EAS508-HW4

Saish Mandavkar

2022-10-20

Lab Code Homework

5.3 Cross Validation Labs

5.3.1 Validation Set Approach

```
# Setting the seed and loading the data

library(ISLR2)
```

```
## Warning: package 'ISLR2' was built under R version 4.0.5
```

```
set.seed(1)
train <- sample(392,196)
```

```
# Fitting a linear regression on the train data using subset option

lm.fit <- lm(mpg ~ horsepower, data = Auto, subset = train)
```

```
# Predicting the estimates for the 392 observations and calculate the MSE for 192 observations

mean((Auto$mpg - predict(lm.fit, Auto))[-train]^2)
```

```
## [1] 23.26601
```

```
# Fitting cubic regression and calculating the MSE

lm.fit2 <- lm(mpg ~poly(horsepower, 2), data = Auto, subset = train)

mean((Auto$mpg - predict(lm.fit2, Auto))[-train]^2)
```

```
## [1] 18.71646
```

```
# Fitting uadratic regression and calculating the MSE

lm.fit3 <- lm(mpg ~ poly(horsepower, 3), data = Auto, subset = train)

mean((Auto$mpg - predict(lm.fit3, Auto))[-train]^2)
```

```
## [1] 18.79401
```

```r
# Using different seed and calculatiing the values for all the three regressions - will result into dif

set.seed(2)
train <- sample(392,196)

# Linear regression MSE

lm.fit <- lm(mpg ~ horsepower, data = Auto, subset = train)

mean((Auto$mpg - predict(lm.fit, Auto))[-train]^2)
```

```
## [1] 25.72651
```

```r
# Cubic regression MSE

lm.fit2 <- lm(mpg ~poly(horsepower, 2), data = Auto, subset = train)

mean((Auto$mpg - predict(lm.fit2, Auto))[-train]^2)
```

```
## [1] 20.43036
```

```r
# Quadratic regression MSE


lm.fit3 <- lm(mpg ~poly(horsepower, 3), data = Auto, subset = train)

mean((Auto$mpg - predict(lm.fit3, Auto))[-train]^2)
```

```
## [1] 20.38533
```

5.3.2 Leave One-Out Cross-Validation

```r
# LOOCV using glm() package

glm.fit <- glm(mpg ~ horsepower, data = Auto)

coef(glm.fit)
```

```
## (Intercept)  horsepower
##  39.9358610  -0.1578447
```

```r
# LOOCV using normal lm() function

lm.fit <- lm(mpg ~ horsepower, data = Auto)

coef(lm.fit)
```

```
## (Intercept)  horsepower
##  39.9358610  -0.1578447
```

```r
# Cross-validation error using glm() package

library(boot)
```

```
## Warning: package 'boot' was built under R version 4.0.5
```

```r
glm.fit <- glm(mpg ~ horsepower, data = Auto)

cv.err <- cv.glm(Auto, glm.fit)

cv.err$delta
```

```
## [1] 24.23151 24.23114
```

```r
# Calculating CV error for for polynomial of order 1 to 10 using a for loop.

cv.error <- rep(0,10)

for (i in  1:10) {

  glm.fit <- glm(mpg ~ poly(horsepower, i), data = Auto)
  cv.error[i] <- cv.glm(Auto, glm.fit)$delta[1]

}

cv.error
```

```
##  [1] 24.23151 19.24821 19.33498 19.42443 19.03321 18.97864 18.83305 18.96115
##  [9] 19.06863 19.49093
```

5.3.2 k-Fold Cross Validation

```r
# Calculating k-fold CV error for for polynomial of order 1 to 10 with k = 10

set.seed(17)
cv.error.10 <- rep(0,10)

for (i in  1:10) {

  glm.fit <- glm(mpg ~ poly(horsepower, i), data = Auto)
  cv.error.10[i] <- cv.glm(Auto, glm.fit, K = 10)$delta[1]

}

cv.error.10
```

```
##  [1] 24.27207 19.26909 19.34805 19.29496 19.03198 18.89781 19.12061 19.14666
##  [9] 18.87013 20.95520
```

5.3.4 The Bootstrap

```
# Creating a function alpha.fn() and which takes in X, Y values and the vector indicating which observa

alpha.fn <- function(data, index) {

  X <- data$X[index]
  Y <- data$Y[index]

  (var(Y) - cov(X,Y)) / (var(X) + var(Y) - 2 * cov(X,Y))
}
```

```
# Using the function on the portfolio database

alpha.fn(Portfolio, 1:100)
```

Estimating the accuracy of a Statistic of Interest

```
## [1] 0.5758321
```

```
# Using sample() function to create a new bootstrap dataset out of Portfolio

set.seed(7)

alpha.fn(Portfolio, sample(100, 100, replace = TRUE))
```

```
## [1] 0.5385326
```

```
# Performing bootstrap analysis using boot() for R = 1000

boot(Portfolio, alpha.fn, R = 1000)
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = Portfolio, statistic = alpha.fn, R = 1000)
##
##
## Bootstrap Statistics :
##      original       bias    std. error
## t1* 0.5758321 0.0007959475  0.08969074
```

Estimating the accuracy of a Linear Regression Model