M6_L2_RomilShah

Romil Shah

July 8, 2016

Read Data and additional packages

```
require(ggplot2)

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.2.5

require(C50)

## Loading required package: C50

## Warning: package 'C50' was built under R version 3.2.5

require(gmodels)

## Loading required package: gmodels

## Warning: package 'gmodels' was built under R version 3.2.5

require(rpart)

## Loading required package: rpart

## Warning: package 'rpart' was built under R version 3.2.5

require(rattle)

## Loading required package: rattle

## Warning: package 'rattle' was built under R version 3.2.5

## Rattle: A free graphical interface for data mining with R.
## Version 4.1.0 Copyright (c) 2006-2015 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.

require(RColorBrewer)

## Loading required package: RColorBrewer

require(tree)

## Loading required package: tree

## Warning: package 'tree' was built under R version 3.2.5

require(party)
```

```
## Loading required package: party

## Warning: package 'party' was built under R version 3.2.5

## Loading required package: grid

## Loading required package: mvtnorm

## Loading required package: modeltools

## Loading required package: stats4

## Loading required package: strucchange

## Warning: package 'strucchange' was built under R version 3.2.5

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.2.5

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Warning: package 'sandwich' was built under R version 3.2.5
```

```
#Chess (King-Rook vs. King-Pawn) Data Set

data_url <- 'https://archive.ics.uci.edu/ml/machine-learning-
databases/chess/king-rook-vs-king-pawn/kr-vs-kp.data'
chess <- read.csv(url(data_url))
chess <- chess[]
head(chess)
```

```
##   f f.1 f.2 f.3 f.4 f.5 f.6 f.7 f.8 f.9 f.10 f.11 l f.12 n f.13 f.14 t
## 1 f   f   f   f   t   f   f   f   f   f    f    f l    f n    f    f t
## 2 f   f   f   f   t   f   t   f   f   f    f    f l    f n    f    f t
## 3 f   f   f   f   f   f   f   f   t   f    f    f l    f n    f    f t
## 4 f   f   f   f   f   f   f   f   f   f    f    f l    f n    f    f t
## 5 f   f   f   f   f   f   f   f   f   f    f    f l    f n    f    f t
## 6 f   f   f   f   f   f   f   f   t   f    f    f l    f n    f    f t
##   f.15 f.16 f.17 f.18 f.19 f.20 f.21 t.1 f.22 f.23 f.24 f.25 f.26 f.27
## 1    f    f    f    f    f    f    f   t    f    f    f    f    f    f
## 2    f    f    f    f    f    f    f   t    f    f    f    f    f    f
## 3    f    f    f    t    f    f    f   t    f    f    f    f    f    f
## 4    f    f    f    f    f    t    f   t    f    f    f    f    f    f
## 5    f    f    f    f    f    t    f   t    f    f    f    t    f    f
## 6    f    f    f    t    f    t    f   t    f    f    f    f    f    f
##   f.28 t.2 t.3 n.1 won
```

```
## 1    f   t   t   n won
## 2    f   t   t   n won
## 3    f   t   t   n won
## 4    f   t   t   n won
## 5    f   t   t   n won
## 6    f   t   t   n won
```

```r
str(chess)
```

```
## 'data.frame':    3195 obs. of  37 variables:
##  $ f   : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.1 : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.2 : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.3 : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.4 : Factor w/ 2 levels "f","t": 2 2 1 1 1 1 2 1 1 2 ...
##  $ f.5 : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.6 : Factor w/ 2 levels "f","t": 1 2 1 1 1 1 1 1 2 2 ...
##  $ f.7 : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.8 : Factor w/ 2 levels "f","t": 1 1 2 1 1 2 1 1 1 1 ...
##  $ f.9 : Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.10: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 2 2 2 2 ...
##  $ f.11: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ l   : Factor w/ 2 levels "g","l": 2 2 2 2 2 2 2 2 2 2 ...
##  $ f.12: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ n   : Factor w/ 3 levels "b","n","w": 2 2 2 2 2 2 2 2 2 2 ...
##  $ f.13: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.14: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ t   : Factor w/ 2 levels "f","t": 2 2 2 2 2 2 2 2 2 2 ...
##  $ f.15: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.16: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.17: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.18: Factor w/ 2 levels "f","t": 1 1 2 1 1 2 1 1 1 1 ...
##  $ f.19: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.20: Factor w/ 2 levels "f","t": 1 1 1 2 2 2 1 1 1 1 ...
##  $ f.21: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ t.1 : Factor w/ 2 levels "f","t": 2 2 2 2 2 2 2 2 2 2 ...
##  $ f.22: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.23: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.24: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.25: Factor w/ 2 levels "f","t": 1 1 1 1 2 1 1 1 1 1 ...
##  $ f.26: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 2 1 1 ...
##  $ f.27: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ f.28: Factor w/ 2 levels "f","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ t.2 : Factor w/ 2 levels "f","t": 2 2 2 2 2 2 2 2 2 2 ...
##  $ t.3 : Factor w/ 2 levels "f","t": 2 2 2 2 2 2 2 2 2 2 ...
##  $ n.1 : Factor w/ 2 levels "n","t": 1 1 1 1 1 1 1 1 1 1 ...
##  $ won : Factor w/ 2 levels "nowin","won": 2 2 2 2 2 2 2 2 2 2 ...
```

```r
table(chess$f)
```

```
##
##    f    t
## 2838  357

#Random sample
set.seed(12345)
chess_rand <- chess[order(runif(3195)), ]

#Comparing samples
summary(chess$won)

## nowin   won
##  1527  1668

summary(chess_rand$won)

## nowin   won
##  1527  1668

head(chess$won)

## [1] won won won won won won
## Levels: nowin won

head(chess_rand$won)

## [1] nowin nowin nowin won   nowin nowin
## Levels: nowin won

chess_train <- chess_rand[1:3000,-38]
chess_test <- chess_rand[3001:3195, ]

prop.table(table(chess_train$f))

##
##         f         t
## 0.8863333 0.1136667

prop.table(table(chess_test$f))

##
##          f          t
## 0.91794872 0.08205128

#Training data
model <- C5.0(chess_train[-1], chess_train$f)
model

##
## Call:
## C5.0.default(x = chess_train[-1], y = chess_train$f)
##
## Classification Tree
## Number of samples: 3000
```

```
## Number of predictors: 36
##
## Tree size: 32
##
## Non-standard options: attempt to group attributes
```

```r
summary(model)
```

```
##
## Call:
## C5.0.default(x = chess_train[-1], y = chess_train$f)
##
##
## C5.0 [Release 2.07 GPL Edition]      Fri Jul 08 14:34:06 2016
## -------------------------------
##
## Class specified by attribute `outcome'
##
## Read 3000 cases (37 attributes) from undefined.data
##
## Decision tree:
##
## f.12 = t: t (14)
## f.12 = f:
## :...n = b:
##     :...t.2 = f: t (57)
##     :   t.2 = t:
##     :   :...won = won: t (42/7)
##     :       won = nowin:
##     :       :...f.11 = t: t (6)
##     :           f.11 = f:
##     :           :...f.25 = f: f (97/7)
##     :               f.25 = t: t (2)
##     n in {n,w}:
##     :...f.10 = f: f (1670/13)
##         f.10 = t:
##         :...f.2 = t:
##             :...f.28 = f: f (3)
##             :   f.28 = t: t (24)
##             f.2 = f:
##             :...f.7 = t:
##                 :...t.2 = t: f (135/10)
##                 :   t.2 = f:
##                 :   :...f.6 = f: f (8)
##                 :       f.6 = t:
##                 :       :...f.9 = t:
##                 :           :...f.17 = f: t (58)
##                 :           :   f.17 = t: f (2)
##                 :           f.9 = f:
##                 :           :...f.4 = f: f (31)
```

```
##                 :               f.4 = t:
##                 :               :...f.5 = f: f (5)
##                 :                   f.5 = t: t (14)
##                 f.7 = f:
##                 :...f.9 = t: f (191)
##                     f.9 = f:
##                     :...t = f:
##                         :...f.3 = f: f (232/5)
##                         :   f.3 = t:
##                         :   :...f.28 = f: f (16)
##                         :       f.28 = t:
##                         :       :...n = n: f (10/2)
##                         :           n = w: t (4)
##                         t = t:
##                         :...f.13 = t: f (12)
##                             f.13 = f:
##                             :...f.8 = t:
##                                 :...f.4 = f: f (97)
##                                 :   f.4 = t:
##                                 :   :...f.5 = f: f (28/2)
##                                 :       f.5 = t:
##                                 :       :...f.16 = f: t (15/3)
##                                 :           f.16 = t: f (2)
##                                 f.8 = f:
##                                 :...won = nowin:
##                                     :...t.2 = f: t (2)
##                                     :   t.2 = t: f (50/4)
##                                     won = won:
##                                     :...f.22 = t: t (3)
##                                         f.22 = f:
##                                         :...f.14 = t: f (16/1)
##                                             f.14 = f:
##                                             :...f.20 = f: f (90/24)
##                                                 f.20 = t: t (64/22)
##
##
## Evaluation on training data (3000 cases):
##
##      Decision Tree
##    ----------------
##    Size      Errors
##
##      32  100( 3.3%)   <<
##
##
##     (a)   (b)    <-classified as
##    ----  ----
##    2627    32    (a): class f
##      68   273    (b): class t
##
```

```
##
##  Attribute usage:
##
##  100.00% f.12
##   99.53% n
##   92.73% f.10
##   37.07% f.2
##   36.17% f.7
##   31.40% f.9
##   21.37% t
##   16.97% t.2
##   12.63% f.13
##   12.40% won
##   12.23% f.8
##    8.73% f.3
##    6.40% f.4
##    5.77% f.22
##    5.67% f.14
##    5.13% f.20
##    3.93% f.6
##    3.50% f.11
##    3.30% f.25
##    2.13% f.5
##    2.00% f.17
##    1.90% f.28
##    0.57% f.16
##
##
## Time: 0.0 secs
```

```r
chess_type_pred <- predict(model, chess_test)

CrossTable(chess_test$f, chess_type_pred, prop.chisq = FALSE, prop.c = FALSE,
prop.r = FALSE, dnn = c('actual type', 'predicted type'))
```

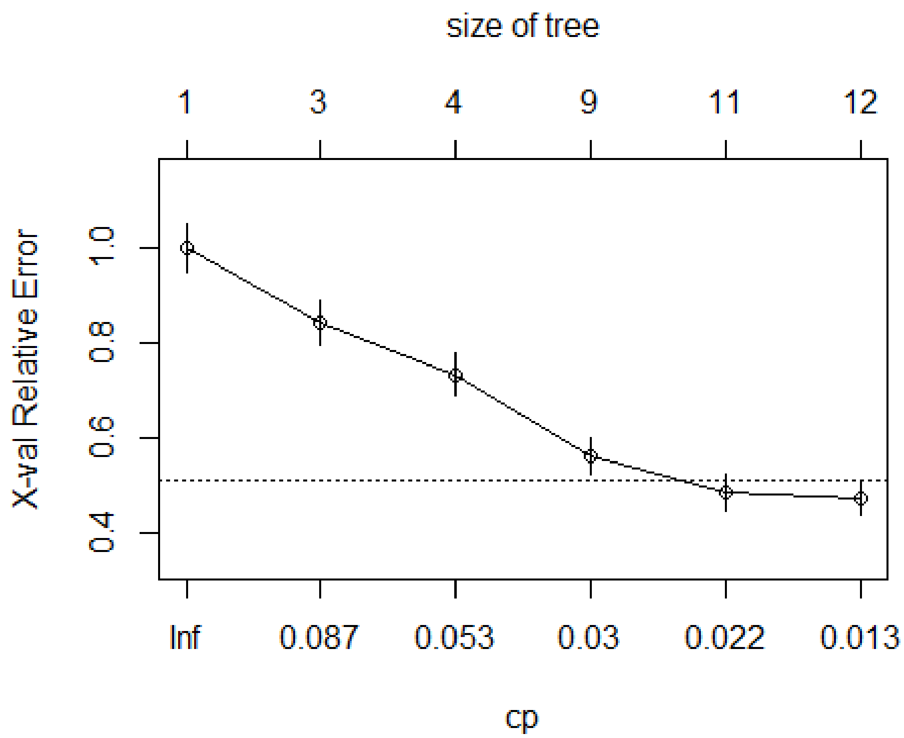
```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  195
##
##
##              | predicted type
##  actual type |         f |         t | Row Total |
## -------------|-----------|-----------|-----------|
```

```
##             f |       174 |         5 |       179 |
##               |     0.892 |     0.026 |           |
## -------------|-----------|-----------|-----------|
##             t |         3 |        13 |        16 |
##               |     0.015 |     0.067 |           |
## -------------|-----------|-----------|-----------|
## Column Total |       177 |        18 |       195 |
## -------------|-----------|-----------|-----------|
##
##
```

```
formula <- f ~
f.1+f.2+f.3+f.4+f.5+f.6+f.7+f.8+f.9+f.10+f.11+l+f.12+n+f.13+f.14+t+f.15+f.16+
f.17+f.18+f.19+f.20+f.21+t.1+f.22+f.23+f.24+f.25+f.26+f.27+f.28+t.2+t.3+n.1+w
on

fit <- rpart(formula, method="class", data=chess_train)
printcp(fit)
```

```
##
## Classification tree:
## rpart(formula = formula, data = chess_train, method = "class")
##
## Variables actually used in tree construction:
##  [1] f.10 f.2  f.24 f.4  f.5  f.7  f.9  n    t.2  won
##
## Root node error: 341/3000 = 0.11367
##
## n= 3000
##
##         CP nsplit rel error  xerror     xstd
## 1 0.092375      0   1.00000 1.00000 0.050983
## 2 0.082111      2   0.81525 0.84164 0.047244
## 3 0.033724      3   0.73314 0.73314 0.044394
## 4 0.026393      8   0.52493 0.56305 0.039313
## 5 0.017595     10   0.47214 0.48680 0.036723
## 6 0.010000     11   0.45455 0.47507 0.036304
```

```
plotcp(fit)
```




```
summary(fit)

## Call:
## rpart(formula = formula, data = chess_train, method = "class")
##   n= 3000
##
##           CP nsplit rel error    xerror       xstd
## 1 0.09237537      0 1.0000000 1.0000000 0.05098253
## 2 0.08211144      2 0.8152493 0.8416422 0.04724445
## 3 0.03372434      3 0.7331378 0.7331378 0.04439370
## 4 0.02639296      8 0.5249267 0.5630499 0.03931282
## 5 0.01759531     10 0.4721408 0.4868035 0.03672307
## 6 0.01000000     11 0.4545455 0.4750733 0.03630352
##
## Variable importance
##    n  t.2 f.10  won  f.2  f.5    t  f.4  f.9  t.3 f.26  f.7  t.1  f.3 f.16
##   17   14    8    7    7    6    6    5    3    3    3    3    3    2    2
## f.24 f.20  f.1 f.22 f.11 f.13  f.8 f.19
##    2    1    1    1    1    1    1    1
##
## Node number 1: 3000 observations,    complexity param=0.09237537
##   predicted class=f  expected loss=0.1136667  P(node) =1
##     class counts:  2659   341
##    probabilities: 0.886 0.114
##   left son=2 (2790 obs) right son=3 (210 obs)
##   Primary splits:
##       n    splits as  RLL, improve=81.35337, (0 missing)
```

```
##       f.10 splits as  LR,  improve=67.25108, (0 missing)
##       t.1  splits as  LR,  improve=34.77388, (0 missing)
##       t.3  splits as  RL,  improve=26.68382, (0 missing)
##       n.1  splits as  RL,  improve=25.47402, (0 missing)
##   Surrogate splits:
##       f.23 splits as  LR, agree=0.93, adj=0.005, (0 split)
##
## Node number 2: 2790 observations,    complexity param=0.03372434
##   predicted class=f  expected loss=0.08172043  P(node) =0.93
##     class counts:  2562   228
##    probabilities: 0.918 0.082
##   left son=4 (1674 obs) right son=5 (1116 obs)
##   Primary splits:
##       f.10 splits as  LR, improve=42.86750, (0 missing)
##       t.3  splits as  RL, improve=21.58485, (0 missing)
##       t.1  splits as  LR, improve=18.60222, (0 missing)
##       f.12 splits as  LR, improve=13.53060, (0 missing)
##       n.1  splits as  RL, improve=13.50111, (0 missing)
##   Surrogate splits:
##       t.3  splits as  RL,  agree=0.766, adj=0.416, (0 split)
##       f.26 splits as  LR,  agree=0.759, adj=0.398, (0 split)
##       t.1  splits as  LR,  agree=0.733, adj=0.332, (0 split)
##       n    splits as  -LR, agree=0.726, adj=0.315, (0 split)
##       f.16 splits as  LR,  agree=0.711, adj=0.277, (0 split)
##
## Node number 3: 210 observations,    complexity param=0.09237537
##   predicted class=t  expected loss=0.4619048  P(node) =0.07
##     class counts:    97   113
##    probabilities: 0.462 0.538
##   left son=6 (147 obs) right son=7 (63 obs)
##   Primary splits:
##       t.2  splits as  RL, improve=38.404080, (0 missing)
##       won  splits as  LR, improve=29.142890, (0 missing)
##       f.9  splits as  RL, improve=14.849110, (0 missing)
##       t    splits as  RL, improve=11.021490, (0 missing)
##       f.20 splits as  LR, improve= 6.192278, (0 missing)
##   Surrogate splits:
##       t    splits as  RL, agree=0.800, adj=0.333, (0 split)
##       f.2  splits as  LR, agree=0.767, adj=0.222, (0 split)
##       f.1  splits as  LR, agree=0.757, adj=0.190, (0 split)
##       f.3  splits as  LR, agree=0.757, adj=0.190, (0 split)
##       f.22 splits as  LR, agree=0.748, adj=0.159, (0 split)
##
## Node number 4: 1674 observations
##   predicted class=f  expected loss=0.01015532  P(node) =0.558
##     class counts:  1657    17
##    probabilities: 0.990 0.010
##
## Node number 5: 1116 observations,    complexity param=0.03372434
##   predicted class=f  expected loss=0.1890681  P(node) =0.372
```

```
##     class counts:   905   211
##    probabilities: 0.811 0.189
##   left son=10 (1087 obs) right son=11 (29 obs)
##   Primary splits:
##       f.2  splits as  LR, improve=29.805430, (0 missing)
##       f.7  splits as  LR, improve=19.071620, (0 missing)
##       f.22 splits as  LR, improve=12.318100, (0 missing)
##       t.2  splits as  RL, improve=11.110070, (0 missing)
##       f.4  splits as  LR, improve= 9.720311, (0 missing)
##
## Node number 6: 147 observations,    complexity param=0.08211144
##   predicted class=f  expected loss=0.3401361  P(node) =0.049
##     class counts:    97    50
##    probabilities: 0.660 0.340
##   left son=12 (105 obs) right son=13 (42 obs)
##   Primary splits:
##       won  splits as  LR, improve=28.605440, (0 missing)
##       f.9  splits as  RL, improve=12.236610, (0 missing)
##       f.28 splits as  RL, improve= 6.970001, (0 missing)
##       f.24 splits as  LR, improve= 4.538219, (0 missing)
##       f.5  splits as  LR, improve= 3.574946, (0 missing)
##   Surrogate splits:
##       f.5  splits as  LR, agree=0.789, adj=0.262, (0 split)
##       f.25 splits as  LR, agree=0.721, adj=0.024, (0 split)
##
## Node number 7: 63 observations
##   predicted class=t  expected loss=0  P(node) =0.021
##     class counts:     0    63
##    probabilities: 0.000 1.000
##
## Node number 10: 1087 observations,    complexity param=0.03372434
##   predicted class=f  expected loss=0.1701932  P(node) =0.3623333
##     class counts:   902   185
##    probabilities: 0.830 0.170
##   left son=20 (834 obs) right son=21 (253 obs)
##   Primary splits:
##       f.7  splits as  LR, improve=15.623920, (0 missing)
##       f.4  splits as  LR, improve=13.523980, (0 missing)
##       f.6  splits as  LR, improve= 6.653843, (0 missing)
##       f.17 splits as  RL, improve= 5.669950, (0 missing)
##       t    splits as  LR, improve= 5.527543, (0 missing)
##   Surrogate splits:
##       f.8 splits as  LR, agree=0.824, adj=0.245, (0 split)
##
## Node number 11: 29 observations
##   predicted class=t  expected loss=0.1034483  P(node) =0.009666667
##     class counts:     3    26
##    probabilities: 0.103 0.897
##
## Node number 12: 105 observations,    complexity param=0.01759531
```

```
##   predicted class=f  expected loss=0.1428571  P(node) =0.035
##     class counts:    90    15
##    probabilities: 0.857 0.143 
##   left son=24 (95 obs) right son=25 (10 obs)
##   Primary splits:
##       f.24 splits as  LR, improve=9.545865, (0 missing)
##       f.19 splits as  LR, improve=6.861654, (0 missing)
##       f.10 splits as  RL, improve=5.427041, (0 missing)
##       f.27 splits as  LR, improve=5.427041, (0 missing)
##       t.3  splits as  RL, improve=1.913265, (0 missing)
##   Surrogate splits:
##       f.11 splits as  LR, agree=0.962, adj=0.6, (0 split)
##       f.19 splits as  LR, agree=0.943, adj=0.4, (0 split)
##       t    splits as  RL, agree=0.914, adj=0.1, (0 split)
## 
## Node number 13: 42 observations
##   predicted class=t  expected loss=0.1666667  P(node) =0.014
##     class counts:     7    35
##    probabilities: 0.167 0.833 
## 
## Node number 20: 834 observations
##   predicted class=f  expected loss=0.1235012  P(node) =0.278
##     class counts:   731   103
##    probabilities: 0.876 0.124 
## 
## Node number 21: 253 observations,    complexity param=0.03372434
##   predicted class=f  expected loss=0.3241107  P(node) =0.08433333
##     class counts:   171    82
##    probabilities: 0.676 0.324 
##   left son=42 (135 obs) right son=43 (118 obs)
##   Primary splits:
##       t.2 splits as  RL, improve=36.191740, (0 missing)
##       f.4 splits as  LR, improve=28.761910, (0 missing)
##       f.5 splits as  RL, improve= 8.513530, (0 missing)
##       f.9 splits as  LR, improve= 8.291211, (0 missing)
##       f.6 splits as  LR, improve= 6.881564, (0 missing)
##   Surrogate splits:
##       t    splits as  RL, agree=0.743, adj=0.449, (0 split)
##       f.4  splits as  LR, agree=0.688, adj=0.331, (0 split)
##       f.3  splits as  LR, agree=0.593, adj=0.127, (0 split)
##       f.13 splits as  LR, agree=0.593, adj=0.127, (0 split)
##       f.20 splits as  LR, agree=0.585, adj=0.110, (0 split)
## 
## Node number 24: 95 observations
##   predicted class=f  expected loss=0.07368421  P(node) =0.03166667
##     class counts:    88     7
##    probabilities: 0.926 0.074 
## 
## Node number 25: 10 observations
##   predicted class=t  expected loss=0.2  P(node) =0.003333333
```

```
##     class counts:     2     8
##    probabilities: 0.200 0.800 
## 
## Node number 42: 135 observations
##   predicted class=f  expected loss=0.07407407  P(node) =0.045
##     class counts:   125    10
##    probabilities: 0.926 0.074 
## 
## Node number 43: 118 observations,    complexity param=0.03372434
##   predicted class=t  expected loss=0.3898305  P(node) =0.03933333
##     class counts:    46    72
##    probabilities: 0.390 0.610 
##   left son=86 (50 obs) right son=87 (68 obs)
##   Primary splits:
##       f.9 splits as  LR, improve=18.916770, (0 missing)
##       f.5 splits as  RL, improve=14.101780, (0 missing)
##       won splits as  RL, improve=13.453910, (0 missing)
##       f.4 splits as  LR, improve=11.886250, (0 missing)
##       t   splits as  LR, improve= 7.634444, (0 missing)
##   Surrogate splits:
##       f.5  splits as  RL, agree=0.814, adj=0.56, (0 split)
##       won  splits as  RL, agree=0.814, adj=0.56, (0 split)
##       f.4  splits as  LR, agree=0.669, adj=0.22, (0 split)
##       f.20 splits as  RL, agree=0.661, adj=0.20, (0 split)
##       t    splits as  LR, agree=0.610, adj=0.08, (0 split)
## 
## Node number 86: 50 observations,    complexity param=0.02639296
##   predicted class=f  expected loss=0.28  P(node) =0.01666667
##     class counts:    36    14
##    probabilities: 0.720 0.280 
##   left son=172 (31 obs) right son=173 (19 obs)
##   Primary splits:
##       f.4  splits as  LR, improve=12.791580, (0 missing)
##       f.5  splits as  LR, improve= 2.754595, (0 missing)
##       f.16 splits as  RL, improve= 2.211282, (0 missing)
##       f.13 splits as  RL, improve= 1.276279, (0 missing)
##       f.26 splits as  RL, improve= 1.276279, (0 missing)
##   Surrogate splits:
##       l    splits as  RL, agree=0.68, adj=0.158, (0 split)
##       f.22 splits as  LR, agree=0.64, adj=0.053, (0 split)
## 
## Node number 87: 68 observations,    complexity param=0.02639296
##   predicted class=t  expected loss=0.1470588  P(node) =0.02266667
##     class counts:    10    58
##    probabilities: 0.147 0.853 
##   left son=174 (9 obs) right son=175 (59 obs)
##   Primary splits:
##       f.5  splits as  RL, improve=15.0927200, (0 missing)
##       f.6  splits as  LR, improve=13.1921600, (0 missing)
##       t    splits as  LR, improve= 4.2016810, (0 missing)
```

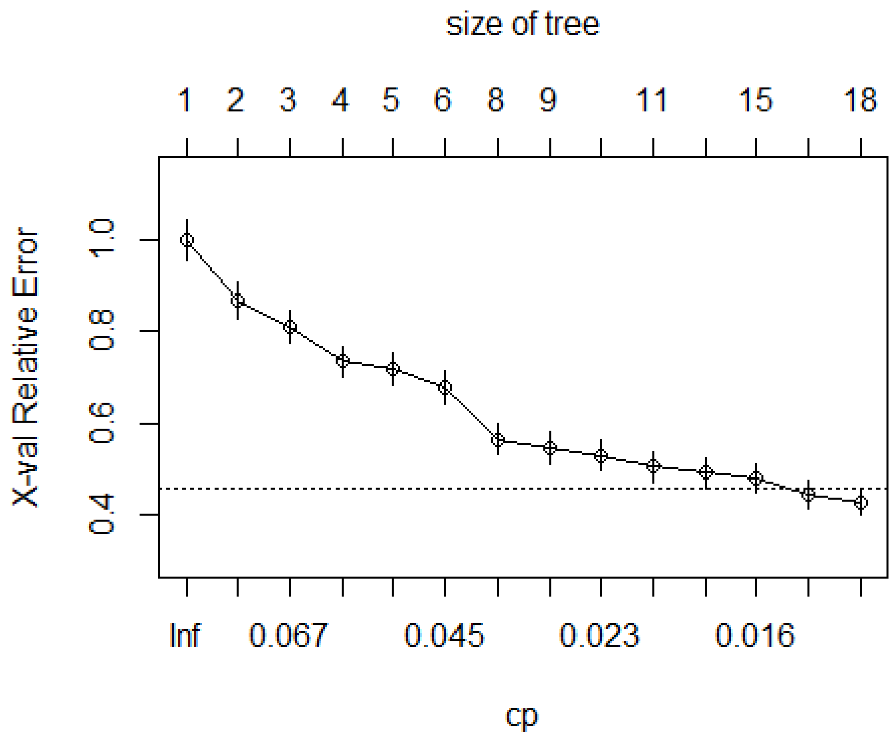
```
##       f.18 splits as  LR, improve= 0.6951872, (0 missing)
##       l    splits as  RL, improve= 0.4486540, (0 missing)
##
## Node number 172: 31 observations
##   predicted class=f  expected loss=0  P(node) =0.01033333
##     class counts:    31     0
##    probabilities: 1.000 0.000
##
## Node number 173: 19 observations
##   predicted class=t  expected loss=0.2631579  P(node) =0.006333333
##     class counts:     5    14
##    probabilities: 0.263 0.737
##
## Node number 174: 9 observations
##   predicted class=f  expected loss=0  P(node) =0.003
##     class counts:     9     0
##    probabilities: 1.000 0.000
##
## Node number 175: 59 observations
##   predicted class=t  expected loss=0.01694915  P(node) =0.01966667
##     class counts:     1    58
##    probabilities: 0.017 0.983
```

```r
#Grow tree
fit <- rpart(formula, method="anova",data=chess_train)
printcp(fit)
```

```
##
## Regression tree:
## rpart(formula = formula, data = chess_train, method = "anova")
##
## Variables actually used in tree construction:
##  [1] f.10 f.2  f.20 f.24 f.4  f.5  f.7  f.8  f.9  n    t    t.2  won
##
## Root node error: 302.24/3000 = 0.10075
##
## n= 3000
##
##          CP nsplit rel error  xerror     xstd
## 1  0.134584      0   1.00000 1.00035 0.044459
## 2  0.070916      1   0.86542 0.86693 0.040925
## 3  0.063532      2   0.79450 0.80850 0.034771
## 4  0.049308      3   0.73097 0.73281 0.033679
## 5  0.047322      4   0.68166 0.71608 0.034819
## 6  0.042860      5   0.63434 0.67633 0.034727
## 7  0.031294      7   0.54862 0.56381 0.033231
## 8  0.024968      8   0.51732 0.54536 0.033748
## 9  0.021161      9   0.49235 0.52879 0.033631
## 10 0.016471     10   0.47119 0.50447 0.033177
## 11 0.015998     12   0.43825 0.49135 0.031916
```

```
## 12 0.015792     14   0.40625 0.47813 0.031437
## 13 0.011966     15   0.39046 0.44413 0.029726
## 14 0.010000     17   0.36653 0.42890 0.029396
```

```r
plotcp(fit)
```




```r
summary(fit)
```

```
## Call:
## rpart(formula = formula, data = chess_train, method = "anova")
##   n= 3000
##
##            CP nsplit rel error    xerror       xstd
## 1  0.13458421      0 1.0000000 1.0003490 0.04445931
## 2  0.07091641      1 0.8654158 0.8669292 0.04092484
## 3  0.06353250      2 0.7944994 0.8084990 0.03477100
## 4  0.04930761      3 0.7309669 0.7328103 0.03367884
## 5  0.04732245      4 0.6816593 0.7160823 0.03481931
## 6  0.04285974      5 0.6343368 0.6763339 0.03472747
## 7  0.03129432      7 0.5486173 0.5638088 0.03323106
## 8  0.02496814      8 0.5173230 0.5453615 0.03374784
## 9  0.02116132      9 0.4923549 0.5287903 0.03363144
## 10 0.01647109     10 0.4711936 0.5044692 0.03317723
## 11 0.01599847     12 0.4382514 0.4913463 0.03191585
## 12 0.01579188     14 0.4062545 0.4781274 0.03143706
## 13 0.01196594     15 0.3904626 0.4441303 0.02972590
## 14 0.01000000     17 0.3665307 0.4288953 0.02939634
```

```
##
## Variable importance
##    n  t.2    t f.10  f.5  won  f.2  f.4  f.9 f.20  t.3 f.26  f.7  t.1  f.8
##   15   13    7    7    7    6    6    6    4    3    3    3    3    2    2
##  f.3 f.16 f.24  f.1 f.22 f.11  f.6 f.13 f.17 f.19
##    2    2    2    1    1    1    1    1    1    1
##
## Node number 1: 3000 observations,    complexity param=0.1345842
##   mean=1.113667, MSE=0.1007466
##   left son=2 (2790 obs) right son=3 (210 obs)
##   Primary splits:
##       n    splits as  RLL, improve=0.13458420, (0 missing)
##       f.10 splits as  LR,  improve=0.11125450, (0 missing)
##       t.1  splits as  LR,  improve=0.05752699, (0 missing)
##       t.3  splits as  RL,  improve=0.04414347, (0 missing)
##       n.1  splits as  RL,  improve=0.04214208, (0 missing)
##   Surrogate splits:
##       f.23 splits as  LR, agree=0.93, adj=0.005, (0 split)
##
## Node number 2: 2790 observations,    complexity param=0.07091641
##   mean=1.08172, MSE=0.0750422
##   left son=4 (1674 obs) right son=5 (1116 obs)
##   Primary splits:
##       f.10 splits as  LR, improve=0.10237370, (0 missing)
##       t.3  splits as  RL, improve=0.05154770, (0 missing)
##       t.1  splits as  LR, improve=0.04442476, (0 missing)
##       f.12 splits as  LR, improve=0.03231299, (0 missing)
##       n.1  splits as  RL, improve=0.03224257, (0 missing)
##   Surrogate splits:
##       t.3  splits as  RL,  agree=0.766, adj=0.416, (0 split)
##       f.26 splits as  LR,  agree=0.759, adj=0.398, (0 split)
##       t.1  splits as  LR,  agree=0.733, adj=0.332, (0 split)
##       n    splits as  -LR, agree=0.726, adj=0.315, (0 split)
##       f.16 splits as  LR,  agree=0.711, adj=0.277, (0 split)
##
## Node number 3: 210 observations,    complexity param=0.0635325
##   mean=1.538095, MSE=0.2485488
##   left son=6 (147 obs) right son=7 (63 obs)
##   Primary splits:
##       t.2  splits as  RL, improve=0.36788870, (0 missing)
##       won  splits as  LR, improve=0.27917190, (0 missing)
##       f.9  splits as  RL, improve=0.14224590, (0 missing)
##       t    splits as  RL, improve=0.10557950, (0 missing)
##       f.20 splits as  LR, improve=0.05931842, (0 missing)
##   Surrogate splits:
##       t    splits as  RL, agree=0.800, adj=0.333, (0 split)
##       f.2  splits as  LR, agree=0.767, adj=0.222, (0 split)
##       f.1  splits as  LR, agree=0.757, adj=0.190, (0 split)
##       f.3  splits as  LR, agree=0.757, adj=0.190, (0 split)
##       f.22 splits as  LR, agree=0.748, adj=0.159, (0 split)
```

```
##
## Node number 4: 1674 observations
##   mean=1.010155, MSE=0.01005219
##
## Node number 5: 1116 observations,    complexity param=0.04930761
##   mean=1.189068, MSE=0.1533214
##   left son=10 (1087 obs) right son=11 (29 obs)
##   Primary splits:
##       f.2  splits as  LR, improve=0.08709608, (0 missing)
##       f.7  splits as  LR, improve=0.05573021, (0 missing)
##       f.22 splits as  LR, improve=0.03599538, (0 missing)
##       t.2  splits as  RL, improve=0.03246534, (0 missing)
##       f.4  splits as  LR, improve=0.02840425, (0 missing)
##
## Node number 6: 147 observations,    complexity param=0.04732245
##   mean=1.340136, MSE=0.2244435
##   left son=12 (105 obs) right son=13 (42 obs)
##   Primary splits:
##       won  splits as  LR, improve=0.43350520, (0 missing)
##       f.9  splits as  RL, improve=0.18544140, (0 missing)
##       f.28 splits as  RL, improve=0.10562790, (0 missing)
##       f.24 splits as  LR, improve=0.06877508, (0 missing)
##       f.5  splits as  LR, improve=0.05417701, (0 missing)
##   Surrogate splits:
##       f.5  splits as  LR, agree=0.789, adj=0.262, (0 split)
##       f.25 splits as  LR, agree=0.721, adj=0.024, (0 split)
##
## Node number 7: 63 observations
##   mean=2, MSE=0
##
## Node number 10: 1087 observations,    complexity param=0.04285974
##   mean=1.170193, MSE=0.1412275
##   left son=20 (834 obs) right son=21 (253 obs)
##   Primary splits:
##       f.7  splits as  LR, improve=0.05088751, (0 missing)
##       f.4  splits as  LR, improve=0.04404798, (0 missing)
##       f.6  splits as  LR, improve=0.02167174, (0 missing)
##       f.17 splits as  RL, improve=0.01846718, (0 missing)
##       t    splits as  LR, improve=0.01800335, (0 missing)
##   Surrogate splits:
##       f.8 splits as  LR, agree=0.824, adj=0.245, (0 split)
##
## Node number 11: 29 observations
##   mean=1.896552, MSE=0.09274673
##
## Node number 12: 105 observations,    complexity param=0.01579188
##   mean=1.142857, MSE=0.122449
##   left son=24 (95 obs) right son=25 (10 obs)
##   Primary splits:
##       f.24 splits as  LR, improve=0.37122810, (0 missing)
```

```
##       f.19 splits as  LR, improve=0.26684210, (0 missing)
##       f.10 splits as  RL, improve=0.21105160, (0 missing)
##       f.27 splits as  LR, improve=0.21105160, (0 missing)
##       t.3  splits as  RL, improve=0.07440476, (0 missing)
##   Surrogate splits:
##       f.11 splits as  LR, agree=0.962, adj=0.6, (0 split)
##       f.19 splits as  LR, agree=0.943, adj=0.4, (0 split)
##       t    splits as  RL, agree=0.914, adj=0.1, (0 split)
##
## Node number 13: 42 observations
##   mean=1.833333, MSE=0.1388889
##
## Node number 20: 834 observations,    complexity param=0.01647109
##   mean=1.123501, MSE=0.1082487
##   left son=40 (191 obs) right son=41 (643 obs)
##   Primary splits:
##       f.9  splits as  RL, improve=0.04185451, (0 missing)
##       t    splits as  LR, improve=0.04008061, (0 missing)
##       f.16 splits as  RL, improve=0.03379648, (0 missing)
##       won  splits as  LR, improve=0.03253955, (0 missing)
##       f.28 splits as  RL, improve=0.01948445, (0 missing)
##
## Node number 21: 253 observations,    complexity param=0.04285974
##   mean=1.324111, MSE=0.2190629
##   left son=42 (135 obs) right son=43 (118 obs)
##   Primary splits:
##       t.2 splits as  RL, improve=0.32650510, (0 missing)
##       f.4 splits as  LR, improve=0.25947660, (0 missing)
##       f.5 splits as  RL, improve=0.07680513, (0 missing)
##       f.9 splits as  LR, improve=0.07479947, (0 missing)
##       f.6 splits as  LR, improve=0.06208229, (0 missing)
##   Surrogate splits:
##       t    splits as  RL, agree=0.743, adj=0.449, (0 split)
##       f.4  splits as  LR, agree=0.688, adj=0.331, (0 split)
##       f.3  splits as  LR, agree=0.593, adj=0.127, (0 split)
##       f.13 splits as  LR, agree=0.593, adj=0.127, (0 split)
##       f.20 splits as  LR, agree=0.585, adj=0.110, (0 split)
##
## Node number 24: 95 observations
##   mean=1.073684, MSE=0.06825485
##
## Node number 25: 10 observations
##   mean=1.8, MSE=0.16
##
## Node number 40: 191 observations
##   mean=1, MSE=0
##
## Node number 41: 643 observations,    complexity param=0.01647109
##   mean=1.160187, MSE=0.1345269
##   left son=82 (262 obs) right son=83 (381 obs)
```

```
##   Primary splits:
##       t    splits as  LR, improve=0.07141939, (0 missing)
##       f.16 splits as  RL, improve=0.04776852, (0 missing)
##       t.2  splits as  LR, improve=0.02744706, (0 missing)
##       f.8  splits as  RL, improve=0.02371414, (0 missing)
##       f.18 splits as  RL, improve=0.02371414, (0 missing)
##   Surrogate splits:
##       t.2  splits as  LR, agree=0.866, adj=0.672, (0 split)
##       f.17 splits as  RL, agree=0.652, adj=0.145, (0 split)
##       f.15 splits as  RL, agree=0.625, adj=0.080, (0 split)
##       f.28 splits as  RL, agree=0.624, adj=0.076, (0 split)
##       f.1  splits as  RL, agree=0.621, adj=0.069, (0 split)
##
## Node number 42: 135 observations
##   mean=1.074074, MSE=0.06858711
##
## Node number 43: 118 observations,    complexity param=0.03129432
##   mean=1.610169, MSE=0.2378627
##   left son=86 (50 obs) right son=87 (68 obs)
##   Primary splits:
##       f.9 splits as  LR, improve=0.3369835, (0 missing)
##       f.5 splits as  RL, improve=0.2512092, (0 missing)
##       won splits as  RL, improve=0.2396680, (0 missing)
##       f.4 splits as  LR, improve=0.2117418, (0 missing)
##       t   splits as  LR, improve=0.1360001, (0 missing)
##   Surrogate splits:
##       f.5  splits as  RL, agree=0.814, adj=0.56, (0 split)
##       won  splits as  RL, agree=0.814, adj=0.56, (0 split)
##       f.4  splits as  LR, agree=0.669, adj=0.22, (0 split)
##       f.20 splits as  RL, agree=0.661, adj=0.20, (0 split)
##       t    splits as  LR, agree=0.610, adj=0.08, (0 split)
##
## Node number 82: 262 observations
##   mean=1.041985, MSE=0.04022202
##
## Node number 83: 381 observations,    complexity param=0.01599847
##   mean=1.24147, MSE=0.1831621
##   left son=166 (142 obs) right son=167 (239 obs)
##   Primary splits:
##       f.8  splits as  RL, improve=0.06621968, (0 missing)
##       f.18 splits as  RL, improve=0.06621968, (0 missing)
##       f.25 splits as  LR, improve=0.06219598, (0 missing)
##       f.16 splits as  RL, improve=0.03883517, (0 missing)
##       won  splits as  LR, improve=0.03586143, (0 missing)
##   Surrogate splits:
##       f.17 splits as  RL, agree=0.717, adj=0.239, (0 split)
##
## Node number 86: 50 observations,    complexity param=0.02116132
##   mean=1.28, MSE=0.2016
##   left son=172 (31 obs) right son=173 (19 obs)
```

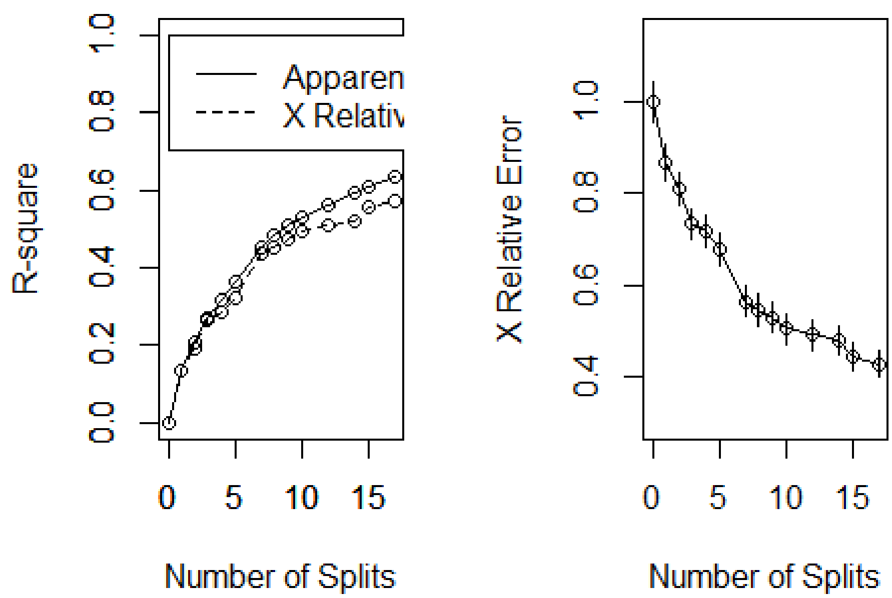
```
##   Primary splits:
##       f.4  splits as  LR, improve=0.63450290, (0 missing)
##       f.5  splits as  LR, improve=0.13663660, (0 missing)
##       f.16 splits as  RL, improve=0.10968660, (0 missing)
##       f.13 splits as  RL, improve=0.06330749, (0 missing)
##       f.26 splits as  RL, improve=0.06330749, (0 missing)
##   Surrogate splits:
##       l    splits as  RL, agree=0.68, adj=0.158, (0 split)
##       f.22 splits as  LR, agree=0.64, adj=0.053, (0 split)
##
## Node number 87: 68 observations,    complexity param=0.02496814
##   mean=1.852941, MSE=0.1254325
##   left son=174 (9 obs) right son=175 (59 obs)
##   Primary splits:
##       f.5  splits as  RL, improve=0.88474580, (0 missing)
##       f.6  splits as  LR, improve=0.77333330, (0 missing)
##       t    splits as  LR, improve=0.24630540, (0 missing)
##       f.18 splits as  LR, improve=0.04075235, (0 missing)
##       l    splits as  RL, improve=0.02630041, (0 missing)
##
## Node number 166: 142 observations,    complexity param=0.01196594
##   mean=1.098592, MSE=0.08887126
##   left son=332 (97 obs) right son=333 (45 obs)
##   Primary splits:
##       f.4  splits as  LR, improve=0.23576390, (0 missing)
##       f.5  splits as  LR, improve=0.07029087, (0 missing)
##       f.16 splits as  RL, improve=0.05762769, (0 missing)
##       f.20 splits as  RL, improve=0.05240885, (0 missing)
##       f.6  splits as  LR, improve=0.03785047, (0 missing)
##
## Node number 167: 239 observations,    complexity param=0.01599847
##   mean=1.32636, MSE=0.2198491
##   left son=334 (155 obs) right son=335 (84 obs)
##   Primary splits:
##       f.20 splits as  LR, improve=0.09610275, (0 missing)
##       won  splits as  LR, improve=0.06744705, (0 missing)
##       f.25 splits as  LR, improve=0.06180182, (0 missing)
##       f.28 splits as  RL, improve=0.03265452, (0 missing)
##       f.4  splits as  RL, improve=0.03078106, (0 missing)
##   Surrogate splits:
##       f.25 splits as  LR, agree=0.699, adj=0.143, (0 split)
##
## Node number 172: 31 observations
##   mean=1, MSE=0
##
## Node number 173: 19 observations
##   mean=1.736842, MSE=0.1939058
##
## Node number 174: 9 observations
##   mean=1, MSE=0
```

```
##
## Node number 175: 59 observations
##   mean=1.983051, MSE=0.01666188
##
## Node number 332: 97 observations
##   mean=1, MSE=0
##
## Node number 333: 45 observations,    complexity param=0.01196594
##   mean=1.311111, MSE=0.214321
##   left son=666 (28 obs) right son=667 (17 obs)
##   Primary splits:
##       f.5  splits as  LR, improve=0.44148630, (0 missing)
##       f.16 splits as  RL, improve=0.20395420, (0 missing)
##       f.20 splits as  RL, improve=0.20395420, (0 missing)
##       f.6  splits as  LR, improve=0.15969470, (0 missing)
##       f.26 splits as  RL, improve=0.09181982, (0 missing)
##   Surrogate splits:
##       f.6 splits as  LR, agree=0.867, adj=0.647, (0 split)
##
## Node number 334: 155 observations
##   mean=1.219355, MSE=0.1712383
##
## Node number 335: 84 observations
##   mean=1.52381, MSE=0.2494331
##
## Node number 666: 28 observations
##   mean=1.071429, MSE=0.06632653
##
## Node number 667: 17 observations
##   mean=1.705882, MSE=0.2076125
```

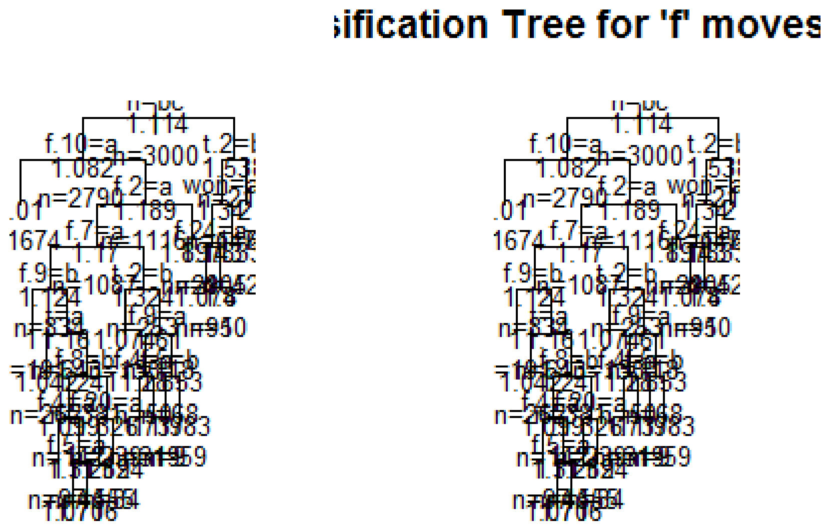
```
#Additional Plots
par(mfrow=c(1,2))
rsq.rpart(fit)
```

```
##
## Regression tree:
## rpart(formula = formula, data = chess_train, method = "anova")
##
## Variables actually used in tree construction:
##  [1] f.10 f.2  f.20 f.24 f.4  f.5  f.7  f.8  f.9  n    t    t.2  won
##
## Root node error: 302.24/3000 = 0.10075
##
## n= 3000
##
##          CP nsplit rel error  xerror     xstd
## 1  0.134584      0   1.00000 1.00035 0.044459
## 2  0.070916      1   0.86542 0.86693 0.040925
## 3  0.063532      2   0.79450 0.80850 0.034771
```

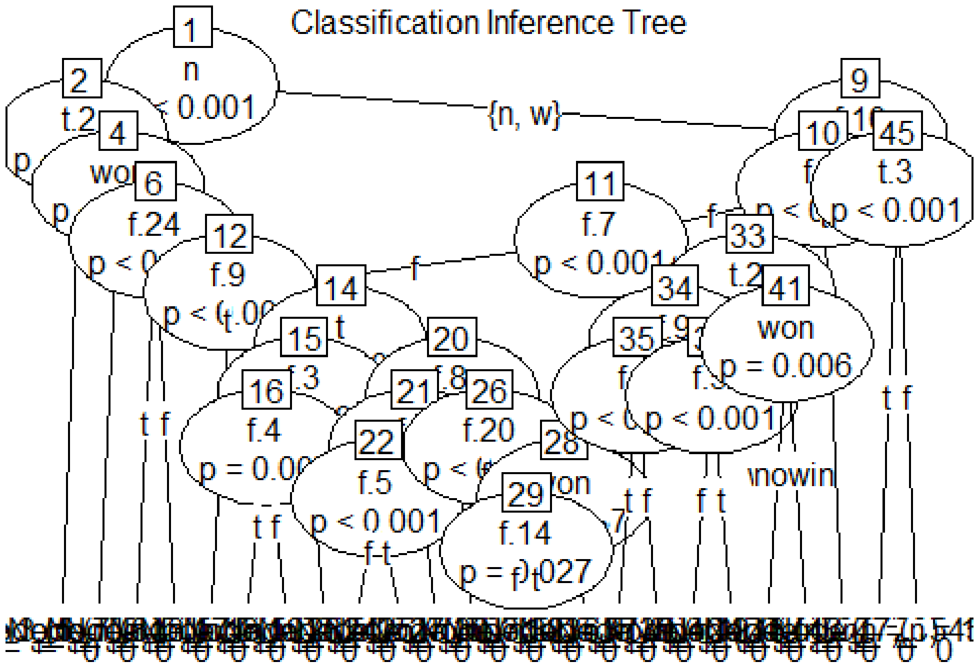
```
## 4  0.049308      3   0.73097 0.73281 0.033679
## 5  0.047322      4   0.68166 0.71608 0.034819
## 6  0.042860      5   0.63434 0.67633 0.034727
## 7  0.031294      7   0.54862 0.56381 0.033231
## 8  0.024968      8   0.51732 0.54536 0.033748
## 9  0.021161      9   0.49235 0.52879 0.033631
## 10 0.016471     10   0.47119 0.50447 0.033177
## 11 0.015998     12   0.43825 0.49135 0.031916
## 12 0.015792     14   0.40625 0.47813 0.031437
## 13 0.011966     15   0.39046 0.44413 0.029726
## 14 0.010000     17   0.36653 0.42890 0.029396
```




```
plot(fit,uniform=TRUE, ain = "Regression Tree for 'f'  ")

## Warning in plot.window(...): "ain" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "ain" is not a graphical parameter

## Warning in title(...): "ain" is not a graphical parameter

text(fit,use.n=TRUE,all=TRUE,cex=0.8)

plot(fit,uniform=T,main="Classification Tree for 'f' moves in chess")
text(fit,use.n=TRUE,all=TRUE,cex=0.8)
```




```r
tr <- tree(formula,data=chess_train)
summary(tr)

##
## Classification tree:
## tree(formula = formula, data = chess_train)
## Variables actually used in tree construction:
##  [1] "f.10" "f.27" "n"    "f.2"  "f.7"  "f.9"  "t"    "f.8"  "f.20" "f.4"
## [11] "t.2"  "f.5"  "won"
## Number of terminal nodes:  20
## Residual mean deviance:  0.2458 = 732.4 / 2980
## Misclassification error rate: 0.051 = 153 / 3000

plot(tr)
text(tr)

ct = ctree(formula,data=chess_train)
plot(ct,main="Classification Inference Tree")
tr.pred = predict(ct,newdata=chess_train, type = "prob")
table(predict(ct), chess_train$f)

##
##         f    t
##    f 2614   82
##    t   45  259
```



Answers:

A(1)

Yes. The size of the data helps in distributing the training and testing data. More the data, it is better in prediction and thus lesser is the error rate. I tried with subsetting the chess data from 3195 to 300 data rows only. The error rate in prediction was higher than that for 3195 data. The observation suggests that model accuracy improves upto certain extent with increase in data size.

A(2)

Yes. The rules make clear sense. This is because the important moves in the game have been recorded in the formula along with the outcome if it is a 'win' or a 'nowin'. Hence the rules help in making a proper decision for the move that is to be played. This function guides which rules are important by how much margin.

```
asRules(fit)

## 
##  Rule number: 4 [f=1.01015531660693 cover=1674 (56%)]
##    n=n,w
##    f.10=f
## 
##  Rule number: 82 [f=1.04198473282443 cover=262 (9%)]
##    n=n,w
```

```
##    f.10=t
##    f.2=f
##    f.7=f
##    f.9=f
##    t=f
##
##  Rule number: 40 [f=1 cover=191 (6%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=f
##    f.9=t
##
##  Rule number: 334 [f=1.21935483870968 cover=155 (5%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=f
##    f.9=f
##    t=t
##    f.8=f
##    f.20=f
##
##  Rule number: 42 [f=1.07407407407407 cover=135 (4%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=t
##    t.2=t
##
##  Rule number: 332 [f=1 cover=97 (3%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=f
##    f.9=f
##    t=t
##    f.8=t
##    f.4=f
##
##  Rule number: 24 [f=1.07368421052632 cover=95 (3%)]
##    n=b
##    t.2=t
##    won=nowin
##    f.24=f
##
##  Rule number: 335 [f=1.52380952380952 cover=84 (3%)]
##    n=n,w
##    f.10=t
##    f.2=f
```

```
##    f.7=f
##    f.9=f
##    t=t
##    f.8=f
##    f.20=t
##
##  Rule number: 7 [f=2 cover=63 (2%)]
##    n=b
##    t.2=f
##
##  Rule number: 175 [f=1.98305084745763 cover=59 (2%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=t
##    t.2=f
##    f.9=t
##    f.5=f
##
##  Rule number: 13 [f=1.83333333333333 cover=42 (1%)]
##    n=b
##    t.2=t
##    won=won
##
##  Rule number: 172 [f=1 cover=31 (1%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=t
##    t.2=f
##    f.9=f
##    f.4=f
##
##  Rule number: 11 [f=1.89655172413793 cover=29 (1%)]
##    n=n,w
##    f.10=t
##    f.2=t
##
##  Rule number: 666 [f=1.07142857142857 cover=28 (1%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=f
##    f.9=f
##    t=t
##    f.8=t
##    f.4=t
##    f.5=f
##
##  Rule number: 173 [f=1.73684210526316 cover=19 (1%)]
```

```
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=t
##    t.2=f
##    f.9=f
##    f.4=t
##
##  Rule number: 667 [f=1.70588235294118 cover=17 (1%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=f
##    f.9=f
##    t=t
##    f.8=t
##    f.4=t
##    f.5=t
##
##  Rule number: 25 [f=1.8 cover=10 (0%)]
##    n=b
##    t.2=t
##    won=nowin
##    f.24=t
##
##  Rule number: 174 [f=1 cover=9 (0%)]
##    n=n,w
##    f.10=t
##    f.2=f
##    f.7=t
##    t.2=f
##    f.9=t
##    f.5=t
```

The rules 'n' and 'f.10' play a significant role in the decision tree. The algorithm generates good rules in order to fit the data based upon the formula and reduce the error in prediction.

A(3)

Decision trees geenrally do not require scaling or normalization. THe same is the case with normalization. Scaling and normalization do not affect decision tree algorithms because they are categorical based.