Machine Learning Hw4

Ekta Chaudhary

20/04/2020

```
library(ISLR)
library(caret)
library(rpart)
library(rpart.plot)
library(party)
library(partykit)
library(randomForest)
library(ranger)
library(gbm)
library(plotmo)
library(pdp)
library(lime)
library(lasso2)
```

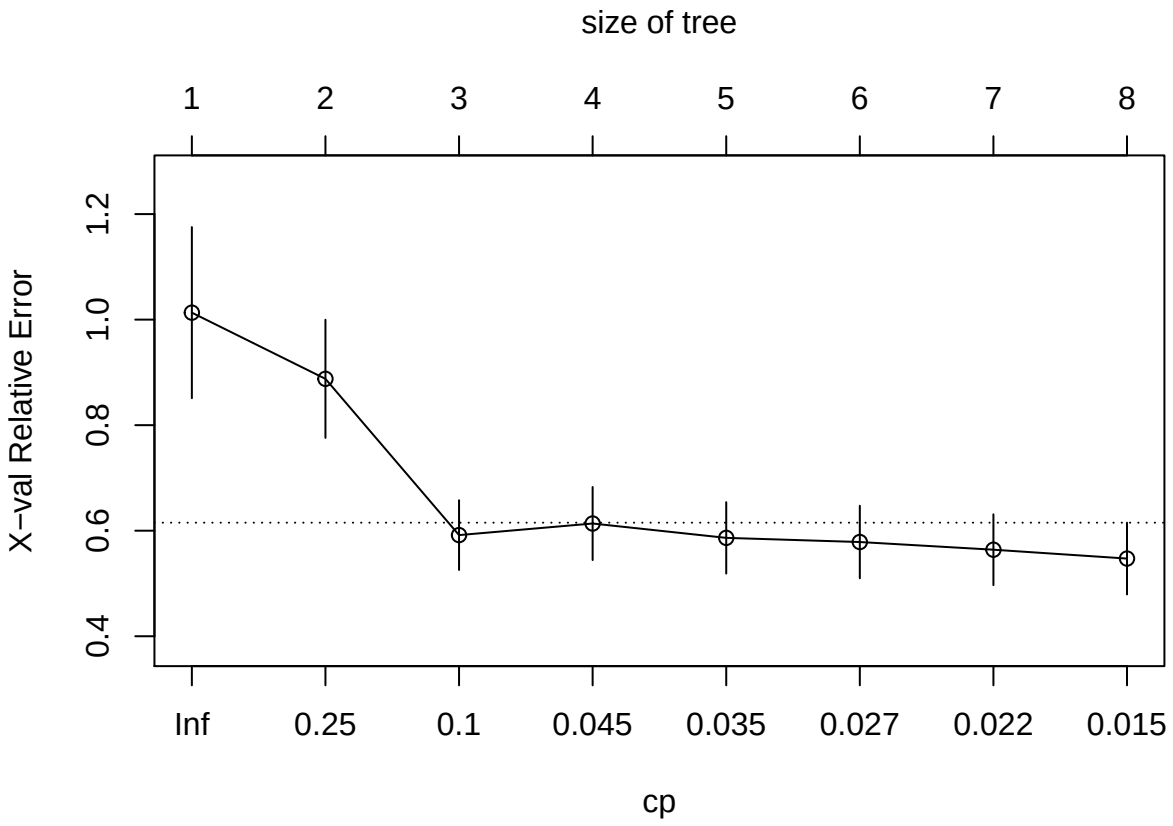
(a) Fit a regression tree with lpsa as the response and the other variables as predictors. Use cross-validation to determine the optimal tree size. Which tree size corresponds to the lowest cross-validation error? Is this the same as the tree size obtained using the 1 SE rule?

```
set.seed(1)
data("Prostate")
ctrl<- trainControl(method = "cv")
```

```
set.seed(1)
tree <- rpart(formula = lpsa~., data = Prostate,
              control = rpart.control(cp = 0.01))
cpTable <- printcp(tree)
```

```
##
## Regression tree:
## rpart(formula = lpsa ~ ., data = Prostate, control = rpart.control(cp = 0.01))
##
## Variables actually used in tree construction:
## [1] lcavol  lweight pgg45
##
## Root node error: 127.92/97 = 1.3187
##
## n= 97
##
##         CP nsplit rel error  xerror     xstd
## 1 0.347108      0   1.00000 1.01323 0.162162
## 2 0.184647      1   0.65289 0.88779 0.111915
## 3 0.059316      2   0.46824 0.59168 0.066102
## 4 0.034756      3   0.40893 0.61359 0.069269
## 5 0.034609      4   0.37417 0.58640 0.067630
## 6 0.021564      5   0.33956 0.57853 0.068772
## 7 0.021470      6   0.31800 0.56398 0.067155
## 8 0.010000      7   0.29653 0.54721 0.068034
```

```
plotcp(tree)
```




```
minErr <- which.min(cpTable[,4])
minErr
```

```
## 8
## 8
```

The tree size 8 corresponds to the lowest cross-validation error.

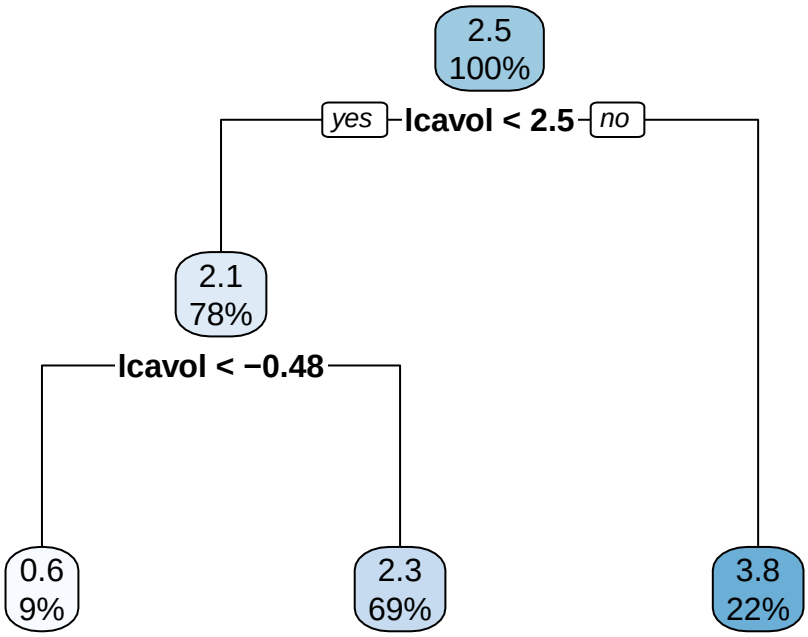
```
cpTable[cpTable[,4]<cpTable[minErr,4]+cpTable[minErr,5],1][1]
```

```
##          3
## 0.05931585
```

The tree size obtained using the 1 SE rule is 3.

(b) Create a plot of the final tree you choose. Pick one of the terminal nodes, and interpret the information displayed.

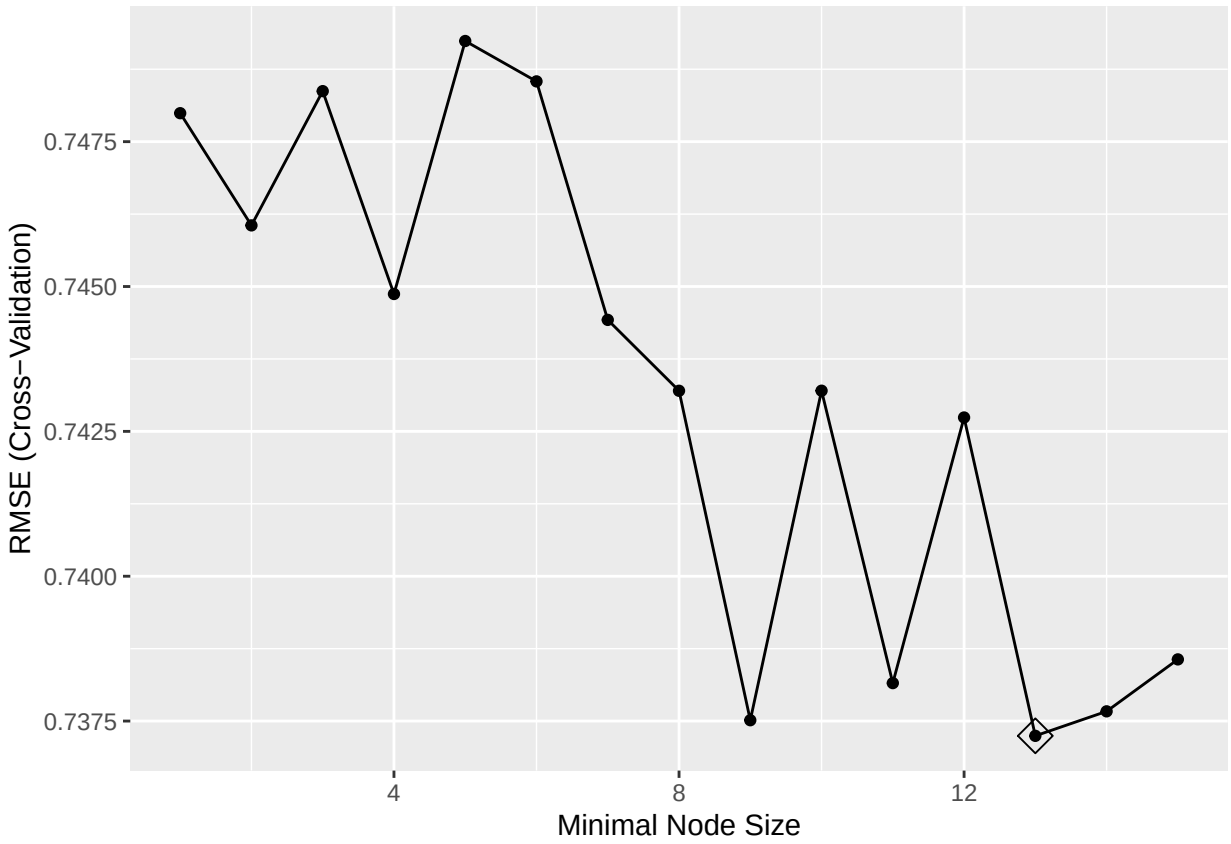
```
tree_a = prune(tree, cp = cpTable[cpTable[,4] < cpTable[minErr,4] + cpTable[minErr,5], 1][1])
rpart.plot(tree_a)
```



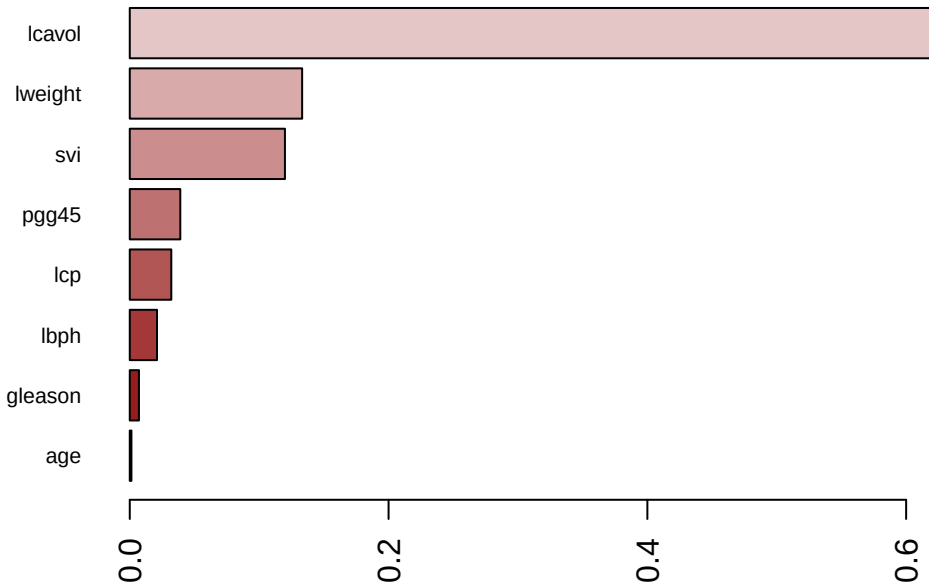
(c) Perform bagging and report the variable importance

```
bagging.grid <- expand.grid(mtry = 6,
                            splitrule = "variance",
                            min.node.size = 1:15)
set.seed(1)
bagging<- train(lpsa~., Prostate,
                method = "ranger",
                tuneGrid = bagging.grid,
                trControl = ctrl,
                importance = "permutation")

ggplot(bagging, highlight = TRUE)
```




```
barplot(sort(ranger::importance(bagging$finalModel), decreasing = FALSE),
        las = 2, horiz = TRUE, cex.names = 0.7,
        col = colorRampPalette(colors = c("darkred","white","darkblue"))(19))
```



The important variables are : lcavol, lweight, svi