Tutorial 6

DSA1101
Introduction to Data Science

October 12, 2018

Exercise 1. *n*-fold cross validation for *decision trees*

Recall that we studied n-fold cross-validation for the k-nearest neighbor classifier, in which the value of k is varied to control the complexity of the decision surface for the classifier. For *decision tree* classification, a similar complexity parameter exists, which is denoted as C_p. Heuristically, smaller values of C_p correspond to decision trees of larger sizes, and hence more complex decision surfaces. For this week's tutorial, we will investigate n-fold cross validation for a *decision tree* classifier.

(a) Consider the dataset 'bank-sample.csv' we discussed in the lectures. For this exercise, we will fit a decision tree with `subscribed` as outcome and `job`, `marital`, `education`, `default`. `housing`, `loan`, `contact` and `poutcome` as feature variables. We want to find the best C_p value in terms of misclassification error rate.

1. Randomly split the entire dataset into 10 mutually exclusive datasets
2. Let C_p take on the values 10^k for $k = -5, -4, -3, ..., 0, ..., 3, 4, 5$
3. At each C_p value, run the following loop for $j = 1, 2, ..., 10$:
   (a) Set the j^{th} group to be the test set
   (b) Fit a decision tree on the other 9 sets with the value of C_p
   (c) Predict the class assignment of `subscribed` for each observation of the test set
   (d) Calculate the number of misclassification(s) by comparing predicted versus actual class labels in the test set
4. Determine the best C_p value in terms of misclassification error rate.

```r
library("rpart")
library("rpart.plot")


#CV for decision tree

banktrain <- read.table("bank-sample.csv",header=TRUE,sep=",")

## drop a few columns to simplify the tree
drops<-c("age", "balance", "day", "campaign",
         "pdays", "previous", "month", "duration")
banktrain <- banktrain [,!(names(banktrain) %in% drops)]

## total records in dataset
n=dim(banktrain)[1]

## Randomly split into 10 datasets
## We have seen this code before
n_folds=10
folds_j <- sample(rep(1:n_folds, length.out = n))
table(folds_j)

cp=10^(-5:5)
misC=rep(0,length(cp))

for(i in 1:length(cp)){

      misclass=0
  for (j in 1:n_folds) {
    test <- which(folds_j == j)
    train=banktrain[-c(test),]
    fit <- rpart(subscribed ~ job + marital +
                education+default + housing +
                loan + contact+poutcome,
                method="class",
                data=train,
                control=rpart.control(cp=cp[i]),
                parms=list(split='information'))

    new.data=data.frame(banktrain[test,c(1:8)])
             ##predict label for test data based on fitted tree
    prd=predict(fit,new.data,type='class')
             misclass = misclass + sum(prd!=banktrain[test,9])
       }
       misC[i]=misclass/n
}

plot(log(cp,base=10),misC,type='b')
```

(b) Plot the *decision tree* fitted with the best C_p value in terms of misclassification rate

```
## determine the best cp in terms of
## misclassification rate

best.cp =cp[which(misC==min(misC))]

## Fit decision tree
fit <- rpart(subscribed ~ job + marital +
             education+default + housing +
             loan + contact+poutcome,
             method="class",
             data=train,
             control=rpart.control(cp=best.cp),
             parms=list(split='information'))

## Plot the tree
rpart.plot(fit, type=4, extra=2)
```