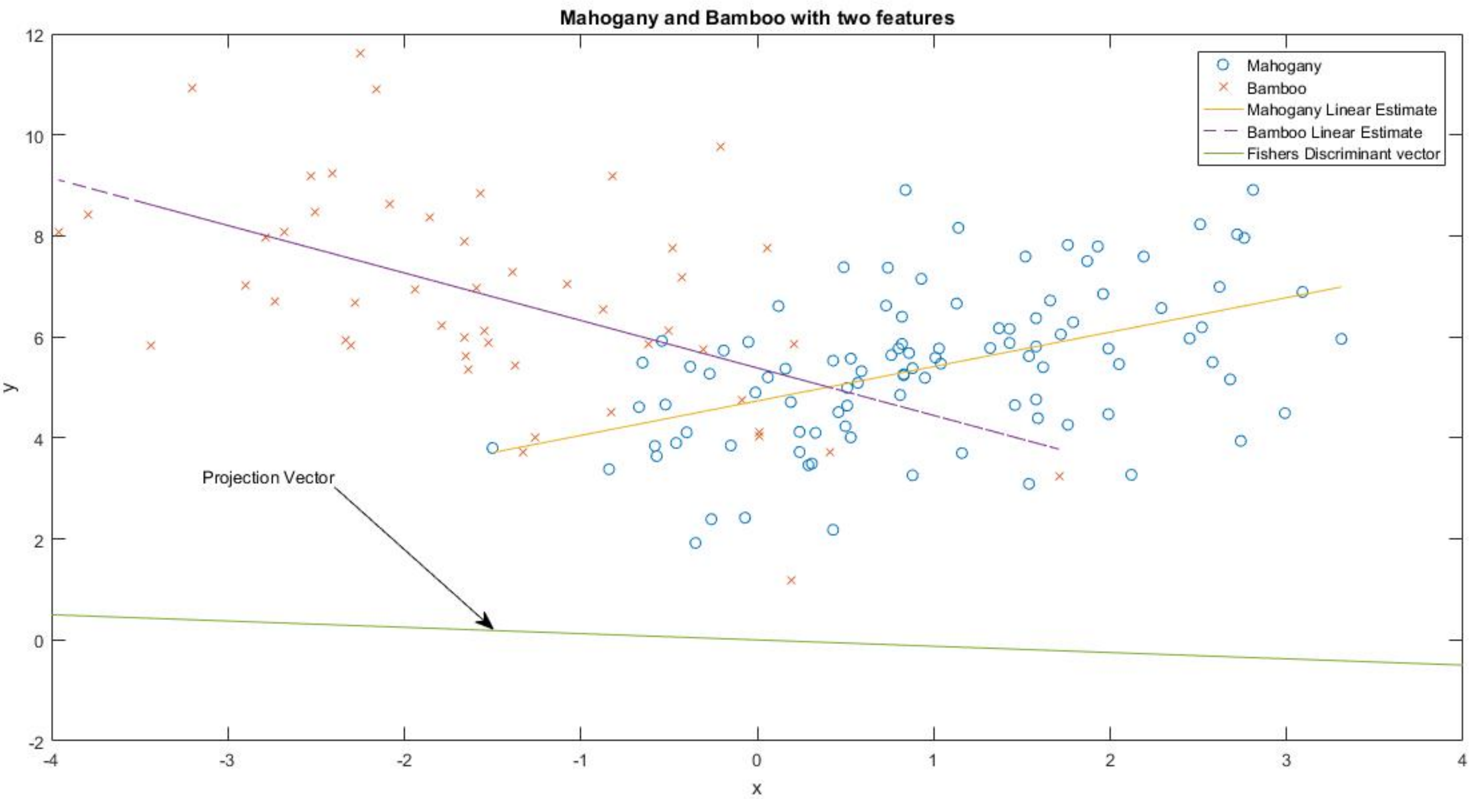

Figure 3: Mahogany and Bamboo with two features including the projection vector

(e)

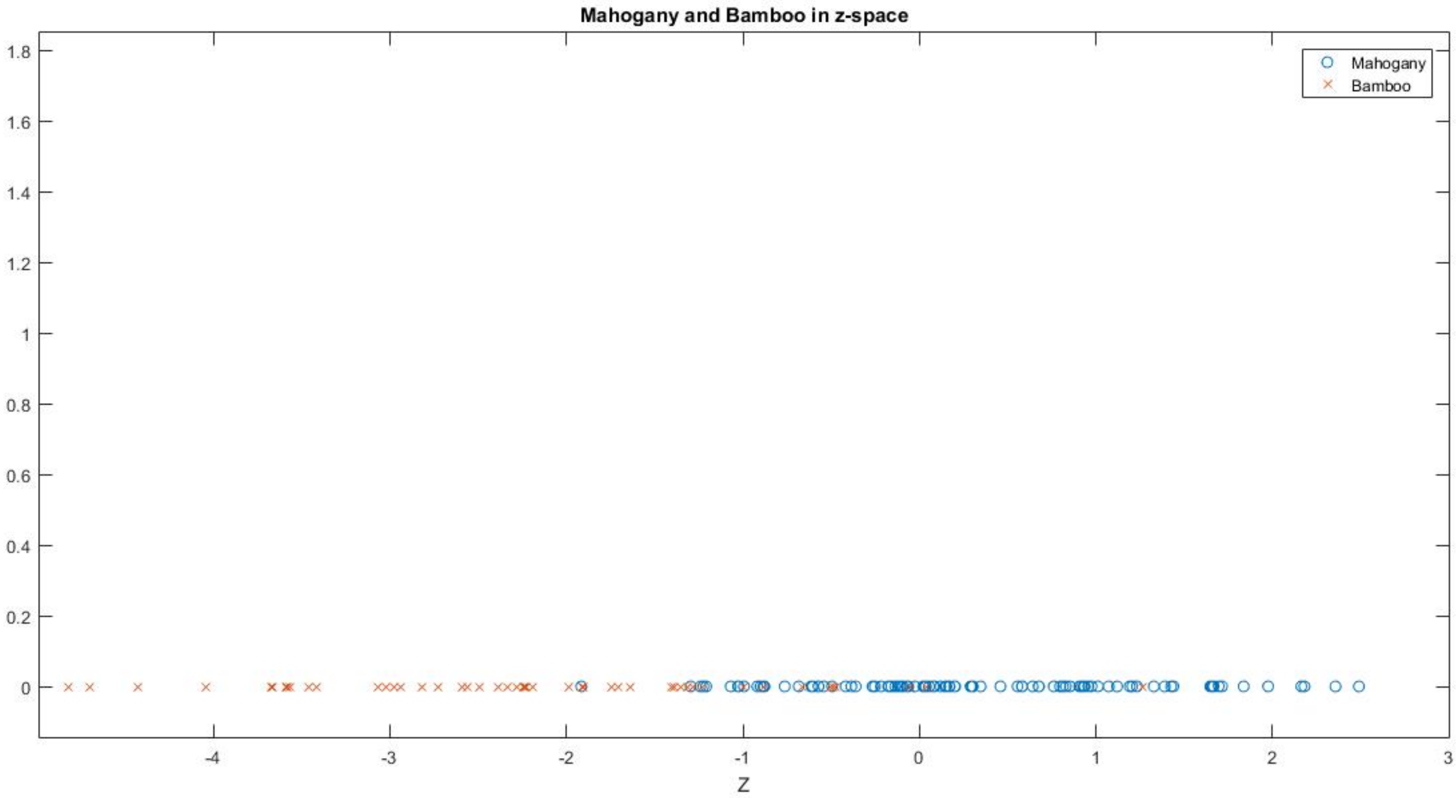

Figure 4: Mahogany and Bamboo projected to z space

$$P[M] = \frac{\#NumberofMahoganyInSampleinz - space}{\#TotalSampleSizeinz - space} = \frac{100}{150} = \frac{2}{3}$$

$$P[B] = \frac{\#NumberofBambooInSampleinz - space}{\#TotalSampleSizeinz - space} = \frac{50}{150} = \frac{1}{3}$$

$$m_M = \frac{1}{M} \sum_z M_z = 0.3191$$

$$m_B = \frac{1}{B} \sum_z B_z = -2.251$$

$$\sigma_M = 0.9383$$

$$\sigma_B = 1.3001$$

Using only the prior prob. we can come up with a simple classifier that will be correct 66% of the time. We say it is a Mahogany every time since prior Prob. of Mahogany 2∗ prior Prob. of Bamboo .

(g) Verify

$$a^T\mu_M = \begin{bmatrix}0.9705 & -0.1211\end{bmatrix} * \begin{bmatrix}1.0044\\5.4160\end{bmatrix} = 0.3191 = \mu_M$$

$$a^TK_Ma = \begin{bmatrix}0.9705 & -0.1211\end{bmatrix} * \begin{bmatrix}1.0852 & 0.7399\\0.7399 & 2.1879\end{bmatrix} * \begin{bmatrix}0.9705\\-0.1211\end{bmatrix} = 0.8804 = \sigma_M^2$$

$$a^T\mu_B = \begin{bmatrix}0.9705 & -0.1211\end{bmatrix} * \begin{bmatrix}-1.4752\\6.7724\end{bmatrix} = -2.2517 = \mu_B$$

$$a^TK_Ba = \begin{bmatrix}0.9705 & -0.1211\end{bmatrix} * \begin{bmatrix}1.3980 & -1.3145\\-1.3145 & 4.4088\end{bmatrix} * \begin{bmatrix}0.9705\\-0.1211\end{bmatrix} = 1.6904 = \sigma_B^2$$

(h)

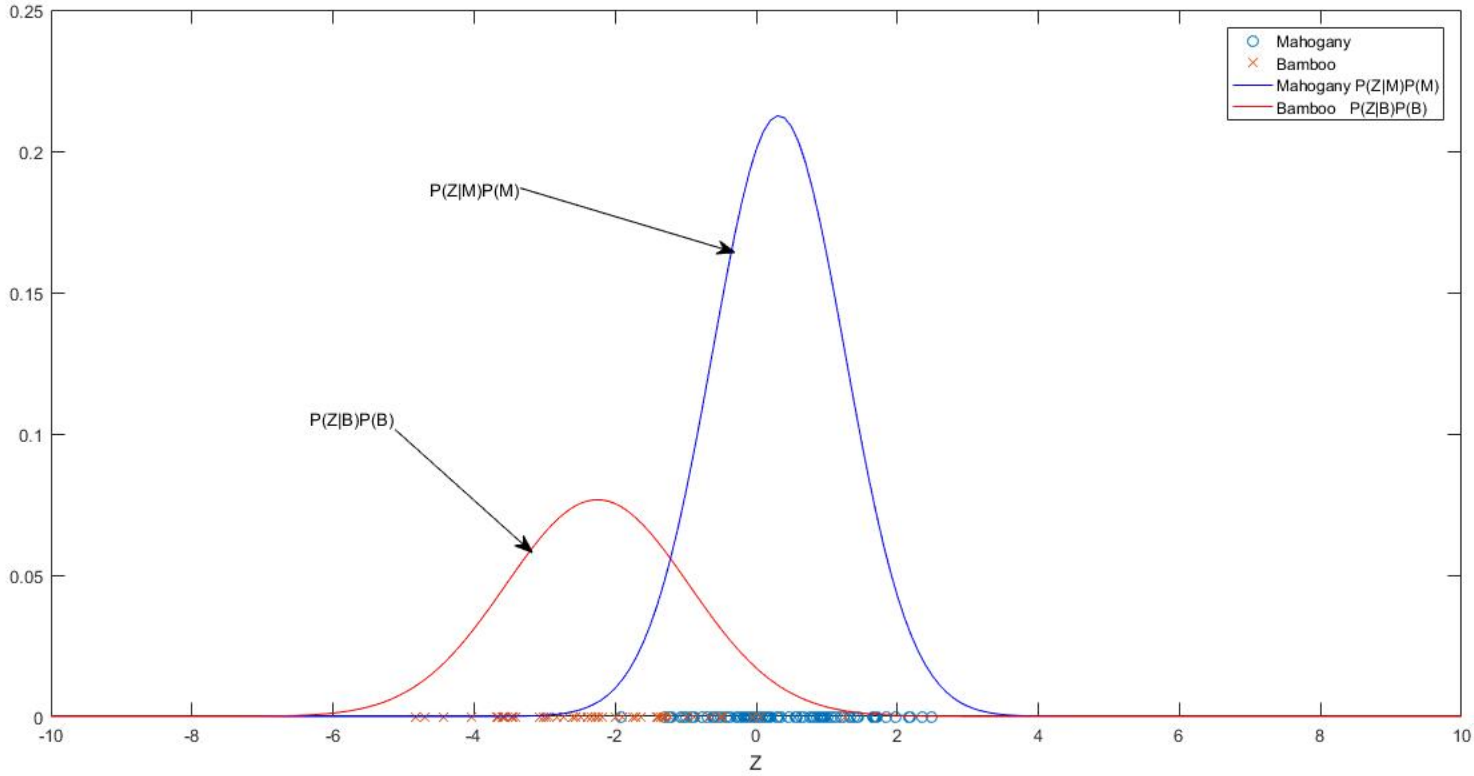

Figure 5: $P[z|M]P[M]$ and $P[z|B]P[B]$

(i)

By solving the quadratic equation we get the following two decision boundary values
$MAP_1 = -1.0656$ and $MAP_2 = 7.2927$

So anything between MAP_1 and MAP_2 is going to be labeled as Mahogany and anything out side of this region is going to be labeled as Bamboo

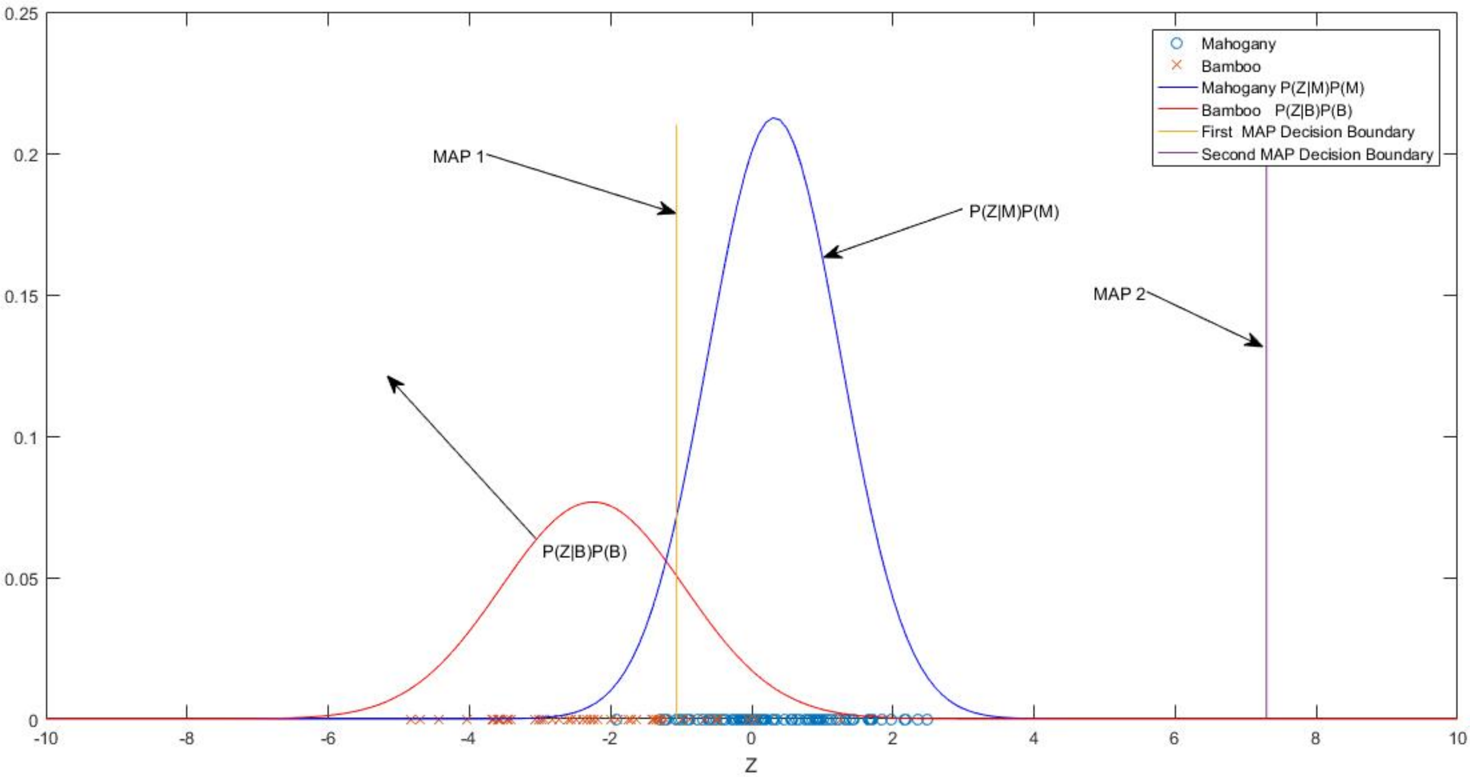

Figure 6: $P[z|M]P[M]$ and $P[z|B]P[B]$ with MAP decision boundaries

	B	M
B	0.2733	0.06
M	0.06	0.6267

Table 1: Confusion Matrix for the data

	B	M
B	0.2731	0.0603
M	0.0603	0.62

Table 2: Confusion Matrix for Normal approximation

(j) Confusion matrix

Using the MAP decision bounderies above we get Table 1

From this we get error rate of

$$errorrate = P[B, M] + P[M, B] = 0.1$$

Using the MAP decision bounderies above we get Table 2(using Q functions refer to code)

From this we get error rate of

$$errorrate = P[B, M] + P[M, B] = 0.1206$$

Code Appendix

```
Mahogany = xlsread('Mahogany.csv');
Bamboo   = xlsread('Bamboo.csv');
%(a)
%****************************************************************
figure;
plot(Mahogany(:,1),Mahogany(:,2),'o','DisplayName','Mahogany');
hold on;
plot(Bamboo(:,1),Bamboo(:,2),'x','DisplayName','Bamboo');

% legend('show');
% title('Mahogany and Bamboo with two features');
% xlabel('x');
% ylabel('y')

%figure;
%hist(Mahogany(:,2),30);

%****************************************************************
%(b)
%****************************************************************
%Prior probabilities
totalNumberOfDataPoints = length(Mahogany)+length(Bamboo);
P_M = length(Mahogany)/totalNumberOfDataPoints;
P_B = length(Bamboo)/totalNumberOfDataPoints;
%mean values
Mahoganyx = Mahogany(:,1);
Mahoganyy = Mahogany(:,2);
Bamboox   = Bamboo(:,1);
Bambooy   = Bamboo(:,2);

muMx = mean(Mahoganyx);
muMy = mean(Mahoganyy);
muBx = mean(Bamboox);
muBy = mean(Bambooy);


sigMx = var(Mahoganyx);
sigMy = var(Mahoganyy);
sigBx = var(Bamboox);
sigBy = var(Bambooy);

muM = [muMx;muMy];
muB = [muBx;muBy];

sigM = [sigMx;sigMy];
```

```matlab
sigB = [sigBx;sigBy];
%covariance matrix
KM  = cov(Mahoganyx,Mahoganyy);
KB  = cov(Bamboox,Bambooy);
% **************************************************************
%(c)
% **************************************************************
Mahoganyrho = corrcoef(Mahoganyx,Mahoganyy);
Mahoganyrho = Mahoganyrho(1,2);

Bamboorho = corrcoef(Bamboox,Bambooy);
Bamboorho = Bamboorho(1,2);

MahoganyLE = muMy + Mahoganyrho*sqrt(sigMy/sigMx)*(Mahoganyx-muMx);
BambooLE = muBy + Bamboorho*sqrt(sigBy/sigBx)*(Bamboox-muBx);

MahoganyLESlope  = Mahoganyrho*sqrt(sigMy/sigMx);
MahoganyLEIntercept  = muMy-Mahoganyrho*sqrt(sigMy/sigMx)*muMx;
BambooLESlope  = Bamboorho*sqrt(sigBy/sigBx);
BambooLEIntercept  = muBy-Bamboorho*sqrt(sigBy/sigBx)*muBx;

fprintf('Mahogany Linear Estimator Slope: %2.2f y-intercept: %2.2f \n'...
    ,MahoganyLESlope,MahoganyLEIntercept);
fprintf('Bamboo Linear Estimator Slope: %2.2f y-intercept: %2.2f \n'...
    ,BambooLESlope,BambooLEIntercept);

hold on;
plot(Mahoganyx,MahoganyLE,'DisplayName','Mahogany Linear Estimate');
hold on;
plot(Bamboox,BambooLE,'--','DisplayName','Bamboo Linear Estimate');

% **************************************************************
%(d)
% **************************************************************
%fisher's Linear Discriminate Vector
Kw = KM + KB;
a = inv(Kw)*(muM-muB);
projVectorSlope = a(2)/a(1);
projx = -4:0.1:4;
projy = projx*projVectorSlope;

```

```
plot(projx,projy,'DisplayName','Fishers Discriminant vector');
legend('show');
title('Mahogany and Bamboo with two features');
xlabel('x');
ylabel('y');
hold off;
%************************************************************
%(e)
%************************************************************
%z-space
Mahoganyz = a'*[Mahoganyx,Mahoganyy]';
Bambooz = a'*[Bamboox,Bambooy]';

figure;
plot(Mahoganyz,zeros(length(Mahoganyz),1),'o','DisplayName','Mahogany');
hold on;
plot(Bambooz,zeros(length(Bambooz),1),'x','DisplayName','Bamboo');
title('Mahogany and Bamboo in z-space')
xlabel('Z');
legend('show');


%************************************************************
%(f)
%************************************************************
%prior Prob.
totalProbInZSpace = length(Mahoganyz)+length(Mahoganyz);
P_M_z =  length(Mahoganyz)/totalProbInZSpace;
P_B_z = length(Bambooz)/totalProbInZSpace;
%mean Values
mM = mean(Mahoganyz);
mB = mean(Bambooz);
%standard deviation
sdM = sqrt(var(Mahoganyz));
sdB = sqrt(var(Bambooz));
%simple classifier always say Mahogany
%************************************************************
%(g)
%************************************************************
mMProj = a'*muM;
```

```
sigMProj = a'*KM*a;

mBProj = a'*muB;
sigBProj = a'*KB*a;

fprintf('mM: %2.2f mMProj: %2.2f \n',mM,mMProj);
fprintf('mB: %2.2f mBProj: %2.2f \n',mB,mBProj);

fprintf('sigM: %2.2f mMProj: %2.2f \n',sdM^2,sigMProj);
fprintf('sigB: %2.2f mBProj: %2.2f \n',sdB^2,sigBProj);
%************************************************************
%(h)
%************************************************************
x = -10:0.1:10;
MahoganyApprox = normpdf(x,mM,sdM);
BambooApprox = normpdf(x,mB,sdB);
figure;
plot(Mahoganyz,zeros(length(Mahoganyz),1),'o','DisplayName','Mahogany');
hold on;
plot(Bambooz,zeros(length(Bambooz),1),'x','DisplayName','Bamboo');
plot(x,MahoganyApprox*P_M_z,'b','DisplayName','Mahogany P(Z|M)P(M)');
hold on;
plot(x,BambooApprox*P_B_z,'r','DisplayName','Bamboo  P(Z|B)P(B)');

%************************************************************
%(i)
%************************************************************
%ax^2+bx+c>0 map decision boundary
a = (1/sigBProj-1/sigMProj);
b = -2*(mBProj/sigBProj-mMProj/sigMProj);
c = (mBProj^2/sigBProj-mMProj^2/sigMProj)+log((P_M_z*sigBProj)/(P_B_z*sigMProj));
MapDecsionBoundaryOne = (-b - sqrt(b^2-4*a*c))/(2*a);
MapDecsionBoundaryTwo = (-b + sqrt(b^2-4*a*c))/(2*a);
%plot([MapDecsionBoundaryTwo,MapDecsionBoundaryTwo],[0,0.21],'DisplayName','First  MAP Decision Boundary');
%plot([MapDecsionBoundaryOne,MapDecsionBoundaryOne],[0,0.21],'DisplayName','Second MAP Decision Boundary');
xlabel('Z');
legend('show');
PBB = sum(Bambooz<MapDecsionBoundaryTwo)+sum(Bambooz>MapDecsionBoundaryOne);
```

```
PBM = sum(Mahoganyz<MapDecsionBoundaryTwo) + sum(Mahoganyz>
    MapDecsionBoundaryOne);
PMB = sum(Bambooz>MapDecsionBoundaryTwo)-sum(Bambooz>
    MapDecsionBoundaryOne);
PMM = sum(Mahoganyz>MapDecsionBoundaryTwo)-sum(Mahoganyz>
    MapDecsionBoundaryOne);
result = [PBB,PMB;PMB,PMM]./totalNumberOfDataPoints;
fprintf('error rate: %2.2f \n',1-sum(diag(result)));
%with the gaussian approximation
PMMG  = (qfunc((MapDecsionBoundaryTwo-mMProj)/sdM)-qfunc((
    MapDecsionBoundaryOne-mMProj)/sdM))*(2/3);
PBBG  = (1-qfunc((MapDecsionBoundaryTwo-mBProj)/sdB)+qfunc((
    MapDecsionBoundaryOne-mBProj)/sdB))*(1/3);
PMBG  = ((qfunc((MapDecsionBoundaryTwo-mBProj)/sdB)- qfunc((
    MapDecsionBoundaryOne-mBProj)/sdB)))*(1/3);
PBMG  = ((1-qfunc((MapDecsionBoundaryTwo-mMProj)/sdM)+ qfunc((
    MapDecsionBoundaryOne-mMProj)/sdM)))*(2/3);
resultG = [PBBG,PMBG;PMBG,PMMG];
```