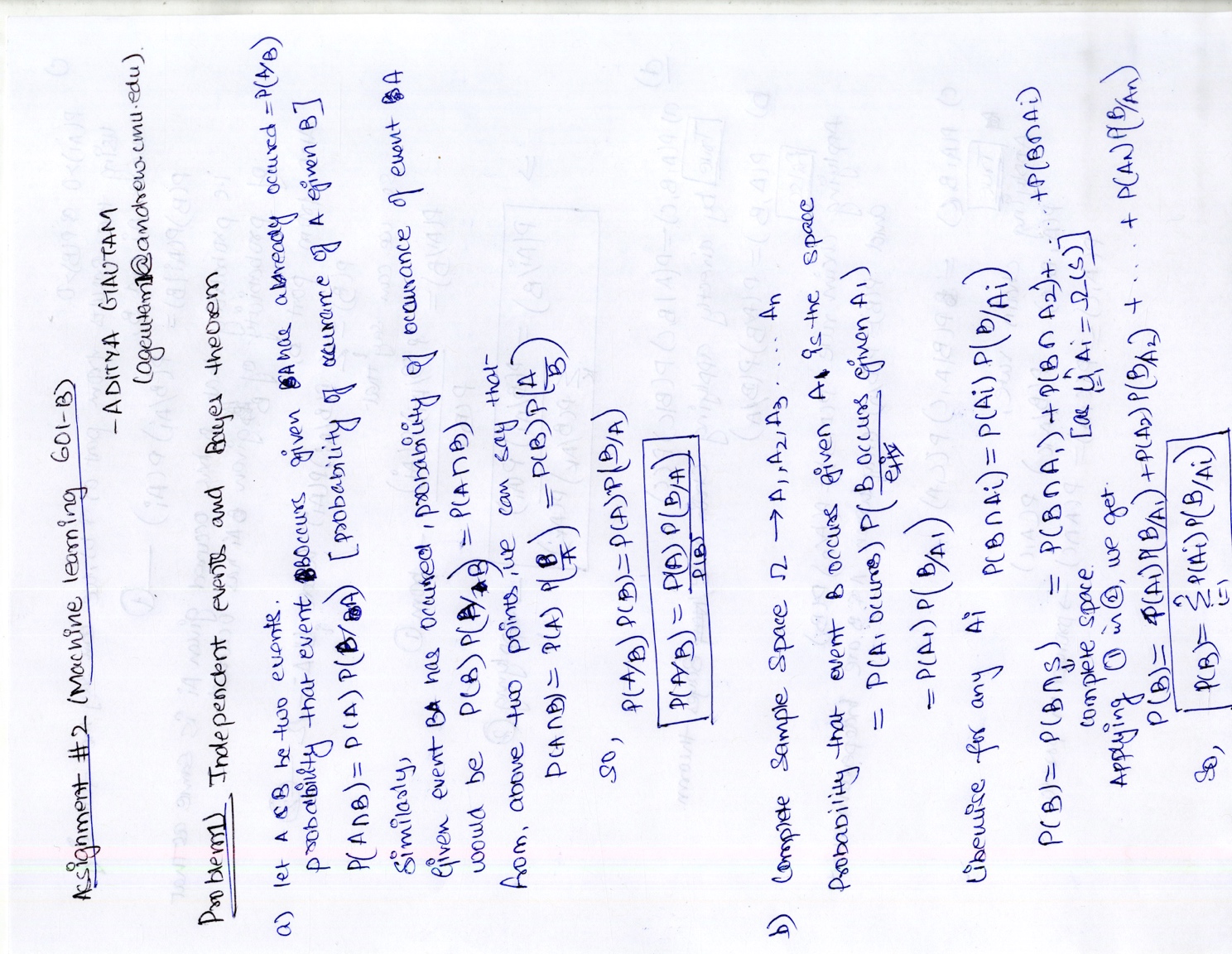
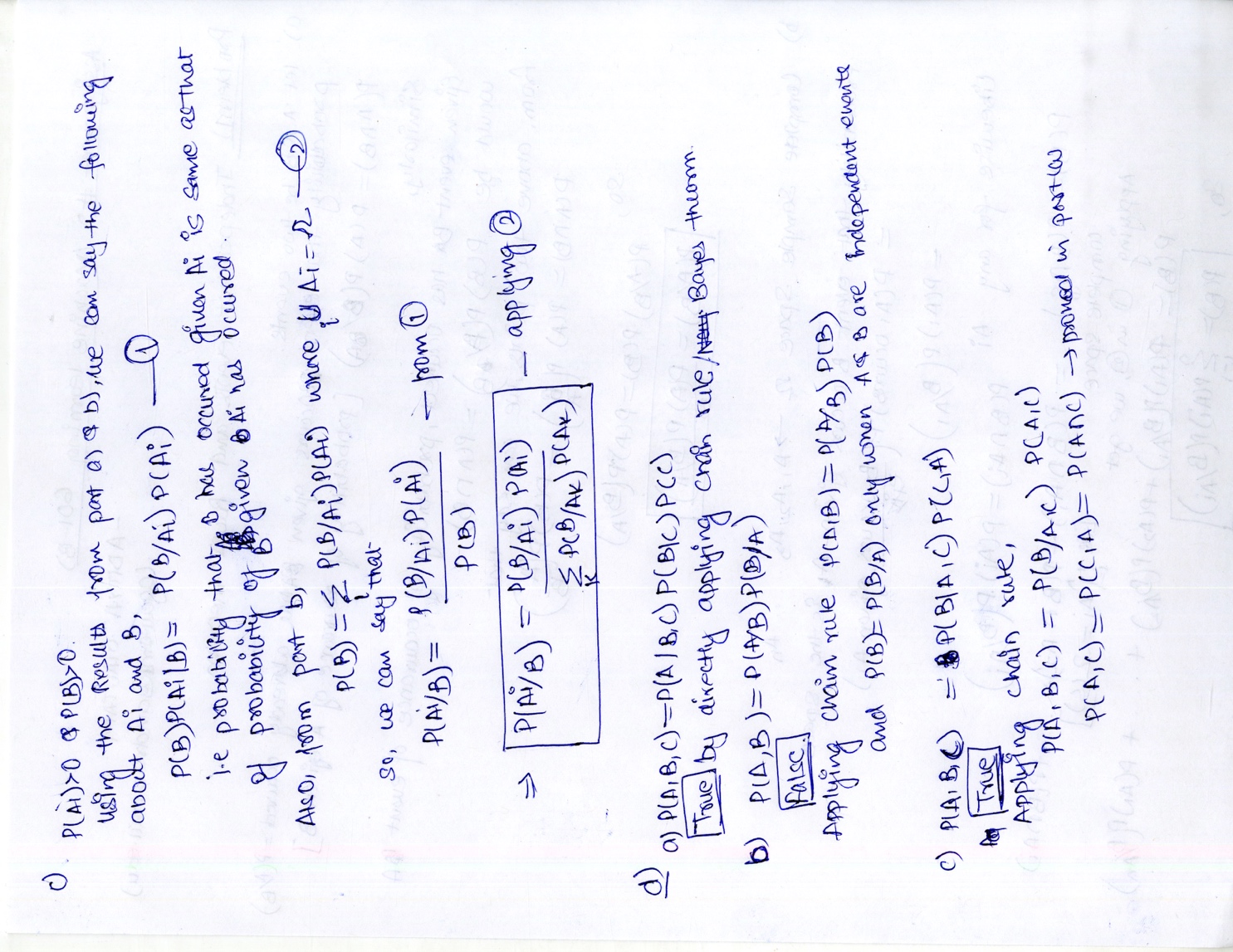
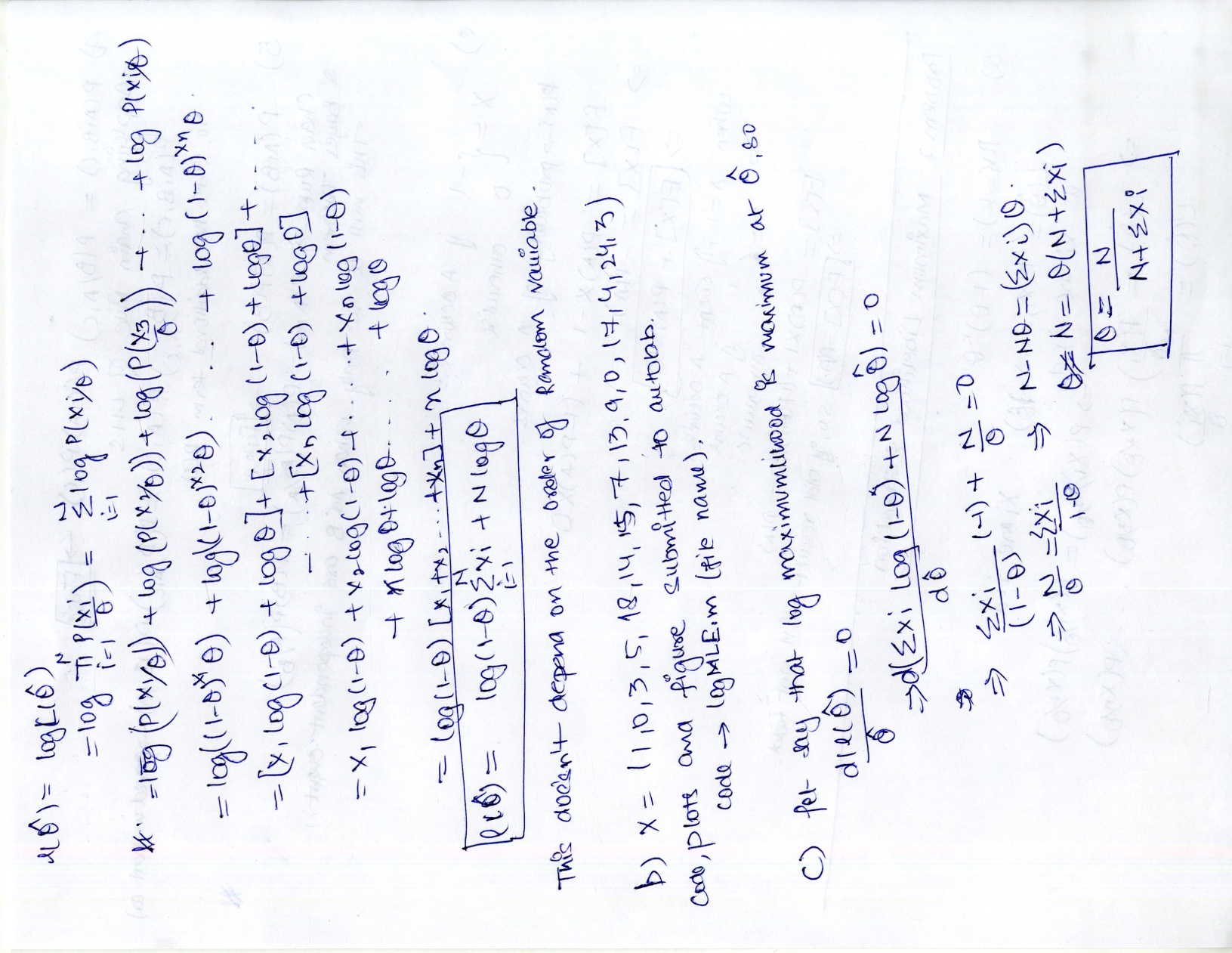
**Machine learning – Assignment#2**

**Aditya Gautam(**[**agautam1@andrew.cmu.edu)**](mailto:agautam1@andrew.cmu.edu))









**Ques 2) Part b):**

**Matlab Code**

function [ ] = logMLE()

theta = 0.01:.01:0.5;

x = [1 0 3 5 18 14 5 7 13 9 0 17 4 24 3];

y5= sum(x(1:5))\*log(1-theta) + 5\*log(theta);

y10 = sum(x(1:10))\*log(1-theta) + 10\*log(theta);

y15 = sum(x(1:15))\*log(1-theta) + 15\*log(theta);

figure

plot(theta,y5)

xlabel('theta');

ylabel('y(log MLE)');

title('Log MLE (5 points)');

figure

plot(theta,y10)

xlabel('theta');

ylabel('y(log MLE)');

title('Log MLE (10 points)');

figure

plot(theta,y15)

xlabel('theta');

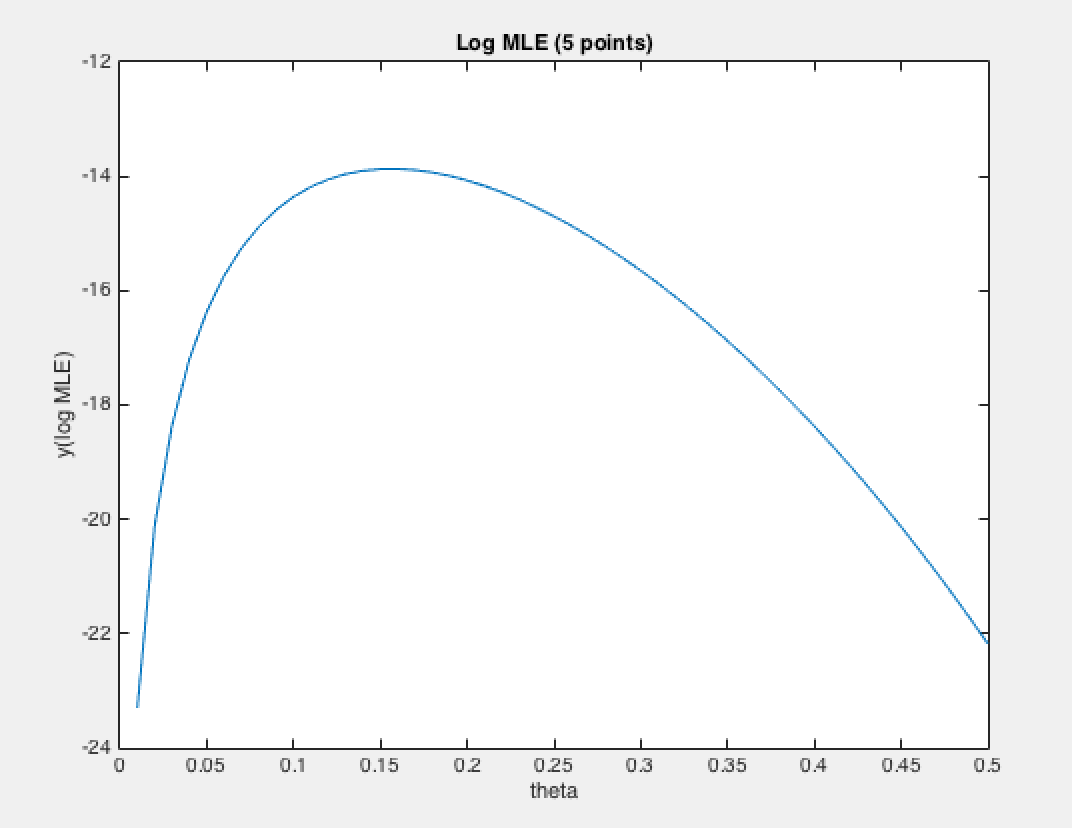
ylabel('y(log MLE)');

title('Log MLE (15 points)');

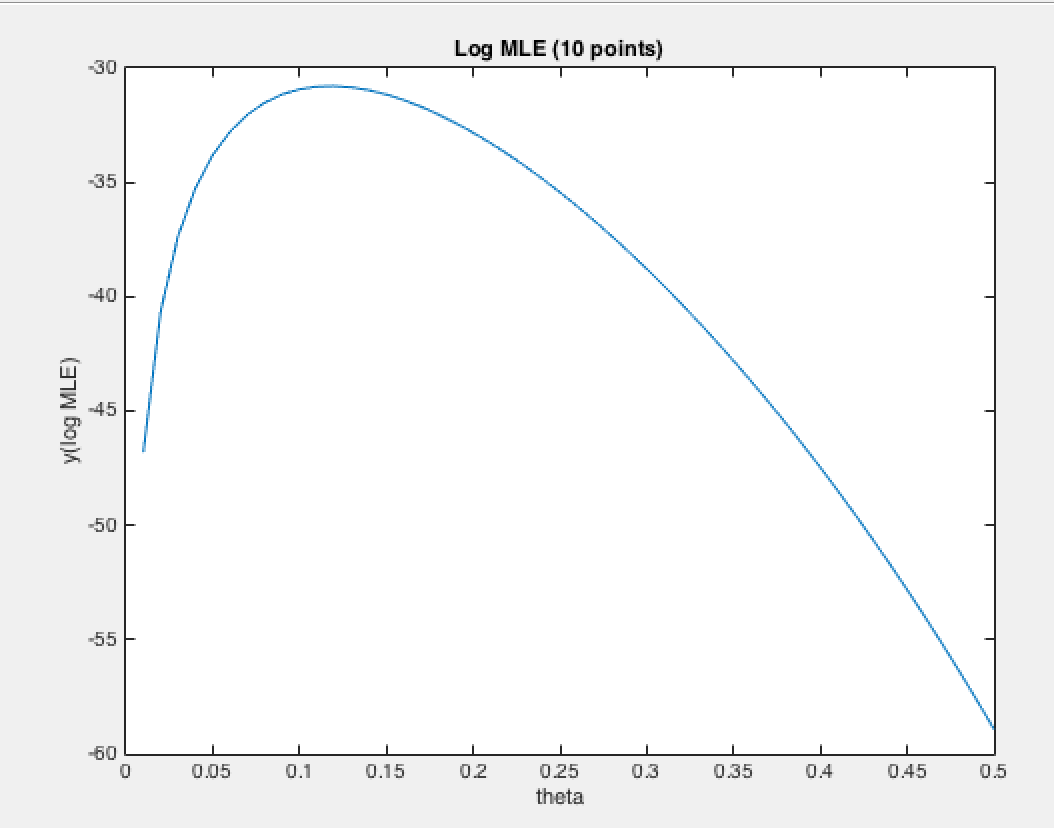
end

**Graphs : ( 2b)**

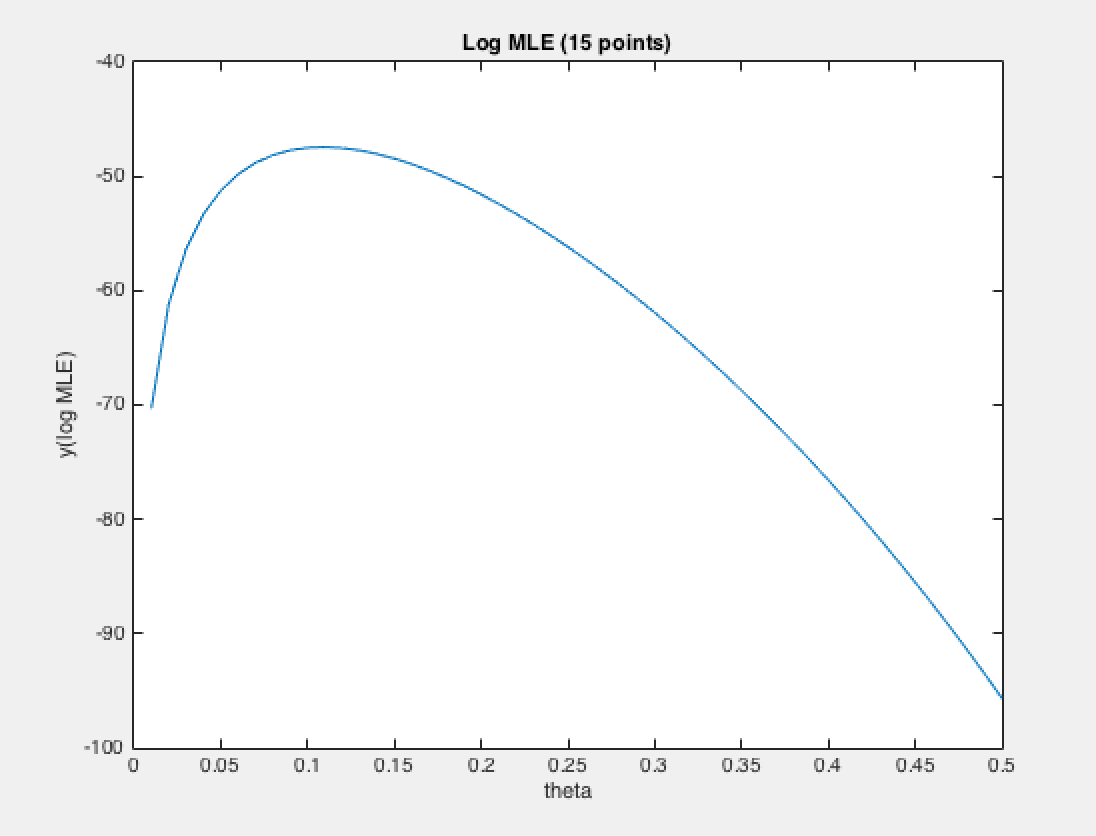
**5 Points**



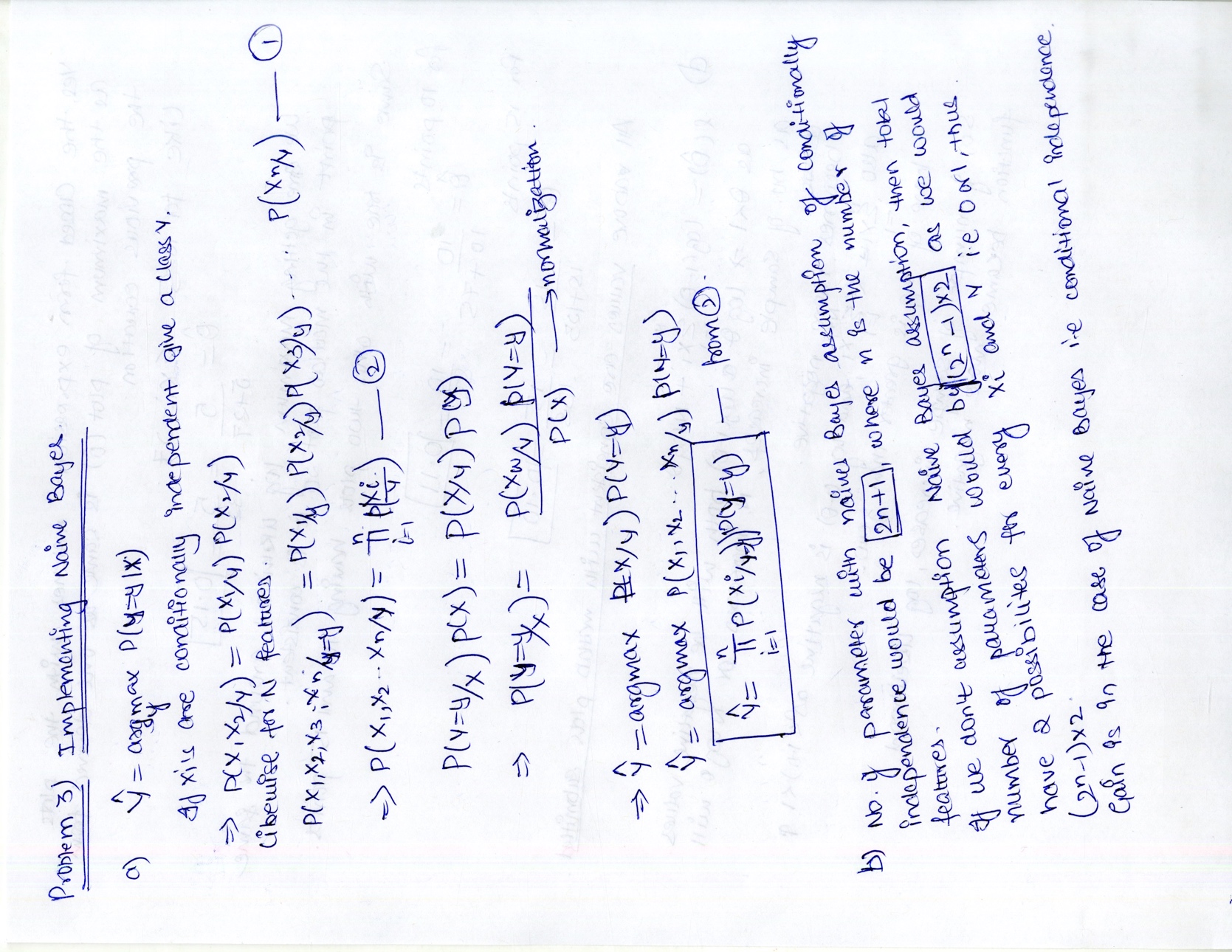
**10 points**

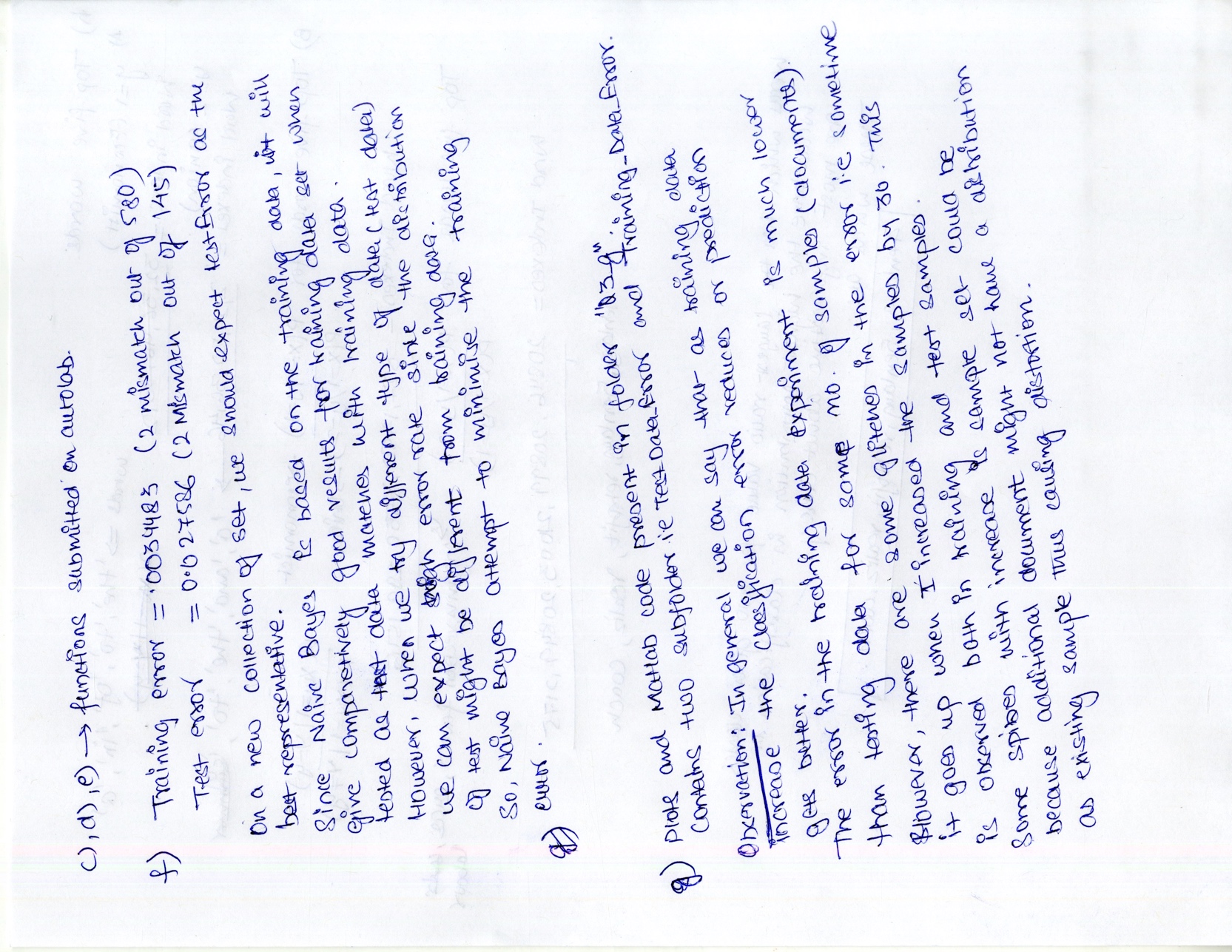


**15 points**

****





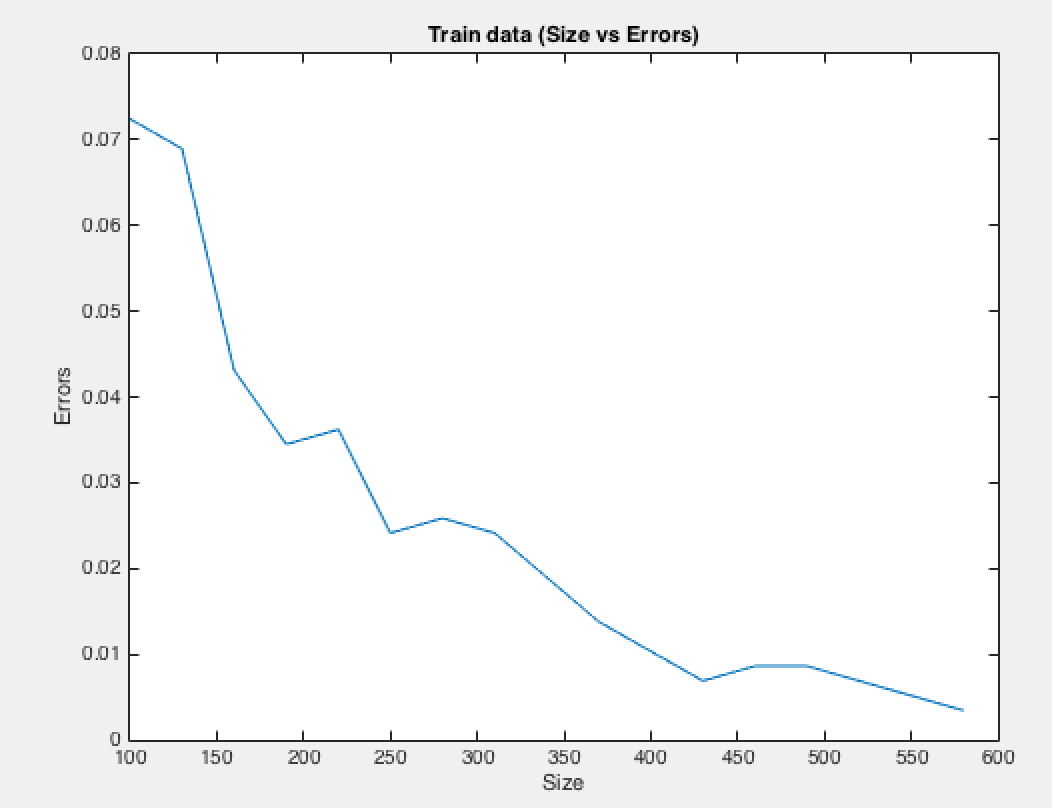


**3 g)**

**Test data errors :**



**Training Data Error:**



**Code for 3(g) :**

function [] = NB\_Size\_Error(XTest,yTest,XTrain,yTrain)

load('HW2Data.mat');

error = zeros([1 length(100:30:580)]);

count =1 ;

for m= 100:30:580

D = NB\_XGivenY\_Size(XTrain,yTrain,m);

p = sum(yTrain(1:m)==1)/length(yTrain);

fprintf('p = %d \n',p);

[yHat] = NB\_Classify(D,p,XTest);

error(count)=sum((yTest~=yHat))/length(yTest);

fprintf('size = %d, error = %d \n',m,error(count));

count = count+1;

end

data\_size = 100:30:580;

figure

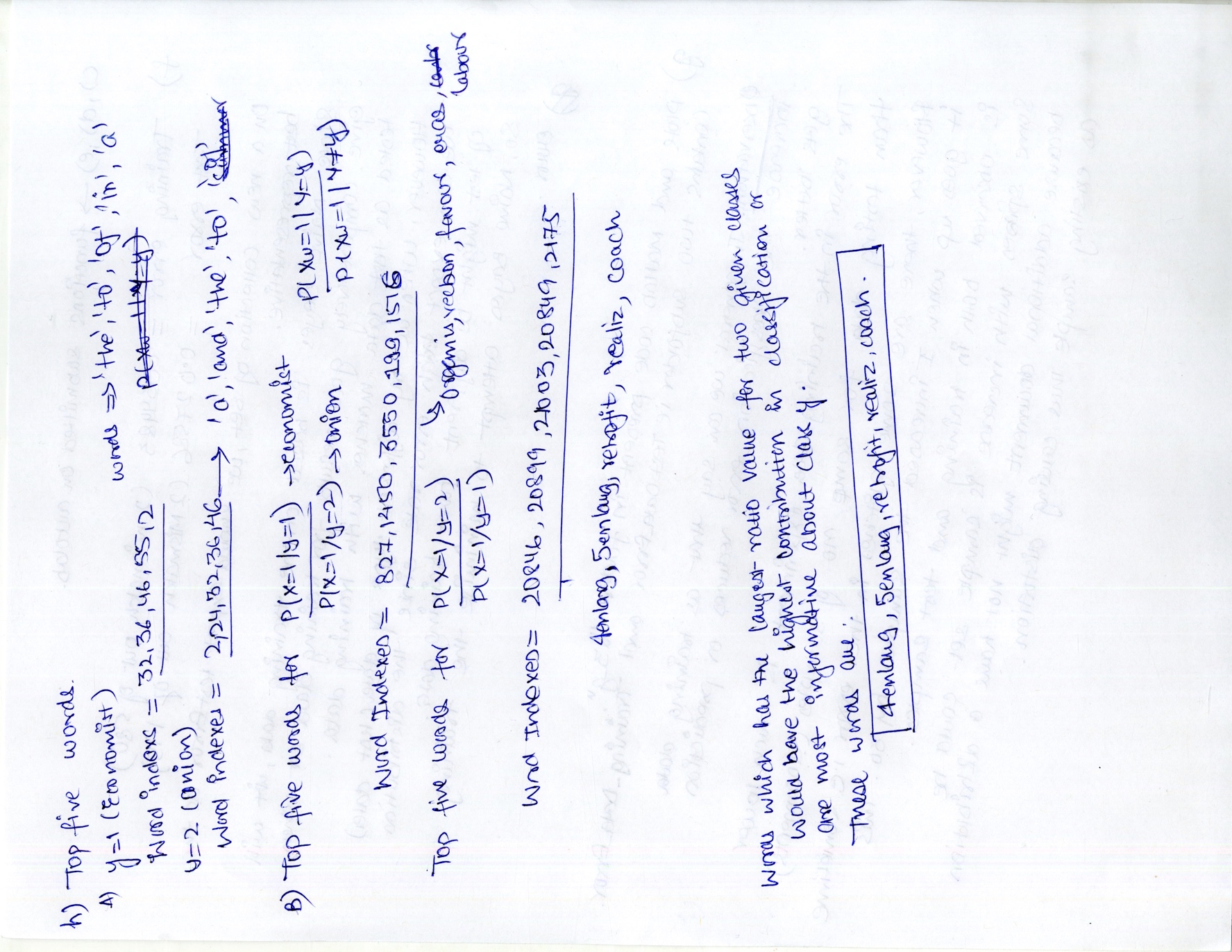
plot(data\_size,error)

xlabel('Size');

ylabel('Errors');

title('Train data (Size vs Errors)');

end



**Code for 3(h):**

function [] = NB\_FreqWords(XTrain,yTrain)

load('HW2Data.mat');

[n, v] = size(XTrain);

D = zeros([2 v]);

word\_count = zeros([2 v]);

[rows,col,value]= find(XTrain);

num\_eco\_docs = sum(yTrain==1);

num\_oni\_docs = sum(yTrain==2);

length\_spare\_matrix = length(rows);

for i = 1:length\_spare\_matrix

if yTrain(rows(i),1)==1

word\_count(1,col(i)) = word\_count(1,col(i))+1;

else

word\_count(2,col(i)) = word\_count(2,col(i))+1;

end

end

D(1,:) = (word\_count(1,:) + .001)/(num\_eco\_docs + .901);

D(2,:) = (word\_count(2,:) + .001)/(num\_oni\_docs + .901);

B = D(1,:);

for i=1:5

[M,I] = max(B);

fprintf(' Economist : val = %d, idx = %d \n',M,I);

B(1,I)=0;

end

B = D(2,:);

for i=1:5

[M,I] = max(B);

fprintf(' Onion : val = %d, idx = %d \n',M,I);

B(1,I)=0;

end

B = D(1,:) ./ D(2,:);

for i=1:5

[M,I] = max(B);

fprintf(' Eco/Onion : val = %d, idx = %d \n',M,I);

B(1,I)=0;

end

B = D(2,:) ./ D(1,:);

for i=1:5

[M,I] = max(B);

fprintf(' Onion/Eco : val = %d, idx = %d \n',M,I);

B(1,I)=0;

end

end