**HOMEWORK #2**

**Submitted by:**

Sreekhar Ale, M.Eng CS,   
Course: Intelligent Data Analysis – fall 2015  
Language used: MATLAB

**ANSWER 1:**

*List of steps*

1. Using xlsread function, read the data into a variable
2. Randomize the data by using randomperm function only in the rows and then copying the randomized data in to the NewData variable
3. Split the first 13020 rows into Training data
4. The next 3000 rows into Validation data
5. The last 3000 rows into Test data

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

**RESULTS:**

NewData = 19020 \* 11 double values  
TrainingData = 13020 \* 11 double values  
ValidationData = 3000 \* 11 double values  
TestData = 3000 \* 11 double values

**SCREEN SHOT:**

Not Applicable

**ANSWER 2:**

*List of steps*

1. Using xlsread function, read the data into a variable
2. Randomize the data by using randomperm function only in the rows and then copying the randomized data in to the NewData variable
3. Split the first 13020 rows into Training data
4. The next 3000 rows into Validation data
5. The last 3000 rows into Test data
6. Split the first ten columns of training data into Features variable
7. Copy the last column of training data into ClassLabels variable

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

**RESULTS:**

NewData = 19020 \* 11 double values  
TrainingData = 13020 \* 11 double values  
ValidationData = 3000 \* 11 double values  
TestData = 3000 \* 11 double values  
Features = 16020 \* 10 double values  
ClassLabels = 16020 \* 1 double values

**SCREEN SHOT:**

Not Applicable

**ANSWER 3:**

*List of steps*

1. Using xlsread function, read the data into a variable
2. Randomize the data by using randomperm function only in the rows and then copying the randomized data in to the NewData variable
3. Split the first 13020 rows into Training data
4. The next 3000 rows into Validation data
5. The last 3000 rows into Test data
6. Split the first ten columns of training data into Features variable
7. Copy the last column of training data into ClassLabels variable
8. Generate a decision tree using the fitctree function as given in the question and set value N = 1200

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

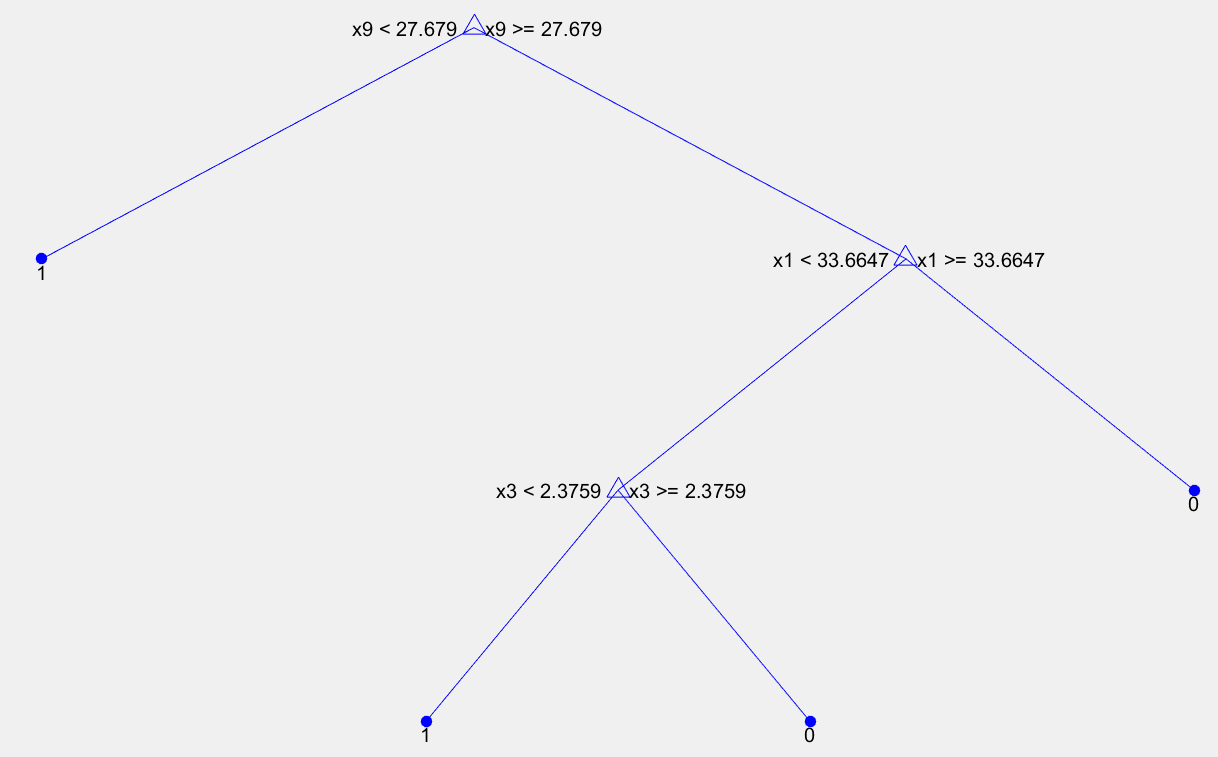
dtr\_1200=fitctree(Features, ClassLabels, 'MinLeafSize', 1200);

view(dtr\_1200,'Mode','graph');

**RESULTS:**

NewData = 19020 \* 11 double values  
TrainingData = 13020 \* 11 double values  
ValidationData = 3000 \* 11 double values  
TestData = 3000 \* 11 double values  
Features = 16020 \* 10 double values  
ClassLabels = 16020 \* 1 double values

**SCREEN SHOT:**



**ANSWER 4:**

*List of steps*

1. To find the predicted Class Labels we need to implement the steps mentioned in Answer 3
2. Copy the first 10 columns data of validation data into a new variable TestFeatures\_1200 for 1200 records in each leaf node
3. For predicting values use predict function with TestFeatures\_1200 and the decision tree
4. Now copy the original values of class labels of validation data into a variable namely OriginalValues\_1200
5. Calculate Accuracy, precision and recall

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

dtr\_1200=fitctree(Features, ClassLabels, 'MinLeafSize', 1200);

view(dtr\_1200,'Mode','graph');

TestFeatures\_1200 = ValidationData(:,1:10);

PredictLabels\_1200 = predict(dtr\_1200,TestFeatures);

OriginalLabels\_1200 = ValidationData(:,11);

TPCount\_1200=0;

FPCount\_1200=0;

FNCount\_1200=0;

TNCount\_1200=0;

for i=1:3000

if(PredictLabels\_1200(i) == 1 && OriginalLabels\_1200(i) == 1)

TPCount\_1200 = TPCount\_1200+1;

elseif(PredictLabels\_1200(i) == 1 && OriginalLabels\_1200(i) == 0)

FPCount\_1200=FPCount\_1200+1;

elseif(PredictLabels\_1200(i) == 0 && OriginalLabels\_1200(i) == 1)

FNCount\_1200=FNCount\_1200+1;

elseif(PredictLabels\_1200(i) == 0 && OriginalLabels\_1200(i) == 0)

TNCount\_1200=TNCount\_1200+1;

end

end

Accuracy\_1200 = (TPCount\_1200+TNCount\_1200)/(TPCount\_1200+TNCount\_1200+FNCount\_1200+FPCount\_1200);

Recall\_1200 = TPCount\_1200/(TPCount\_1200+FNCount\_1200);

Precision\_1200 = TPCount\_1200/(TPCount\_1200+FPCount\_1200);

**RESULTS:**

Accuracy\_1200 = 0.782333333333333  
Precision\_1200 = 0.796506137865911  
Recall\_1200 = 0.883708748035621  
FNCount\_1200 = 222  
FPCount\_1200 = 431  
TNCount\_1200 = 660   
TPCount\_1200 = 1687

**SCREEN SHOT:**

Not applicable

**ANSWER 5:**

**5 (a) bit answer**

*List of steps*

1. To find the graphical view of the decision tree, implement the steps mentioned in the answer 3 above by replacing the value of N with 1000

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

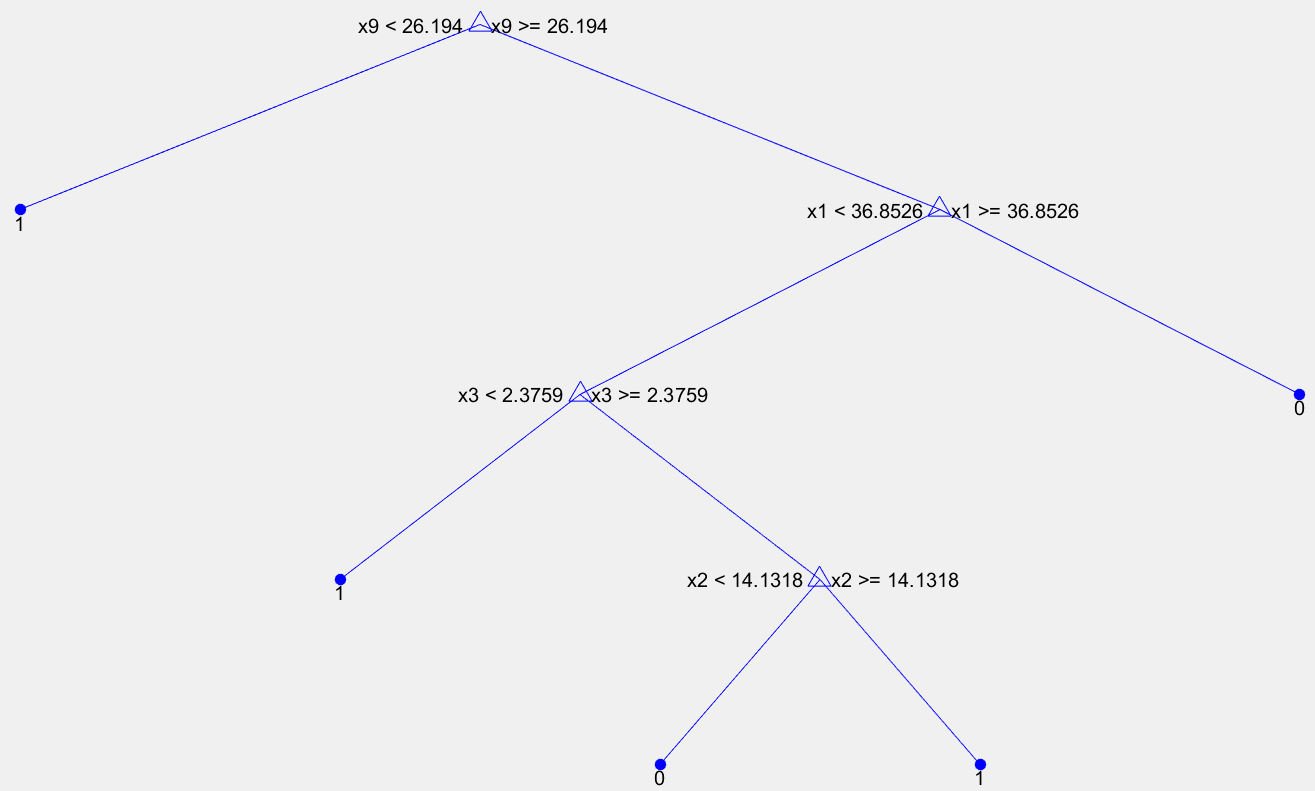
dtr\_1000=fitctree(Features, ClassLabels, 'MinLeafSize', 1000);

view(dtr\_1000,'Mode','graph');

**RESULTS:**

NewData = 19020 \* 11 double values  
TrainingData = 13020 \* 11 double values  
ValidationData = 3000 \* 11 double values  
TestData = 3000 \* 11 double values  
Features = 16020 \* 10 double values  
ClassLabels = 16020 \* 1 double values

**SCREEN SHOT:**



**5 (b) bit answer**

*List of steps*

1. To find the predicted labels and determine accuracy, precision and recall values and also the numbers of TP, TN, FN and FP we need to implement the steps mentioned above in Answer 4
2. But with 1000 as N value as given in the question.
3. Also we need to determine accuracy, precision and recall by using TrainingData obtained

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

dtr\_1000=fitctree(Features, ClassLabels, 'MinLeafSize', 1000);

view(dtr\_1000,'Mode','graph');

TestFeatures\_Test\_1000 = TrainingData(:,1:10);

PredictLabels\_Test\_1000 = predict(dtr\_1000,TestFeatures\_Test\_1000);

OriginalLabels\_Test\_1000 = TrainingData(:,11);

TPCount\_Test\_1000=0;

FPCount\_Test\_1000=0;

FNCount\_Test\_1000=0;

TNCount\_Test\_1000=0;

for i=1:3000

if(PredictLabels\_Test\_1000(i) == 1 && OriginalLabels\_Test\_1000(i) == 1)

TPCount\_Test\_1000 = TPCount\_Test\_1000+1;

elseif(PredictLabels\_Test\_1000(i) == 1 && OriginalLabels\_Test\_1000(i) == 0)

FPCount\_Test\_1000=FPCount\_Test\_1000+1;

elseif(PredictLabels\_Test\_1000(i) == 0 && OriginalLabels\_Test\_1000(i) == 1)

FNCount\_Test\_1000=FNCount\_Test\_1000+1;

elseif(PredictLabels\_Test\_1000(i) == 0 && OriginalLabels\_Test\_1000(i) == 0)

TNCount\_Test\_1000=TNCount\_Test\_1000+1;

end

end

Accuracy\_Test\_1000 = (TPCount\_Test\_1000+TNCount\_Test\_1000)/(TPCount\_Test\_1000+TNCount\_Test\_1000+FNCount\_Test\_1000+FPCount\_Test\_1000);

Recall\_Test\_1000 = TPCount\_Test\_1000/(TPCount\_Test\_1000+FNCount\_Test\_1000);

Precision\_Test\_1000 = TPCount\_Test\_1000/(TPCount\_Test\_1000+FPCount\_Test\_1000); **RESULTS:**

Accuracy\_Test \_1000 = 0.802666666666667  
Precision\_Test \_1000 = 0.797718297498903  
Recall\_Test \_1000 = 0.932786044125192  
FNCount\_Test \_1000 = 131  
FPCount\_Test \_1000 = 461  
TNCount\_Test \_1000 = 590  
TPCount\_Test \_1000 = 1818

**SCREEN SHOT:**

Not applicable

**5 (c) bit answer**

*List of steps*

1. To find the predicted labels and determine accuracy, precision and recall values and also the numbers of TP, TN, FN and FP we need to implement the steps mentioned above in Answer 4
2. But with 1000 as N value as given in the question.
3. We need to determine accuracy, precision and recall by using ValidationData and TestData obtained together

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

dtr\_1000=fitctree(Features, ClassLabels, 'MinLeafSize', 1000);

view(dtr\_1000,'Mode','graph');

TestFeatures\_Val\_1000 = ValidationData(:,1:10);

PredictLabels\_Val\_1000 = predict(dtr\_1000,TestFeatures\_Val\_1000);

OriginalLabels\_Val\_1000 = ValidationData(:,11);

TPCount\_Val\_1000=0;

FPCount\_Val\_1000=0;

FNCount\_Val\_1000=0;

TNCount\_Val\_1000=0;

for i=1:3000

if(PredictLabels\_Val\_1000(i) == 1 && OriginalLabels\_Val\_1000(i) == 1)

TPCount\_Val\_1000 = TPCount\_Val\_1000+1;

elseif(PredictLabels\_Val\_1000(i) == 1 && OriginalLabels\_Val\_1000(i) == 0)

FPCount\_Val\_1000=FPCount\_Val\_1000+1;

elseif(PredictLabels\_Val\_1000(i) == 0 && OriginalLabels\_Val\_1000(i) == 1)

FNCount\_Val\_1000=FNCount\_Val\_1000+1;

elseif(PredictLabels\_Val\_1000(i) == 0 && OriginalLabels\_Val\_1000(i) == 0)

TNCount\_Val\_1000=TNCount\_Val\_1000+1;

end

end

Accuracy\_Val\_1000 = (TPCount\_Val\_1000+TNCount\_Val\_1000)/(TPCount\_Val\_1000+TNCount\_Val\_1000+FNCount\_Val\_1000+FPCount\_Val\_1000);

Recall\_Val\_1000 = TPCount\_Val\_1000/(TPCount\_Val\_1000+FNCount\_Val\_1000);

Precision\_Val\_1000 = TPCount\_Val\_1000/(TPCount\_Val\_1000+FPCount\_Val\_1000);

**RESULTS:**

Accuracy\_Val \_1000 = 0.807000000000000  
Precision\_Val \_1000 = 0.800873362445415  
Recall\_Val \_1000 = 0.937148696985181  
FNCount\_Val \_1000 = 123  
FPCount\_Val \_1000 = 456  
TNCount\_Val \_1000 = 587  
TPCount\_Val \_1000 = 1834

**SCREEN SHOT:**

Not applicable

**ANSWER 6:**

**6 (a) bit answer**

*List of steps*

1. To find the graphical view of the decision tree, implement the steps mentioned in the answer 3 above by replacing the value of N with 20

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

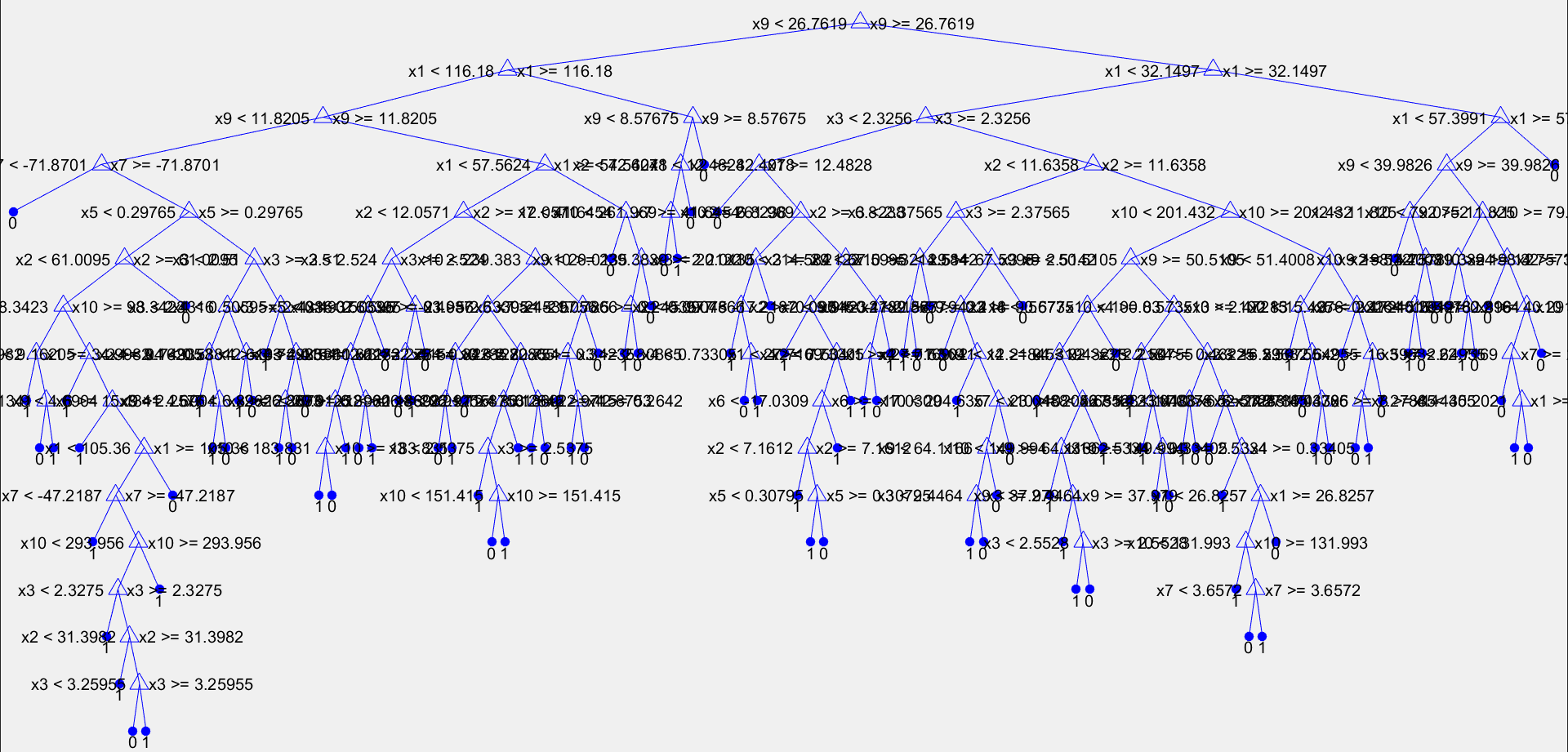
dtr\_20=fitctree(Features, ClassLabels, 'MinLeafSize', 20);

view(dtr\_20,'Mode','graph');

**RESULTS:**

NewData = 19020 \* 11 double values  
TrainingData = 13020 \* 11 double values  
ValidationData = 3000 \* 11 double values  
TestData = 3000 \* 11 double values  
Features = 16020 \* 10 double values  
ClassLabels = 16020 \* 1 double values

**SCREEN SHOT:**



*Please find attachment for the matlab image generated*

**6 (b) bit answer**

*List of steps*

1. To find the predicted labels and determine accuracy, precision and recall values and also the numbers of TP, TN, FN and FP we need to implement the steps mentioned above in Answer 4
2. But with 20 as N value as given in the question.
3. Also we need to determine accuracy, precision and recall by using TrainingData obtained

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

dtr\_20=fitctree(Features, ClassLabels, 'MinLeafSize', 1000);

view(dtr\_20,'Mode','graph');

TrainFeatures\_ Train \_20 = TrainingData(:,1:10);

PredictLabels\_ Train \_20 = predict(dtr\_20, TrainFeatures \_ Train \_20);

OriginalLabels\_ Train \_20 = TrainingData(:,11);

TPCount\_ Train \_20=0;

FPCount\_ Train \_20=0;

FNCount\_ Train \_20=0;

TNCount\_ Train \_20=0;

for i=1:3000

if(PredictLabels\_ Train \_20(i) == 1 && OriginalLabels\_ Train \_20(i) == 1)

TPCount\_ Train \_20 = TPCount\_ Train \_20+1;

elseif(PredictLabels\_ Train \_20(i) == 1 && OriginalLabels\_ Train \_20(i) == 0)

FPCount\_ Train \_20=FPCount\_ Train \_20+1;

elseif(PredictLabels\_ Train \_20(i) == 0 && OriginalLabels\_ Train \_20(i) == 1)

FNCount\_ Train \_20=FNCount\_ Train \_20+1;

elseif(PredictLabels\_ Train \_20(i) == 0 && OriginalLabels\_ Train \_20(i) == 0)

TNCount\_ Train \_20=TNCount\_ Train \_20+1;

end

end

Accuracy\_ Train \_20 = (TPCount\_Train\_20+TNCount\_Test\_20)/(TPCount\_ Train\_\_20+TNCount\_ Train \_20+FNCount\_ Train\_20+FPCount\_ Train \_20);

Recall\_ Train \_20 = TPCount\_ Train \_20/(TPCount\_ Train \_20+FNCount\_ Train\_20);

Precision\_ Train \_20 = TPCount\_ Train\_20/(TPCount\_ Train \_20+FPCount\_ Train\_20);

**RESULTS:**

Accuracy\_Train\_20 = 0.841666666666667  
Precision\_ Train \_20 = 0.871365204534253  
Recall\_ Train \_20 = 0.892028254288597  
FNCount\_ Train \_20 = 214  
FPCount\_ Train \_20 = 261  
TNCount\_ Train \_20 = 757  
TPCount\_ Train \_20 = 1768

**SCREEN SHOT:**

Not applicable

**6 (c) bit answer**

*List of steps*

1. To find the predicted labels and determine accuracy, precision and recall values and also the numbers of TP, TN, FN and FP we need to implement the steps mentioned above in Answer 4
2. But with 20 as N value as given in the question.
3. We need to determine accuracy, precision and recall by using ValidationData and TestData obtained together

**MATLAB SOURCE CODE:**

data = xlsread('D:\My work\M.Eng 1st sem\IDA\Assignment\Homework 2\magic04.xlsx');

NewData = data(randperm(19020),:);

TrainingData = NewData(1:13020,:);

ValidationData = NewData(13021:16020,:);

TestData = NewData(16021:end,:);

Features = TrainingData(:,1:10);

ClassLabels = TrainingData(:,11);

dtr\_20=fitctree(Features, ClassLabels, 'MinLeafSize', 1000);

view(dtr\_20,'Mode','graph');

TestFeatures\_Val\_20 = ValidationData(:,1:10);

PredictLabels\_Val\_20 = predict(dtr\_20,TestFeatures\_Val\_20);

OriginalLabels\_Val\_20 = ValidationData(:,11);

TPCount\_Val\_20=0;

FPCount\_Val\_20=0;

FNCount\_Val\_20=0;

TNCount\_Val\_20=0;

for i=1:3000

if(PredictLabels\_Val\_20(i) == 1 && OriginalLabels\_Val\_20(i) == 1)

TPCount\_Val\_20 = TPCount\_Val\_20+1;

elseif(PredictLabels\_Val\_20(i) == 1 && OriginalLabels\_Val\_20(i) == 0)

FPCount\_Val\_20=FPCount\_Val\_20+1;

elseif(PredictLabels\_Val\_20(i) == 0 && OriginalLabels\_Val\_20(i) == 1)

FNCount\_Val\_20=FNCount\_Val\_20+1;

elseif(PredictLabels\_Val\_20(i) == 0 && OriginalLabels\_Val\_20(i) == 0)

TNCount\_Val\_20=TNCount\_Val\_20+1;

end

end

Accuracy\_Val\_20 = (TPCount\_Val\_20+TNCount\_Val\_20)/(TPCount\_Val\_20+TNCount\_Val\_20+FNCount\_Val\_20+FPCount\_Val\_20);

Recall\_Val\_20 = TPCount\_Val\_20/(TPCount\_Val\_20+FNCount\_Val\_20);

Precision\_Val\_20 = TPCount\_Val\_20/(TPCount\_Val\_20+FPCount\_Val\_20);

**RESULTS:**

Accuracy\_Val \_20 = 0.831333333333333  
Precision\_Val \_20 = 0.855182926829268  
Recall\_Val \_20 = 0.883928571428571  
FNCount\_Val \_20 = 221  
FPCount\_Val \_20 = 285  
TNCount\_Val \_20 = 811  
TPCount\_Val \_20 = 1683

**SCREEN SHOT:**

Not applicable

**ANSWER 7:**

*List of steps*

1. Firstly we need to implement the steps mentioned in Answer 4 above and repeat for all the values of N as mentioned in the question.
2. Then we should find accuracy of the training data for the set of N as mentioned and store it in an array of accuracy data
3. Later we need to find accuracy of the validation data for the set of N as mentioned and store it in an array of Validation data
4. Simultaneously we should find the number of nodes of the decision tree for the various values of N as mentioned in the Question and store it in an array of Nodes array
5. Using plot function we need to plot all the graphs

**MATLAB SOURCE CODE:**

Below is the part of the code we use to plot the graphs, all the remaining steps are the same of previous problems.

Total\_accuracy\_train = [accuracy\_train\_5,accuracy\_train\_10,accuracy\_train\_20,accuracy\_train\_50,accuracy\_train\_100,accuracy\_train\_125,accuracy\_train\_250,accuracy\_train\_500,accuracy\_train\_750,accuracy\_1000];

Total\_accuracy\_validation = [accuracy\_vald\_5,accuracy\_vald\_10,accuracy\_vald\_20,accuracy\_vald\_50,accuracy\_vald\_100,accuracy\_vald\_125,accuracy\_vald\_250,accuracy\_vald\_500,accuracy\_vald\_750,accuracy\_vd\_1000];

Total\_noof\_nodes = [num\_nodes\_train\_5,num\_nodes\_train\_10,num\_nodes\_train\_20,num\_nodes\_train\_50,num\_nodes\_train\_100,num\_nodes\_train\_125,num\_nodes\_train\_250,num\_nodes\_train\_500,num\_nodes\_train\_750,num\_nodes\_training\_data\_dtr\_1000];

figure()

hold all

N = [1000,750,500,250,125,100,50,20,10,5];

plot(Total\_accuracy\_train,Total\_accuracy\_validation);

figure()

hold all

plot(Total\_noof\_nodes,Total\_accuracy\_train);

figure()

hold all

plot(Total\_noof\_nodes,Total\_accuracy\_validation);

figure()

hold all

plot(Total\_noof\_nodes,N);

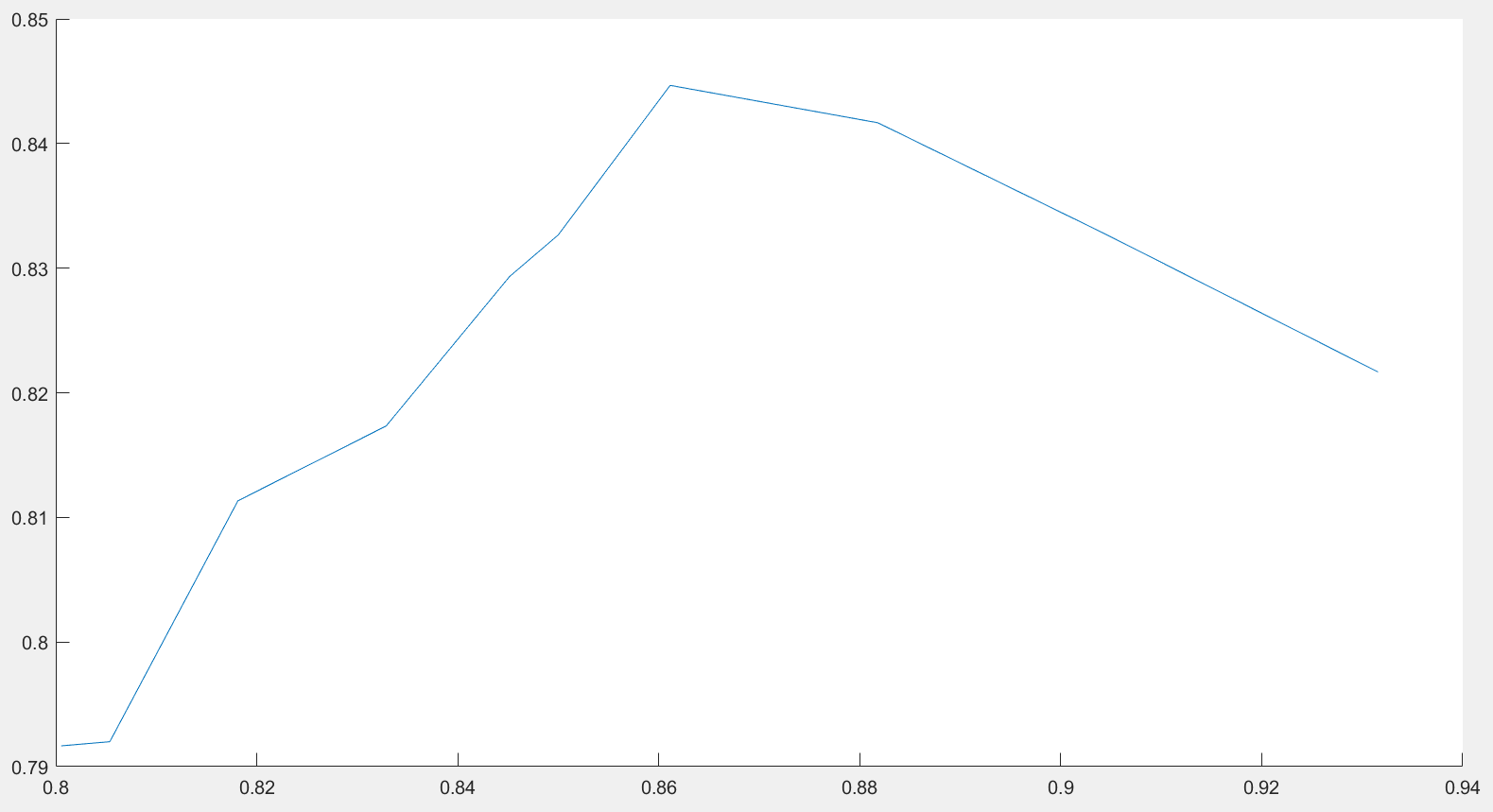
**RESULTS:**

Total\_accuracy\_train = [0.931566820276498,0.904608294930876,0.881797235023042,0.861136712749616,0.850000000000000,0.845161290322581,0.832872503840246,0.818125960061444,0.805376344086022,0.800537634408602]

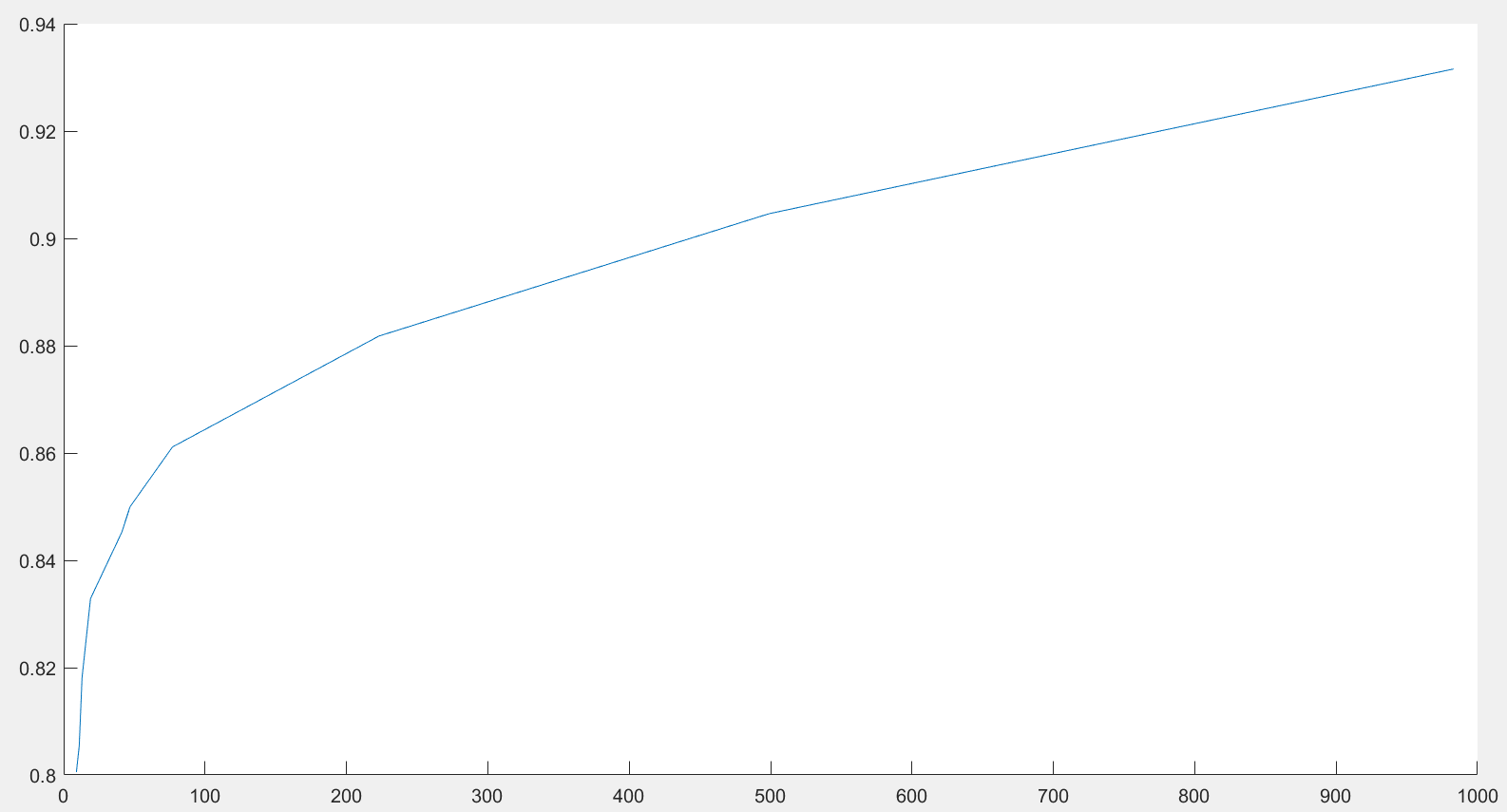
Total\_accuracy\_validation = [0.821666666666667,0.832666666666667,0.841666666666667,0.844666666666667,0.832666666666667,0.829333333333333,0.817333333333333,0.811333333333333,0.792000000000000,0.791666666666667]

Total\_noof\_nodes= [983,499,223,77,47,41,19,13,11,9]

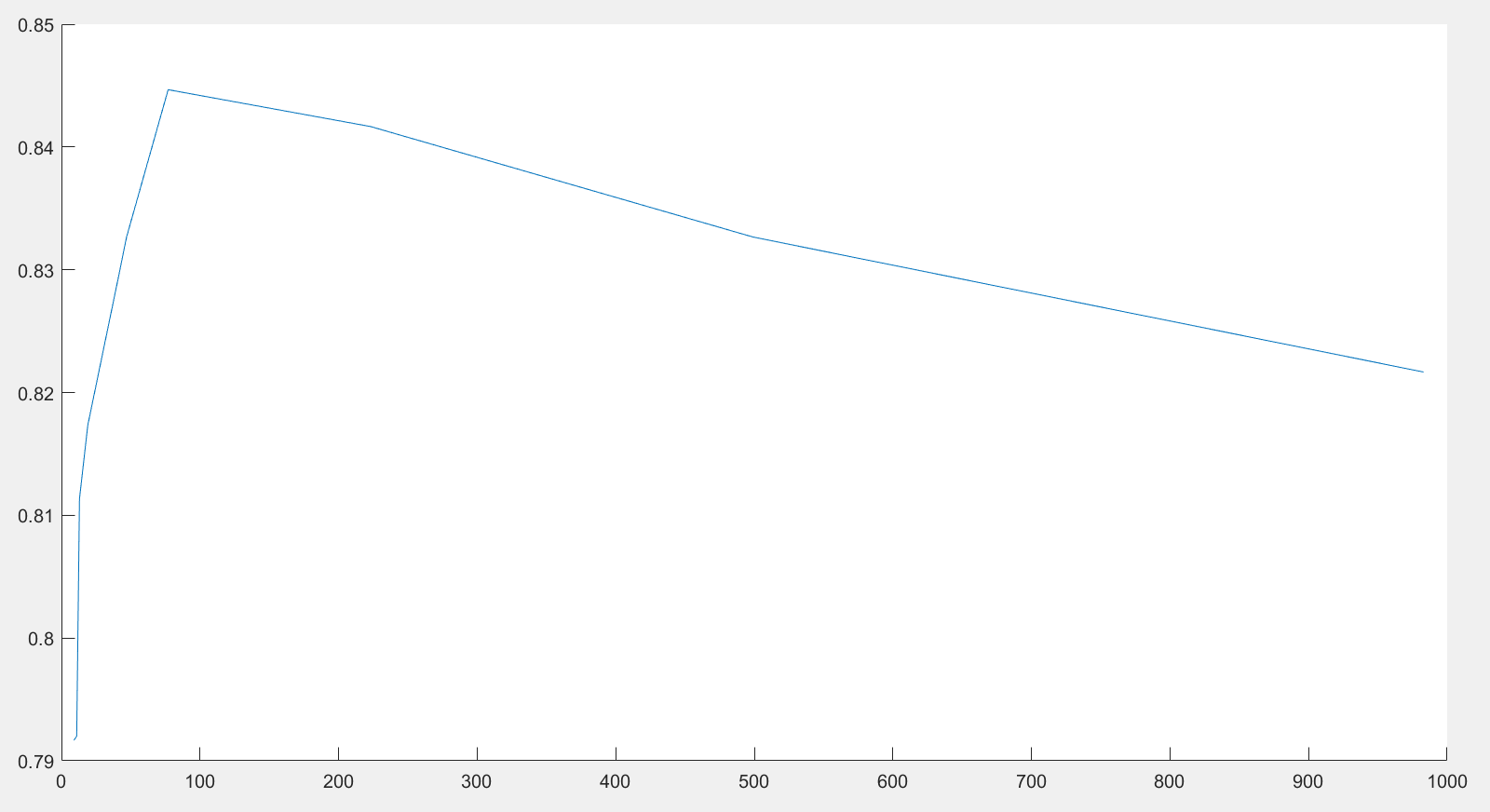
**SCREEN SHOT:**



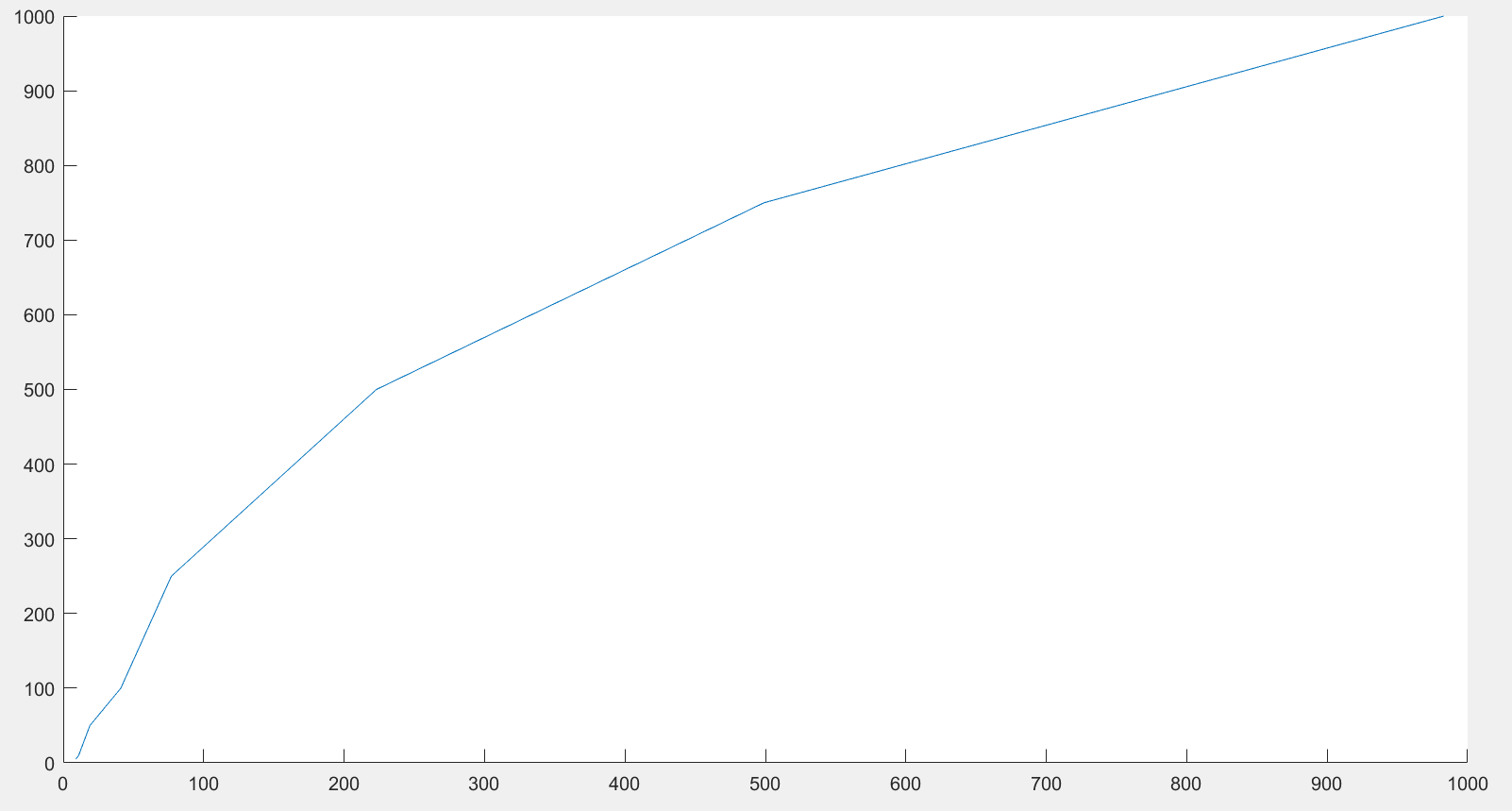
Plot for accuracy with training data v/s accuracy with validation data



Plot for number of nodes v/s accuracy with training data



Plot for number of nodes v/s accuracy with validation data



Plot for number of nodes v/s N number of records

**ANSWER 8:**

*List of steps*

1. From the results obtained in Answer 7, we plot the graph.
2. From the graph we get find the number of nodes for which the accuracy of data is high
3. Later implement the steps mentioned in answer 4 for N=3000 records and find accuracy, precision and recall

**MATLAB SOURCE CODE:**

Below is the part of the code we use to plot the graphs, all the remaining code is same as the above questions.

Total\_accuracy\_train = [accuracy\_train\_5,accuracy\_train\_10,accuracy\_train\_20,accuracy\_train\_50,accuracy\_train\_100,accuracy\_train\_125,accuracy\_train\_250,accuracy\_train\_500,accuracy\_train\_750,accuracy\_1000];

Total\_accuracy\_validation = [accuracy\_vald\_5,accuracy\_vald\_10,accuracy\_vald\_20,accuracy\_vald\_50,accuracy\_vald\_100,accuracy\_vald\_125,accuracy\_vald\_250,accuracy\_vald\_500,accuracy\_vald\_750,accuracy\_vd\_1000];

Total\_noof\_nodes = [num\_nodes\_train\_5,num\_nodes\_train\_10,num\_nodes\_train\_20,num\_nodes\_train\_50,num\_nodes\_train\_100,num\_nodes\_train\_125,num\_nodes\_train\_250,num\_nodes\_train\_500,num\_nodes\_train\_750,num\_nodes\_training\_data\_dtr\_1000];

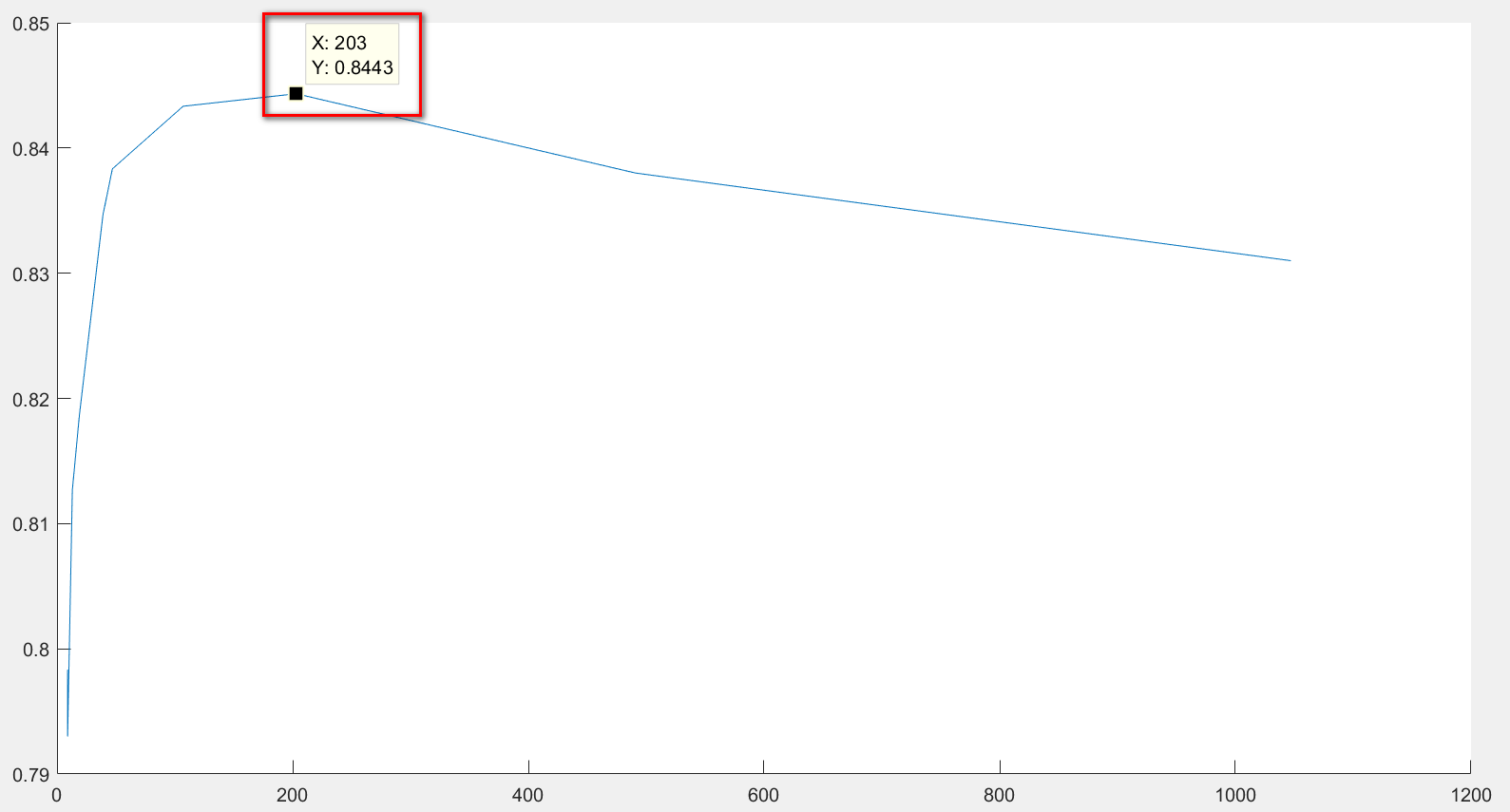
figure()

hold all

plot(Total\_noof\_nodes,Total\_accuracy\_validation);

*//Remaining code is same as code in Answer for just replacing N=3000*

**SCREEN SHOT:**



**RESULTS:**

From the graph obtained we get to know that for number of **nodes = 203** we get the decision tree as the best model for the training data.

Accuracy\_test \_3000 = 0.859666666666667  
Precision\_test\_3000 = 0.876561719140430  
Recall\_test \_3000 = 0.909751037344398