COMS 7350 DataMining
Professor Cen Li
Project 1: Decision Tree & Naive Bayes

Xin Yang

February 19, 2014

1 Experiment One:

1.1 Data 1: Acute Inflammations

The main idea of this data set is to prepare the algorithm of the expert system, which will perform the presumptive diagnosis of diseases of urinary system. This data includes both continuous and discrete data.

```
---------------------------------------------------------------------------
Attribute Information:

Number of Instances: 120
Number of Attributes: 6
Missing Values? No

a1 Temperature of patient { 35C-42C }
a2 Occurrence of nausea { yes, no }
a3 Lumbar pain { yes, no }
a4 Urine pushing (continuous need for urination) { yes, no }
a5 Micturition pains { yes, no }
a6 Burning of urethra, itch, swelling of urethra outlet { yes,no }
d1 decision: Inflammation of urinary bladder { yes, no }

For example, '35.9 no no yes yes yes yes no'
---------------------------------------------------------------------------
```

1.2 Data 2: Breast Cancer

This data sets are used to study the early stage of the breast cancer, which includes only discrete value. There are 16 values are missing. These missing values do affect the algorithm, in order to get the algorithm works, I have

set all missing values into 0. Because 0 is not among the attribute range 1-10, so it will not contribute anything to the results.

```
-------------------------------------------------------------------------
Attribute Information:

Number of Instances: 699
Number of Attributes: 19
Missing Values? Yes(16)

 1. Sample code number            id number
 2. Clump Thickness               1 - 10
 3. Uniformity of Cell Size       1 - 10
 4. Uniformity of Cell Shape      1 - 10
 5. Marginal Adhesion             1 - 10
 6. Single Epithelial Cell Size   1 - 10
 7. Bare Nuclei                   1 - 10
 8. Bland Chromatin               1 - 10
 9. Normal Nucleoli               1 - 10
 10. Mitoses                      1 - 10
 11. Class:                       (2 for benign, 4 for malignant)
-------------------------------------------------------------------------
```

1.3 Data Division

Randomly withdraw 10%, 20%, 30% , 40%, 50%, 60%, 70%, 80%, 90%, 100%. This is done in C++, the source file name is **DividData.cpp**.

Before division, we need to preprocess the data. All preprocess of the data is done by regular expression in Notepad.

- The breast cancer data is separated by tab , which should be changed to space.
- Remove the first column which is useless.
- Add value for the missing values.

Directory **DividData1** is used to divide the acute data.
Directory **DividData2** is used to divide the breast cancer data.

1.4 Plot the learning curve

In order to plot the ROC, we use 10 cross validation in Decision Tree classification. **csh xval.sh data 10** . The results for both data are shown following. The results.data file are seperately in directory **DeTreeData1** and **DeTreeData2**.

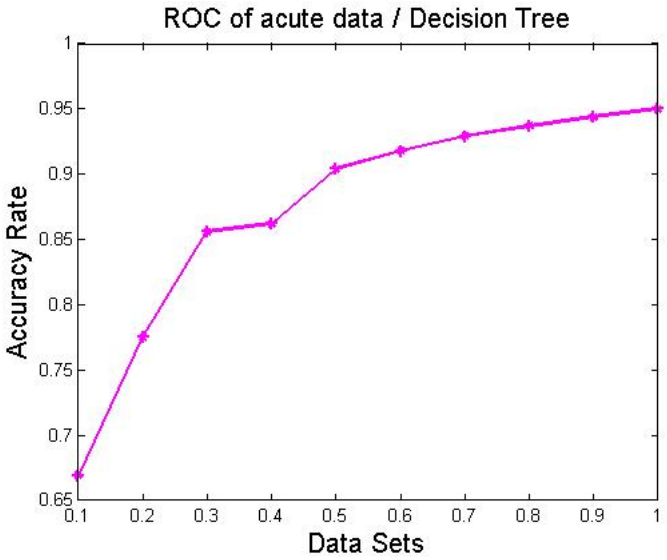

Figure 1: Acute Data

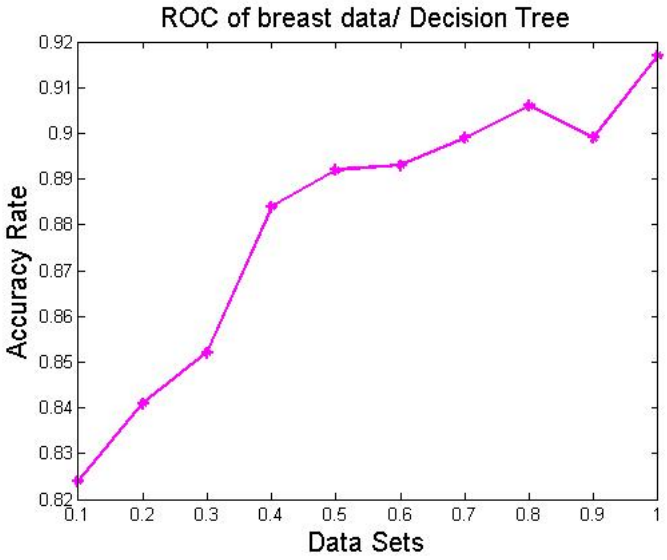

Figure 2: Breast Data

From the curve, we notice that when the training data is increasing , the accuracy rate is getting better and better. The area under the curve is bigger and bigger. This is agree with what we expected. The more information provide, we can learn better classifier.

2 Experiment Two

2.1 c4.5 -f breast

```
Simplified Decision Tree:

CellSize = 1: 2 (384.0/6.2)
CellSize = 2: 2 (45.0/10.5)
CellSize = 4: 4 (40.0/11.5)
CellSize = 5: 4 (30.0/1.4)
CellSize = 6: 4 (27.0/3.7)
CellSize = 7: 4 (19.0/2.5)
CellSize = 8: 4 (28.0/2.6)
CellSize = 9: 4 (6.0/2.3)
CellSize = 10: 4 (67.0/1.4)
CellSize = 3:
|   BareNuclei = 1: 2 (19.0/1.3)
|   BareNuclei = 2: 2 (5.0/2.3)
|   BareNuclei = 4: 4 (2.0/1.0)
|   BareNuclei = 5: 4 (4.0/2.2)
|   BareNuclei = 6: 2 (0.0)
|   BareNuclei = 7: 4 (1.0/0.8)
|   BareNuclei = 8: 4 (1.0/0.8)
|   BareNuclei = 9: 4 (1.0/0.8)
|   BareNuclei = 10: 4 (13.0/2.5)
|   BareNuclei = 3:
|   |   ClumpThickness = 1: 4 (0.0)
|   |   ClumpThickness = 2: 4 (0.0)
|   |   ClumpThickness = 3: 2 (2.0/1.0)
|   |   ClumpThickness = 4: 4 (0.0)
|   |   ClumpThickness = 5: 4 (3.0/1.1)
|   |   ClumpThickness = 6: 4 (0.0)
|   |   ClumpThickness = 7: 4 (1.0/0.8)
|   |   ClumpThickness = 8: 4 (0.0)
|   |   ClumpThickness = 9: 4 (0.0)
|   |   ClumpThickness = 10: 4 (0.0)

Tree saved
```

```
Evaluation on training data (698 items):

 Before Pruning           After Pruning
----------------   --------------------------
Size      Errors   Size      Errors   Estimate

 151    8( 1.1%)     31   29( 4.2%)    ( 8.1%)   <<
```

2.2 c4.5rules -f breast

```
------------------
Processing tree 0

Final rules from tree 0:

Rule 1:
     CellSize = 1
     BareNuclei = 1
->  class 2  [99.6%]

Rule 2:
     CellSize = 1
     Nucleoli = 1
->  class 2  [99.3%]

Rule 7:
     BareNuclei = 1
     Nucleoli = 1
->  class 2  [98.6%]

Rule 22:
     ClumpThickness = 5
     Chromatin = 2
->  class 2  [95.3%]

Rule 36:
     Nucleoli = 2
->  class 2  [77.1%]

Rule 29:
     ClumpThickness = 10
->  class 4  [98.0%]
```

```
Rule 38:
     CellSize = 10
->  class 4  [98.0%]

Rule 6:
     Nucleoli = 10
->  class 4  [97.8%]

Rule 13:
     BareNuclei = 10
->  class 4  [96.2%]

Rule 33:
     Adhesion = 10
->  class 4  [95.3%]

Rule 35:
     CellSize = 8
->  class 4  [90.9%]

Rule 28:
     ClumpThickness = 9
->  class 4  [90.6%]

Rule 16:
     ClumpThickness = 7
->  class 4  [89.0%]

Rule 19:
     BareNuclei = 9
->  class 4  [85.7%]

Rule 4:
     Nucleoli = 4
->  class 4  [85.2%]

Rule 12:
     BareNuclei = 8
->  class 4  [82.3%]

Rule 5:
     Nucleoli = 7
->  class 4  [77.1%]
```

```
Rule 15:
     ClumpThickness = 5
     BareNuclei = 3
->  class 4  [70.7%]

Rule 11:
     BareNuclei = 7
->  class 4  [70.0%]
```

2.3 consult -f breast

```
----------------------------------------
CellSize: 3
BareNuclei: 2

Decision:
2  CF = 0.80  [ 0.55 - 1.00 ]
----------------------------------------
CellSize: 1

Decision:
2  CF = 0.99  [ 0.98 - 1.00 ]
----------------------------------------
CellSize: 10

Decision:
4  CF = 1.00  [ 0.98 - 1.00 ]
----------------------------------------
CellSize: 3
BareNuclei: 5

Decision:
4  CF = 0.75  [ 0.45 - 1.00 ]
----------------------------------------
CellSize: 2

Decision:
2  CF = 0.82  [ 0.77 - 1.00 ]
```

2.4 Analysis

- From the results, the obtained rules are agree with what I expected. For example, when the clumpthickness, Nucleoli, BareNuclei these attribute are very small, the the patient is more likely to be benign. On the contrary, the bigger values of these attributes indicate high possibility of malignant.
- Interesting discovery of breast cancer data. I did not expect Adhesion will be an indicator for malignant. Because normally, the cell will expand to all over when the cancer goes to deteriorate. So I thought the Adhesion is very small when the patient is malignant.

```
    Rule 33:
     Adhesion = 10
->  class 4  [95.3%]
```

- For the testing, most of the classification is agree with the answer. However, there are still rare cases are not right. Because the predictions are based on decision rules, sometimes there are unexpected things beyond decision rule. Simple decision trees tend to over fit the training data.

3 Experiment Three: Decision tree VS Naive Bayes

3.1 Decision Tree

Decision trees are a class of predictive data mining tools which predict either a categorical or continuous response variable.A decision tree is comprised of nodes and splits to the data. Decision trees are easy to debug, because it will guide you to find if there is a statistical relationship between a given input and the output and how strong that relationship is. So Decision trees can be used a research tool to help you learn classifiers.

- Advantage
  1. Flexible, easy to understand, the predictions are based on decision rules, not involve mathematical calculations.
  2. They can handle both continuous value and discrete values.
  3. It can build a classifier directly from the data file without needing any design work.
- Disadvantage
  1. Simple decision trees tend to over fit the training data, you generally have to do tree pruning.

3.2 Naive Bayes

A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations.

- Advantage

    1. Bayes can perform quite well
    2. It doesn't over fit, so there is no need to prune or process the network.
    3. Naive bayes does quite well when the training data doesn't contain all possibilities so it can be very good with low amounts of data.

- Disadvantage

    1. You have to design the probabilities and build the classification by hand.
    2. They are harder to debug and understand because each node contributes to the combined probability, so you have to be careful to test.

3.3 Decision tree VS Naive Bayes

From the results, we can see that Naive Bayes has better accuracy than decision tree based on the same data sets. This is also agree with what we expected. Naive Bayes has better performance than decision tree, however the former will pay cost to the complex probability design.

Figure 3: Decision Tree

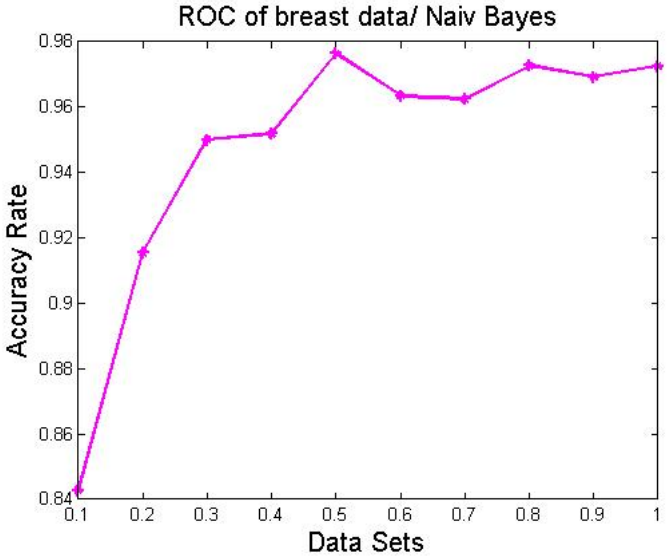

Figure 4: Naive Bayes