Midterm Supplement

Christopher Daigle

Nov. 5 2018

```
rm(list = ls())
library(boot)
library(ISLR)
set.seed(1)
```

Cross validation Exercise #### Cross validation can be used to estimate the test error for a classification problem. Run a logit model with the insurance data. The dependent variable is lowCharge and independent variables are age, sex, bmi, smoker, and region.

```
setwd('~/Git/MachineLearningAndBigDataWithR/Data')
dataName <- 'insurance.csv'
df <- read.csv(dataName, stringsAsFactors = FALSE)
str(df)
```

```
## 'data.frame':    1338 obs. of  7 variables:
##  $ age     : int  19 18 28 33 32 31 46 37 37 60 ...
##  $ sex     : chr  "female" "male" "male" "male" ...
##  $ bmi     : num  27.9 33.8 33 22.7 28.9 ...
##  $ children: int  0 1 3 0 0 0 1 3 2 0 ...
##  $ smoker  : chr  "yes" "no" "no" "no" ...
##  $ region  : chr  "southwest" "southeast" "southeast" "northwest" ...
##  $ charges : num  16885 1726 4449 21984 3867 ...
```

```
df$lowCharge <- 0
df$lowCharge[df$charges < 7000] <- 1
head(df)
```

```
##   age    sex    bmi children smoker    region   charges lowCharge
## 1  19 female 27.900        0    yes southwest 16884.924         0
## 2  18   male 33.770        1     no southeast  1725.552         1
## 3  28   male 33.000        3     no southeast  4449.462         1
## 4  33   male 22.705        0     no northwest 21984.471         0
## 5  32   male 28.880        0     no northwest  3866.855         1
## 6  31 female 25.740        0     no southeast  3756.622         1
```

```
y <- df$lowCharge
y <- as.integer(y)

X1 <- df$age
X1 <- as.integer(X1)

X2 <- df$sex
X2[X2 == 'male'] <- 1
X2[X2 == 'female'] <- 0
X2 <- as.integer(X2)

X3 <- df$bmi

X4 <- df$smoker
```

```r
X4[X4 == 'yes'] <- 1
X4[X4 == 'no'] <- 0

X5 <- df$region
unique(X5)
```

```
## [1] "southwest" "southeast" "northwest" "northeast"
```

```r
X5[X5 == 'southwest'] <- 1
X5[X5 == 'southeast'] <- 2
X5[X5 == 'northwest'] <- 3
X5[X5 == 'northeast'] <- 4

glmFit <-
  glm(y ~ X1 + X2, family = binomial, data = df)
summary(glmFit)
```

```
##
## Call:
## glm(formula = y ~ X1 + X2, family = binomial, data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0537  -0.5742  -0.2635   0.6934   1.8619
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.370108   0.264845  16.501   <2e-16 ***
## X1          -0.135320   0.007135 -18.965   <2e-16 ***
## X2           0.045047   0.147953   0.304    0.761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1770.6  on 1337  degrees of freedom
## Residual deviance: 1140.8  on 1335  degrees of freedom
## AIC: 1146.8
##
## Number of Fisher Scoring iterations: 5
```

Compare this model with the following models using K-Fold cross-validation with K=10 lowCharge ~age+sex+bmi, lowCharge ~age+sex+bmi+smoker, lowCharge ~age+sex+bmi+smoker+region

```r
# Creating K-Fold 'bins' ####
n <- nrow(df)
x <- 1:n
cv.error <- matrix(NA, 4, 11)
rownames(cv.error) <- c('Model1', 'Model2', 'Model3', 'Model4')
colnames(cv.error) <-
  c(
    'MSEK-Fold1',
    'MSEK-Fold2',
    'MSEK-Fold3',
    'MSEK-Fold4',
```

```
    'MSEK-Fold5',
    'MSEK-Fold6',
    'MSEK-Fold7',
    'MSEK-Fold8',
    'MSEK-Fold9',
    'MSEK-Fold10',
    'MeanMSE'
  )

for (i in 1:4) {
  for (j in 1:10) {
    glmFit1 <-
      glm(y ~ X1 + X2, family = binomial, data = df)
    glmFit2  <-
      glm(y ~ X1 + X2 + X3,
          family = binomial,
          data = df)
    glmFit3  <-
      glm(y ~ X1 + X2 + X3 + X4,
          family = binomial,
          data = df)
    glmFit4  <-
      glm(y ~ X1 + X2 + X3 + X4 + X5,
          family = binomial,
          data = df)
    cv.error[i, j] <- cv.glm(df, glmFit1, K = 10)$delta[1]
  }
  cv.error[i, 11] <- mean(cv.error[i, ], na.rm = TRUE)
}
```

```
## Warning: 'newdata' had 134 rows but variables found have 1338 rows

## Warning in y - yhat: longer object length is not a multiple of shorter
## object length

## Warning: 'newdata' had 134 rows but variables found have 1338 rows

## Warning in y - yhat: longer object length is not a multiple of shorter
## object length

## Warning: 'newdata' had 134 rows but variables found have 1338 rows

## Warning in y - yhat: longer object length is not a multiple of shorter
## object length

## Warning: 'newdata' had 134 rows but variables found have 1338 rows

## Warning in y - yhat: longer object length is not a multiple of shorter
## object length

## Warning: 'newdata' had 134 rows but variables found have 1338 rows

## Warning in y - yhat: longer object length is not a multiple of shorter
## object length

## Warning: 'newdata' had 134 rows but variables found have 1338 rows

## Warning in y - yhat: longer object length is not a multiple of shorter
## object length
```

```
## Warning in y - yhat: longer object length is not a multiple of shorter
## object length
```

```
## Warning: 'newdata' had 134 rows but variables found have 1338 rows
```

```
## Warning in y - yhat: longer object length is not a multiple of shorter
## object length
```

```
## Warning: 'newdata' had 134 rows but variables found have 1338 rows
```

```
## Warning in y - yhat: longer object length is not a multiple of shorter
## object length
```

```
cv.error
```

```
##        MSEK-Fold1 MSEK-Fold2 MSEK-Fold3 MSEK-Fold4 MSEK-Fold5 MSEK-Fold6
## Model1  0.3321263  0.3323346  0.3328502  0.3367259  0.3357122  0.3391525
## Model2  0.3354028  0.3331473  0.3333654  0.3346214  0.3282419  0.3324946
## Model3  0.3341613  0.3330023  0.3336209  0.3343223  0.3302717  0.3352746
## Model4  0.3341919  0.3290136  0.3346000  0.3349257  0.3357813  0.3330569
##        MSEK-Fold7 MSEK-Fold8 MSEK-Fold9 MSEK-Fold10   MeanMSE
## Model1  0.3355702  0.3323632  0.3332616   0.3348176 0.3344914
## Model2  0.3314926  0.3314377  0.3315959   0.3347781 0.3326578
## Model3  0.3316854  0.3317962  0.3329543   0.3367571 0.3333846
## Model4  0.3345868  0.3339362  0.3298470   0.3331851 0.3333124
```

```
numbers = c('first', 'second', 'third', 'fourth')
for (i in 1:4) {
  print(paste('The MSE for the', numbers[i], 'model is:', cv.error[i, 11]))
}
```

```
## [1] "The MSE for the first model is: 0.33449142121637"
## [1] "The MSE for the second model is: 0.332657768784132"
## [1] "The MSE for the third model is: 0.33338461680326"
## [1] "The MSE for the fourth model is: 0.333312443232973"
```