DTCreditApprovalCRAN.R

ai

Mon Jun 5 22:20:27 2017

```r
# Reference for data source (
# @misc{Lichman:2013 ,
# author = "M. Lichman",
# year = "2013",
# title = "{UCI} Machine Learning Repository",
# url = "http://archive.ics.uci.edu/ml",
# institution = "University of California, Irvine, School of Information and Computer Sciences" })

# Decision Trees
# Source of Data Set:- UCI Repository - Wine Quality Data(https://archive.ics.uci.edu/ml/datasets/wine+

# required libraries
# # The rpart package can be installed via the install.packages("rpart") and
# # loaded with the library(rpart) command.
library(rpart) #recursive and partitioning trees

# # The plotly package can be installed via the install.packages("plotly") and
# # loaded with the library(plotly) command.
library(plotly) #data visualization
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```r
# # The rpart.plot package can be installed via the install.packages("rpart.plot") and
# # loaded with the library(rpart.plot) command.
library(rpart.plot)

# # The rattle package can be installed via the install.packages("rattle") and
# # loaded with the library(rattle) command.
library(rattle)
```

```
## Rattle: A free graphical interface for data mining with R.
## Version 5.0.8 Copyright (c) 2006-2017 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```r
# # The RColorBrewer package can be installed via the install.packages("RColorBrewer") and
# # loaded with the library(RColorBrewer) command.
library(RColorBrewer)


# # The RWeka package can be installed via the install.packages("RWeka") and
# # loaded with the library(RWeka) command.
library(RWeka)


# Step 01: Collecting data
# Download data from UCI repository
CreditDataUrl <- "https://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german/german.data"

# Read the url html file into a data frame titled CreditData.
CreditData <- read.table(CreditDataUrl)

# Assging attribute information
# The target function column name is class
colnames(CreditData) <- c("chk_status", "mth_duration", "credit_history", "purpose", "credit_amount", "s

# Write a CSV file from CreditData
Credit_Data <- write.csv(CreditData, file = "CreditData.csv", row.names = FALSE)

# Exploring and preparing the data
# Step 2: Exploring and preparing the data
# Read the csv file into a data frame titled CreditData.
CreditData <- read.csv("CreditData.csv", header=TRUE)

# Class columns convert into facator
CreditData$class <- ifelse(CreditData$class==1, "good","bad")
CreditData$class = as.factor(CreditData$class)

# Displays description of each variable

head(CreditData)
```

```
##   chk_status mth_duration credit_history purpose credit_amount saving
## 1        A11            6            A34     A43          1169    A65
## 2        A12           48            A32     A43          5951    A61
## 3        A14           12            A34     A46          2096    A61
## 4        A11           42            A32     A42          7882    A61
## 5        A11           24            A33     A40          4870    A61
## 6        A14           36            A32     A46          9055    A65
##   employ_time pct_dpi status_gender other_debts residency_time property
## 1         A75       4           A93        A101              4     A121
## 2         A73       2           A92        A101              2     A121
## 3         A74       2           A93        A101              3     A121
## 4         A74       2           A93        A103              4     A122
## 5         A73       3           A93        A101              4     A124
## 6         A73       2           A93        A101              4     A124
##   age other_installments housing existing_credits  job dependents_num
## 1  67               A143    A152                2 A173              1
## 2  22               A143    A152                1 A173              1
```

```
## 3  49               A143    A152               1 A172               2
## 4  45               A143    A153               1 A173               2
## 5  53               A143    A153               2 A173               2
## 6  35               A143    A153               1 A172               2
##   phone foreign class
## 1  A192    A201  good
## 2  A191    A201   bad
## 3  A191    A201  good
## 4  A191    A201  good
## 5  A191    A201   bad
## 6  A192    A201  good
```

```r
# Data preparation - creating random training and test datasets
# Create random sample
# Divide the data into a training set and a test set randomly with ratio 80:20

set.seed(123)
train_sample <- sample(nrow(CreditData), 0.9 * nrow(CreditData))
CreditData_train <- CreditData[train_sample, ]
CreditData_test <- CreditData[-train_sample, ]

# Check whether data set fairly even split
prop.table(table(CreditData_train$class))
```

```
##
##       bad      good
## 0.2966667 0.7033333
```

```r
prop.table(table(CreditData_test$class))
```

```
##
##  bad good
## 0.33 0.67
```

```r
# Train model - Regression Tree
# Build the model with recursive partitioning trees
CreditData_model <- rpart(class ~. , data = CreditData_train)

summary(CreditData_model)
```

```
## Call:
## rpart(formula = class ~ ., data = CreditData_train)
##   n= 900
##
##           CP nsplit rel error    xerror       xstd
## 1 0.04119850      0 1.0000000 1.0000000 0.05132453
## 2 0.02247191      4 0.8089888 0.9662921 0.05080949
## 3 0.01622971      6 0.7640449 0.9513109 0.05057089
## 4 0.01498127     10 0.6966292 0.9588015 0.05069095
## 5 0.01310861     13 0.6516854 0.9625468 0.05075041
## 6 0.01000000     16 0.6067416 0.9513109 0.05057089
##
## Variable importance
##       chk_status    credit_amount   credit_history          purpose
##               30               11               11               11
##     mth_duration           saving         property existing_credits
```

```
##             10              7              6              3
##            age  status_gender    employ_time        pct_dpi
##              3              1              1              1
##            job        housing
##              1              1
##
## Node number 1: 900 observations,    complexity param=0.0411985
##   predicted class=good  expected loss=0.2966667  P(node) =1
##     class counts:   267   633
##    probabilities: 0.297 0.703
##   left son=2 (488 obs) right son=3 (412 obs)
##   Primary splits:
##       chk_status     splits as  LLRR,        improve=46.703630, (0 missing)
##       credit_history splits as  LLRRR,       improve=14.313530, (0 missing)
##       saving         splits as  LLRRR,       improve=13.547950, (0 missing)
##       mth_duration   < 15.5    to the right, improve=10.540210, (0 missing)
##       credit_amount  < 3913.5  to the right, improve= 8.573889, (0 missing)
##   Surrogate splits:
##       saving         splits as  LLRRR,       agree=0.610, adj=0.148, (0 split)
##       credit_history splits as  LLLLR,       agree=0.594, adj=0.114, (0 split)
##       purpose        splits as  LRLLRRLLLL,  agree=0.576, adj=0.073, (0 split)
##       age            < 30.5    to the left,  agree=0.559, adj=0.036, (0 split)
##       mth_duration   < 10.5    to the right, agree=0.557, adj=0.032, (0 split)
##
## Node number 2: 488 observations,    complexity param=0.0411985
##   predicted class=good  expected loss=0.4446721  P(node) =0.5422222
##     class counts:   217   271
##    probabilities: 0.445 0.555
##   left son=4 (230 obs) right son=5 (258 obs)
##   Primary splits:
##       mth_duration   < 20.5    to the right, improve=9.258267, (0 missing)
##       credit_history splits as  LLRRR,       improve=8.405602, (0 missing)
##       property       splits as  RLLL,        improve=7.901285, (0 missing)
##       saving         splits as  LLRRR,       improve=6.821161, (0 missing)
##       other_debts    splits as  LLR,         improve=5.591242, (0 missing)
##   Surrogate splits:
##       credit_amount  < 2665    to the right, agree=0.752, adj=0.474, (0 split)
##       purpose        splits as  RLLRRRRLRL,  agree=0.633, adj=0.222, (0 split)
##       property       splits as  RRLL,        agree=0.627, adj=0.209, (0 split)
##       credit_history splits as  LLRLR,       agree=0.592, adj=0.135, (0 split)
##       housing        splits as  RRL,         agree=0.578, adj=0.104, (0 split)
##
## Node number 3: 412 observations
##   predicted class=good  expected loss=0.1213592  P(node) =0.4577778
##     class counts:    50   362
##    probabilities: 0.121 0.879
##
## Node number 4: 230 observations,    complexity param=0.0411985
##   predicted class=bad   expected loss=0.4521739  P(node) =0.2555556
##     class counts:   126   104
##    probabilities: 0.548 0.452
##   left son=8 (191 obs) right son=9 (39 obs)
##   Primary splits:
##       saving        splits as  LLLRR,       improve=6.634630, (0 missing)
```

```
##      pct_dpi       < 2.5     to the right, improve=3.763873, (0 missing)
##      credit_amount < 2165.5  to the left,  improve=3.720369, (0 missing)
##      purpose       splits as  LRLLLLLL-L,  improve=3.635620, (0 missing)
##      mth_duration  < 47.5    to the right, improve=2.538354, (0 missing)
##
## Node number 5: 258 observations,    complexity param=0.0411985
##   predicted class=good  expected loss=0.3527132  P(node) =0.2866667
##     class counts:    91   167
##    probabilities: 0.353 0.647
##   left son=10 (24 obs) right son=11 (234 obs)
##   Primary splits:
##       credit_history splits as  LLRRR,       improve=8.353210, (0 missing)
##       property       splits as  RLLL,        improve=6.056202, (0 missing)
##       purpose        splits as  LLRLRLLLRR,  improve=5.267674, (0 missing)
##       other_debts    splits as  LLR,         improve=4.117618, (0 missing)
##       credit_amount  < 7423    to the right, improve=3.661637, (0 missing)
##
## Node number 8: 191 observations,    complexity param=0.02247191
##   predicted class=bad   expected loss=0.3979058  P(node) =0.2122222
##     class counts:   115    76
##    probabilities: 0.602 0.398
##   left son=16 (34 obs) right son=17 (157 obs)
##   Primary splits:
##       mth_duration  < 47.5    to the right, improve=5.205473, (0 missing)
##       purpose       splits as  LRRLLLLL-L,  improve=3.964865, (0 missing)
##       pct_dpi       < 2.5     to the right, improve=2.692034, (0 missing)
##       credit_amount < 11788   to the right, improve=2.534153, (0 missing)
##       age           < 25.5    to the left,  improve=1.834814, (0 missing)
##   Surrogate splits:
##       credit_amount < 13319.5 to the right, agree=0.838, adj=0.088, (0 split)
##       purpose       splits as  RRRRRLRR-R,  agree=0.827, adj=0.029, (0 split)
##
## Node number 9: 39 observations,    complexity param=0.01498127
##   predicted class=good  expected loss=0.2820513  P(node) =0.04333333
##     class counts:    11    28
##    probabilities: 0.282 0.718
##   left son=18 (18 obs) right son=19 (21 obs)
##   Primary splits:
##       chk_status     splits as  LR--,        improve=7.239316, (0 missing)
##       credit_amount  < 2079    to the left,  improve=5.750427, (0 missing)
##       credit_history splits as  RLLRR,       improve=2.757835, (0 missing)
##       purpose        splits as  LRLRR--L-R,  improve=2.074872, (0 missing)
##       pct_dpi        < 2.5     to the right, improve=1.641026, (0 missing)
##   Surrogate splits:
##       credit_history splits as  LRLRR,       agree=0.692, adj=0.333, (0 split)
##       credit_amount  < 1681    to the left,  agree=0.692, adj=0.333, (0 split)
##       mth_duration   < 45      to the left,  agree=0.641, adj=0.222, (0 split)
##       purpose        splits as  LLRLR--R-R,  agree=0.641, adj=0.222, (0 split)
##       employ_time    splits as  RLRRL,       agree=0.641, adj=0.222, (0 split)
##
## Node number 10: 24 observations
##   predicted class=bad   expected loss=0.25  P(node) =0.02666667
##     class counts:    18     6
##    probabilities: 0.750 0.250
```

```
##
## Node number 11: 234 observations,    complexity param=0.01622971
##   predicted class=good  expected loss=0.3119658  P(node) =0.26
##     class counts:    73   161
##    probabilities: 0.312 0.688
##   left son=22 (141 obs) right son=23 (93 obs)
##   Primary splits:
##       property      splits as  RLLL,        improve=4.328535, (0 missing)
##       credit_amount < 7423    to the right, improve=4.289367, (0 missing)
##       purpose       splits as  LRRLRLLLLR,  improve=3.458028, (0 missing)
##       mth_duration  < 11.5    to the right, improve=3.385334, (0 missing)
##       other_debts   splits as  LLR,         improve=3.012226, (0 missing)
##   Surrogate splits:
##       purpose       splits as  LLLLRLLLLL, agree=0.667, adj=0.161, (0 split)
##       other_debts   splits as  LLR,        agree=0.667, adj=0.161, (0 split)
##       job           splits as  LRLL,       agree=0.658, adj=0.140, (0 split)
##       status_gender splits as  LLLR,       agree=0.632, adj=0.075, (0 split)
##       foreign       splits as  LR,         agree=0.620, adj=0.043, (0 split)
##
## Node number 16: 34 observations
##   predicted class=bad   expected loss=0.1470588  P(node) =0.03777778
##     class counts:    29     5
##    probabilities: 0.853 0.147
##
## Node number 17: 157 observations,    complexity param=0.02247191
##   predicted class=bad   expected loss=0.4522293  P(node) =0.1744444
##     class counts:    86    71
##    probabilities: 0.548 0.452
##   left son=34 (133 obs) right son=35 (24 obs)
##   Primary splits:
##       purpose       splits as  LRRLL-LL-L,  improve=5.024041, (0 missing)
##       credit_amount < 2313    to the left,  improve=3.205512, (0 missing)
##       pct_dpi       < 2.5     to the right, improve=2.543577, (0 missing)
##       employ_time   splits as  RLLRR,       improve=1.889131, (0 missing)
##       other_debts   splits as  LLR,         improve=1.789666, (0 missing)
##
## Node number 18: 18 observations
##   predicted class=bad   expected loss=0.3888889  P(node) =0.02
##     class counts:    11     7
##    probabilities: 0.611 0.389
##
## Node number 19: 21 observations
##   predicted class=good  expected loss=0  P(node) =0.02333333
##     class counts:     0    21
##    probabilities: 0.000 1.000
##
## Node number 22: 141 observations,    complexity param=0.01622971
##   predicted class=good  expected loss=0.3900709  P(node) =0.1566667
##     class counts:    55    86
##    probabilities: 0.390 0.610
##   left son=44 (57 obs) right son=45 (84 obs)
##   Primary splits:
##       credit_amount    < 1373    to the left,  improve=3.552098, (0 missing)
##       purpose          splits as  LRRRRLLLRR,  improve=3.186922, (0 missing)
```

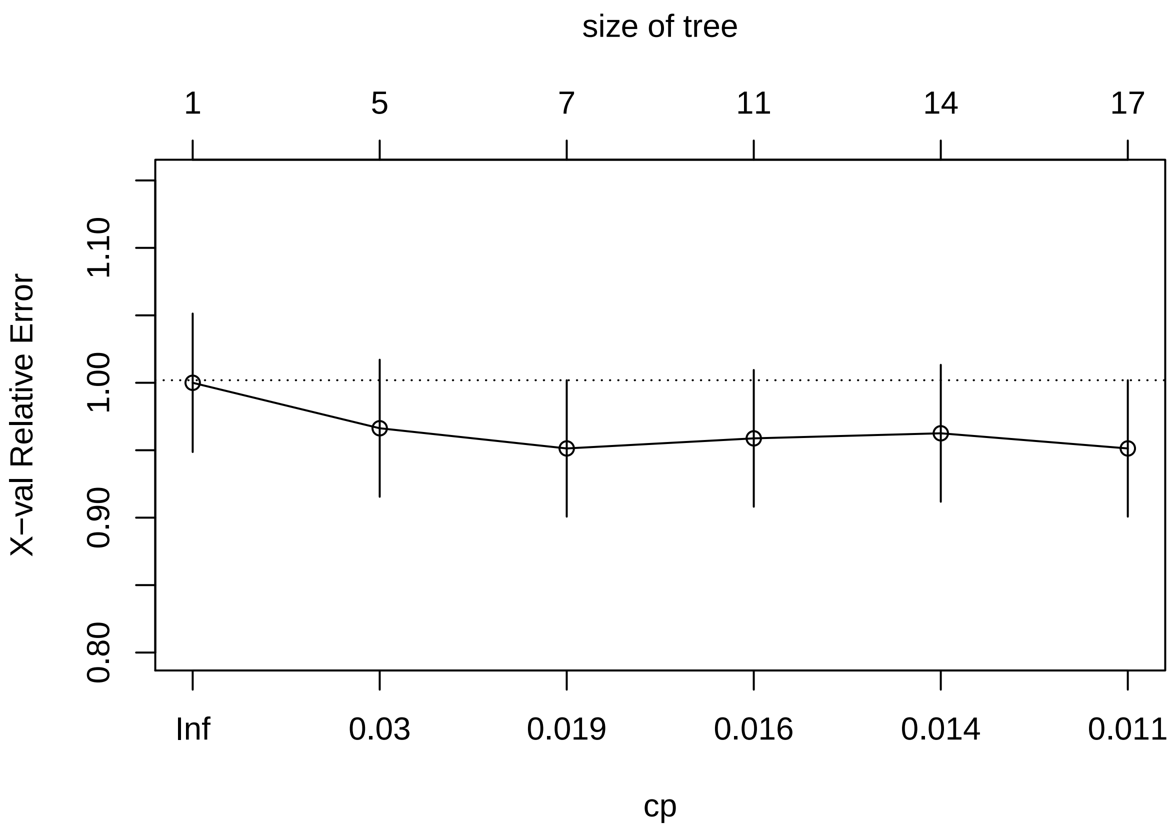
```
##      credit_history   splits as  --LRR,       improve=3.165402, (0 missing)
##      existing_credits < 1.5     to the left,  improve=2.509724, (0 missing)
##      employ_time      splits as  LLLRL,       improve=1.683934, (0 missing)
##  Surrogate splits:
##      purpose      splits as  RRRRRRLLLR,  agree=0.667, adj=0.175, (0 split)
##      mth_duration < 9.5     to the left,  agree=0.645, adj=0.123, (0 split)
##      pct_dpi      < 3.5     to the right, agree=0.631, adj=0.088, (0 split)
##      age          < 21.5    to the left,  agree=0.624, adj=0.070, (0 split)
##      phone        splits as  LR,          agree=0.624, adj=0.070, (0 split)
##
## Node number 23: 93 observations
##   predicted class=good  expected loss=0.1935484  P(node) =0.1033333
##     class counts:    18    75
##    probabilities: 0.194 0.806
##
## Node number 34: 133 observations,    complexity param=0.01310861
##   predicted class=bad   expected loss=0.3984962  P(node) =0.1477778
##     class counts:    80    53
##    probabilities: 0.602 0.398
##   left son=68 (36 obs) right son=69 (97 obs)
##   Primary splits:
##      credit_amount < 2313    to the left,  improve=2.176924, (0 missing)
##      pct_dpi       < 2.5     to the right, improve=2.136210, (0 missing)
##      job           splits as  RLRL,        improve=1.471917, (0 missing)
##      employ_time   splits as  LLLRR,       improve=1.155885, (0 missing)
##      housing       splits as  LRL,         improve=1.038691, (0 missing)
##   Surrogate splits:
##      foreign splits as  RL, agree=0.737, adj=0.028, (0 split)
##
## Node number 35: 24 observations
##   predicted class=good  expected loss=0.25  P(node) =0.02666667
##     class counts:     6    18
##    probabilities: 0.250 0.750
##
## Node number 44: 57 observations,    complexity param=0.01622971
##   predicted class=bad   expected loss=0.4736842  P(node) =0.06333333
##     class counts:    30    27
##    probabilities: 0.526 0.474
##   left son=88 (39 obs) right son=89 (18 obs)
##   Primary splits:
##      existing_credits < 1.5     to the left,  improve=4.865497, (0 missing)
##      age              < 37.5    to the left,  improve=4.704836, (0 missing)
##      property         splits as  -RLR,        improve=3.996735, (0 missing)
##      mth_duration     < 8.5     to the right, improve=3.932164, (0 missing)
##      purpose          splits as  LRRRRLRLRR,  improve=3.341053, (0 missing)
##   Surrogate splits:
##      credit_history splits as  --LLR,       agree=0.860, adj=0.556, (0 split)
##      property       splits as  -LLR,        agree=0.737, adj=0.167, (0 split)
##      age            < 50      to the left,  agree=0.737, adj=0.167, (0 split)
##      housing        splits as  LLR,         agree=0.737, adj=0.167, (0 split)
##      purpose        splits as  LLRLLLLLRL,  agree=0.719, adj=0.111, (0 split)
##
## Node number 45: 84 observations,    complexity param=0.01622971
##   predicted class=good  expected loss=0.297619  P(node) =0.09333333
```

```
##     class counts:    25    59
##    probabilities: 0.298 0.702
##   left son=90 (7 obs) right son=91 (77 obs)
##   Primary splits:
##       credit_amount < 7341    to the right, improve=4.781385, (0 missing)
##       employ_time   splits as  RRRRL,       improve=3.886659, (0 missing)
##       age           < 36.5    to the right, improve=3.440476, (0 missing)
##       property      splits as  -RRL,        improve=2.519048, (0 missing)
##       phone         splits as  RL,          improve=1.682859, (0 missing)
##
## Node number 68: 36 observations
##   predicted class=bad   expected loss=0.25  P(node) =0.04
##     class counts:    27     9
##    probabilities: 0.750 0.250
##
## Node number 69: 97 observations,    complexity param=0.01310861
##   predicted class=bad   expected loss=0.4536082  P(node) =0.1077778
##     class counts:    53    44
##    probabilities: 0.546 0.454
##   left son=138 (48 obs) right son=139 (49 obs)
##   Primary splits:
##       credit_amount      < 3962    to the right, improve=2.749141, (0 missing)
##       pct_dpi            < 2.5     to the right, improve=2.056500, (0 missing)
##       phone              splits as  RL,          improve=1.378578, (0 missing)
##       credit_history     splits as  LRRRR,       improve=1.062272, (0 missing)
##       other_installments splits as  LRL,         improve=0.900656, (0 missing)
##   Surrogate splits:
##       credit_history   splits as  LRRLL,       agree=0.691, adj=0.375, (0 split)
##       mth_duration     < 27.5    to the right, agree=0.660, adj=0.312, (0 split)
##       pct_dpi          < 3.5     to the left,  agree=0.649, adj=0.292, (0 split)
##       existing_credits < 1.5     to the right, agree=0.629, adj=0.250, (0 split)
##       purpose          splits as  L--RR-RL-L,  agree=0.608, adj=0.208, (0 split)
##
## Node number 88: 39 observations,    complexity param=0.01498127
##   predicted class=bad   expected loss=0.3333333  P(node) =0.04333333
##     class counts:    26    13
##    probabilities: 0.667 0.333
##   left son=176 (18 obs) right son=177 (21 obs)
##   Primary splits:
##       purpose      splits as  LR-RRLRLLR,  improve=5.158730, (0 missing)
##       property     splits as  -RLR,        improve=4.470175, (0 missing)
##       mth_duration < 7.5     to the right, improve=2.476190, (0 missing)
##       saving       splits as  LLR-R,       improve=1.712366, (0 missing)
##       employ_time  splits as  LLLLR,       improve=1.712366, (0 missing)
##   Surrogate splits:
##       employ_time        splits as  RLRLR,       agree=0.641, adj=0.222, (0 split)
##       age                < 22.5    to the right, agree=0.641, adj=0.222, (0 split)
##       pct_dpi            < 3.5     to the right, agree=0.615, adj=0.167, (0 split)
##       property           splits as  -RLL,        agree=0.615, adj=0.167, (0 split)
##       other_installments splits as  LRR,         agree=0.615, adj=0.167, (0 split)
##
## Node number 89: 18 observations
##   predicted class=good  expected loss=0.2222222  P(node) =0.02
##     class counts:     4    14
```

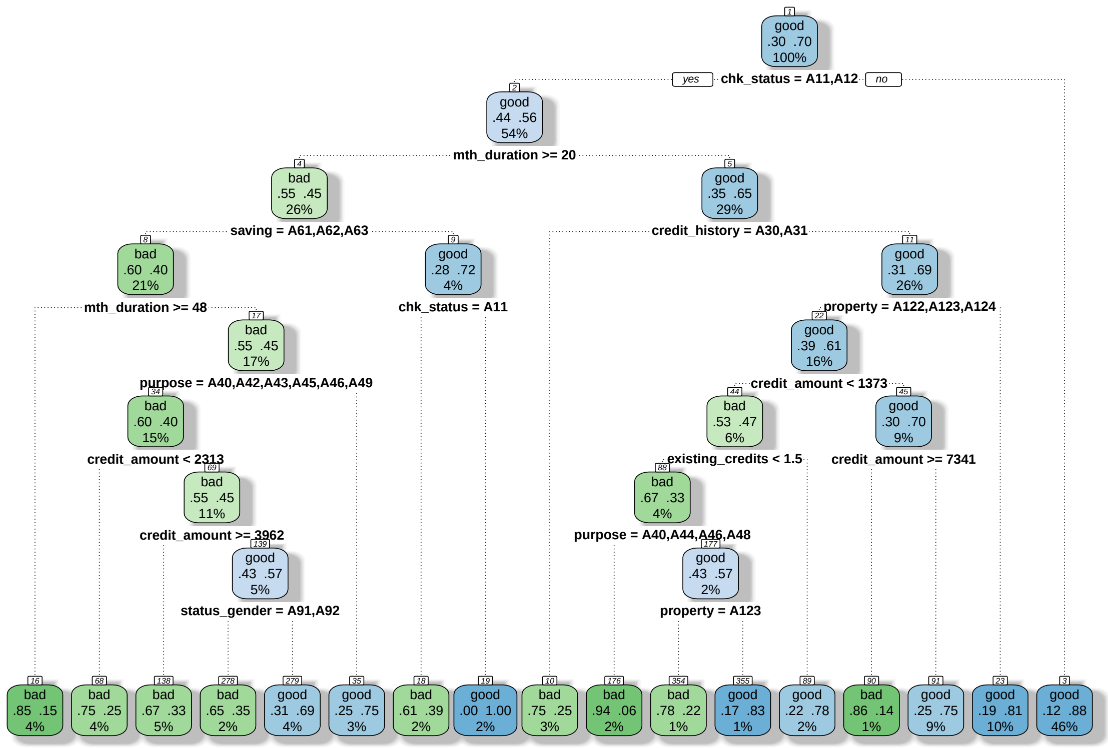
```
##    probabilities: 0.222 0.778
##
## Node number 90: 7 observations
##   predicted class=bad   expected loss=0.1428571  P(node) =0.007777778
##     class counts:     6     1
##    probabilities: 0.857 0.143
##
## Node number 91: 77 observations
##   predicted class=good  expected loss=0.2467532  P(node) =0.08555556
##     class counts:    19    58
##    probabilities: 0.247 0.753
##
## Node number 138: 48 observations
##   predicted class=bad   expected loss=0.3333333  P(node) =0.05333333
##     class counts:    32    16
##    probabilities: 0.667 0.333
##
## Node number 139: 49 observations,    complexity param=0.01310861
##   predicted class=good  expected loss=0.4285714  P(node) =0.05444444
##     class counts:    21    28
##    probabilities: 0.429 0.571
##   left son=278 (17 obs) right son=279 (32 obs)
##   Primary splits:
##       status_gender      splits as  LLRR,        improve=2.485294, (0 missing)
##       housing            splits as  LRR,         improve=2.461538, (0 missing)
##       pct_dpi            < 2.5     to the right, improve=2.258824, (0 missing)
##       employ_time        splits as  RLLRR,       improve=1.939394, (0 missing)
##       other_installments splits as  RRL,         improve=1.727273, (0 missing)
##   Surrogate splits:
##       pct_dpi        < 1.5     to the left,  agree=0.694, adj=0.118, (0 split)
##       age            < 21.5    to the left,  agree=0.694, adj=0.118, (0 split)
##       mth_duration   < 22.5    to the left,  agree=0.673, adj=0.059, (0 split)
##       credit_history splits as  LRRRR,       agree=0.673, adj=0.059, (0 split)
##       purpose        splits as  L--RR-RR-R,  agree=0.673, adj=0.059, (0 split)
##
## Node number 176: 18 observations
##   predicted class=bad   expected loss=0.05555556  P(node) =0.02
##     class counts:    17     1
##    probabilities: 0.944 0.056
##
## Node number 177: 21 observations,    complexity param=0.01498127
##   predicted class=good  expected loss=0.4285714  P(node) =0.02333333
##     class counts:     9    12
##    probabilities: 0.429 0.571
##   left son=354 (9 obs) right son=355 (12 obs)
##   Primary splits:
##       property       splits as  -RL-,        improve=3.8412700, (0 missing)
##       purpose        splits as  -R-RL-L--R,  improve=1.1220780, (0 missing)
##       employ_time    splits as  RLLLR,       improve=0.8241758, (0 missing)
##       residency_time < 3.5     to the left,  improve=0.8241758, (0 missing)
##       chk_status     splits as  RL--,        improve=0.5079365, (0 missing)
##   Surrogate splits:
##       chk_status    splits as  RL--,        agree=0.810, adj=0.556, (0 split)
##       age           < 25      to the right, agree=0.714, adj=0.333, (0 split)
```

```
##       job           splits as  RRL-,        agree=0.714, adj=0.333, (0 split)
##       purpose       splits as  -R-RL-R--R,  agree=0.667, adj=0.222, (0 split)
##       credit_amount < 755     to the right, agree=0.667, adj=0.222, (0 split)
##
## Node number 278: 17 observations
##   predicted class=bad   expected loss=0.3529412  P(node) =0.01888889
##     class counts:    11     6
##    probabilities: 0.647 0.353
##
## Node number 279: 32 observations
##   predicted class=good  expected loss=0.3125  P(node) =0.03555556
##     class counts:    10    22
##    probabilities: 0.312 0.688
##
## Node number 354: 9 observations
##   predicted class=bad   expected loss=0.2222222  P(node) =0.01
##     class counts:     7     2
##    probabilities: 0.778 0.222
##
## Node number 355: 12 observations
##   predicted class=good  expected loss=0.1666667  P(node) =0.01333333
##     class counts:     2    10
##    probabilities: 0.167 0.833
```

```
# plot the cost complexity parameters
plotcp(CreditData_model)
```

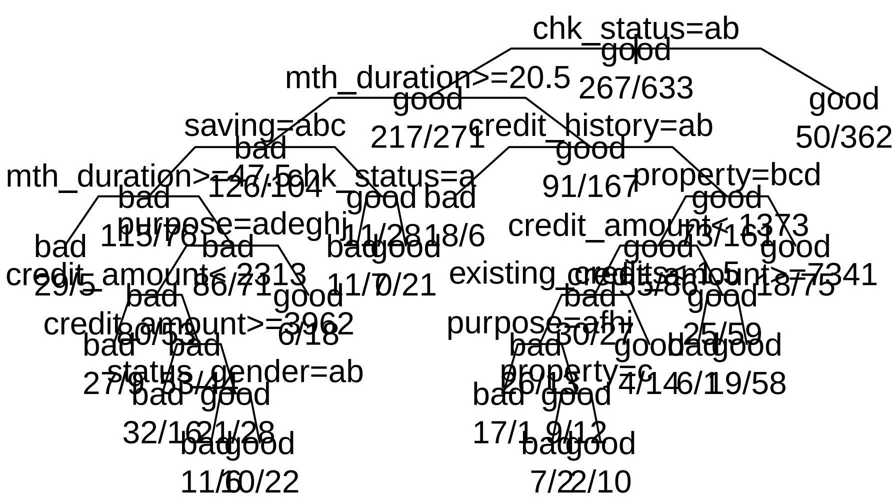



```
# Visualizing Decision Trees
fancyRpartPlot(CreditData_model)
```



Rattle 2017−Jun−05 22:20:29 ai

```r
# Visualize the classification tree
plot(CreditData_model, uniform=TRUE, branch=0.6, margin=0.1)
text(CreditData_model, all=TRUE, use.n = TRUE)
```



```r
# Model Evaluation using test data
CreditData_predict <- predict(CreditData_model, CreditData_test, type="class")

# Use the table function to generate a classification table for testing dataset
table(CreditData_test$class, CreditData_predict)
```

```
##         CreditData_predict
##          bad good
```

```
##   bad    9   24
##   good   9   58
```

```r
# Accuracy : Measures of performance
library(caret)
```

```
## Loading required package: lattice
```

```r
confusionMatrix(table(CreditData_predict, CreditData_test$class))
```

```
## Confusion Matrix and Statistics
##
##
## CreditData_predict bad good
##               bad    9    9
##               good  24   58
##
##                Accuracy : 0.67
##                  95% CI : (0.5688, 0.7608)
##     No Information Rate : 0.67
##     P-Value [Acc > NIR] : 0.54705
##
##                   Kappa : 0.1564
##  Mcnemar's Test P-Value : 0.01481
##
##             Sensitivity : 0.2727
##             Specificity : 0.8657
##          Pos Pred Value : 0.5000
##          Neg Pred Value : 0.7073
##              Prevalence : 0.3300
##          Detection Rate : 0.0900
##    Detection Prevalence : 0.1800
##       Balanced Accuracy : 0.5692
##
##        'Positive' Class : bad
##
```

```r
# Pruning a recursive partitioning tree
# Find minimum cross-validation error of the classification tree model
min(CreditData_model$cptable[,"xerror"])
```

```
## [1] 0.9513109
```

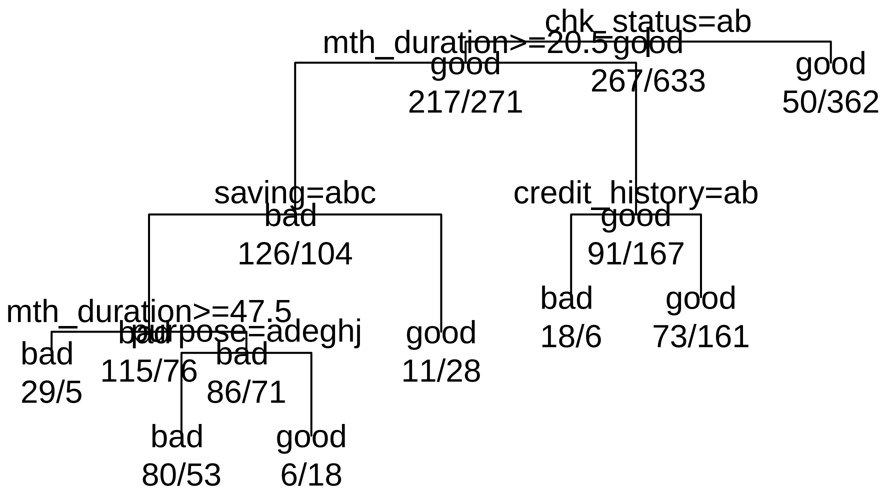
```r
# Locate the record with the minimum cross-validation errors
value = which.min(CreditData_model$cptable[,"xerror"])

# Get the cost complexity parameter of the record with the minimum cross-validation errors
CreditData_model_CP = CreditData_model$cptable[value, "CP"]
CreditData_model_CP
```

```
## [1] 0.01622971
```

```r
# Prune the tree by setting the cp parameter to the CP value of the record with minimum cross-validatio
prune_tree = prune(CreditData_model, cp=CreditData_model_CP)

# Visualize the classification tree by using the plot and text function
plot(prune_tree, margin=0.1)
text(prune_tree, all=TRUE, use.n=TRUE)
```




```r
# Generate a classification table based on the pruned classification tree model
predictions = predict(prune_tree, CreditData_test, type="class")
table(CreditData_test$class, predictions)
```

```
##       predictions
##        bad good
##   bad   11   22
##   good   4   63
```

```r
# Generate confusion matrix
confusionMatrix(table(predictions, CreditData_test$class))
```

```
## Confusion Matrix and Statistics
##
##
## predictions bad good
##        bad   11    4
##        good  22   63
##
##                Accuracy : 0.74
##                  95% CI : (0.6427, 0.8226)
##     No Information Rate : 0.67
##     P-Value [Acc > NIR] : 0.0814644
##
##                   Kappa : 0.3176
##  Mcnemar's Test P-Value : 0.0008561
##
##             Sensitivity : 0.3333
##             Specificity : 0.9403
##          Pos Pred Value : 0.7333
##          Neg Pred Value : 0.7412
##              Prevalence : 0.3300
##          Detection Rate : 0.1100
##    Detection Prevalence : 0.1500
##       Balanced Accuracy : 0.6368
##
##        'Positive' Class : bad
##
```

```r
# # Model Improvement using M5P from RWeka
# # Build Model
# WineData_model_M5P <- M5P(quality ~. , data= WineData_train)

# # Model Evaluation using test data
# WineData_predict_M5P <- predict(WineData_model_M5P, WineData_test)

# WineData_model_M5P

# MAE(WineData_test$quality, WineData_predict_M5P)
```