R Notebook

Code

Hide

```
df <- read.csv("smoking.csv")
str(df)
```

```
'data.frame':   55692 obs. of  27 variables:
 $ ID                 : int  0 1 2 3 4 5 6 7 9 10 ...
 $ gender             : chr  "F" "F" "M" "M" ...
 $ age                : int  40 40 55 40 40 30 40 45 50 45 ...
 $ height.cm.         : int  155 160 170 165 155 180 160 165 150 175 ...
 $ weight.kg.         : int  60 60 60 70 60 75 60 90 60 75 ...
 $ waist.cm.          : num  81.3 81 80 88 86 85 85.5 96 85 89 ...
 $ eyesight.left.     : num  1.2 0.8 0.8 1.5 1 1.2 1 1.2 0.7 1 ...
 $ eyesight.right.    : num  1 0.6 0.8 1.5 1 1.2 1 1 0.8 1 ...
 $ hearing.left.      : num  1 1 1 1 1 1 1 1 1 1 ...
 $ hearing.right.     : num  1 1 1 1 1 1 1 1 1 1 ...
 $ systolic           : num  114 119 138 100 120 128 116 153 115 113 ...
 $ relaxation         : num  73 70 86 60 74 76 82 96 74 64 ...
 $ fasting.blood.sugar: num  94 130 89 96 80 95 94 158 86 94 ...
 $ Cholesterol        : num  215 192 242 322 184 217 226 222 210 198 ...
 $ triglyceride       : num  82 115 182 254 74 199 68 269 66 147 ...
 $ HDL                : num  73 42 55 45 62 48 55 34 48 43 ...
 $ LDL                : num  126 127 151 226 107 129 157 134 149 126 ...
 $ hemoglobin         : num  12.9 12.7 15.8 14.7 12.5 16.2 17 15 13.7 16 ...
 $ Urine.protein      : num  1 1 1 1 1 1 1 1 1 1 ...
 $ serum.creatinine   : num  0.7 0.6 1 1 0.6 1.2 0.7 1.3 0.8 0.8 ...
 $ AST                : num  18 22 21 19 16 18 21 38 31 26 ...
 $ ALT                : num  19 19 16 26 14 27 27 71 31 24 ...
 $ Gtp                : num  27 18 22 18 22 33 39 111 14 63 ...
 $ oral               : chr  "Y" "Y" "Y" "Y" ...
 $ dental.caries      : int  0 0 0 0 0 0 1 0 0 0 ...
 $ tartar             : chr  "Y" "Y" "N" "Y" ...
 $ smoking            : int  0 0 1 0 0 0 1 0 0 0 ...
```

Hide

```
df <- subset(df, select=-c(ID,oral))
```

Format columns.

Hide

```
df$gender <- factor(df$gender)
# df$oral <- factor(df$oral) #single factor
df$dental.caries <- factor(df$dental.caries)
df$tartar <- factor(df$tartar)
df$smoking <- factor(df$smoking)
df$hearing.left. <- factor(df$hearing.left.)
df$hearing.right. <- factor(df$hearing.right.)

levels(df$dental.caries) <- c("N","Y")
levels(df$smoking) <- c("N","Y")
```

Training and testing data.

Hide

```
set.seed(1234)
i <- sample(nrow(df), 0.75*nrow(df), replace=FALSE)
train <- df[i,]
test <- df[-i,]
```

Calculate decision tree.

Hide

```
library(rpart)
tree1 <- rpart(smoking~., data=train, method="class")
summary(tree1)
```

```
Call:
rpart(formula = smoking ~ ., data = train, method = "class")
  n= 41769

          CP nsplit rel error    xerror        xstd
1 0.19151704      0 1.0000000 1.0000000 0.006411875
2 0.05295342      1 0.8084830 0.8084830 0.006078279
3 0.01580796      2 0.7555295 0.7559849 0.005957905
4 0.01000000      4 0.7239136 0.7346474 0.005905086

Variable importance
          gender       height.cm.       hemoglobin       weight.kg. serum.creatinine
Gtp              ALT              age              AST     triglyceride
              27               18               18               12               11
10                1                1                1                1                1

Node number 1: 41769 observations,    complexity param=0.191517
  predicted class=N  expected loss=0.3680241  P(node) =1
    class counts: 26397 15372
   probabilities: 0.632 0.368
  left son=2 (15225 obs) right son=3 (26544 obs)
  Primary splits:
      gender     splits as  LR,        improve=5116.533, (0 missing)
      height.cm. < 162.5 to the left,  improve=3276.694, (0 missing)
      hemoglobin < 14.25 to the left,  improve=2997.453, (0 missing)
      Gtp        < 25.5  to the left,  improve=1992.068, (0 missing)
      weight.kg. < 62.5  to the left,  improve=1685.664, (0 missing)
  Surrogate splits:
      height.cm.       < 162.5 to the left,  agree=0.883, adj=0.678, (0 split)
      hemoglobin       < 14.15 to the left,  agree=0.875, adj=0.656, (0 split)
      weight.kg.       < 57.5  to the left,  agree=0.804, adj=0.462, (0 split)
      serum.creatinine < 0.85  to the left,  agree=0.788, adj=0.418, (0 split)
      Gtp              < 17.5  to the left,  agree=0.734, adj=0.271, (0 split)

Node number 2: 15225 observations
  predicted class=N  expected loss=0.04124795  P(node) =0.3645048
    class counts: 14597   628
   probabilities: 0.959 0.041

Node number 3: 26544 observations,    complexity param=0.05295342
  predicted class=Y  expected loss=0.4445449  P(node) =0.6354952
    class counts: 11800 14744
   probabilities: 0.445 0.555
  left son=6 (14188 obs) right son=7 (12356 obs)
  Primary splits:
      Gtp           < 34.5  to the left,  improve=431.57690, (0 missing)
      triglyceride  < 113.5 to the left,  improve=284.15680, (0 missing)
      age           < 62.5  to the right, improve=159.24450, (0 missing)
      tartar        splits as  LR,        improve=128.79720, (0 missing)
      dental.caries splits as  LR,        improve= 85.88246, (0 missing)
  Surrogate splits:
      ALT                 < 26.5  to the left,  agree=0.707, adj=0.370, (0 split)
```

```
      AST                < 25.5  to the left,  agree=0.672, adj=0.295, (0 split)
      triglyceride       < 145.5 to the left,  agree=0.659, adj=0.267, (0 split)
      waist.cm.          < 86.45 to the left,  agree=0.627, adj=0.199, (0 split)
      fasting.blood.sugar < 103.5 to the left,  agree=0.597, adj=0.135, (0 split)

Node number 6: 14188 observations,    complexity param=0.01580796
  predicted class=N  expected loss=0.4713138  P(node) =0.3396778
    class counts:  7501  6687
   probabilities: 0.529 0.471
  left son=12 (833 obs) right son=13 (13355 obs)
  Primary splits:
      age          < 62.5  to the right, improve=75.99166, (0 missing)
      triglyceride < 88.5  to the left,  improve=73.34851, (0 missing)
      tartar       splits as  LR,        improve=72.47402, (0 missing)
      Gtp          < 23.5  to the left,  improve=62.88057, (0 missing)
      AST          < 21.5  to the right, improve=51.64929, (0 missing)
  Surrogate splits:
      height.cm.       < 152.5 to the left,  agree=0.942, adj=0.005, (0 split)
      serum.creatinine < 1.85  to the right, agree=0.941, adj=0.002, (0 split)
      systolic         < 173   to the right, agree=0.941, adj=0.001, (0 split)
      relaxation       < 127.5 to the right, agree=0.941, adj=0.001, (0 split)

Node number 7: 12356 observations
  predicted class=Y  expected loss=0.3479281  P(node) =0.2958175
    class counts:  4299  8057
   probabilities: 0.348 0.652

Node number 12: 833 observations
  predicted class=N  expected loss=0.2641056  P(node) =0.01994302
    class counts:   613   220
   probabilities: 0.736 0.264

Node number 13: 13355 observations,    complexity param=0.01580796
  predicted class=N  expected loss=0.4842381  P(node) =0.3197347
    class counts:  6888  6467
   probabilities: 0.516 0.484
  left son=26 (6357 obs) right son=27 (6998 obs)
  Primary splits:
      age           < 37.5  to the left,  improve=74.94439, (0 missing)
      tartar        splits as  LR,        improve=65.58848, (0 missing)
      triglyceride  < 88.5  to the left,  improve=63.08678, (0 missing)
      Gtp           < 23.5  to the left,  improve=61.25796, (0 missing)
      dental.caries splits as  LR,        improve=44.38722, (0 missing)
  Surrogate splits:
      height.cm.          < 172.5 to the right, agree=0.611, adj=0.183, (0 split)
      fasting.blood.sugar < 96.5  to the left,  agree=0.603, adj=0.167, (0 split)
      triglyceride        < 91.5  to the left,  agree=0.576, adj=0.110, (0 split)
      eyesight.left.      < 1.05  to the right, agree=0.570, adj=0.097, (0 split)
      LDL                 < 104.5 to the left,  agree=0.569, adj=0.095, (0 split)

Node number 26: 6357 observations
  predicted class=N  expected loss=0.4286613  P(node) =0.1521942
```

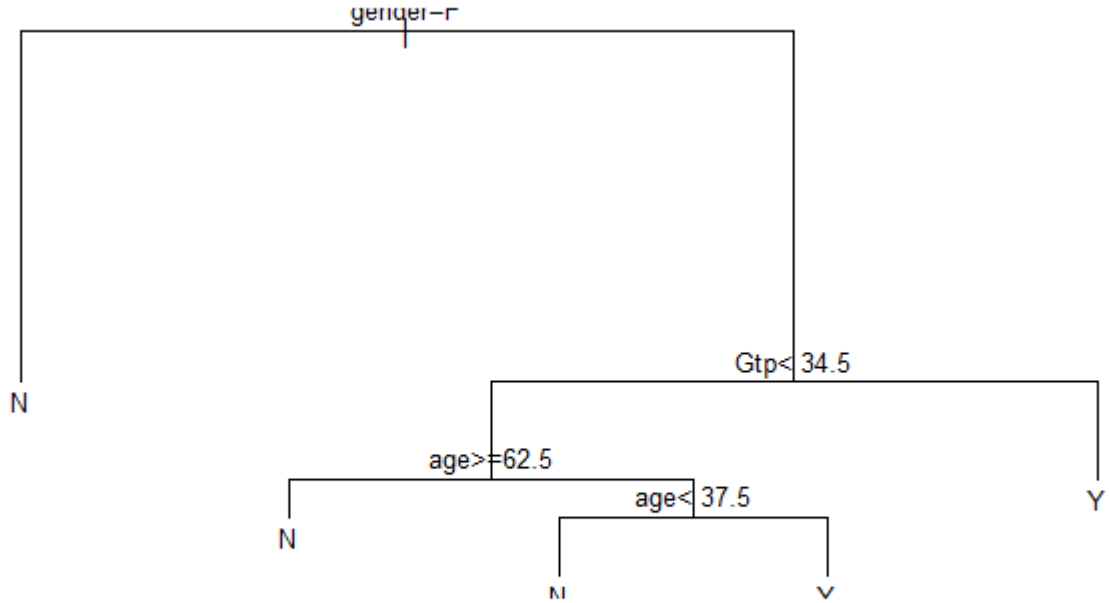
```
    class counts:  3632  2725
   probabilities: 0.571 0.429

Node number 27: 6998 observations
  predicted class=Y  expected loss=0.4652758  P(node) =0.1675405
    class counts:  3256  3742
   probabilities: 0.465 0.535
```

Plot tree.

Hide

```
plot(tree1)
text(tree1, cex=0.8, pretty=0)
```



Calculate accuracy

Hide

```
library(mltools)
```

```
Registered S3 method overwritten by 'data.table':
  method           from
  print.data.table

Attaching package: ‘mltools’

The following object is masked from ‘package:e1071’:

    skewness
```

Hide

```
pred <- predict(tree1, newdata=test, type="class")
acc_t <- mean(pred==test$smoking)
mcc_t <- mcc(factor(pred),test$smoking)
print(paste("Accuracy = ", acc_t))
```

```
[1] "Accuracy =  0.728650434532787"
```

Hide

```
print(paste("mcc = ", mcc_t))
```

```
[1] "mcc =  0.456092798004974"
```

Random Forest

Hide

```
library(randomForest)
```

```
randomForest 4.7-1.1
Type rfNews() to see new features/changes/bug fixes.
```

Hide

```
set.seed(1234)
rf <- randomForest(smoking~., data=train, importance=TRUE)
rf
```

```
Call:
 randomForest(formula = smoking ~ ., data = train, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 17.1%
Confusion matrix:
      N     Y class.error
N 22357  4040   0.1530477
Y  3101 12271   0.2017304
```

Evaluate Forest

Hide

```
pred <- predict(rf, newdata=test, type="response")
acc_rf <- mean(pred==test$smoking)
mcc_rf <- mcc(factor(pred),test$smoking)
print(paste("Accuracy = ", acc_rf))
```

```
[1] "Accuracy =  0.830568124685772"
```

Hide

```
print(paste("mcc = ", mcc_rf))
```

```
[1] "mcc =  0.641749373503481"
```

Accuracy increased for the forest over the tree.

Hide

```
library(xgboost)
train_label = ifelse(train$smoking=="Y",1,0)
train_matrix = data.matrix(subset(train,select=-c(smoking)))
model <- xgboost(data=train_matrix,label=train_label,nrounds=100,objective="binary:logistic")
```

```
[1] train-logloss:0.599024
[2] train-logloss:0.545346
[3] train-logloss:0.511779
[4] train-logloss:0.489393
[5] train-logloss:0.473748
[6] train-logloss:0.462328
[7] train-logloss:0.454266
[8] train-logloss:0.447463
[9] train-logloss:0.442383
[10]    train-logloss:0.437945
[11]    train-logloss:0.433428
[12]    train-logloss:0.430051
[13]    train-logloss:0.427314
[14]    train-logloss:0.423531
[15]    train-logloss:0.420867
[16]    train-logloss:0.418409
[17]    train-logloss:0.416098
[18]    train-logloss:0.413740
[19]    train-logloss:0.411437
[20]    train-logloss:0.410234
[21]    train-logloss:0.407516
[22]    train-logloss:0.405939
[23]    train-logloss:0.403908
[24]    train-logloss:0.402059
[25]    train-logloss:0.400740
[26]    train-logloss:0.399838
[27]    train-logloss:0.398220
[28]    train-logloss:0.396992
[29]    train-logloss:0.396807
[30]    train-logloss:0.394506
[31]    train-logloss:0.392787
[32]    train-logloss:0.391148
[33]    train-logloss:0.390478
[34]    train-logloss:0.388359
[35]    train-logloss:0.386563
[36]    train-logloss:0.384929
[37]    train-logloss:0.383671
[38]    train-logloss:0.382331
[39]    train-logloss:0.381672
[40]    train-logloss:0.381134
[41]    train-logloss:0.379261
[42]    train-logloss:0.377979
[43]    train-logloss:0.377429
[44]    train-logloss:0.377033
[45]    train-logloss:0.376545
[46]    train-logloss:0.375696
[47]    train-logloss:0.374178
[48]    train-logloss:0.372486
[49]    train-logloss:0.370867
[50]    train-logloss:0.369149
[51]    train-logloss:0.368131
[52]    train-logloss:0.366718
```

```
[53]    train-logloss:0.364809
[54]    train-logloss:0.364198
[55]    train-logloss:0.363082
[56]    train-logloss:0.362218
[57]    train-logloss:0.361109
[58]    train-logloss:0.360377
[59]    train-logloss:0.360039
[60]    train-logloss:0.359513
[61]    train-logloss:0.358620
[62]    train-logloss:0.357704
[63]    train-logloss:0.356981
[64]    train-logloss:0.356430
[65]    train-logloss:0.355661
[66]    train-logloss:0.354708
[67]    train-logloss:0.353604
[68]    train-logloss:0.353146
[69]    train-logloss:0.352289
[70]    train-logloss:0.351757
[71]    train-logloss:0.350532
[72]    train-logloss:0.349914
[73]    train-logloss:0.349121
[74]    train-logloss:0.348403
[75]    train-logloss:0.347136
[76]    train-logloss:0.345795
[77]    train-logloss:0.343903
[78]    train-logloss:0.342439
[79]    train-logloss:0.341777
[80]    train-logloss:0.340328
[81]    train-logloss:0.339173
[82]    train-logloss:0.338779
[83]    train-logloss:0.337830
[84]    train-logloss:0.337070
[85]    train-logloss:0.335271
[86]    train-logloss:0.333833
[87]    train-logloss:0.333599
[88]    train-logloss:0.332666
[89]    train-logloss:0.331975
[90]    train-logloss:0.331914
[91]    train-logloss:0.330897
[92]    train-logloss:0.329966
[93]    train-logloss:0.328682
[94]    train-logloss:0.327927
[95]    train-logloss:0.326332
[96]    train-logloss:0.325646
[97]    train-logloss:0.324674
[98]    train-logloss:0.323615
[99]    train-logloss:0.322951
[100]   train-logloss:0.321981
```

Evaluate.

Hide

```
test_label = ifelse(test$smoking=="Y",1,0)
test_matrix = data.matrix(subset(test,select=-c(smoking)))

probs <- predict(model, test_matrix)
pred <- ifelse(probs>0.5, 1, 0)

acc_xg <- mean(pred==test_label)
mcc_xg <- mcc(pred, test_label)

print(paste("accuracy = ", acc_xg))
```

```
[1] "accuracy =  0.780794369029663"
```

Hide

```
print(paste("mcc = ", mcc_xg))
```

```
[1] "mcc =  0.532711729259617"
```

XGBoost did better than the original tree, but the forest still had higher accuracy.

Try Adabag

Hide

```
library(adabag)
```

```
Loading required package: caret
Loading required package: ggplot2

Attaching package: ‘ggplot2’

The following object is masked from ‘package:randomForest’:

    margin

Loading required package: lattice
Loading required package: foreach
Loading required package: doParallel
Loading required package: iterators
Loading required package: parallel
```

Hide

```
adab1 <- boosting(smoking~., data=train, boos=TRUE, mfinal=20, coeflearn='Breiman')
summary(adab1)
```

```
           Length Class   Mode
formula         3  formula call
trees          20  -none-  list
weights        20  -none-  numeric
votes       83538  -none-  numeric
prob        83538  -none-  numeric
class       41769  -none-  character
importance     24  -none-  numeric
terms           3  terms   call
call            6  -none-  call
```

Evaluate.

Hide

```
pred <- predict(adab1, newdata=test, type="response")
acc_adabag <- mean(pred$class==test$smoking)
mcc_adabag <- mcc(factor(pred$class), test$smoking)
print(paste("accuracy=", acc_adabag))
print(paste("mcc=", mcc_adabag))
```

This one took significantly longer to run, but only had slightly better accuracy than the original decision tree.