Homework 3

ECE 6790 - Information Processing Models in Neural Systems

Harsh Bhate (903424029)

Problem 1

(a). The estimated marginal distribution is:

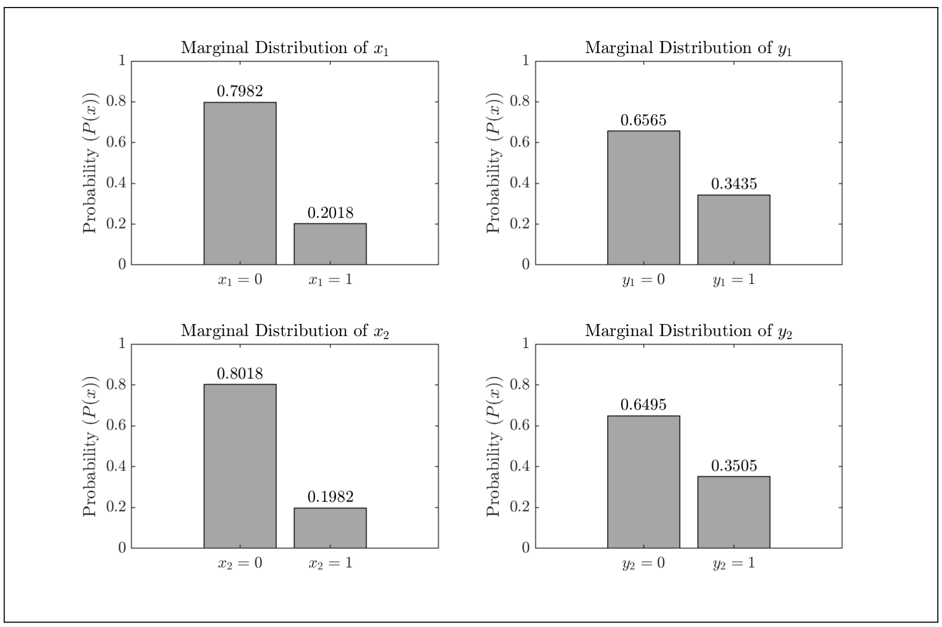
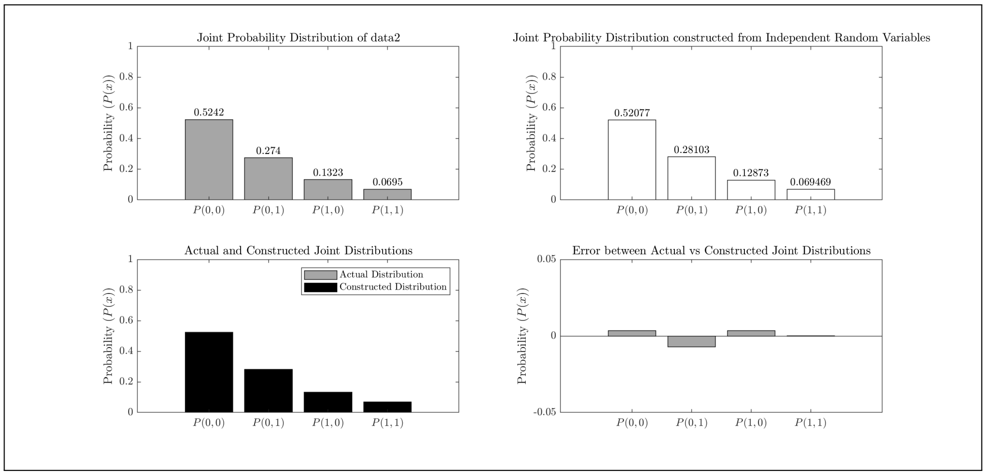


Figure 1: Marginal Probability Distribution of x_1, y_1, x_2 and y_2

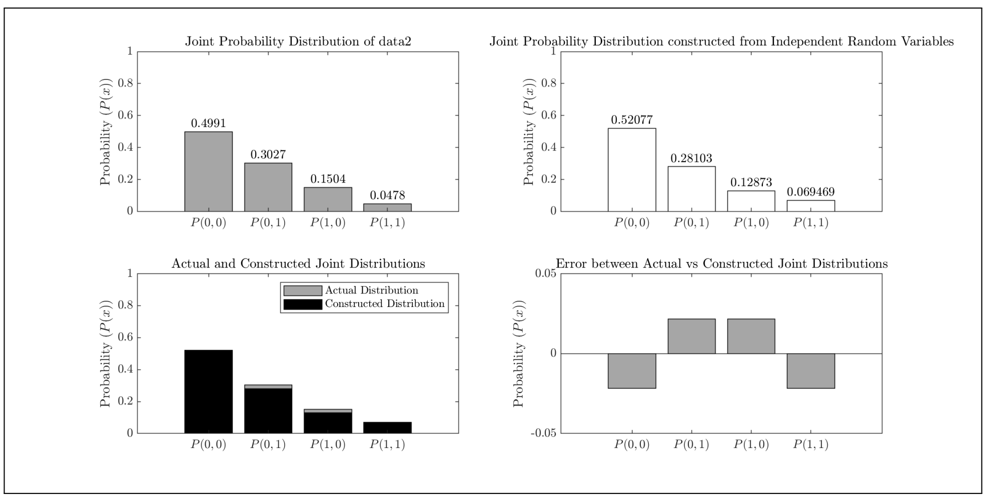
(b). The Probability distribution is said to be independent when,

$$P(X,Y) = P(X) \cdot P(Y)$$

Assuming both datasets are indepenent, we compare the reconstructed distribution assuming independence and actual distribution. The results of the comparison are as follows:



(a) Actual vs Constructed Distribution for Data 1



(b) Actual vs Constructed Distribution for Data 2

Figure 2: Actual vs Constructed Distribution for Data 1 and Data 2

From observing the error difference between the actual and reconstructed distributions in Data 2, we can safely conclude that $P(X,Y) \neq P(X) \cdot P(Y)$ for data 2 and thus, data 2 is not independent.

Problem 2

(a). The Generated Data is:

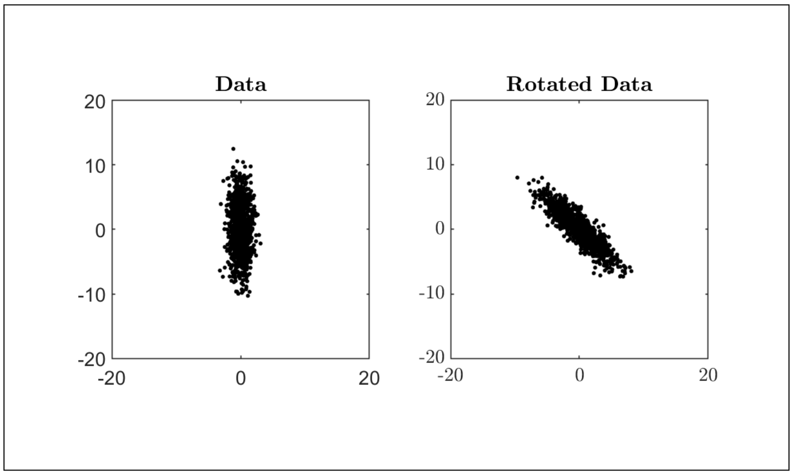


Figure 3: Original and Rotated Distribution

The marginal Distribution of the generated data is:

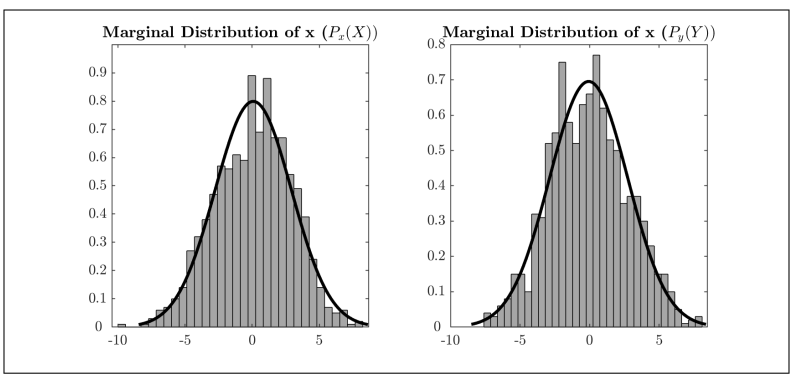


Figure 4: Marginal Distributions

(b). The covariance of a rotated data set with given covariance (Σ_{org}) is:

$$\begin{aligned}\Sigma &= A^{-1}\Sigma_o rgA \\ &= \begin{bmatrix} 8.5 & -7.5 \\ -7.5 & 8.5 \end{bmatrix}\end{aligned}$$

The MATLAB estimated Covariance is:

$$\Sigma = \begin{bmatrix} 8.5136 & -7.4786 \\ -7.4786 & 8.4395 \end{bmatrix}$$

(c). The Eigenvalues of the covariance Matrix are:

$$\lambda_{\mathbf{1}} = 0.9978, \lambda_{\mathbf{2}} = 15.9552]$$

The Corresponding Eigenvectors are:

$$\mathbf{V} = \begin{bmatrix} -0.7054 & -0.7089 \\ -0.7089 & 0.7054 \end{bmatrix}$$

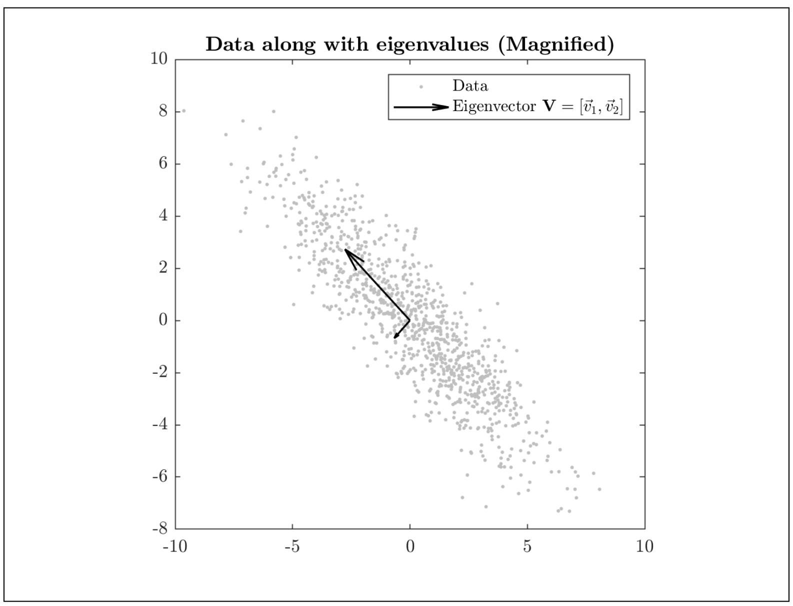


Figure 5: Scaled Eigenvectors of Covariance Matrix overlapped on Data

(d). The new Dataset is:

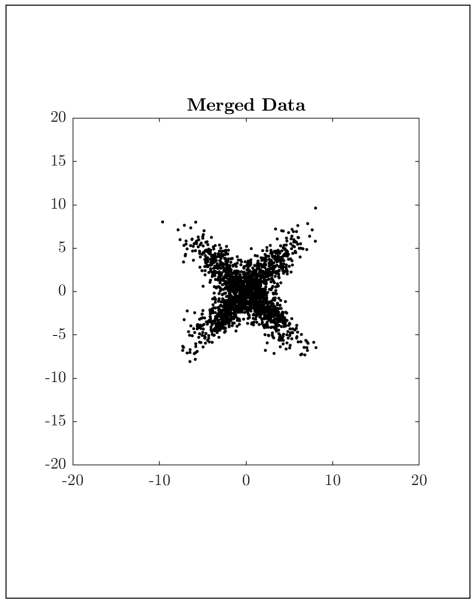


Figure 6: Merged Distribution

The Corresponding Marginal distribution is:

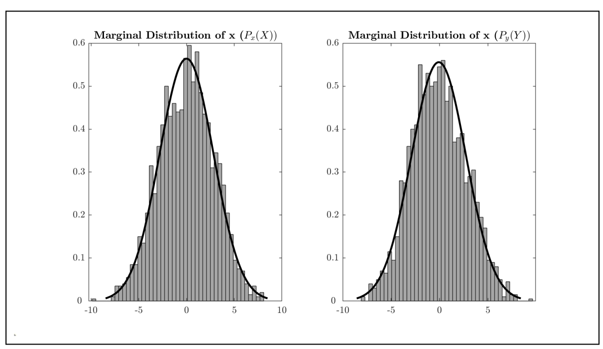


Figure 7: Marginal distribution of the Merged Data

The Data is not Gaussian Distributed anymore.

(d). The Eigenvalues for the new dataset are:

$$_1 = 8;_2 = 8$$

The corresponding Eigenvectors are:

$$\mathbf{V} = \begin{bmatrix} 0.0649 & -0.9979 \\ -0.9979 & -0.0649 \end{bmatrix}$$

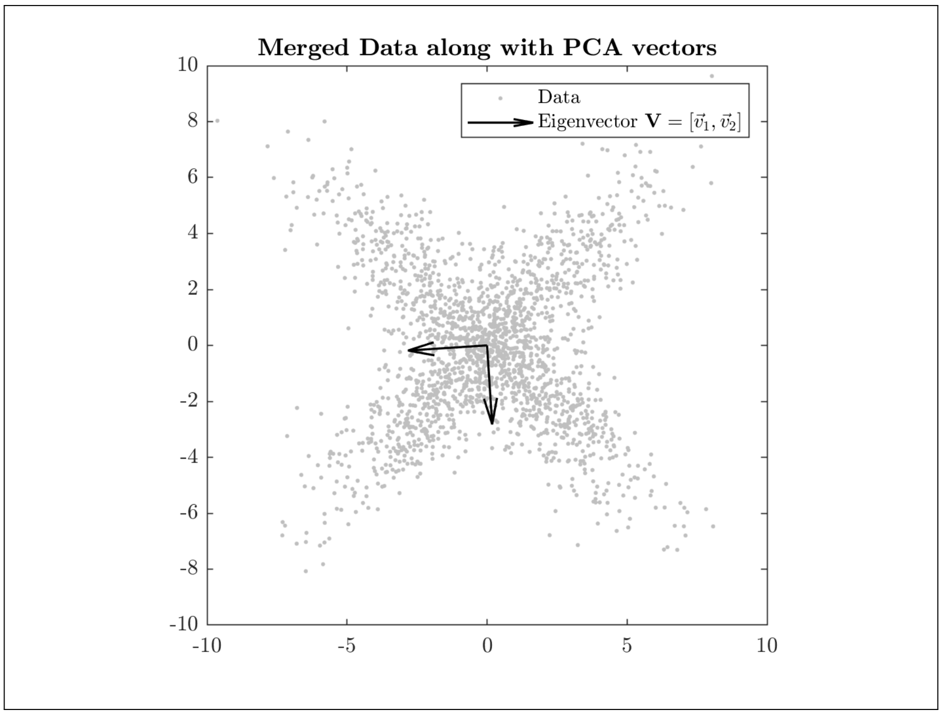


Figure 8: PCA Vectors along with merged data

Problem 3

(a).

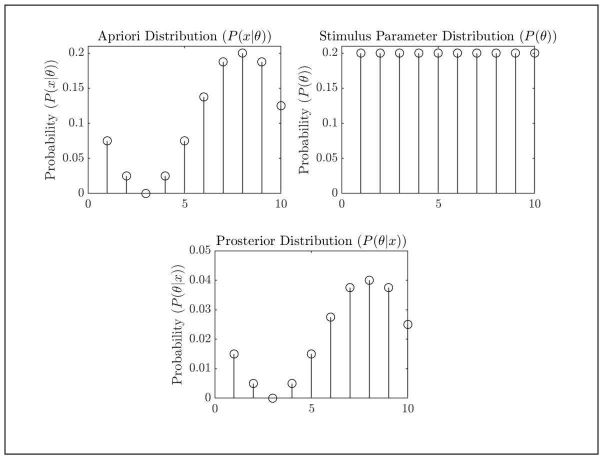


Figure 9: Prosterior Distribution

(b).

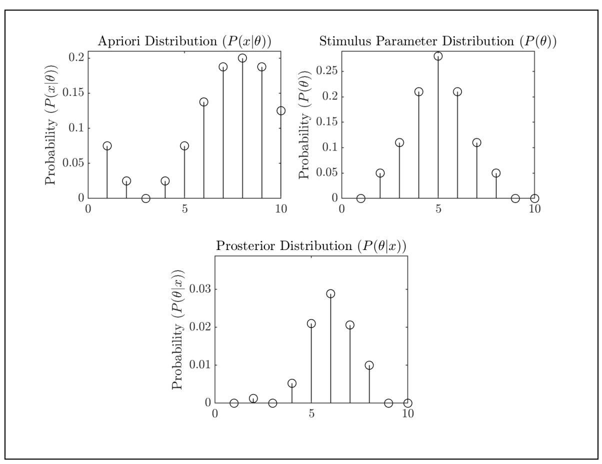


Figure 10: Prosterior Distribution

(c).

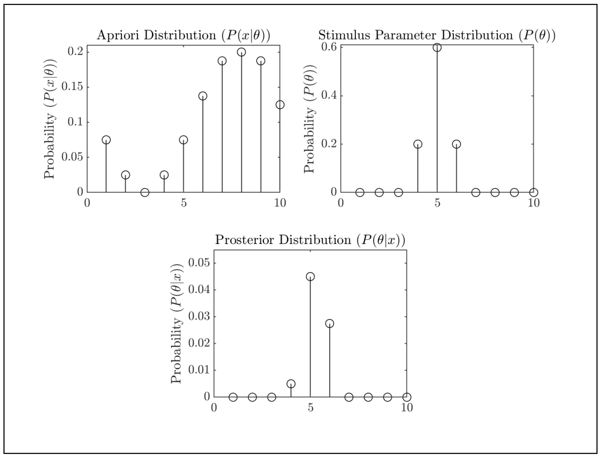


Figure 11: Prosterior Distribution

(d). The MLE and MAP of the distribution is defined as:

$$MLE = \underset{\theta}{argmax} P(x|\theta)$$

$$MAP = \underset{\theta}{argmax} P(\theta|x)$$

In case (a), $P(\theta)$ is equally distributed. Therefore, all the values of θ are equally likely. Thus, the MLE and MAP are the equal ($MLE = MAP = 8$).

In case (b), $P(\theta)$ is more concentrated towards the centre between values 4,5 and 6. However, $P(x|\theta)$ is more concentrated towards values 7, 8 and 9. Thus, the MAP is not equal to MLE as the latter considers A priori distribution which is different from the Prosterior distribution. Here, $MLE = 8, MAP = 6$.

In case (c), $P(\theta)$ is centrally concentrated at 5. Thus, considering the prosterior distribution ($P(\theta|x)$), the MAP would be skewed towards 5. Here, $MLE = 8, MAP = 5$.

In all the cases, since the A Priori distribution is the same, the MLE remains the same. ($MLE = 8$).

Appendix

Code for Problem 1

```
%% Homework - 3
%  Question: 1
%  Author: Harsh Bhate
%% Pre-loading and Pre-processing.
%  Loading Data to MATLAB
clear all;
clc;
load ('HW_corr_data.mat');
%% Part 1(a). Estimating Marginal Distribution from Joint Distribution
[px1,py1,px2,py2] = marginal (data1, data2);
marginal_dist = [px1,py1,px2,py2]
%% Part 1(b). Checking for Independence
[p00,p01,p10,p11] = conditional (data1);
conditional_dist = [p00,p01,p10,p11]
actualVsMarginal (px2,py2, conditional_dist)
%% Marginal PDF
function [fx1,fy1,fx2,fy2] = marginal(dist1, dist2)
    %   Function to find the marginal PDF and plot it for
    %   two distributions dist1 and dist2.

    %   Extracting the x and y points for each distribution
    x1 = dist1(:,1);
    y1 = dist1(:,2);
    x2 = dist2(:,1);
    y2 = dist2(:,2);
    %   Computing the PDF
    fx1 = [(length(x1) - sum(x1))/length(x1), sum(x1)/length(x1)];
    fy1 = [(length(y1) - sum(y1))/length(y1), sum(y1)/length(y1)];
    fx2 = [(length(x2) - sum(x2))/length(x2), sum(x2)/length(x2)];
    fy2 = [(length(y2) - sum(y2))/length(y2), sum(y2)/length(y2)];
    %   Setting the labels for plotting
    x1Labels ={'$x_{1} = 0$'; '$x_{1} = 1$'};
    y1Labels ={'$y_{1} = 0$'; '$y_{1} = 1$'};
    x2Labels ={'$x_{2} = 0$'; '$x_{2} = 1$'};
```

```
y2Labels ={'$y_{2} = 0$'; '$y_{2} = 1$'};
%   Plotting the graph
figure();
subplot (2,2,1);
bar(fx1, 'stacked','FaceColor',[0.65 0.65 0.65]);
set(gca,'xticklabel',x1Labels)
title ('Marginal Distribution of $x_{1}$','Interpreter','latex')
ylabel ('Probability ($P(x)$)','Interpreter','latex')
ylim([0.0 1.0])
text(1:length(fx1),fx1,num2str(fx1') ,...
    'vert','bottom','horiz','center', 'Interpreter','latex');
set(gca,'TickLabelInterpreter','latex')
subplot (2,2,2);
bar(fy1, 'stacked','FaceColor',[0.65 0.65 0.65]);
set(gca,'xticklabel',y1Labels)
title ('Marginal Distribution of $y_{1}$', 'Interpreter','latex')
ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
ylim([0.0 1.0])
text(1:length(fy1),fy1,num2str(fy1') ,...
    'vert','bottom','horiz','center', 'Interpreter','latex');
set(gca,'TickLabelInterpreter','latex')
subplot (2,2,3);
bar(fx2, 'stacked','FaceColor',[0.65 0.65 0.65]);
set(gca,'xticklabel',x2Labels)
title ('Marginal Distribution of $x_{2}$', 'Interpreter','latex')
ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
ylim([0.0 1.0])
text(1:length(fx2),fx2,num2str(fx2') ,...
    'vert','bottom','horiz','center', 'Interpreter','latex');
set(gca,'TickLabelInterpreter','latex')
subplot (2,2,4);
bar(fy2, 'stacked','FaceColor',[0.65 0.65 0.65]);
set(gca,'xticklabel',y2Labels)
title ('Marginal Distribution of $y_{2}$','Interpreter','latex')
ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
ylim([0.0 1.0])
```

```
    text(1:length(fy2),fy2,num2str(fy2'),...
        'vert','bottom','horiz','center', 'Interpreter','latex');
    set(gca,'TickLabelInterpreter','latex')
end
%% Function to plot continuous PDF
function [p00, p01, p10, p11] = conditional(dist)
    %   Function to plot conditional prob of binomial distribution dist

    %   Calculating the conditional PDF
    p00 = sum(dist(:, 1) == 0 & dist(:, 2) == 0)/length(dist);
    p01 = sum(dist(:, 1) == 0 & dist(:, 2) == 1)/length(dist);
    p10 = sum(dist(:, 1) == 1 & dist(:, 2) == 0)/length(dist);
    p11 = sum(dist(:, 1) == 1 & dist(:, 2) == 1)/length(dist);
    %   Labels
    xLabels ={'$P(0,0)$';'$P(0,1)$';'$P(1,0)$'; '$P(1,1)$'};
    %   Plotting
    figure();
    F = [p00, p01, p10, p11];
    bar(F, 'stacked','FaceColor',[0.65 0.65 0.65]);
    set(gca,'xticklabel',xLabels)
    title ('Joint Probability Distribution', ...
        'Interpreter','latex')
    ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
    ylim([0.0 1.0])
    text(1:length(F),F,num2str(F'),...
        'vert','bottom','horiz','center', 'Interpreter','latex');
    set(gca,'TickLabelInterpreter','latex')
end
%% Function to plot Conditional PDF
function actualVsMarginal (Fx, Fy, conditional)
    %   Function to plot the marginal and conditional prob

    %   Extracting Distributions
    px0 = Fx(1)
    px1 = Fx(2)
    py0 = Fy(1)
```

```
    py1 = Fy(2)
    %   Plotting independence assumed distributions
    p_00 = px0*py0;
    p_01 = px0*py1;
    p_10 = px1*py0;
    p_11 = px1*py1;
    P_ind = [p_00, p_01, p_10, p_11]
    %   Labels
    xLabels ={'$P(0,0)$';'$P(0,1)$';'$P(1,0)$'; '$P(1,1)$'};
    %   Plotting
    figure();
    subplot(2,2,1)
    bar(conditional, 'stacked','FaceColor',[0.65 0.65 0.65]);
    set(gca,'xticklabel',xLabels)
    title ('Joint Probability Distribution of data2', ...
        'Interpreter','latex')
    ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
    ylim([0.0 1.0])
    text(1:length(conditional),conditional,num2str(conditional'),...
        'vert','bottom','horiz','center', 'Interpreter','latex');
    set(gca,'TickLabelInterpreter','latex')

    subplot(2,2,2)
    bar(P_ind, 'stacked','FaceColor',[1.0 1.0 1.0]);
    set(gca,'xticklabel',xLabels)
    title ('Joint Probability Distribution constructed from Independent Random Variables', ...
        'Interpreter','latex')
    ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
    ylim([0.0 1.0])
    text(1:length(P_ind),P_ind,num2str(P_ind'),...
        'vert','bottom','horiz','center', 'Interpreter','latex');
    set(gca,'TickLabelInterpreter','latex')

    subplot(2,2,3)
    bar(conditional, 'stacked','FaceColor',[0.65 0.65 0.65]);
    hold on
    bar(P_ind, 'stacked','FaceColor',[0.0 0.0 0.0]);
```

```
    set(gca,'xticklabel',xLabels)
    title ('Actual and Constructed Joint Distributions',...
        'Interpreter', 'Latex')
    ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
    ylim([0.0 1.0])
    set(gca,'TickLabelInterpreter','latex')
    leg = legend('Actual Distribution','Constructed Distribution');
    set(leg,'Interpreter','latex');

    subplot(2,2,4)
    diff = conditional - P_ind
    bar(diff, 'stacked','FaceColor',[0.65 0.65 0.65]);
    set(gca,'xticklabel',xLabels)
    title ('Error between Actual vs Constructed Joint Distributions', ...
        'Interpreter','latex')
    ylim([-0.05 0.05])
    ylabel ('Probability ($P(x)$)', 'Interpreter','latex')
    set(gca,'TickLabelInterpreter','latex')
end
```

Code for Problem 2

```
%% Homework - 3
%  Question: 2
%  Author: Harsh Bhate
%% Cleaning Up
clear all;
clc;
%% Part 2(a) Generating Dataset
mu = [0 0];
sigma = [1 0; 0 16];
R = chol(sigma);
z = repmat(mu,1000,1) + randn(1000,2)*R;
theta = - pi/4;
A = [cos(theta) -sin(theta); sin(theta) cos(theta)];
data = z*A;
figure();
subplot(1,2,1);
plot (z(:,1),z(:,2),'.k');
title ('\textbf{Data}', ...
        'Interpreter','latex')
ylim([-20 20]);
xlim([-20 20]);
axis square;
subplot(1,2,2);
plot (data(:,1),data(:,2),'.k');
title ('\textbf{Rotated Data}', ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
ylim([-20 20]);
xlim([-20 20]);
set(gca,'TickLabelInterpreter','latex')
axis square;

%   Plotting the histograms
figure()
subplot(1,2,1);
h1 = histfit(data(:,1));
set(h1(1),'facecolor',[0.65,0.65,0.65]); set(h1(2),'color
    ','k')
yt = get(gca, 'YTick');
set(gca, 'YTick', yt, 'YTickLabel', yt/100)
title ('\textbf{Marginal Distribution of x ($P_{x}(X)$})'
    , ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
```

```matlab

subplot(1,2,2);
h2 = histfit(data(:,2));
set(h2(1),'facecolor',[0.65,0.65,0.65]); set(h2(2),'color','k')
yt = get(gca, 'YTick');
set(gca, 'YTick', yt, 'YTickLabel', yt/100)
title ('\textbf{Marginal Distribution of x ($P_{y}(Y)$})', ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
%% Part 2(b) Estimating Covariance
sigma_estimated = cov (data);
disp('The Estimated Covariance is:');
disp(sigma_estimated);
%% Part 2(c) Eigenvalue
[eigenVectors, D] = eig(sigma_estimated);
disp('The Eigenvalues are:');
disp(D);
disp('The Eigenvectors are:');
disp(eigenVectors);
v1 = eigenVectors(:,1);
v2 = eigenVectors(:,2);
eigen = sqrt(diag(D));
figure();
plot (data(:,1),data(:,2) ,...
    '.', 'color', [0.75 0.75 0.75]);
hold on;
quiver(0, 0, v1(1), v1(2),eigen(1) ,...
    'LineWidth', 1, 'MaxHeadSize', 1, 'color', [0 0 0]);
hold on;
quiver(0, 0, v2(1), v2(2),eigen(2) ,...
    'LineWidth', 1, 'MaxHeadSize', 1, 'color', [0 0 0]);
hold on;
title ('\textbf{Data along with eigenvalues (Magnified)}', ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
axis square
set(gca,'TickLabelInterpreter','latex')
axis square;
leg = legend('Data','Eigenvector $\mathbf{V} = [\vec{v}_1, \vec{v}_2]$');
    set(leg,'Interpreter','latex');
%% Part (d). Generating New Data
theta = + pi/4;
```

```
A = [cos(theta) -sin(theta); sin(theta) cos(theta)];
data2 = z*A;
merged_data = vertcat(data,data2);

figure();
plot (merged_data(:,1),merged_data(:,2),'.k');
title ('\textbf{Merged Data}', ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
ylim([-20 20]);
xlim([-20 20]);
set(gca,'TickLabelInterpreter','latex')
axis square;
%   Plotting the histograms
figure()
subplot(1,2,1);
h1 = histfit(merged_data(:,1));
set(h1(1),'facecolor',[0.65,0.65,0.65]); set(h1(2),'color','k')
yt = get(gca, 'YTick');
set(gca, 'YTick', yt, 'YTickLabel', yt/200)
title ('\textbf{Marginal Distribution of x ($P_{x}(X)$})', ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')

subplot(1,2,2);
h2 = histfit(merged_data(:,2));
set(h2(1),'facecolor',[0.65,0.65,0.65]); set(h2(2),'color','k')
yt = get(gca, 'YTick');
set(gca, 'YTick', yt, 'YTickLabel', yt/200)
title ('\textbf{Marginal Distribution of x ($P_{y}(Y)$})', ...
        'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
%% Part (e) PCA
sigma_new = cov(merged_data);
[Vectors, D] = eig(sigma_new);
disp(Vectors)
v1 = Vectors(:,1);
v2 = Vectors(:,2);
eigen = sqrt(diag(D));
v1 = v1;
v2 = v2;
figure();
```

```
plot (merged_data(:,1),merged_data(:,2) ,...
      '.', 'color', [0.75 0.75 0.75]);
hold on;
quiver(0, 0, v1(1), v1(2), eigen(1), ...
     'LineWidth', 1, 'MaxHeadSize', 1, 'color', [0 0 0]);
hold on;
quiver(0, 0, v2(1), v2(2), eigen(2), ...
     'LineWidth', 1, 'MaxHeadSize', 1, 'color', [0 0 0]);
hold off;
title ('\textbf{Merged Data along with PCA vectors}', ...
         'Interpreter','latex')
set(gca,'TickLabelInterpreter','latex')
axis square
set(gca,'TickLabelInterpreter','latex')
axis square;
leg = legend('Data','Eigenvector $\mathbf{V} = [\vec{v}_1
    , \vec{v}_2]$');
     set(leg,'Interpreter','latex');
```

Code for Problem 3

```matlab
%% Homework - 3
%  Question: 3
%  Author: Harsh Bhate
%% Cleaning Stuff
clc;
clear all;
%% Common Declarations
p_xGivenTheta = [0.075 0.025 0 0.025 0.075 0.1375 0.1875 0.2 0.1875 0.125];
p_theta1 = [0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2];
p_theta2 = [0 0.05 0.11 0.21 0.28 0.21 0.11 0.05 0 0];
p_theta3 = [0 0 0 0.2 0.6 0.2 0 0 0 0];
%% Prob 1(a)
p_theta1Givenx = prosteriorProb(p_xGivenTheta, p_theta1);
%% Prob 1(b)
p_theta2Givenx = prosteriorProb(p_xGivenTheta, p_theta2);
%% Prob 1(c)
p_theta3Givenx = prosteriorProb(p_xGivenTheta, p_theta3);
%% Prob 1(d)
[~, MLE] = max(p_xGivenTheta);
[~, MAP_theta1] = max(p_theta1Givenx);
[~, MAP_theta2] = max(p_theta2Givenx);
[~, MAP_theta3] = max(p_theta3Givenx);
txt1 = sprintf('MLE = %d', MLE);
txt2 = sprintf('MAP(theta1) = %d', MAP_theta1);
txt3 = sprintf('MAP(theta2) = %d', MAP_theta2);
txt4 = sprintf('MAP(theta3) = %d', MAP_theta3);
disp (txt1);
disp (txt2);
disp (txt3);
disp (txt4);
%% Function to compute prosterior probability
function prosterior = prosteriorProb(apriori, parameter)
    prosterior = apriori.*parameter;
    figure();
    X = [1 2 3 4 5 6 7 8 9 10];
    h(1) = subplot(2,2,1);
    stem(X, apriori, 'k')
    title ('Apriori Distribution ($P(x|\theta)$)', ...
        'Interpreter','latex')
    ylabel ('Probability ($P(x|\theta)$)', 'Interpreter', 'latex')
    set(gca,'TickLabelInterpreter','latex')
    ylim([0 max(apriori)+0.01])
```

```
    h(2) = subplot(2,2,2);
    stem(X, parameter, 'k')
    title('Stimulus Parameter Distribution ($P(\theta)$)', ...
        'Interpreter','latex')
    ylabel('Probability ($P(\theta)$)', 'Interpreter','latex')
    set(gca,'TickLabelInterpreter','latex')
    ylim([0 max(parameter)+0.01])

    h(3) = subplot(2,2,3);
    stem(X, prosterior, 'k')
    title('Prosterior Distribution ($P(\theta|x)$)', ...
        'Interpreter','latex')
    ylabel('Probability ($P(\theta|x)$)', 'Interpreter','latex')
    set(gca,'TickLabelInterpreter','latex')
    ylim([0 max(prosterior)+0.01])

    pos = get(h,'Position');
    new = mean(cellfun(@(v)v(1),pos(1:2)));
    set(h(3),'Position',[new,pos{end}(2:end)])
end
```