Week 11

Jinhong Du

目录

1 Problem 1

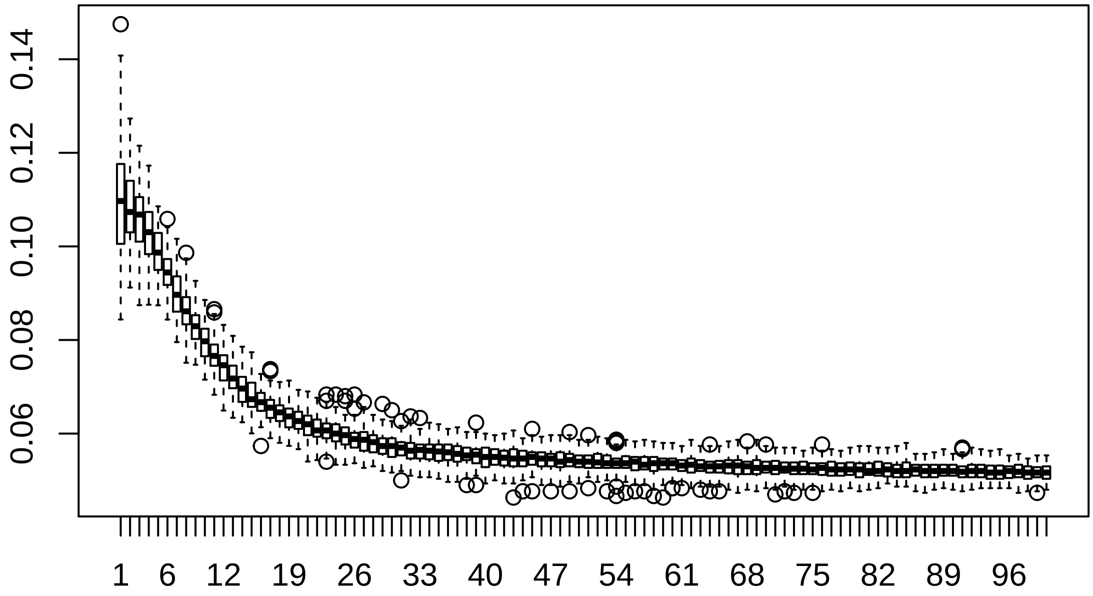
Use a random forest to classify the **spambase** data. Repeat the analysis 100 times using different random seeds to start each replication. For each repetition, find the OOB misclassification rate and draw the boxplot for OOB misclassification rates. Repeat this for different values of m (number of variables selected as candidates for splitting) and B (number of bootstrap trees in the forest). What can you say about the effect of m and B on the OOB misclassification rate?

```
load('spamtrain.rda')
load('spamtest.rda')

library(randomForest)
```

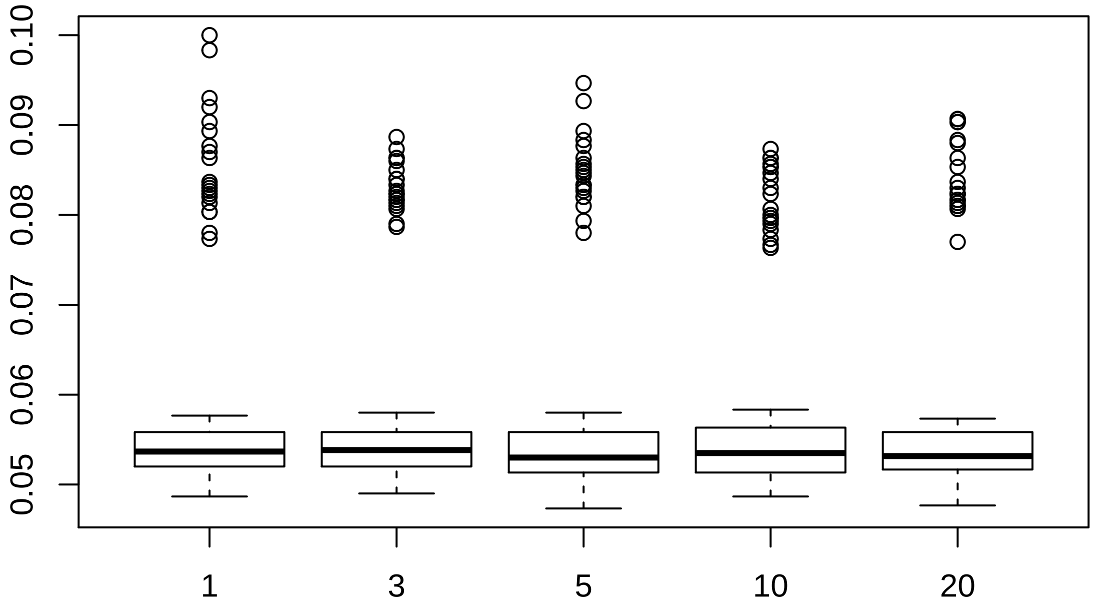
```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
OOB_rate <- matrix(0,100,100)
for (i in 1:100) {
    set.seed(i)
    spam.rf <- randomForest(class~.,
                            data = spam.train,
                            ntree = 100)
    OOB_rate[i,] <- spam.rf$err.rate[,1]
}

dataset <- data.frame(value = as.vector(OOB_rate),
                      group = factor(rep(c(1:100), each = 100)))
boxplot(value ~ group, dataset)
```



```
m_list <- c(1,3,5,10,20)
OOB_rate <- matrix(0, 5, 100)
B <- 100
for (m in 1:5) {
    for (i in 1:100) {
        spam.rf <- randomForest(class~.,
                                data = spam.train,
                                ntree = B,
                                mtry = m_list[m])
        OOB_rate[m,i] <- spam.rf$err.rate[B,1]
    }
}

dataset <- data.frame(value = as.vector(OOB_rate),
                      group = factor(rep(m_list, each = 100)))
boxplot(value ~ group, dataset)
```



2 Problem 2

2.1 Bagging Classified Tree

```
load('spamtrain.rda')
load('spamtest.rda')

library(ipred)
library(rpart)
spam.bag <- bagging(class~.,
                    data = spam.train,
                    nbagg = 100, # Bootstrap 样本抽样次数
                    control = rpart.control(minsplit = 2,
                                            cp = 0,
                                            xval = 0))

pre.bag <- predict(spam.bag,
                   newdata = spam.test)
mean(pre.bag != spam.test$class)
```

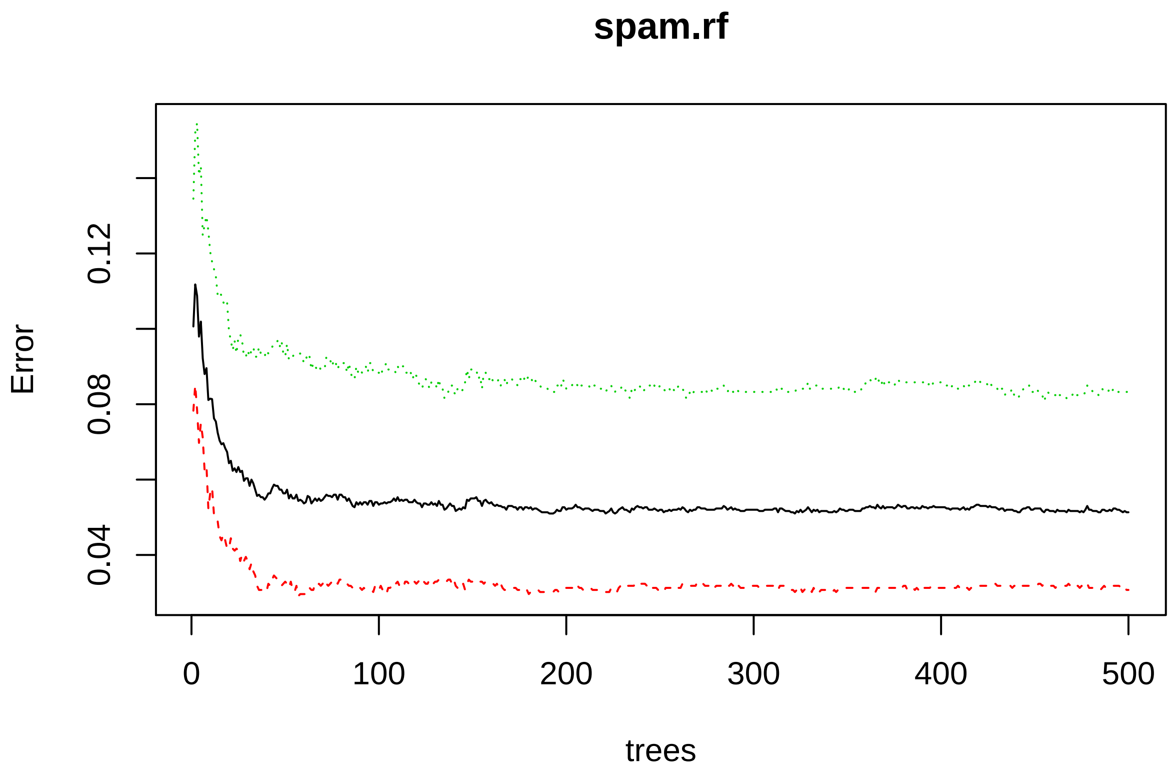
```
## [1] 0.06121174
```

```
# 添加 coob = T 来得到 OOB 估计
spam.bag <-  bagging(class~.,
                     data = spam.train,
                     nbagg = 100,
                     control = rpart.control(minsplit = 2,
                                             cp = 0,
                                             xval = 0),
                     coob = T)
spam.bag
```

```
##
## Bagging classification trees with 100 bootstrap replications
##
## Call: bagging.data.frame(formula = class ~ ., data = spam.train, nbagg = 100,
##     control = rpart.control(minsplit = 2, cp = 0, xval = 0),
##     coob = T)
##
## Out-of-bag estimate of misclassification error:  0.0573
```

2.2 Random Forest

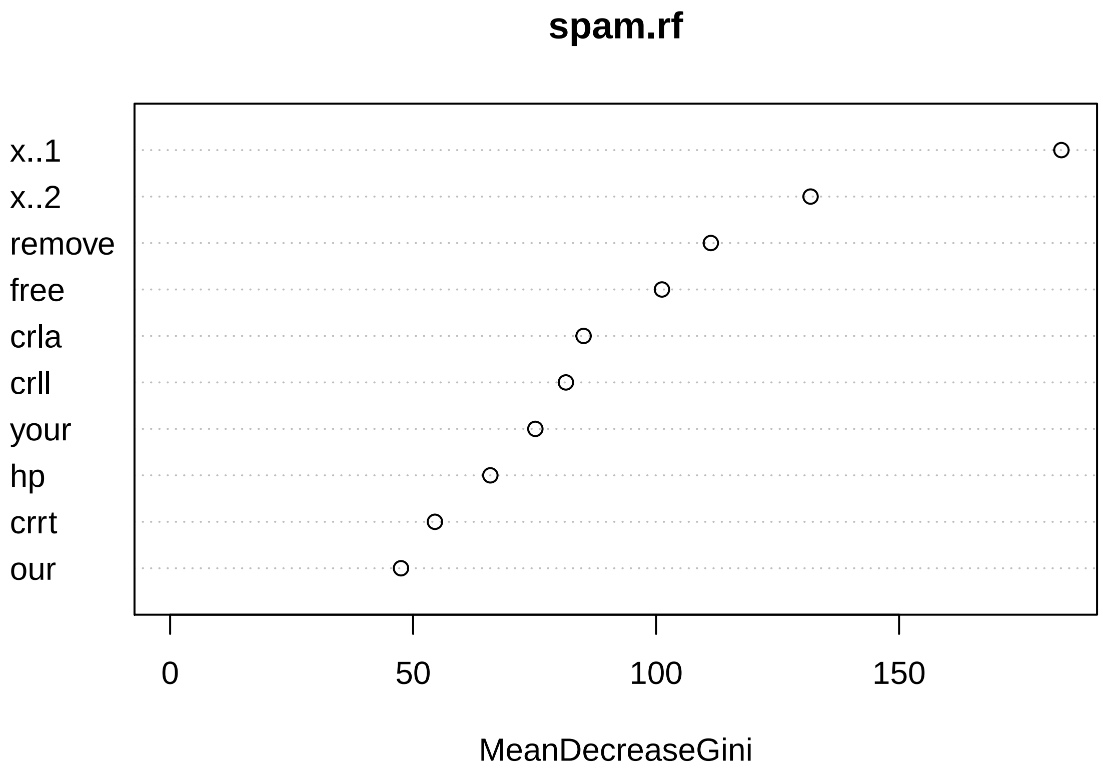
```
library(randomForest)
spam.rf <- randomForest(class~.,
                          data = spam.train)
plot(spam.rf)
```



```
pre.rf <- predict(spam.rf,
                  newdata = spam.test)
mean(pre.bag != spam.test$class)
```

```
## [1] 0.06121174
```

```
varImpPlot(spam.rf,
           n.var = 10)
```



2.3 Adaboost

```
library(ada)
spam.ada <- ada(class~.,
                data = spam.train,
                iter = 50,
                loss = "logistic",
                type = "discrete")

pre.ada <- predict(spam.ada,
                   newdata = spam.test,
                   type = "vector")
mean(pre.ada != spam.test$class)
```

```
## [1] 0.04996877
```