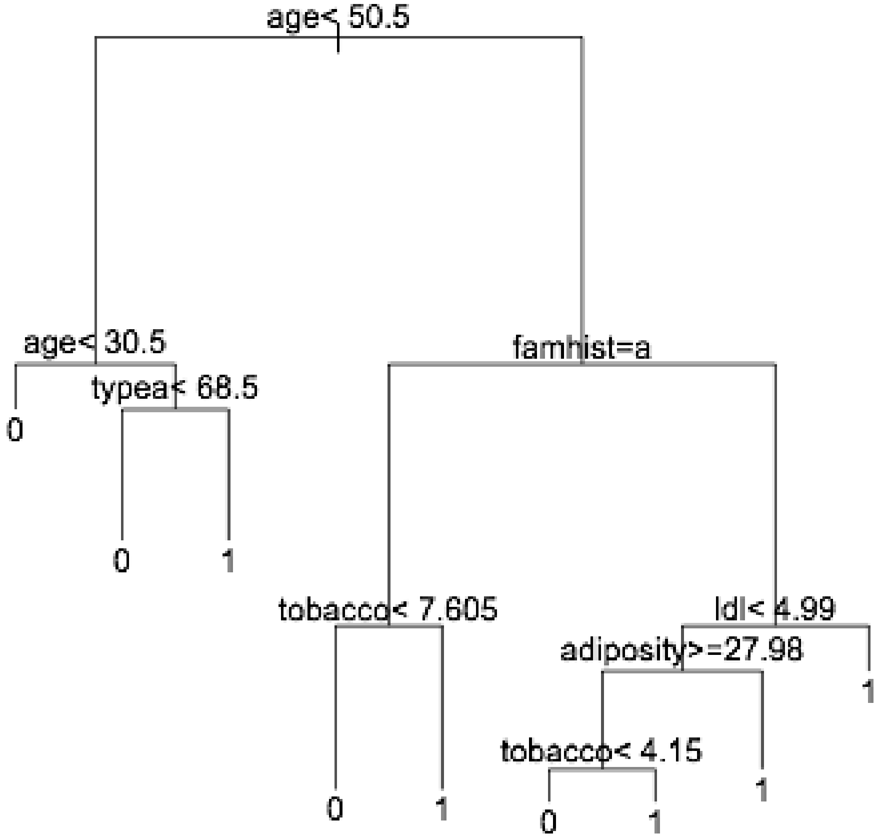
```
# **************************************************
#
#
# **************************************************
#
#  Classification tree

> data(SAheart)
> names(SAheart)

 [1] "sbp"       "tobacco"   "ldl"       "adiposity" "famhist"
 [6] "typea"     "obesity"   "alcohol"   "age"       "chd"

> (heartree = rpart(chd ~ ., data = SAheart, method="class"))

## output follows
##
n= 462

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 462 160 0 (0.653680 0.346320)
   2) age< 50.5 290  64 0 (0.779310 0.220690)
     4) age< 30.5 108   8 0 (0.925926 0.074074) *
     5) age>=30.5 182  56 0 (0.692308 0.307692)
      10) typea< 68.5 170  46 0 (0.729412 0.270588) *
      11) typea>=68.5 12   2 1 (0.166667 0.833333) *
   3) age>=50.5 172  76 1 (0.441860 0.558140)
     6) famhist=Absent 82  33 0 (0.597561 0.402439)
      12) tobacco< 7.605 58  16 0 (0.724138 0.275862) *
      13) tobacco>=7.605 24   7 1 (0.291667 0.708333) *
     7) famhist=Present 90  27 1 (0.300000 0.700000)
      14) ldl< 4.99 39  18 1 (0.461538 0.538462)
        28) adiposity>=27.985 20   7 0 (0.650000 0.350000)
          56) tobacco< 4.15 10   1 0 (0.900000 0.100000) *
          57) tobacco>=4.15 10   4 1 (0.400000 0.600000) *
        29) adiposity< 27.985 19   5 1 (0.263158 0.736842) *
      15) ldl>=4.99 51   9 1 (0.176471 0.823529) *

> plot(heartree, margin = .10)
> text(heartree) # depth of branches proportional to reduction in error
> plot(heartree, margin = .10, compress = T, uniform = T, branch = 0.4)
> text(heartree, use.n = T) # depth of branches is uniform
> post(heartree) # makes a file called heartree.ps in the local directory
> printcp(heartree)

Classification tree:
rpart(formula = chd ~ ., data = SAheart, method = "class")

Variables actually used in tree construction:
[1] adiposity age      famhist   ldl       tobacco   typea

Root node error: 160/462 = 0.346

n= 462

      CP nsplit rel error xerror   xstd
1 0.1250      0     1.000  1.000 0.0639
2 0.1000      1     0.875  1.056 0.0647
3 0.0625      2     0.775  1.000 0.0639
4 0.0250      3     0.713  0.863 0.0615
5 0.0188      5     0.663  0.831 0.0608
6 0.0125      7     0.625  0.875 0.0617
7 0.0100      8     0.613  0.931 0.0628

> table(actual=SAheart$chd,predicted=predict(heartree,type="class"))
      predicted
actual   0   1
     0 275  27
     1  71  89
> 1-sum(diag(.Last.value))/sum(.Last.value)
[1] 0.21212

## this is on the training data, not new test data, so is overly optimistic

> heartlogreg = glm(chd ~ sbp+tobacco+ldl+famhist+obesity+alcohol+age,
data=SAheart, family=binomial)

> table(SAheart$chd, predict(heartlogreg, type="response")>0.5)


    FALSE TRUE
  0   255   47
  1    78   82
> 1-sum(diag(.Last.value))/sum(.Last.value)
[1] 0.27056

## so we've done a bit better; but true test is on test data

data(fgl)
```

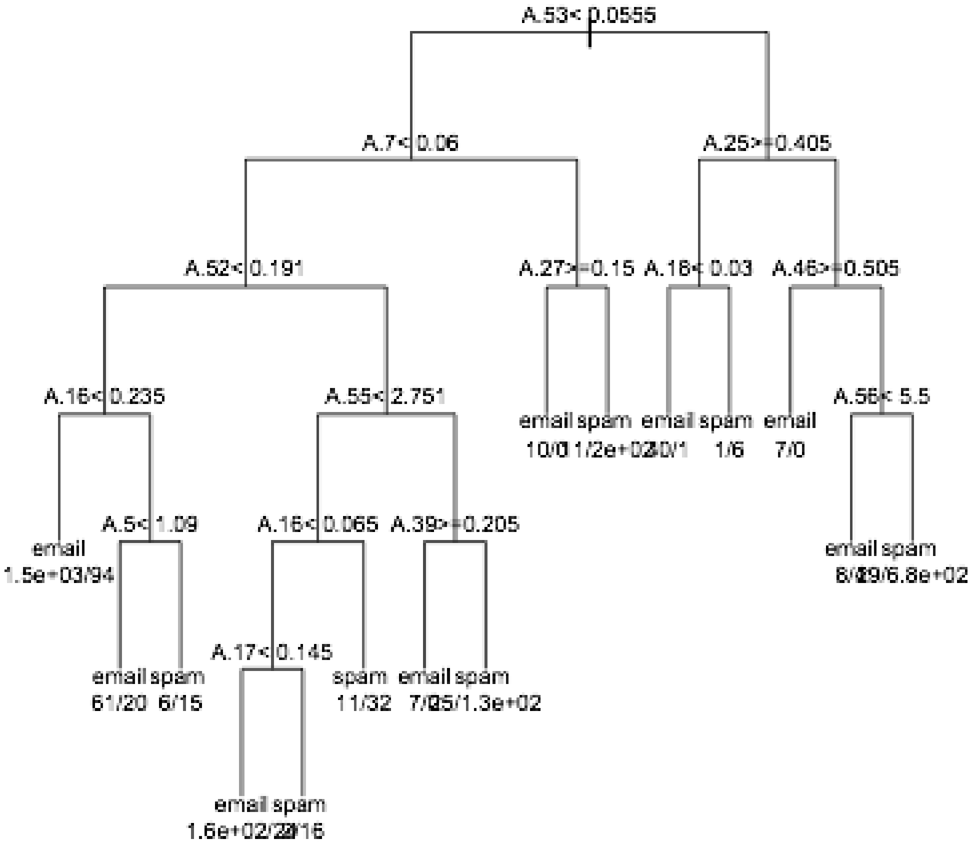
```
dim(fgl)
#[1] 214  10

fgl[1:4,]

# **************************************************
#      RI    Na   Mg   Al    Si    K   Ca Ba Fe type
# 1  3.01 13.64 4.49 1.10 71.78 0.06 8.75  0  0 WinF
# 2 -0.39 13.89 3.60 1.36 72.73 0.48 7.83  0  0 WinF
# 3 -1.82 13.53 3.55 1.54 72.99 0.39 7.78  0  0 WinF
# 4 -0.34 13.21 3.69 1.29 72.61 0.57 8.22  0  0 WinF
# **************************************************

levels(fgl$type)
# **************************************************
# [1] "WinF"  "WinNF" "Veh"   "Con"   "Tabl"  "Head"
# **************************************************
#
 set.seed(123)  # since xerror is randomly chosen, results will differ with different
seeds

 fgl.rp = rpart(type ~ .,data = fgl, cp = .001)
 plotcp(fgl.rp)
 printcp(fgl.rp)

#Classification tree:
#rpart(formula = type ~ ., data = fgl, cp = 0.001)
#
#Variables actually used in tree construction:
#[1] Al Ba Ca Fe Mg Na RI
#
#Root node error: 138/214 = 0.64486
#
#n= 214
#
#        CP nsplit rel error  xerror     xstd
#1 0.206522      0   1.00000 1.00000 0.050729
#2 0.072464      2   0.58696 0.60145 0.051652
#3 0.057971      3   0.51449 0.59420 0.051536
#4 0.036232      4   0.45652 0.53623 0.050419
#5 0.032609      5   0.42029 0.53623 0.050419
#6 0.010870      7   0.35507 0.50725 0.049733
#7 0.001000      9   0.33333 0.50725 0.049733

## try 8 splits, cp = 0.02
```

```
 fgl.rp2 = prune(fgl.rp, cp = 0.02)
plot(fgl.rp2, uniform = T); text(fgl.rp2, use.n = T, cex = .8)
fgl.rp2
#n= 214
#
#node), split, n, loss, yval, (yprob)
 #     * denotes terminal node
#
# 1) root 214 138 WinNF (0.33 0.36 0.079 0.061 0.042 0.14)
#   2) Ba< 0.335 185 110 WinNF (0.37 0.41 0.092 0.065 0.049 0.016)
#     4) Al< 1.42 113  50 WinF (0.56 0.27 0.12 0.0088 0.027 0.018)
#       8) Ca< 10.48 101  38 WinF (0.62 0.21 0.13 0 0.02 0.02)
#       16) RI>=-0.93 85  25 WinF (0.71 0.2 0.071 0 0.012 0.012)
#          32) Mg< 3.865 77  18 WinF (0.77 0.14 0.065 0 0.013 0.013) *
#          33) Mg>=3.865 8   2 WinNF (0.12 0.75 0.12 0 0 0) *
#        17) RI< -0.93 16   9 Veh (0.19 0.25 0.44 0 0.062 0.062) *
#       9) Ca>=10.48 12   2 WinNF (0 0.83 0 0.083 0.083 0) *
#     5) Al>=1.42 72  28 WinNF (0.083 0.61 0.056 0.15 0.083 0.014)
#      10) Mg>=2.26 52  11 WinNF (0.12 0.79 0.077 0 0.019 0) *
#      11) Mg< 2.26 20   9 Con (0 0.15 0 0.55 0.25 0.05)
#        22) Na< 13.495 12   1 Con (0 0.083 0 0.92 0 0) *
#        23) Na>=13.495 8   3 Tabl (0 0.25 0 0 0.62 0.12) *
#   3) Ba>=0.335 29   3 Head (0.034 0.034 0 0.034 0 0.9) *
```



```
> library(ElemStatLearn)
> data(spam)
> dim(spam)
[1] 4601   58
> names(spam)
 [1] "A.1"  "A.2"  "A.3"  "A.4"  "A.5"  "A.6"  "A.7"  "A.8"
 [9] "A.9"  "A.10" "A.11" "A.12" "A.13" "A.14" "A.15" "A.16"
[17] "A.17" "A.18" "A.19" "A.20" "A.21" "A.22" "A.23" "A.24"
[25] "A.25" "A.26" "A.27" "A.28" "A.29" "A.30" "A.31" "A.32"
[33] "A.33" "A.34" "A.35" "A.36" "A.37" "A.38" "A.39" "A.40"
```

```
[41] "A.41" "A.42" "A.43" "A.44" "A.45" "A.46" "A.47" "A.48"
[49] "A.49" "A.50" "A.51" "A.52" "A.53" "A.54" "A.55" "A.56"
[57] "A.57" "spam"
> spamtest = scan("2008-9/spam.traintest")
Read 4601 items
> levels(spamtest)
NULL
> spamtest[1:5]
[1] 1 0 1 0 0
> sum(spamtest)
[1] 1536
> is.factor(spam$spam)
[1] TRUE
> spamtree = rpart(spam ~ ., data=spam[spamtest==0,], cp = .001)
> printcp(spamtree)

Classification tree:
rpart(formula = spam ~ ., data = spam[spamtest == 0, ], cp = 0.001)

Variables actually used in tree construction:
 [1] A.12 A.16 A.17 A.18 A.19 A.21 A.24 A.25 A.27 A.39 A.42 A.45
[13] A.46 A.5  A.50 A.52 A.53 A.55 A.56 A.57 A.6  A.7  A.9

Root node error: 1218/3065 = 0.397

n= 3065

        CP nsplit rel error xerror    xstd
1  0.49343      0     1.000  1.000 0.0222
2  0.14450      1     0.507  0.507 0.0182
3  0.04187      2     0.362  0.363 0.0160
4  0.02791      4     0.278  0.300 0.0147
5  0.01724      5     0.250  0.276 0.0142
6  0.01149      6     0.233  0.253 0.0137
7  0.00821      7     0.222  0.245 0.0135
8  0.00575      8     0.213  0.227 0.0130
9  0.00411     10     0.202  0.227 0.0130
10 0.00369     11     0.198  0.232 0.0131
11 0.00328     13     0.190  0.232 0.0131
12 0.00246     14     0.187  0.227 0.0130
13 0.00219     20     0.172  0.236 0.0132
14 0.00164     23     0.166  0.239 0.0133
15 0.00103     31     0.153  0.235 0.0132
16 0.00100     44     0.136  0.237 0.0133
> plot(spamtree, margin = .1, unif=T)
> text(spamtree, cex=.6)
> spamtree2 = rpart(spam ~ ., data = spam[spamtest==0,], cp=0.0043)
> spamtree2$cptable[,2]
 1  2  3  4  5  6  7  8  9
 0  1  2  4  5  6  7  8 10
+ table(predict(spamtree2,spam[spamtest==1,],type="class"),
+       spam[spamtest==1,58])/sum(spamtest)

          email     spam
  email 0.579427 0.061849
  spam  0.033203 0.325521
> spamtree3 = rpart(spam ~ ., data = spam[spamtest==0,], cp = .0025)
> table(predict(spamtree3,
spam[spamtest==1,],type="class"),spam[spamtest==1,58])/sum(spamtest)

          email     spam
  email 0.580729 0.054688
  spam  0.031901 0.332682
> plot(spamtree3, uniform = T);text(spamtree3,use.n=T, cex=.7)
```


A.53< 0.0555
A.7< 0.06
A.25>=0.405
A.52< 0.191
A.27>=0.15
A.18< 0.03
A.46>=0.505
A.16< 0.235
A.55< 2.751
email
10/0
spam
1/2e+02
email
20/1
spam
1/6
email
7/0
A.56< 5.5
email
1.5e+03/94
A.5< 1.09
A.16< 0.065
A.39>=0.205
email
6/0
spam
29/6.8e+02
A.17< 0.145
email
61/20
spam
6/15
spam
11/32
email
7/0
spam
25/1.3e+02
email
1.6e+02/22
spam
2/16