Decision Tree Example: Statlog Heart

We will use statlog heart dataset again for decision tree model practice. Statlog (Heart) Data Set is downloaded from UCI machine learning repository. It has 13 different attributes and 1 class variable. The detailed feature attributes are presented in the below. You could also find it here: https://archive.ics.uci.edu/ml/datasets/Statlog+%28Heart%29

The attributes include: 1. Age; 2. Sex (female =0, male =1); 3. chestpain = chest pain type (4 values); 4. restbloodpr = resting blood pressure; 5. serumchol = serum cholestoral in mg/dl; 6. blsugar.120 = fasting blood sugar > 120 mg/dl (have =1, don’t have =0); 7. restelectro = resting electrocardiographic results (values 0,1,2); 8. maxheartrate = maximum heart rate achieved; 9. angina = exercise induced angina (don’t have =0, have =1); 10. oldpeak = ST depression induced by exercise relative to rest; 11. st.segment = the slope of the peak exercise ST segment; 12. majorvessels = number of major vessels (0-3) colored by flourosopy; 13. thal: 3 = normal; 6 = fixed defect; 7 = reversable defect;

The class variable is statlogheart: 1=Absence of heart disease; 2= presence of heart disease;

Among all attributes, we have real attributes: 1,4,5,8,10,12; ordered attributes: 11; binary attributes: 2,6,9 and nominal attributes: 3, 7, 13;

1. Import dataset to R and check the data structure of the dataset.

Ans :

'data.frame': 270 obs. of 14 variables:

$ age : int 70 67 57 64 74 65 56 59 60 63 ...

$ sex : int 1 0 1 1 0 1 1 1 1 0 ...

$ chestpain : int 4 3 2 4 2 4 3 4 4 4 ...

$ restbloodpr : int 130 115 124 128 120 120 130 110 140 150 ...

$ serumchol : int 322 564 261 263 269 177 256 239 293 407 ...

$ blsugar.120 : int 0 0 0 0 0 0 1 0 0 0 ...

$ restelectro : int 2 2 0 0 2 0 2 2 2 2 ...

$ maxheartrate: int 109 160 141 105 121 140 142 142 170 154 ...

$ angina : int 0 0 0 1 1 0 1 1 0 0 ...

$ oldpeak : num 2.4 1.6 0.3 0.2 0.2 0.4 0.6 1.2 1.2 4 ...

$ st.segment : int 2 2 1 2 1 1 2 2 2 2 ...

$ majorvessels: int 3 0 0 1 1 0 1 1 2 3 ...

$ thal : int 3 7 7 7 3 7 6 7 7 7 ...

$ statlogheart: int 2 1 2 1 1 1 2 2 2 2 ...

Code:

statlog <- read.csv(file.choose(), stringsAsFactors = F)

str(statlog)

1. Change attributes 2,3,6,7,9,13, and 14 to factor variables. Hint: dt[c(2,3,6,7,9,13,14)]<-data.frame(lapply(dt[c(2,3,6,7,9,13,14)],factor))

Ans :

'data.frame': 270 obs. of 14 variables:

$ age : int 70 67 57 64 74 65 56 59 60 63 ...

$ sex : Factor w/ 2 levels "0","1": 2 1 2 2 1 2 2 2 2 1 ...

$ chestpain : Factor w/ 4 levels "1","2","3","4": 4 3 2 4 2 4 3 4 4 4 ...

$ restbloodpr : int 130 115 124 128 120 120 130 110 140 150 ...

$ serumchol : int 322 564 261 263 269 177 256 239 293 407 ...

$ blsugar.120 : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 2 1 1 1 ...

$ restelectro : Factor w/ 3 levels "0","1","2": 3 3 1 1 3 1 3 3 3 3 ...

$ maxheartrate: int 109 160 141 105 121 140 142 142 170 154 ...

$ angina : Factor w/ 2 levels "0","1": 1 1 1 2 2 1 2 2 1 1 ...

$ oldpeak : num 2.4 1.6 0.3 0.2 0.2 0.4 0.6 1.2 1.2 4 ...

$ st.segment : int 2 2 1 2 1 1 2 2 2 2 ...

$ majorvessels: int 3 0 0 1 1 0 1 1 2 3 ...

$ thal : Factor w/ 3 levels "3","6","7": 1 3 3 3 1 3 2 3 3 3 ...

$ statlogheart: Factor w/ 2 levels "1","2": 2 1 2 1 1 1 2 2 2 2 ...

Code:

statlog[c(2,3,6,7,9,13,14)]<-data.frame(lapply(statlog[c(2,3,6,7,9,13,14)],factor))

statlog[c(2,3,6,7,9,13,14)]

1. Add labels (No disease, Heart disease) to statlogheart

Ans :

age sex chestpain restbloodpr serumchol blsugar.120 restelectro maxheartrate angina oldpeak st.segment majorvessels

1 70 1 4 130 322 0 2 109 0 2.4 2 3

2 67 0 3 115 564 0 2 160 0 1.6 2 0

3 57 1 2 124 261 0 0 141 0 0.3 1 0

4 64 1 4 128 263 0 0 105 1 0.2 2 1

5 74 0 2 120 269 0 2 121 1 0.2 1 1

6 65 1 4 120 177 0 0 140 0 0.4 1 0

thal statlogheart

1 3 Heart disease

2 7 No disease

3 7 Heart disease

4 7 No disease

5 3 No disease

6 7 No disease

Code:

statlog$statlogheart <- factor(statlog$statlogheart , levels=c(1,2), labels=c("No disease", "Heart disease"))

str(statlog$statlogheart)

head(statlog)

1. Create a training dataset with 99% of records, and use it for the rest of the analysis. (Discussion: Typically, you will not want to keep such a high percentage of records in training set. I made this exception because results were not stable at lower percentage split.)

Ans :

'data.frame': 267 obs. of 14 variables:

$ age : int 58 50 58 69 42 67 57 62 60 63 ...

$ sex : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 2 1 1 1 ...

$ chestpain : Factor w/ 4 levels "1","2","3","4": 4 2 3 1 3 4 3 4 1 4 ...

$ restbloodpr : int 128 120 140 140 130 100 150 150 150 150 ...

$ serumchol : int 216 244 211 239 180 299 126 244 240 407 ...

$ blsugar.120 : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 2 1 1 1 ...

$ restelectro : Factor w/ 3 levels "0","1","2": 3 1 3 1 1 3 1 1 1 3 ...

$ maxheartrate: int 131 162 165 151 150 125 173 154 171 154 ...

$ angina : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 2 1 1 ...

$ oldpeak : num 2.2 1.1 0 1.8 0 0.9 0.2 1.4 0.9 4 ...

$ st.segment : int 2 1 1 1 1 2 1 2 1 2 ...

$ majorvessels: int 3 0 0 2 0 2 1 0 0 3 ...

$ thal : Factor w/ 3 levels "3","6","7": 3 1 1 1 1 1 3 1 1 3 ...

$ statlogheart: Factor w/ 2 levels "No disease","Heart disease": 2 1 1 1 1 2 1 2 1 2 ...

Code:

nrows <- nrow(statlog)

nrows

train.size <- floor(nrows \* 0.99)

train.size

set.seed(45678)

train.index <- sample(1:nrows, train.size, replace = F)

statlog.train <- statlog[train.index,]

str(statlog.train)

1. Build a decision tree model with statlogheart as outcome variable and the rest variables as predictors. Then display simple facts about the tree. What tree size does this decision tree model have?

Ans :

Classification Tree

Number of samples: 267

Number of predictors: 13

Tree size: 15

Code:

install.packages("C50")

library(C50)

model <- C5.0(statlog.train[-14],statlog.train$statlogheart)

model

1. Summarize this decision tree model and display the detailed information of the tree model. How many patients have been misclassified? How many attributes have been utilized to generate the tree.

Ans :

# 14 attributes

# 23( 8.6%) have been misclassified

Code:

summary(model)

1. If a patient is female, has one major vessel colored by flourosopy (majorvessels) and has thal = 6, does this patient have heart disease base on the decision tree?

Ans :

#Yes , this patient have heart disease base on the decision tree.(65 records,6 incorrect)

1. If a patient has no major vessel colored, angina is one, oldpeak is less than 1.5, has thal = 7, serumchol is less than 255, does this patient have heart disease base on the decision tree?

Ans : No heart disease

1. Boost the accuracy of decision trees by boosted decision tree with 10 trials. What is the average tree size for 10 iterations? How many patients in the boosted model are misclassified?

Ans :

#Average tree size: 12

#5 patients have been misclassified

Code:

credit\_boost10 <- C5.0(statlog.train[-14], statlog.train$statlogheart,

trials = 10)

credit\_boost10

summary(credit\_boost10)

1. Create a cost sensitive decision tree on training data. The sequence of the costs should be c (0, 1, 10, 0) in the R syntax.

Code:

# build the matrix

error\_cost <- matrix(c(0, 1, 10, 0), nrow = 2)

error\_cost

statlog\_cost <- C5.0(statlog.train[-14], statlog.train$statlogheart,

costs = error\_cost)

statlog\_cost

1. Use the above cost sensitive decision tree to create a factor vector of predictions for in-sample data. How many cases of false negatives are there? Is there any difference in CrossTable results and contingency matrix in summary command?

Ans :

#61

#No there is no difference

Code:

statlog\_cost\_pred <- predict(statlog\_cost,statlog.train)

summary(statlog\_cost\_pred)

install.packages('gmodels')

library(gmodels)

CrossTable(statlog.train$statlogheart, statlog\_cost\_pred, dnn=c("Actual", "Predicted"))