

Homework 4

Yanhao Li

Contents

| | |
|------------|----|
| Question 1 | 2 |
| Question 2 | 15 |

```
library(tidyverse)
library(ISLR)
library(lasso2)
library(ISLR)
library(rpart)
library(rpart.plot)
library(randomForest)
library(ranger)
library(caret)
library(gbm)
```

Question 1

Load, clean, and tidy data

```
data("Prostate")

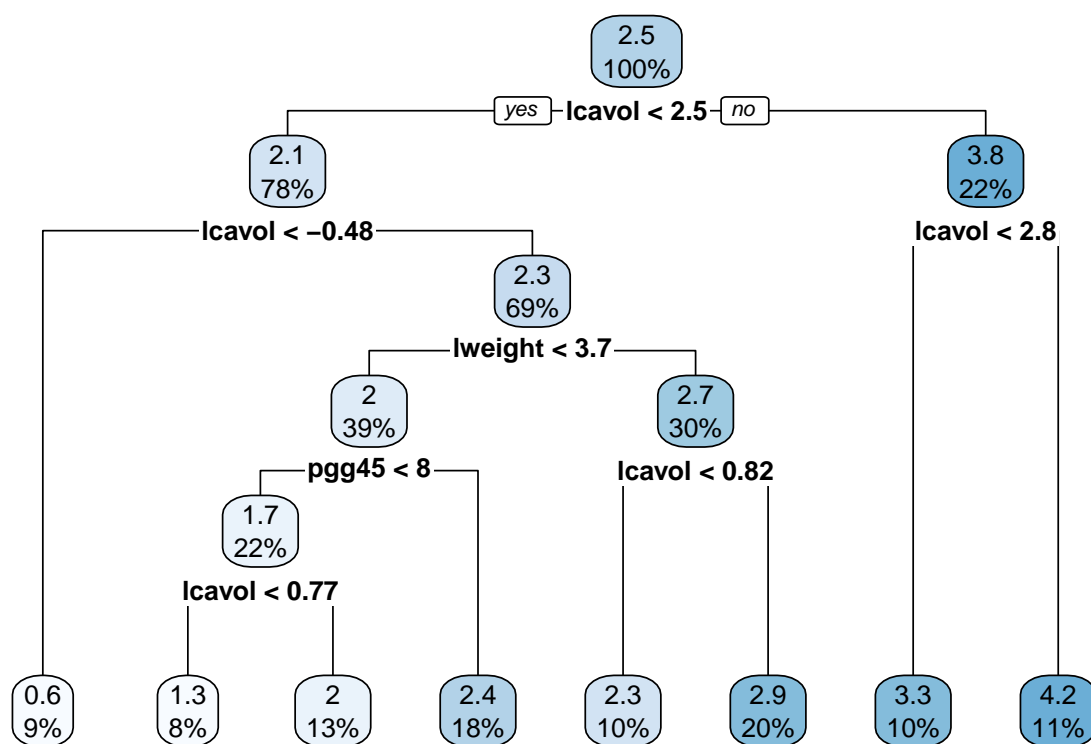
prostate = Prostate %>%
  janitor::clean_names()
```

Question a

```
set.seed(1)

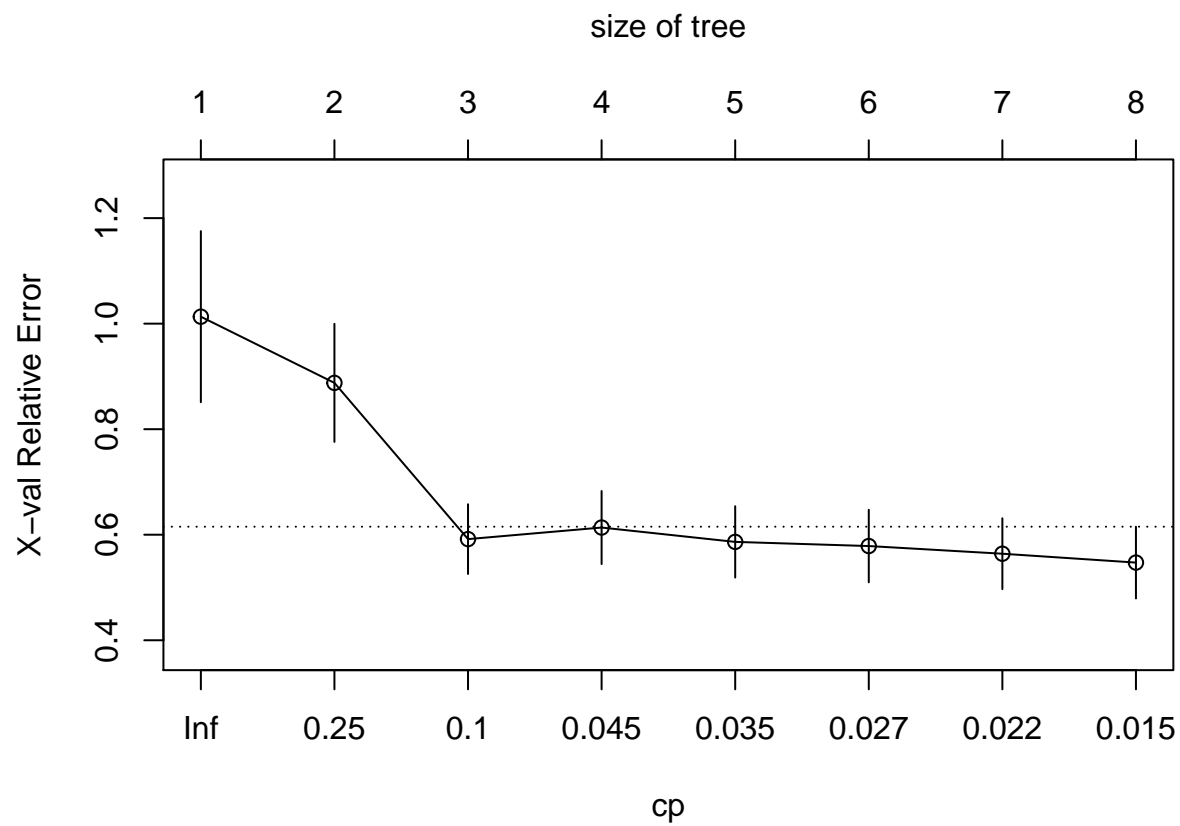
tree1 = rpart(formula = lpsa ~ .,
              data = prostate)

rpart.plot(tree1)
```



```
cpTable <- tree1$cptable
```

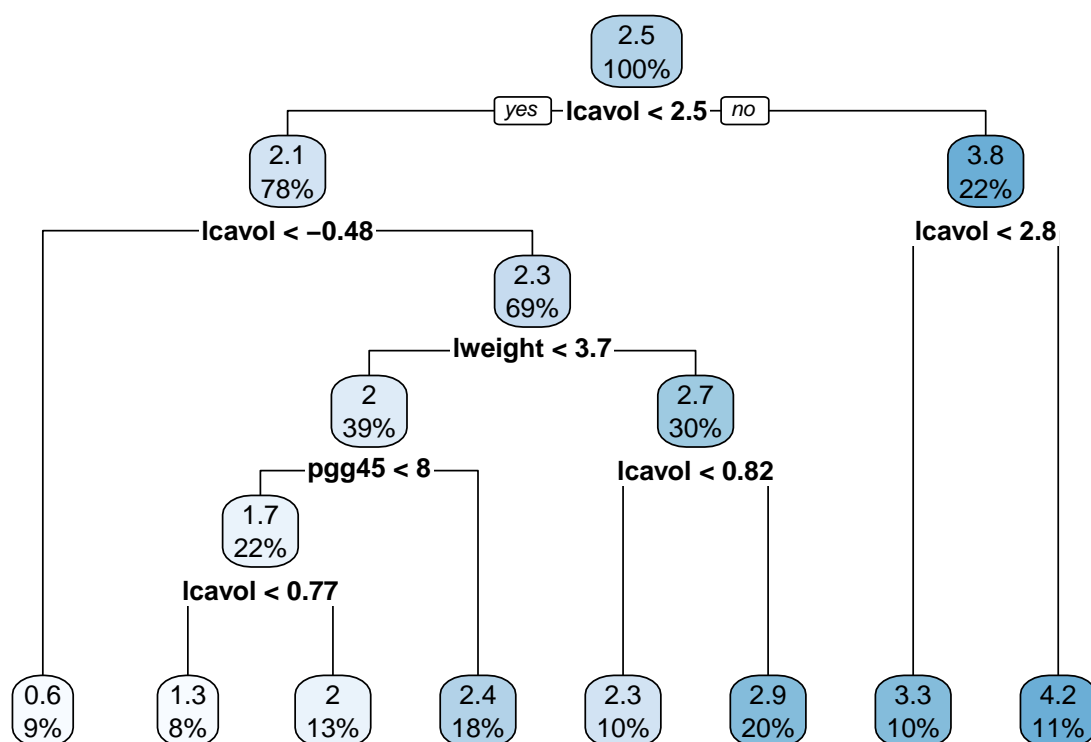
```
plotcp(tree1)
```



```
# minimum cross-validation error
minErr <- which.min(cpTable[,4])

tree2 <- prune(tree1, cp = cpTable[minErr,1])

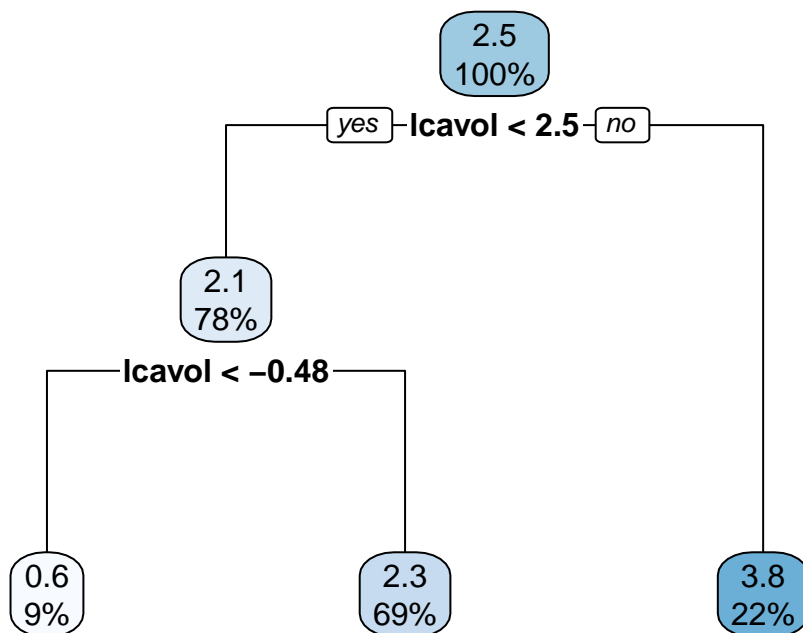
rpart.plot(tree2)
```



```

# 1SE rule
tree3 <- prune(tree1, cp = cpTable[cpTable[,4] < cpTable[minErr,4] + cpTable[minErr,5],1][1])
rpart.plot(tree3)

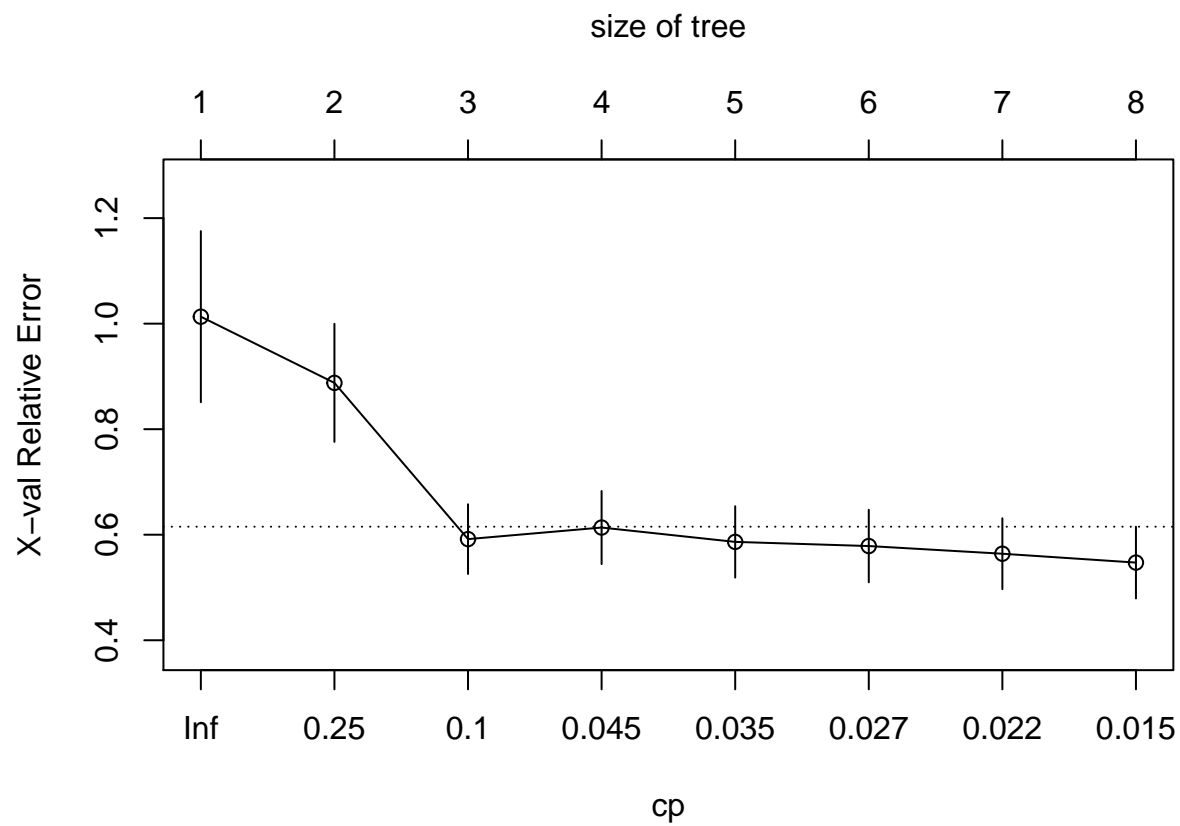
```



Tree size corresponds to the lowest cross-validation error is 8. It is different from the tree size obtained using the 1 SE rule, which is 3.

Question b

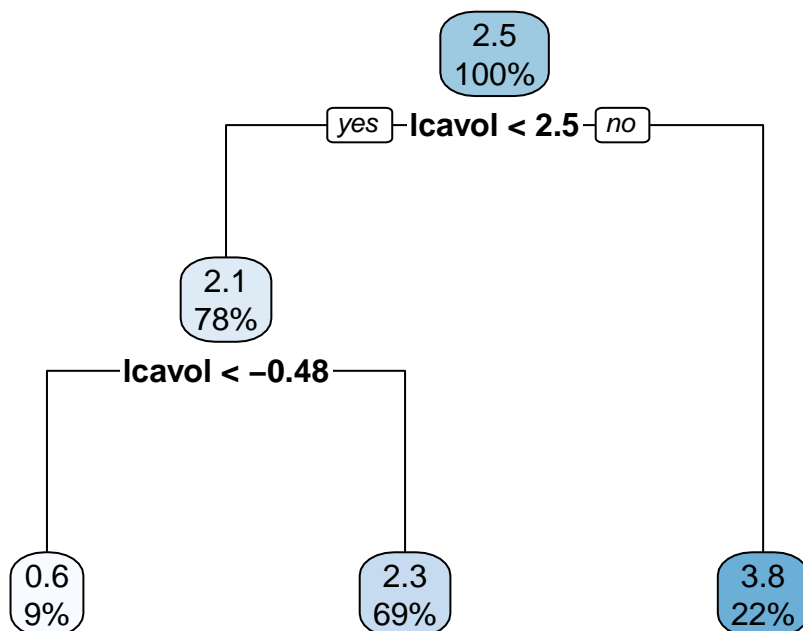
```
plotcp(tree1)
```



```
set.seed(1)

tree4 <- rpart(formula = lpsa ~ .,
               data = prostate,
               control = rpart.control(cp = 0.1))

rpart.plot(tree4)
```



A good choice of cp for pruning is often the leftmost value for which the mean lies below the horizontal line. According to the plot, I choose cp equals to 0.1 and size of tree equals to 3.

In terminal node where $lcavol$ is less than -0.48, the mean $lpsa$ is 0.6. This node contains 9% of the sample.

Question c

```

ctrl <- trainControl(method = "cv")

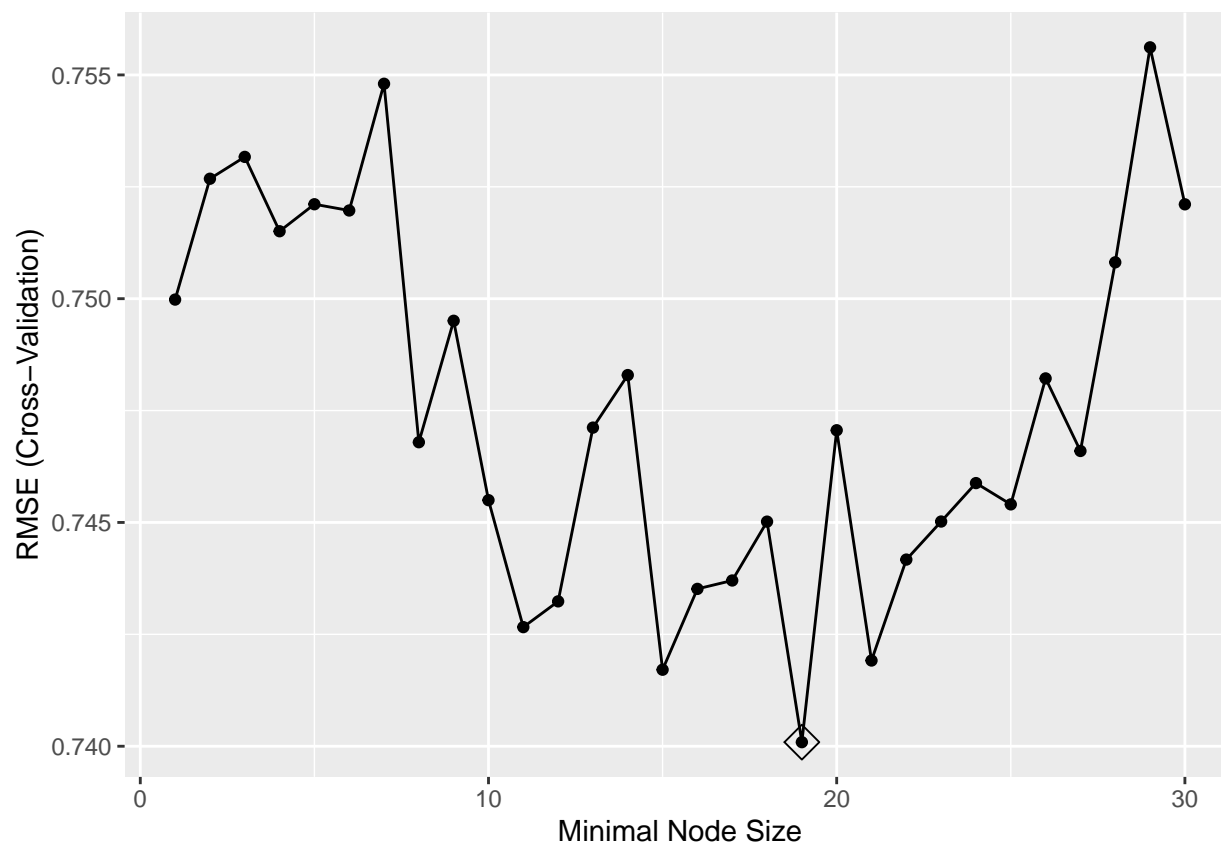
bag.grid <- expand.grid(mtry = 8,
  splitrule = "variance",
  min.node.size = 1:30)

set.seed(1)

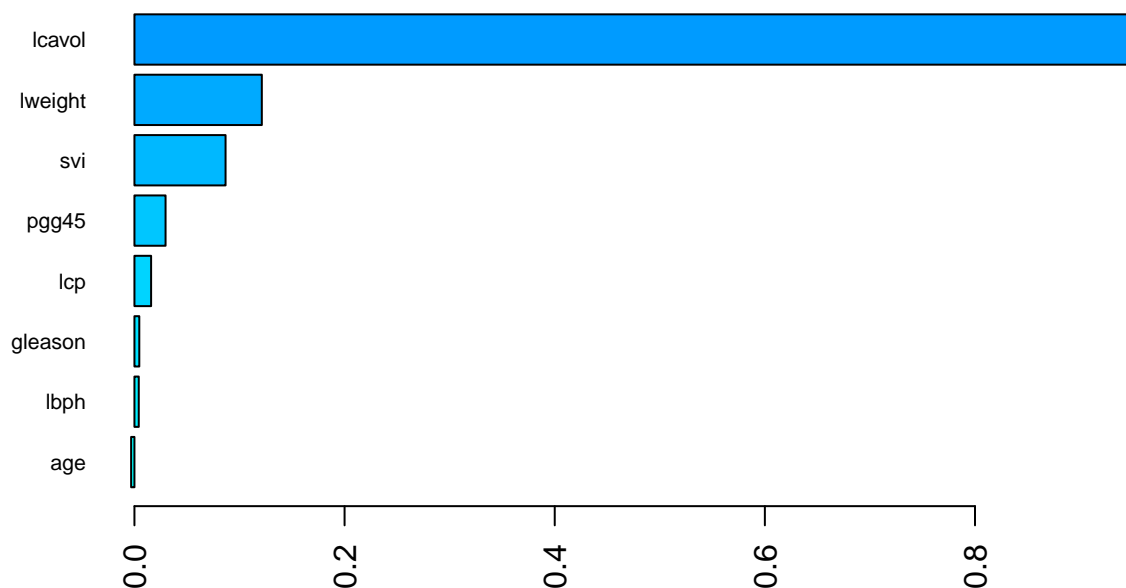
bag.fit <- train(lpsa~.,
  Prostate,
  method = "ranger",
  tuneGrid = bag.grid,
  trControl = ctrl,
  importance = "permutation")

ggplot(bag.fit, highlight = TRUE)

```

```
barplot(sort(ranger::importance(bag.fit$finalModel), decreasing = FALSE),
  las = 2, horiz = TRUE, cex.names = 0.7,
  col = colorRampPalette(colors = c("cyan", "blue"))(19))
```



According to the plot, variable importance from highest to lowest is lcavol, lweight, svi, pgg45, lcp, gleason, lbph, and age.

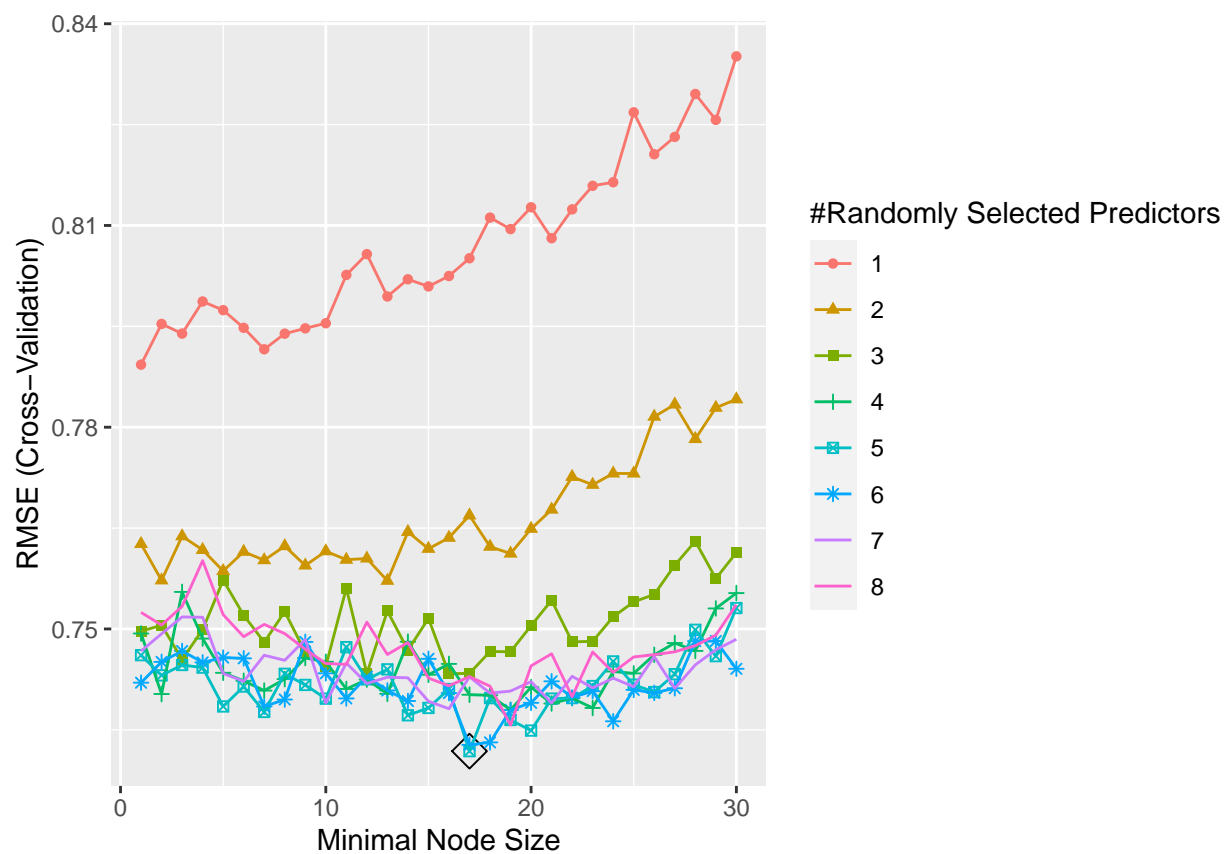
Question d

```
rf.grid <- expand.grid(mtry = 1:8,
                      splitrule = "variance",
                      min.node.size = 1:30)

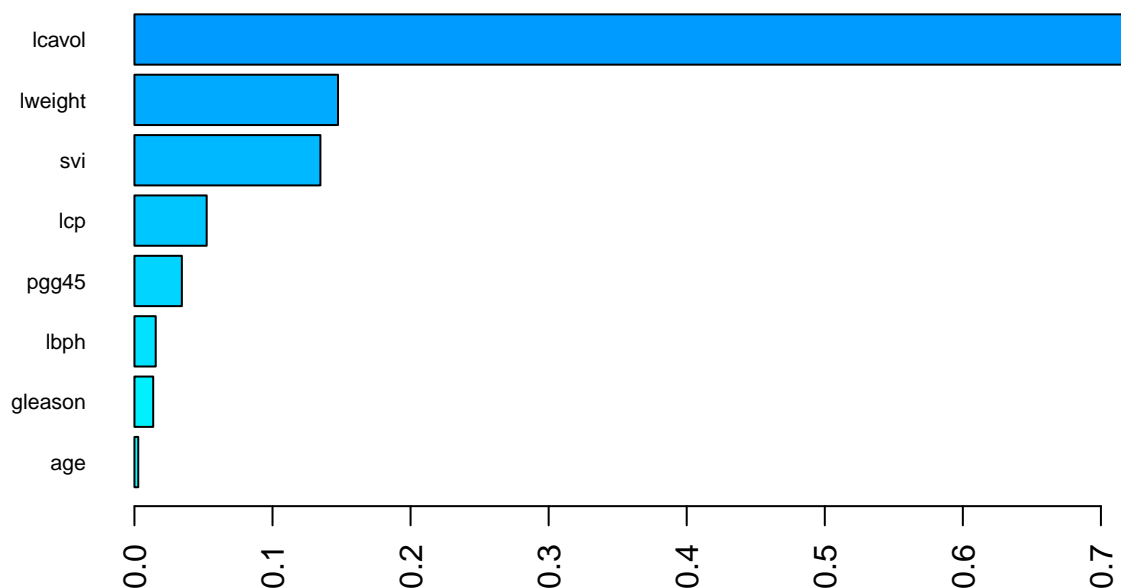
set.seed(1)

rf.fit <- train(lpsa ~ . ,
                prostate,
                method = "ranger",
                tuneGrid = rf.grid,
                trControl = ctrl,
                importance = "permutation")

ggplot(rf.fit, highlight = TRUE)
```



```
barplot(sort(ranger::importance(rf.fit$finalModel), decreasing = FALSE),
  las = 2, horiz = TRUE, cex.names = 0.7,
  col = colorRampPalette(colors = c("cyan","blue"))(19))
```



According to the plot, variable importance from highest to lowest is lcavol, lweight, svi, lcp, pgg45, lbph, gleason, and age.

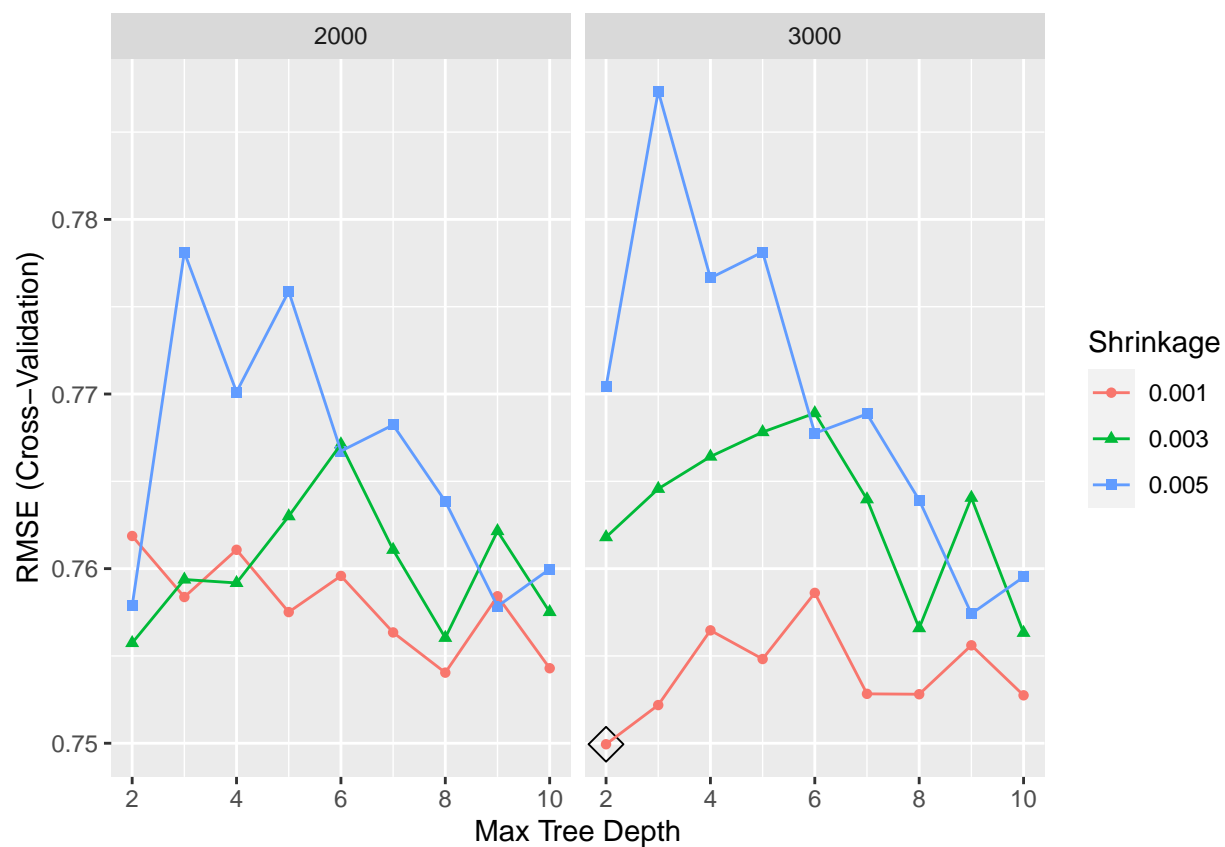
Question e

```
gbm.grid <- expand.grid(n.trees = c(2000,3000),
                      interaction.depth = 2:10,
                      shrinkage = c(0.001,0.003,0.005),
                      n.minobsinnode = 1)

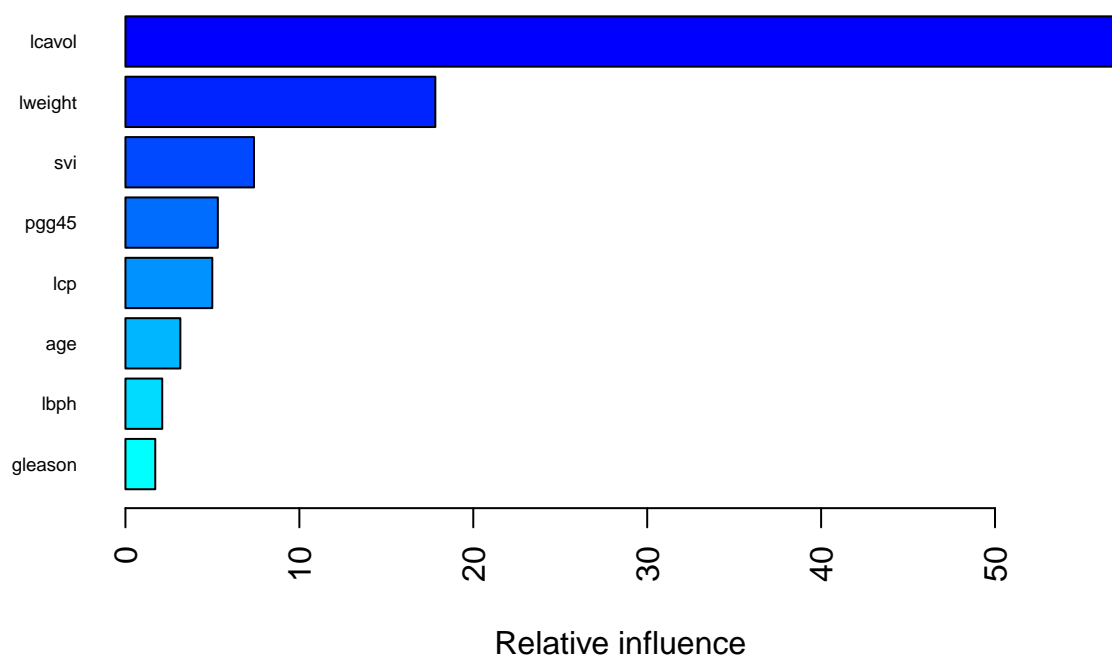
set.seed(1)

gbm.fit <- train(lpsa ~ . ,
                prostate,
                method = "gbm",
                tuneGrid = gbm.grid,
                trControl = ctrl,
                verbose = FALSE)

ggplot(gbm.fit, highlight = TRUE)
```



```
summary(gbm.fit$finalModel, las = 2, cBars = 19, cex.names = 0.6)
```



```
##          var  rel.inf
## lcavol  lcavol 57.494112
## lweight lweight 17.818879
## svi      svi   7.392286
## pgg45    pgg45  5.312763
## lcp      lcp   4.994192
## age      age   3.157564
## lbph     lbph  2.115772
## gleason  gleason 1.714432
```

According to the plot, variable importance from highest to lowest is lcavol, lweight, svi, pgg45, lcp, age, lbph, and gleason.

Question f

```
resamp <- resamples(list(bag = bag.fit, rf = rf.fit, bst = gbm.fit))

summary(resamp)
```

```
##
## Call:
## summary.resamples(object = resamp)
##
```

```
## Models: bag, rf, bst
## Number of resamples: 10
##
## MAE
##      Min.    1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## bag 0.4616225 0.5496355 0.6181923 0.6041023 0.6575394 0.7234704    0
## rf  0.4570936 0.5415374 0.6148593 0.6000239 0.6699930 0.7046284    0
## bst 0.4988605 0.5382599 0.6238535 0.6060614 0.6541746 0.7189084    0
##
## RMSE
##      Min.    1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## bag 0.5421937 0.6587298 0.7315537 0.7400910 0.8227738 0.9183309    0
## rf  0.5406496 0.6394524 0.7253799 0.7318267 0.8257673 0.9340790    0
## bst 0.6127814 0.6613586 0.7369285 0.7499428 0.8330904 0.9285269    0
##
## Rsquared
##      Min.    1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## bag 0.4337901 0.5292513 0.6201345 0.6193841 0.6972059 0.7870307    0
## rf  0.3511685 0.5449248 0.6009057 0.6156774 0.7326923 0.7714574    0
## bst 0.3912307 0.4940316 0.6253058 0.6103438 0.7211938 0.8171347    0
```

According to the table, random forest has lower mean RMSE. Consequently, I will choose random forest.

Question 2

Load, clean, and tidy data

Question a

Question b

Question c