Feature Selection on Sonar dataset

Mengyao Wang

2/22/2021

Libraries to load

```
library(mlbench)
```

Feature selection and classification

The task is to train a network to discriminate between sonar signals bounced off a metal cylinder and those bounced off a roughly cylindrical rock. Each pattern is a set of 60 numbers in the range 0.0 to 1.0. Each number represents the energy within a particular frequency band, integrated over a certain period of time. The integration aperture for higher frequencies occur later in time, since these frequencies are transmitted later during the chirp. The label associated with each record contains the letter “R” if the object is a rock and “M” if it is a mine (metal cylinder). The numbers in the labels are in increasing order of aspect angle, but they do not encode the angle directly.

UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science.

```
# load data
library(mlbench)
library(class)
data(Sonar)
dim(Sonar)
```

```
## [1] 208  61
```

Partition the data into training and testing sets

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
set.seed(2021)
inTrain <- createDataPartition(Sonar$Class, p = 0.5)[[1]]
SonarTrain <- Sonar[ inTrain,]
SonarTest  <- Sonar[-inTrain,]
```

Wrapper feature selection

Forward stepwise selection

```r
selectFeature <- function(train, test, cls.train, cls.test, features) {
  ## identify a feature to be selected
  current.best.accuracy <- -Inf
  selected.i <- NULL
  for(i in seq(ncol(train))) {
    current.f <- colnames(train)[i]
    if(!current.f %in% features) {
      model <- knn(train = cbind(train[, features], train[, current.f]),
                   test = cbind(test[, features], test[, current.f]), cl = cls.train, k = 3)
      test.acc <- sum(model == cls.test) / length(cls.test)

      if(test.acc > current.best.accuracy) {
        current.best.accuracy <- test.acc
        selected.i <- colnames(train)[i]
      }
    }
  }
  return(selected.i)
}


##
library(caret)
set.seed(1)
inTrain <- createDataPartition(Sonar$Class, p = .6)[[1]]
allFeatures <- colnames(Sonar)[-61]
train <- Sonar[ inTrain,-61]
test  <- Sonar[-inTrain,-61]
cls.train <- Sonar$Class[inTrain]
cls.test <- Sonar$Class[-inTrain]

features <- NULL
# select the 1 to 10 best features using knn as a wrapper classifier
for (j in 1:10) {
  selected.i <- selectFeature(train, test, cls.train, cls.test, features)
  print(selected.i)

  # add the best feature from current run
  features <- c(features, selected.i)
}
```

```
## [1] "V48"
## [1] "V1"
## [1] "V34"
## [1] "V45"
## [1] "V36"
## [1] "V43"
## [1] "V51"
## [1] "V50"
## [1] "V52"
## [1] "V53"
```

Classify on the two types of samples using the full dataset compared to using top 10 wrapper selected features

Fitting the classifier on top 10 wrapper selected features.

```r
knn.fit3 <- knn(train = SonarTrain[, features], test = SonarTest[, features],
                cl = SonarTrain$Class, k = 5, prob = TRUE)
table(knn.fit3, SonarTest$Class)
```

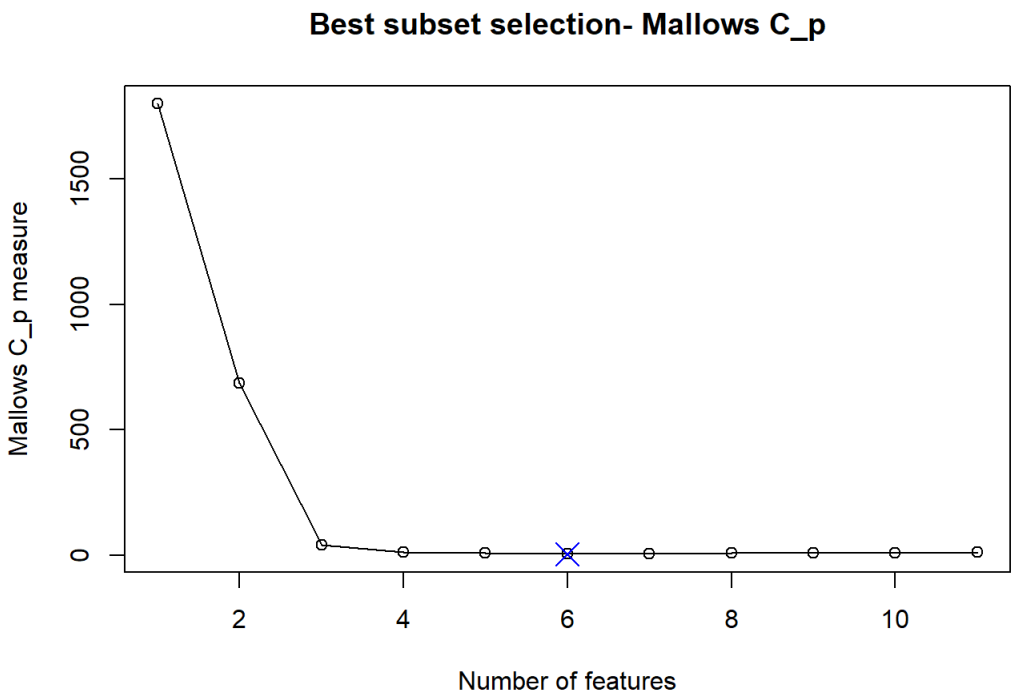
```
##
## knn.fit3  M  R
##        M 40 13
##        R 15 35
```

Starting with exhaustive search (all possible subsets).

```r
# 'leaps' the helper package for subset selection and stepwise search
library(leaps)
library(ISLR)
leaps.credit <- regsubsets(Balance ~ . - ID, data = Credit,
                           method = "exhaustive", nvmax = 11)
summary.leaps.credit <- summary(leaps.credit)
summary.leaps.credit
```

```
## Subset selection object
## Call: regsubsets.formula(Balance ~ . - ID, data = Credit, method = "exhaustive",
##     nvmax = 11)
## 11 Variables  (and intercept)
##                    Forced in Forced out
## Income                 FALSE      FALSE
## Limit                  FALSE      FALSE
## Rating                 FALSE      FALSE
## Cards                  FALSE      FALSE
## Age                    FALSE      FALSE
## Education              FALSE      FALSE
## GenderFemale           FALSE      FALSE
## StudentYes             FALSE      FALSE
## MarriedYes             FALSE      FALSE
## EthnicityAsian         FALSE      FALSE
## EthnicityCaucasian     FALSE      FALSE
## 1 subsets of each size up to 11
## Selection Algorithm: exhaustive
##           Income Limit Rating Cards Age Education GenderFemale StudentYes
## 1  ( 1 )  " "    " "   "*"    " "   " " " "       " "          " "
## 2  ( 1 )  "*"    " "   "*"    " "   " " " "       " "          " "
## 3  ( 1 )  "*"    " "   "*"    " "   " " " "       " "          "*"
## 4  ( 1 )  "*"    "*"   " "    "*"   " " " "       " "          "*"
## 5  ( 1 )  "*"    "*"   "*"    "*"   " " " "       " "          "*"
## 6  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       " "          "*"
## 7  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 8  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 9  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 10  ( 1 ) "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 11  ( 1 ) "*"    "*"   "*"    "*"   "*" "*"       "*"          "*"
##           MarriedYes EthnicityAsian EthnicityCaucasian
## 1  ( 1 )  " "        " "            " "
## 2  ( 1 )  " "        " "            " "
## 3  ( 1 )  " "        " "            " "
## 4  ( 1 )  " "        " "            " "
## 5  ( 1 )  " "        " "            " "
## 6  ( 1 )  " "        " "            " "
## 7  ( 1 )  " "        " "            " "
## 8  ( 1 )  " "        "*"            " "
## 9  ( 1 )  "*"        "*"            " "
## 10  ( 1 ) "*"        "*"            "*"
## 11  ( 1 ) "*"        "*"            "*"
```

```r
min.cp <- which.min(summary.leaps.credit$cp)
plot(summary.leaps.credit$cp, type = 'l',
     main = "Best subset selection- Mallows C_p",
     ylab = "Mallows C_p measure",
     xlab = "Number of features")
points(summary.leaps.credit$cp)
points(min.cp, summary.leaps.credit$cp[min.cp], pch = 4, col = "blue", cex = 2)
```

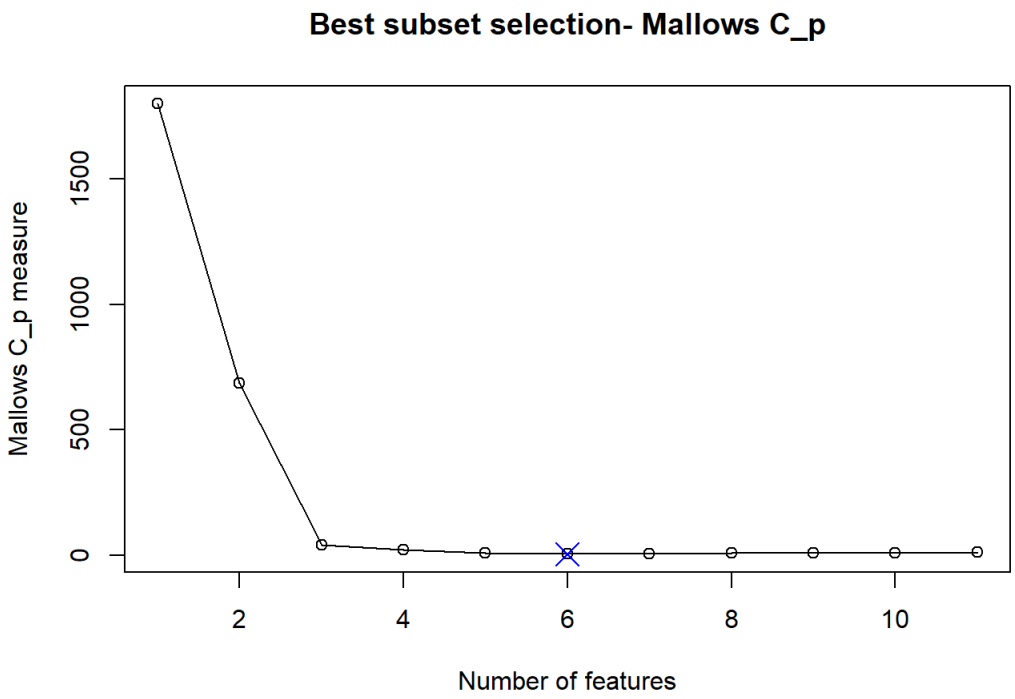


Consider also the forward stepwise search

```
forward.credit <- regsubsets(Balance ~ . - ID, data = Credit,
                             method = "forward", nvmax = 11)
summary.forward.credit <- summary(forward.credit)
summary.forward.credit
```

```
## Subset selection object
## Call: regsubsets.formula(Balance ~ . - ID, data = Credit, method = "forward",
##     nvmax = 11)
## 11 Variables  (and intercept)
##                    Forced in Forced out
## Income                 FALSE      FALSE
## Limit                  FALSE      FALSE
## Rating                 FALSE      FALSE
## Cards                  FALSE      FALSE
## Age                    FALSE      FALSE
## Education              FALSE      FALSE
## GenderFemale           FALSE      FALSE
## StudentYes             FALSE      FALSE
## MarriedYes             FALSE      FALSE
## EthnicityAsian         FALSE      FALSE
## EthnicityCaucasian     FALSE      FALSE
## 1 subsets of each size up to 11
## Selection Algorithm: forward
##           Income Limit Rating Cards Age Education GenderFemale StudentYes
## 1  ( 1 )  " "    " "   "*"    " "   " " " "       " "          " "
## 2  ( 1 )  "*"    " "   "*"    " "   " " " "       " "          " "
## 3  ( 1 )  "*"    " "   "*"    " "   " " " "       " "          "*"
## 4  ( 1 )  "*"    "*"   "*"    " "   " " " "       " "          "*"
## 5  ( 1 )  "*"    "*"   "*"    "*"   " " " "       " "          "*"
## 6  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       " "          "*"
## 7  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 8  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 9  ( 1 )  "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 10  ( 1 ) "*"    "*"   "*"    "*"   "*" " "       "*"          "*"
## 11  ( 1 ) "*"    "*"   "*"    "*"   "*" "*"       "*"          "*"
##           MarriedYes EthnicityAsian EthnicityCaucasian
## 1  ( 1 )  " "        " "            " "
## 2  ( 1 )  " "        " "            " "
## 3  ( 1 )  " "        " "            " "
## 4  ( 1 )  " "        " "            " "
## 5  ( 1 )  " "        " "            " "
## 6  ( 1 )  " "        " "            " "
## 7  ( 1 )  " "        " "            " "
## 8  ( 1 )  " "        "*"            " "
## 9  ( 1 )  "*"        "*"            " "
## 10  ( 1 ) "*"        "*"            "*"
## 11  ( 1 ) "*"        "*"            "*"
```

```
min.cp <- which.min(summary.forward.credit$cp)
plot(summary.forward.credit$cp, type = 'l',
     main = "Best subset selection- Mallows C_p",
     ylab = "Mallows C_p measure",
     xlab = "Number of features")
points(summary.forward.credit$cp)
points(min.cp, summary.forward.credit$cp[min.cp], pch = 4, col = "blue", cex = 2)
```



I share my learning journey into Data Science with my amazing LinkedIn friends, please let me know if you would like to see more small samples like this, thanks for your support! Mengyao Wang