Cross-Validation

Jason

Monday, May 25, 2015

```
library(ISLR)
#Our Data
data(Auto)
summary(Auto)
```

```
##       mpg          cylinders      displacement     horsepower   
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0  
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0  
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5  
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5  
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0  
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0  
##                                                                 
##      weight      acceleration        year           origin     
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000  
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000  
##  Median :2804   Median :15.50   Median :76.00   Median :1.000  
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577  
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000  
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000  
##                                                                
##                  name    
##  amc matador       :  5  
##  ford pinto        :  5  
##  toyota corolla    :  5  
##  amc gremlin       :  4  
##  amc hornet        :  4  
##  chevrolet chevette:  4  
##  (Other)           :365
```

1. Validation

```
set.seed(1)
#Index of train data
train <- sample(392, 196)
training <- Auto[train, ]
testing <- Auto[-train, ]

#Linear Regression
m1 <- lm(mpg ~ horsepower, data=training)
```

```
MSE1 <- mean((testing$mpg - predict(m1, testing))^2)
MSE1
```

```
## [1] 26.14142
```

```
#Different seeds
set.seed(2)
#Index of train data
train <- sample(392, 196)
training <- Auto[train, ]
testing <- Auto[-train, ]

#Linear Regression
m2 <- lm(mpg ~ horsepower, data=training)
```

```
MSE2 <- mean((testing$mpg - predict(m1, testing))^2)
MSE2
```

```
## [1] 22.64484
```

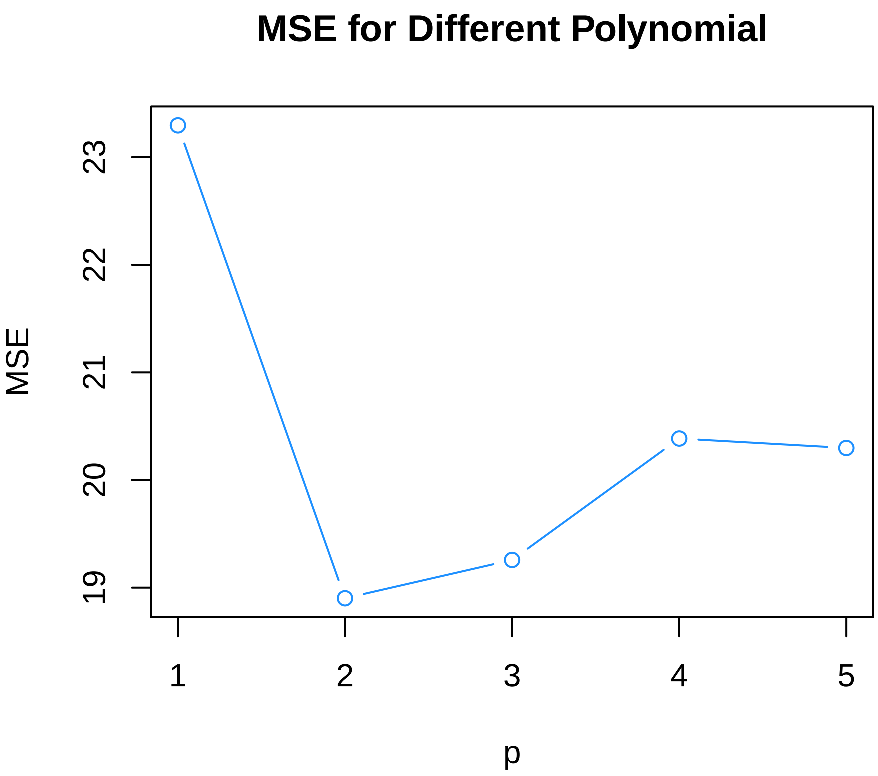
```
set.seed(2)
#Index of train data
train <- sample(392, 196)
training <- Auto[train, ]
testing <- Auto[-train, ]

#Polynomial terms
#p=1
m1 <- lm(mpg ~ horsepower, data=training)
#p=2
m2 <- lm(mpg ~ poly(horsepower, 2), data=training)
#p=3
m3 <- lm(mpg ~ poly(horsepower, 3), data=training)
#p=4
m4 <- lm(mpg ~ poly(horsepower, 4), data=training)
#p=5
m5 <- lm(mpg ~ poly(horsepower, 5), data=training)
```

```
#p=1
MSE_p1 <- mean((testing$mpg - predict(m1, testing))^2)
#p=2
MSE_p2 <- mean((testing$mpg - predict(m2, testing))^2)
#p=3
MSE_p3 <- mean((testing$mpg - predict(m3, testing))^2)
#p=4
MSE_p4 <- mean((testing$mpg - predict(m4, testing))^2)
#p=5
MSE_p5 <- mean((testing$mpg - predict(m5, testing))^2)
MSE_all <- data.frame(p=c(1, 2, 3, 4, 5),
                      MSE=c(MSE_p1, MSE_p2, MSE_p3,
                            MSE_p4, MSE_p5))
MSE_all
```

```
##   p      MSE
## 1 1 23.29559
## 2 2 18.90124
## 3 3 19.25740
## 4 4 20.38538
## 5 5 20.29775
```

```r
plot(MSE_all, main="MSE for Different Polynomial", type="b", col="dodgerblue")
```



2. Cross-Validation - LOOCV

```r
set.seed(1)
#Linear Regression
glm1 <- glm(mpg ~ horsepower, data=Auto)
#The package which provides function to do cross validation
library(boot)
cv1 <- cv.glm(Auto, glm1)
cv1$delta
```

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

```r
#Another method by leverage hi
mean(((glm1$y - fitted(glm1))/(1 - hatvalues(glm1)))^2)
```

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

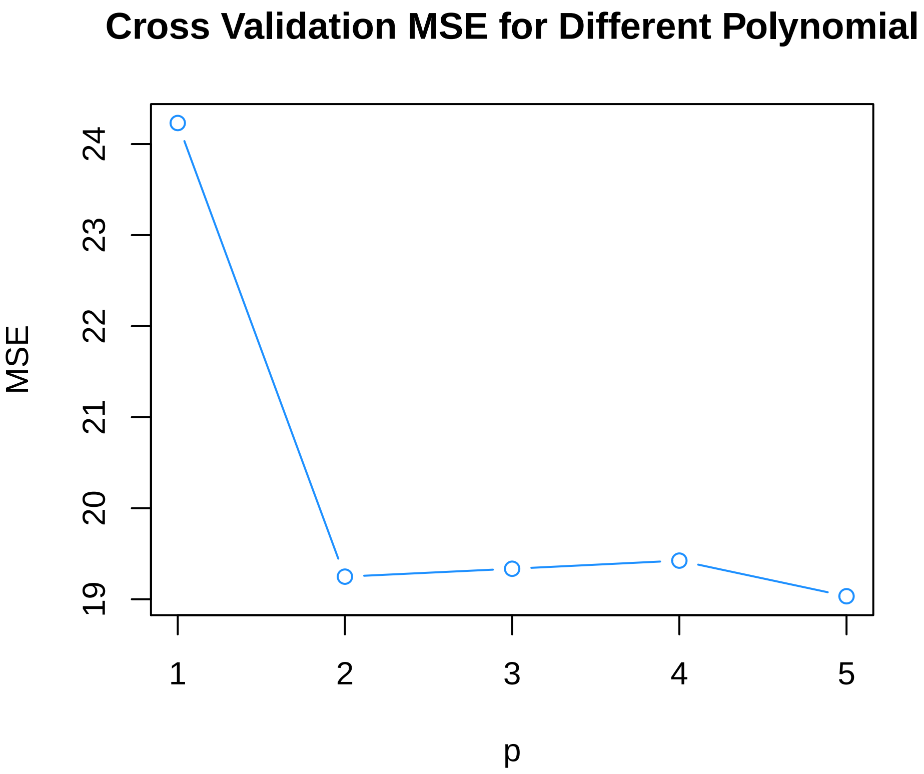
```
#For different polynomial
cv_error <- c()

for(i in 1:5){
  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
```

```
plot(1:5, cv_error, main="Cross Validation MSE for Different Polynomial",
     xlab="p", ylab="MSE", type="b", col="dodgerblue")
```



3. Cross-Validation - K-fold

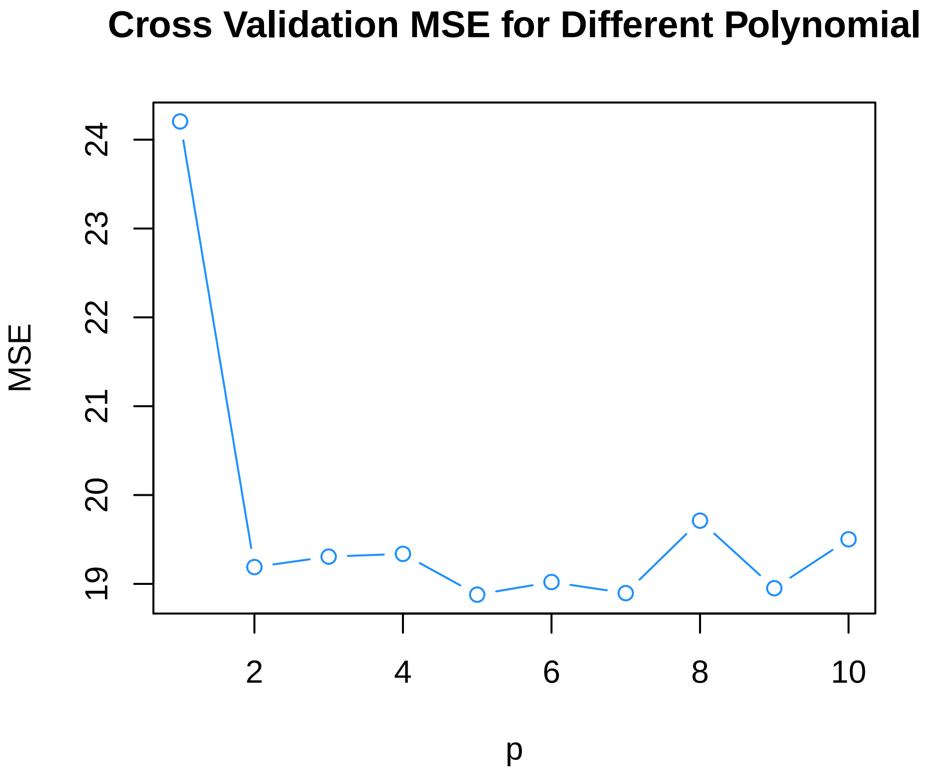
```
set.seed(17)
#For different polynomial
cv_error <- c()
```

```
for(i in 1:10){
  glm.fit <- glm(mpg ~ poly(horsepower, i), data=Auto)
  cv_error[i] <- cv.glm(Auto, glm.fit, K=10)$delta[1]
}

cv_error
```

```
##  [1] 24.20520 19.18924 19.30662 19.33799 18.87911 19.02103 18.89609
##  [8] 19.71201 18.95140 19.50196
```

```
plot(1:10, cv_error, main="Cross Validation MSE for Different Polynomial",
     xlab="p", ylab="MSE", type="b", col="dodgerblue")
```


Cross Validation MSE for Different Polynomial
MSE
p