Department of Statistics
University of Wisconsin, Madison
Technical Report 989
March 3, 1998

CRUISE User Manual

Hyunjoong Kim
Department of Statistics
University of Wisconsin–Madison

Wei-Yin Loh
Department of Statistics
University of Wisconsin–Madison

\[\text{Research partially supported by U.S. Army Research Office grant DAAH04-94-G-0042, a grant from Pfizer, Inc. and a University of Wisconsin Vilas Associateship.}\]
1 Introduction

The CRUISE is a program for tree-structured classification. CRUISE stands for “Classification Rule with Unbiased Interaction Selection”. It consists of several algorithms for the construction of classification trees. Its main features are:

- Recursive linear discriminant analysis,
- Multiple splits for more than 2 classes,
- Unbiased variable selection via bootstrap calibration,
- Detection of pairwise interactions among predictor variables,
- Choice of split and tree selection criteria,
- Choice of tree pruning by cross-validation or by a test sample,
- Missing value treatment by global imputation or by nodewise imputation,
- Use of alternate split or proxy split for future data with missing values, and
- Box-Cox transformations for univariate splits.

The basic algorithm for CRUISE version 1.0 is described in Kim (1998). This user manual explains how the program is executed and how the output is interpreted.

2 Installation and execution

The CRUISE is distributed in a compressed file, called cruise.zip. Carry out the following steps to install the program on a PC.


2. Uncompress the file cruise.zip.

When uncompressed, cruise.zip produces the following files:
1. `cruise.ps`. This user manual in postscript format.

2. `cruise.exe`. The Win95/NT executable file. Executables are available on other platforms, including DEC Alpha (Digital Unix) and SUN Ultra (SUN OS). See [http://www.stat.wisc.edu/~hkim/cruise/cruise.html](http://www.stat.wisc.edu/~hkim/cruise/cruise.html).

3. `heart.dat`, `heart.dsc`. Sample data and variable description files used in this user manual.

The CRUISE program can be invoked with the command `cruise` or by double-clicking on the `cruise` icon.

## 3 Input files in CRUISE

Two text files (three if test data are available) are needed to run CRUISE.

**Data file:** This file contains the learning (or training) samples. Each sample consists of observations on the class variable and the predictor variables. The entries in each sample record should be comma or space delimited. Each record can occupy one or more lines in the file, but each record must begin on a new line. Record values should be numerical. Class variable and categorical variables are required to take integer values, not necessarily consecutive.

**Test file:** If a test file is available, it must have the same format as that of the data file.

**Description file:** This file is used to provide information to the program about the name of the data file, the names and column locations of the variables, and their data types in the analysis. The file `heart.dsc` is an example description file. Its contents are:

```
heart.dat
?
column, varname, vartype
  1  age  n
  2  sex  c
  3 chestpain c
  4 bloodpres n
  5 cholestor n
  6 bloodsuga c
  7 restcardi c
  8 heartrate n
  9 exercise c
 10 oldpeak  n
 11 slope  c
 12 vessels  c
 13 thal  c
 14 diagnosis  d
```

The first line of the file gives the name of the learning sample file. The second line gives the code that denotes a missing value in the data. The missing value code can be up to 25 characters long. A missing value code must appear in the second line of the file even if there are no missing values in the data (in which case any character string can be used). If no
missing values are found in the data with this string, CRUISE will notify the user but still proceed with the tree construction. The third line contains three character strings to indicate the column headers for the subsequent lines. The position, name and role of each variable comes next (in that order), with one line for each variable. Variable names longer than 9 characters are truncated.
The following roles for the variables are permitted:

- **c** This is a categorical variable.
- **n** This is a numerical variable.
- **d** This is a class or a dependent variable. Only one variable can have the d type.

## 4 Running CRUISE

The CRUISE program prompts the user for answers during the construction of the classification tree. For each question, CRUISE prints out the range of possible answers within brackets (e.g. [0:1]) and a suggested answer (e.g. <cr>=1). Any answer out of the range will cause the question to be asked again. For example,

```
Input 1 to overwrite it, 0 to choose another name ([0:1], <cr>=1): 2
Error, value out of range
Input 1 to overwrite it, 0 to choose another name ([0:1], <cr>=1): 
```

A carriage return will result in the choice of the default value. Following is an annotated example session log for the UNIX version (annotations are printed in *italics*).

```
> cruise

CRUISE (beta)
Copyright (c) 1997, by HyunJoong Kim and Wei-Yin Loh

Q1
Enter name of file to store results: heart.out
   A file by that name already exists
Input 1 to overwrite it, 0 to choose another name ([0:1], <cr>=1):

Q2
Enter name of variable description file: heart.dsc

Q3
Input 1 for univariate splits, 2 for linear combination splits ([1:2], <cr>=1):

Q4
Input 1 for variable selection via interaction tests,
   2 for variable selection via F-and-Levene tests ([1:2], <cr>=1):

Q5
Input 1 for equal priors,
   2 for estimated priors,

March 3, 1998
3 for your choice of priors ([1:3], <cr>=2):

Q6
Input 1 for equal, 2 for unequal misclassification costs ([1:2], <cr>=1):

Q7
Input 1 for pruning via cross-validation,
2 for pruning via test sample,
3 for direct stopping ([1:3], <cr>=1):

Q8
Input number of SEs for pruning ([0.00:50.00], <cr>=0.00): 1

Q9
Enter a mindat value ([1:303], <cr>=5):

Q10
Input 1 if you have a test sample, 2 otherwise ([1:2], <cr>=2):

Q11
Input number of folds for cross-validation ([2:303], <cr>=10):

Q12
Missing values are found in the learning sample
Input 1 to fit (complete cases) and impute,
2 to impute and fit,
3 to fit (available cases) and impute (univariate split),
4 to impute at root node only and fit ([1:4], <cr>=3):

Q13
Missing values are found in pruning (test or cross-validation) sample
Choose one of the following options for handling missing values.
Input 1 for Root node imputation,
2 for Grand mean/mode imputation for each node,
3 for Estimate the class and impute,
4 for Go down if possible,
5 for Alternate Split,
6 for Proxy Split ([1:6], <cr>=5):

Building Initial Trees...
Calibrating...........................................................
Pruning Trees..............
Elapsed time in seconds:
cpu = 13.6 user = 13.0 sys = 0.571 wall = 53.8

!!! Good Bye !!!
Explanation of questions

Following is a brief explanation of the questions asked by the program.

Q1. This asks for the name of a file to store the results. If a file by that name already exists, the user is asked whether he or she would like to overwrite it or choose another name.

Q2. This asks for the name of the description file. If the file does not exist, then the program prompts again to input existing filename.

Q3. The user can choose to use univariate splits or linear combination splits. If linear combination splits are desired, the user must choose option 2 here.

Q4. If the user choose to use univariate splits, the program asks the user to choose between the unbiased interaction selection method and the unbiased F-and-Levene test method.

Q5. This allows the user to choose various prior probabilities for each class. If the user wants to specify the priors, option 3 should be chosen.

Q6. The user can choose between equal or unequal misclassification costs. If the user wishes to specify his/her own costs, choice 2 should be selected.

Q7. The user can choose the method of tree simplification. If a test data file is available for pruning, the user can make choice 2. Otherwise, option 1 is the recommended selection.

Q8. The number of SEs control the size of the pruned tree. 0-SE gives the tree with the smallest cross-validation estimate of error. We choose the 1-SE tree in this example.

Q9. mindat is the smallest number of samples in a node during tree construction. A node will not be split if all the classes contain fewer observations than mindat. The smaller the value of mindat, the larger the initial tree will be prior to pruning.

Q10. If a test data file is available for evaluating the trees, the user can select option 1.

Q11. This asks for the value of V in V-fold cross-validation. The larger the value of V, the longer time it takes the program to finish. 10-fold is the default.

Q12. When missing values are found during tree construction in the learning sample, the user can choose an imputation method. Option 3 is the default for univariate splits but is not available for linear combination splits. For linear combination splits, option 2 is the default.

Q13. When missing values are found in the pruning stage or in the test data file, the user can choose the method for imputation. Generally, options 5 or 6 are recommended.

5 Sample output file

This section contains a sample output file with annotations given in italics.
Learning sample data file is: heart.dat
Number of learning samples = 303
Number of classes = 2
Number of covariates = 13
Number of categorical variables = 8

Split method: Univariate splits
Variable Selection method: Interaction Detection and Calibration
Prior selection: Estimated prior probability
Cost selection: Equal misclassification cost
Imputation option for building trees: Fit (available cases) and impute (univariate splits) method

Tree size determination: Pruning by cross-validation
S.E rule used = 1.0
Minimum size of each node (mindat): 5
Number of folds for cross-validation: 10
Imputation option for test, CV, or pruning sample: Alternate Split

Calibrated scale for hypothesis tests = 1.3038

Tree Structures:

Node 1: oldpeak <= 0.831296
  Node 2: vessels = 4 3 2
    Node 4: sex = 1
      Node 8: chestpain = 4 2
        Node 16: Terminal Node, predicted class = 1
          Class label: 1 2
          Class size: 20 3
      Node 8: chestpain = 3 1
        Node 17: Terminal Node, predicted class = 2
          Class label: 1 2
          Class size: 5 7
    Node 4: sex = 0
      Node 9: Terminal Node, predicted class = 2
        Class label: 1 2
        Class size: 2 14
Node 2: vessels = 1
   Node 5: Terminal Node, predicted class = 2
      Class label : 1 2
      Class size : 18 94

Node 1: oldpeak > 0.831296
   Node 3: thal = 3 2
      Node 6: chestpain = 2 4 3
         Node 12: Terminal Node, predicted class = 1
            Class label : 1 2
            Class size : 73 6
      Node 6: chestpain = 1
         Node 13: Terminal Node, predicted class = 2
            Class label : 1 2
            Class size : 2 5

Node 3: thal = 1
   Node 7: heartrate <= 129.064
      Node 14: sex = 1
         Node 28: Terminal Node, predicted class = 1
            Class label : 1 2
            Class size : 8 1
      Node 14: sex = 0
         Node 29: Terminal Node, predicted class = 2
            Class label : 1 2
            Class size : 2 6
   Node 7: heartrate > 129.064
      Node 15: Terminal Node, predicted class = 2
            Class label : 1 2
            Class size : 9 28

P6
Detailed Description of the Tree:

<table>
<thead>
<tr>
<th>Nodes</th>
<th>No.</th>
<th>Subnode</th>
<th>Interact</th>
<th>Split</th>
<th>Split</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>cases</td>
<td>label</td>
<td>variable</td>
<td>variable</td>
<td>value</td>
</tr>
<tr>
<td>1</td>
<td>303</td>
<td>2</td>
<td>heartrate</td>
<td>oldpeak ≤</td>
<td>0.83130</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>&lt; infinity</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td># obs</td>
<td>mean/mode of oldpeak</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Class 1 :</td>
<td>139 1.57410</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Class 2 :</td>
<td>164 0.586585</td>
</tr>
<tr>
<td>2</td>
<td>163</td>
<td>4</td>
<td>sex</td>
<td>vessels =</td>
<td>4 3 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

March 3, 1998
# obs mean/mode of vessels
Class 1 : 45  1
Class 2 : 118  1

4  51  8  sex = 1
9  = 0

# obs mean/mode of sex
Class 1 : 27  1
Class 2 : 24  0

8  35  16  oldpeak  chestpain= 4 2
17  = 3 1

# obs mean/mode of chestpain
Class 1 : 25  4
Class 2 : 10  3

16  23  **** terminal, predicted class :  1

# obs
Class 1 : 20
Class 2 :  3

17  12  **** terminal, predicted class :  2

# obs
Class 1 :  5
Class 2 :  7

9  16  **** terminal, predicted class :  2

# obs
Class 1 :  2
Class 2 : 14

5  112  **** terminal, predicted class :  2

# obs
Class 1 : 18
Class 2 : 94

3  140  6  chestpain  thal = 3 2
7  = 1

# obs mean/mode of thal
Class 1 : 94  3
Class 2 : 46  1

6  86  12  oldpeak  chestpain= 2 4 3
13  = 1

# obs mean/mode of chestpain
Class 1 : 75  4
Class 2 : 11  1
** terminal, predicted class : 1
  # obs
  Class 1 : 73
  Class 2 : 6

** terminal, predicted class : 2
  # obs
  Class 1 : 2
  Class 2 : 5

54 14 age heartrate <= 129.06
  15 < infinity
  # obs mean/mode of heartrate
  Class 1 : 19 131.526
  Class 2 : 35 152.686

17 28 sex = 1
  29 = 0
  # obs mean/mode of sex
  Class 1 : 10 1
  Class 2 : 7 0

** terminal, predicted class : 1
  # obs
  Class 1 : 8
  Class 2 : 1

** terminal, predicted class : 2
  # obs
  Class 1 : 2
  Class 2 : 6

** terminal, predicted class : 2
  # obs
  Class 1 : 9
  Class 2 : 28

P7
Number of nodes in maximum tree = 27
Number of nodes in final tree = 17
Number of terminal nodes in final tree = 9

P8
Classification Matrix :
  predicted class
           1  2
  Class
  #obs Prior
  ------------------

9

March 3, 1998
1 139 .459 101 38
2 164 .541 10 154

Total obs = 303,  # correct = 255
Estimate of misclassification cost = 0.1584
S.E. of misclassification cost = 0.2008E-01

P9
Cross-validation error cost acquired by pruning : 0.2574
S.E. of CV misclassification cost = 0.2501E-01

P10
Elapsed time in seconds:
cpu = 13.6  user = 13.0  sys = 0.571  wall = 53.8

Explanation of annotations

P1. This paragraph displays the information obtained from the learning data including total number of observations, number of classes, number of predictors, and the number of categorical variables.

P2. Information obtained from the user during the interactive session, such as split method, variable selection method, prior probabilities, misclassification costs, and imputation methods for missing values are listed here.

P3. Tree simplification options are given here. Pruning method, SE rule, size of mindat, V-fold, and imputation method while pruning are included.

P4. Bootstrap calibration coefficient for unbiased interaction selection method is displayed.

P5. This paragraph displays the tree structure in outline form suitable for importing into flow-chart programs such as allCLEAR or CLEAR Org Charts.

P6. Detailed descriptions of the splits are given here. Split variable, split point, interacting variable with the split variable, and the sample mean or mode of the selected variable are also given.

P7. Total number of nodes and the number of terminal nodes in final tree are listed.

P8. Classification result for the learning sample is given. The resubstitution error rate along with its standard error are also listed.


P10. UNIX versions of the program also report the total CPU time taken by the run.

The resulting classification tree is given in Figure 1 (as drawn with CLEAR Org Charts).
Figure 1: A classification tree with univariate splits using interaction detection method. Intermediate nodes are represented by circles with the split variables given. A case goes down to a node if it satisfies the condition of the node. The number on the right of each node is the node class sizes. Terminal nodes are represented by rectangles with the predicted class inside.

6 Other outputs

6.1 Univariate splits

F-and-Levene tests based splits can be selected as follows:

Input 1 for variable selection via interaction tests,
   2 for variable selection via F-and-Levene tests ([1:2], <cr>=1): 2

Using the defaults along with this option, we get the following output.

CRUISE (beta)
Copyright (c) 1997-1998, by HyunJoong Kim and Wei-Yin Loh
This version was updated on : February 09, 1998
This job was completed on 2/11/98 at 20:16

Learning sample data file is: hea.dat
Number of learning samples = 303
Number of classes = 2
Number of covariates = 13
Number of categorical variables = 8
Split method: Univariate splits
Variable Selection method: F & Levene test
alpha-threshold used = 0.500E-01
Prior selection: Estimated prior probability
Cost selection: Equal misclassification cost
Imputation option for building trees: Fit(available cases) and impute (univariate splits) method

Tree size determination: Pruning by cross-validation
S.E rule used = 0.0
Minimum size of each nodes (mindat): 5
Number of folds for cross-validation: 10
Imputation option for test, CV, or pruning sample: Alternate Split

Tree Structures:

Node 1: thal = 3 2
   Node 2: chestpain = 4 3
      Node 4: Terminal Node, predicted class = 1
         Class label : 1 2
         Class size : 94 21

   Node 2: chestpain = 2 1
      Node 5: Terminal Node, predicted class = 2
         Class label : 1 2
         Class size : 8 13

Node 1: thal = 1
   Node 3: vessels = 4 3
      Node 6: Terminal Node, predicted class = 1
         Class label : 1 2
         Class size : 12 8

   Node 3: vessels = 2 1
      Node 7: vessels = 2
         Node 14: chestpain = 4
            Node 28: Terminal Node, predicted class = 1
               Class label : 1 2
               Class size : 9 1

            Node 14: chestpain = 2 1 3
               Node 29: Terminal Node, predicted class = 2
                  Class label : 1 2
                  Class size : 3 16

   Node 7: vessels = 3 4 1
      Node 15: Terminal Node, predicted class = 2
         Class label : 1 2

12

March 3, 1998
Class size : 13 105

Detailed Description of the Tree :

<table>
<thead>
<tr>
<th>Nodes</th>
<th>No. cases</th>
<th>Subnode label</th>
<th>Split stat.</th>
<th>Split variable</th>
<th>Split value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>303</td>
<td>2</td>
<td>F</td>
<td>thal</td>
<td>3 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td># obs mean/mode of thal</td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>139</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>164</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>136</td>
<td>4</td>
<td>F</td>
<td>chestpain</td>
<td>4 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td># obs mean/mode of chestpain</td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>102</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>34</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>115</td>
<td>*** terminal, predicted class : 1</td>
<td># obs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>94</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>*** terminal, predicted class : 2</td>
<td># obs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>167</td>
<td>6</td>
<td>F</td>
<td>vessels</td>
<td>4 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td># obs mean/mode of vessels</td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>37</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>130</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>*** terminal, predicted class : 1</td>
<td># obs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>147</td>
<td>14</td>
<td>F</td>
<td>vessels</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 4 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td># obs mean/mode of vessels</td>
<td></td>
</tr>
<tr>
<td>Class 1 :</td>
<td>25</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class 2 :</td>
<td>122</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>29</td>
<td>28</td>
<td>F</td>
<td>chestpain</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 1 3</td>
</tr>
</tbody>
</table>
# obs mean/mode of chestpain
Class 1 : 12  4
Class 2 : 17  3

28  10  **** terminal, predicted class :  1
    # obs
    Class 1 :  9
    Class 2 :  1

29  19  **** terminal, predicted class :  2
    # obs
    Class 1 :  3
    Class 2 : 16

15  118  **** terminal, predicted class :  2
    # obs
    Class 1 : 13
    Class 2 : 105

Number of nodes in maximum tree = 23
Number of nodes in final tree = 11
Number of terminal nodes in final tree = 6

Classification Matrix : predicted class
                        1   2
Class  #obs  Prior   ----------
 1   139 .459  115  24
 2   164 .541  30 134

Total obs = 303,  # correct = 249
Estimate of misclassification cost = 0.1782
S.E. of misclassification cost = 0.2198E-01

Cross-validation error cost acquired by pruning : 0.2310
S.E. of CV misclassification cost = 0.2403E-01
Elapsed time in seconds:
cpu = 2.64  user = 2.27  sys = 0.368  wall = 16.8

6.2 Linear combination splits
Linear combination splits can be obtained with the following option.
Input 1 for univariate splits, 2 for linear combination splits ([1:2],<cr>=1):2
Using default values, the output for linear combination splits is as follows.

CRUISE (beta)
Copyright (c) 1997-1998, by HyunJoong Kim and Wei-Yin Loh

14  March 3, 1998
Learning sample data file is: hea.dat
Number of learning samples = 303
Number of classes = 2
Number of covariates = 13
Number of categorical variables = 8
Split method: Linear combination splits
Prior selection: Estimated prior probability
Cost selection: Equal misclassification cost
Imputation option for building trees: Impute and fit method

Tree size determination: Pruning by cross-validation
S.E rule used = 0.0
Minimum size of each nodes(mindat): 5
Number of folds for cross-validation: 10
Imputation option for test, CV, or pruning sample: Alternate Split

Tree Structures:
Node 1:
Node 2:
  Node 4: Terminal Node, predicted class = 1
  Class label : 1 2
  Class size : 110 10

Node 2:
  Node 5: Terminal Node, predicted class = 2
  Class label : 1 2
  Class size : 3 7

Node 1:
Node 3:
  Node 6: Terminal Node, predicted class = 1
  Class label : 1 2
  Class size : 6 2

Node 3:
  Node 7: Terminal Node, predicted class = 2
  Class label : 1 2
  Class size : 20 145

Detailed Description of the Tree:

<table>
<thead>
<tr>
<th>Nodes No.</th>
<th>Class label</th>
<th>Subnode label</th>
</tr>
</thead>
<tbody>
<tr>
<td>cases</td>
<td>sizes</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>303</td>
<td>139 2</td>
</tr>
<tr>
<td></td>
<td>164</td>
<td>3</td>
</tr>
</tbody>
</table>
Category for sex : Crimcoords
0  0.615704E-01
1  -0.615704E-01

Category for chestpain : Crimcoords
1  0.113943E-01
2  0.389832E-01
3  0.324819E-01
4  -0.828594E-01

Category for bloodsuga : Crimcoords
0  0.807747E-01
1  -0.807747E-01

Category for restcardi : Crimcoords
1  0.115799
2  -0.124192
3  0.839342E-02

Category for exercise : Crimcoords
0  0.612433E-01
1  -0.612433E-01

Category for slope : Crimcoords
1  0.706261E-01
2  -0.469647E-01
3  -0.236614E-01

Category for vessels : Crimcoords
1  0.918092E-01
2  -0.611532E-02
3  -0.388186E-01
4  -0.468753E-01

Category for thal : Crimcoords
1  0.716107E-01
2  -0.253449E-01
3  -0.462659E-01

Coefficients of class 1 2
Const:  -105.568  -106.426
age     :  1.06186  1.08719
sex     : -64.7369  -54.2135
chestpain: -57.6984 -40.3826
bloodpres:  0.343440  0.323354
cholestor:  0.632653E-010.594862E-01
bloodsuga:  44.9612  41.8761
restcardi:  55.6152  59.7589
heartrate:  0.531798  0.549444
exercise:  -27.9557 -20.3774
oldpeak:  0.774697  0.498678
slope:  -57.8430 -48.6420
vessels:  35.2158  58.6349
thal:  31.3034  47.7564
2 130 113 4
17 5
Category for sex : Crimcoords
0 -.124138
1  .124138
Category for chestpain : Crimcoords
1  .279740
2 -.147499
3 -.498446E-01
4 -.823962E-01
Category for bloodsuga : Crimcoords
0  .116955
1 -.116955
Category for restcardi : Crimcoords
1  .184137
2 -.215089
3  .309524E-01
Category for exercise : Crimcoords
0  .901388E-01
1 -.901388E-01
Category for slope : Crimcoords
1  .146603
2 -.601869E-01
3 -.864165E-01
Category for vessels : Crimcoords
1  .140323
2 -.442735E-01
3 -.762561E-01
4 -.197930E-01
Category for thal : Crimcoords
1  .848393E-01
2  .408427E-01
3 -.125682
Coefficients of class 1 2
Const: -105.242 -105.251
age  : 0.909073  0.973343
sex :  65.2400   63.1130
chestpain : -47.7443 -31.4702
bloodpres : 0.385652  0.375006
cholestor : 0.104552  0.968287E-01
bloodsuga : 28.6242  29.9034
restcardi : 33.6005  39.3006
hearrate : 0.469024  0.456655
exercise : -39.3840  -34.0157
oldpeak : 1.17332  0.815765
slope : -35.1571  -26.4607
vessels : 9.65635  22.8932
thal : 22.1488    29.5442

4  120  **** terminal, predicted class : 1
    # obs
Class 1 : 110
Class 2 : 10

5  10   **** terminal, predicted class : 2
    # obs
Class 1 : 3
Class 2 : 7

3  173  26  6
    147  7
Category for sex : Crimcoords
  0        0.763983E-01
  1        -0.763983E-01
Category for chestpain : Crimcoords
  1        -0.836014E-01
  2        0.561616E-01
  3        0.100729
  4        -0.732892E-01
Category for bloodsuga : Crimcoords
  0        -0.111965
  1        0.111965
Category for restcardi : Crimcoords
  1        -0.832382E-01
  2        0.304616
  3        -0.221378
Category for exercise : Crimcoords
  0        0.121578
  1        -0.121578
Category for slope : Crimcoords
  1        0.472517E-01
  2        -0.117437
  3        0.701858E-01
Category for vessels : Crimcoords
  1        0.207344
  2        0.755886E-01
  3        -0.101904E-02
  4        -0.281914
Category for thal : Crimcoords
  1        -0.221558E-01
  2        0.233489
  3        -0.211333
Coefficients of class 1 2
Const: -124.243    -123.245
age : 1.30110  1.30509
sex : -12.0983  -1.91786
chestpain : 16.1506  25.0855
bloodpres : 0.347026  0.313960
cholestor : 0.406699E-02 0.253173E-02
bloodsuga : -45.6990  -39.9635
restcardi : -13.2547  -10.9847
heartRate : 0.716122  0.746797
exercise : 29.8441  28.3294
oldpeak : -.630592  -.553702
slope : -33.4844  -27.2705
vessels : 41.2085  51.7154
thal : -7.71215  -2.54237

6     8     **** terminal, predicted class : 1
       # obs
Class 1 :  6
Class 2 :  2

7   165     **** terminal, predicted class : 2
       # obs
Class 1 :  20
Class 2 : 145

Number of nodes in maximum tree = 13
Number of nodes in final tree = 7
Number of terminal nodes in final tree = 4

Classification Matrix :       predicted class
                          1   2
Class       #obs Prior  -----------
1          139 .459   116  23
2          164 .541   12  152

Total obs = 303,    # correct = 268
Estimate of misclassification cost = 0.1155
S.E. of misclassification cost = 0.1817E-01

Cross-validation error cost acquired by pruning : 0.1650
S.E. of CV misclassification cost = 0.2115E-01
Elapsed time in seconds:
cpu = 5.38         user = 4.86      sys = 0.520      wall = 15.5

References