Analytics 512 Homework 5

Arif Ali

03/24/2016

Analytics 512 HW #5

Exercise 6.8 2b

The ridge regression relative to leasr squares is: less flexible and hence will give improved prediction accu- racy when its increase in bias is less than its decrease in variance. As λ increases, the flexibility of the ridge regression fit decreases.

Exercise 6.8 3

Part A

Decrease Steadily, as S increases,the model better fits to the training data, which lowers the value of the residuals, thus lowering the RSS.

Part B

Decrease initially, and then eventually start increasing in a U shape. This is because at a certain point, the model will be overfitted, which results in an increase in the value of the residuals.

Part C

Increase Steadily. As the value of S increase, there are more non-zero values of β_i, the more the variance increases.

Exercise 6.8 9

Part A

In [3]:

```
library(ISLR)
attach(College)
train = sample(nrow(College), nrow(College)*.70, replace = F)
training = College[train,]
testing = College[-train,]
```

Part B

In [4]:

```
college.lm = lm(Apps~.,data=training)
est.lm = predict(college.lm, testing)
#MSE
mean((testing$Apps - est.lm)^2)
```

Out[4]: 939913.886192582

Part C

In [5]:

```
library(glmnet)
grid=10^seq(10,-2,length=100)

x=model.matrix(Apps~., data = training)
y = training$Apps
college.ridge = cv.glmnet(x,y,alpha=0,lambda=grid)
ridge.pred = predict(college.ridge,
                     newx=model.matrix(Apps~., data = testing),
                     s=college.ridge$lambda.min)
mean((testing$Apps - ridge.pred)^2)
```

```
Loading required package: Matrix
Loading required package: foreach
Loaded glmnet 2.0-2
```

Out[5]: 939926.831254838

Part D

In [6]:

```
college.lasso = cv.glmnet(x,y,alpha=1,lambda=grid,thresh=1e-12)
college.lasso$lambda.min
```

Out[6]: 18.7381742286039

In [7]:

```
lasso.pred = predict(college.lasso,
                     newx=model.matrix(Apps~., data = testing),
                     s=college.lasso$lambda.min)
mean((testing$Apps - lasso.pred)^2)
```

Out[7]: 966737.526000242

Exercise 6.8 11

```
In [8]: library(MASS)
        Boston$crim = log10(as.numeric(Boston$crim))
        dim(Boston)
```

```
Out[8]:    506   14
```

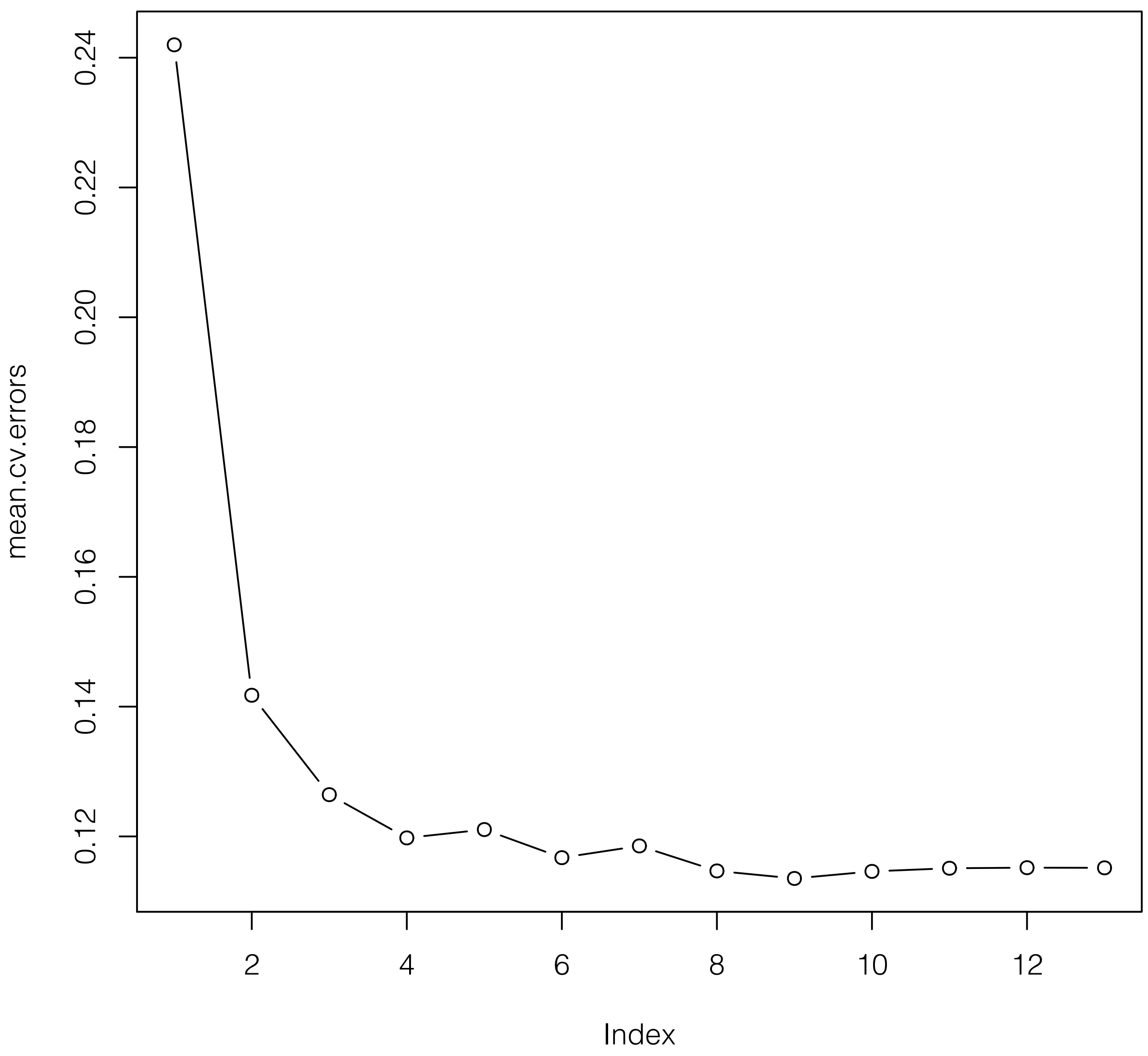
Part A

```
In [9]: library(leaps)
        Boston.bestsubset = regsubsets(crim~.,data=Boston)
```

```
In [10]: predict.regsubsets <- function(object, newdata, id, ...) {
             form <- as.formula(object$call[[2]])
             mat <- model.matrix(form, newdata)
             coefi <- coef(object, id = id)
             xvars <- names(coefi)
             mat[, xvars] %*% coefi
         }

         k=10
         set.seed (1)
         folds=sample(1:k,nrow(Boston),replace=TRUE)
         cv.errors=matrix(NA,k,13, dimnames=list(NULL, paste(1:13)))
```

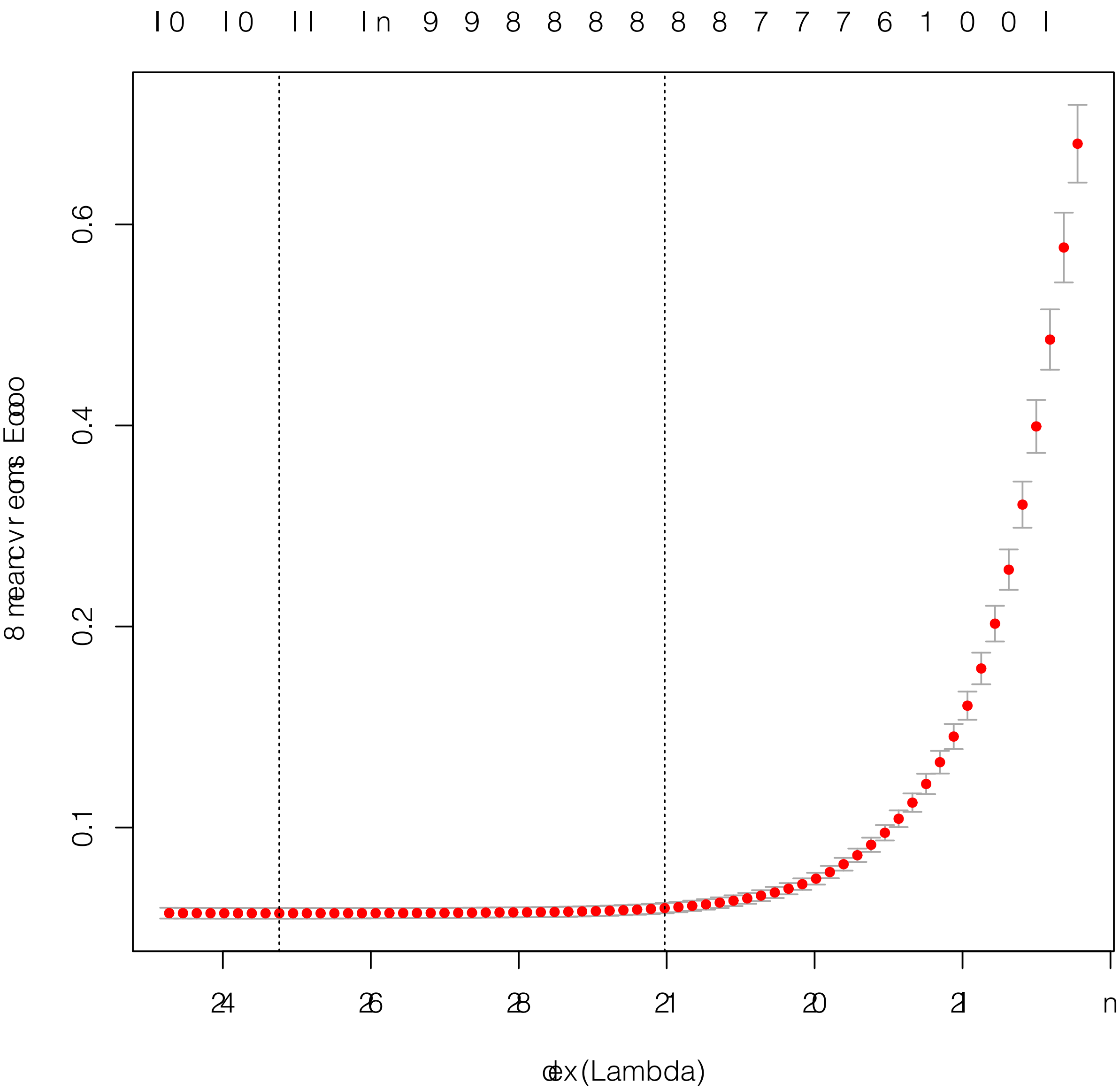
```
In [11]: for(j in 1:k){
             best.fit = regsubsets(crim~.,data=Boston[folds!=j,],
                                   nvmax=13)
             for(i in 1:13){
                 pred = predict(best.fit,Boston[folds==j,],id=i)
                 cv.errors[j,i] = mean( (Boston$crim[folds==j]-pred)^2)
                 }
             }
         mean.cv.errors=apply(cv.errors ,2,mean)
         plot(mean.cv.errors ,type='b')
```



As the number of variables increase, the mean cv errors tend to lower, especially with respect from 1 to 2 variables. It's interesting to see that a 9 variable best subset selection, dips lower than the other number of variables.

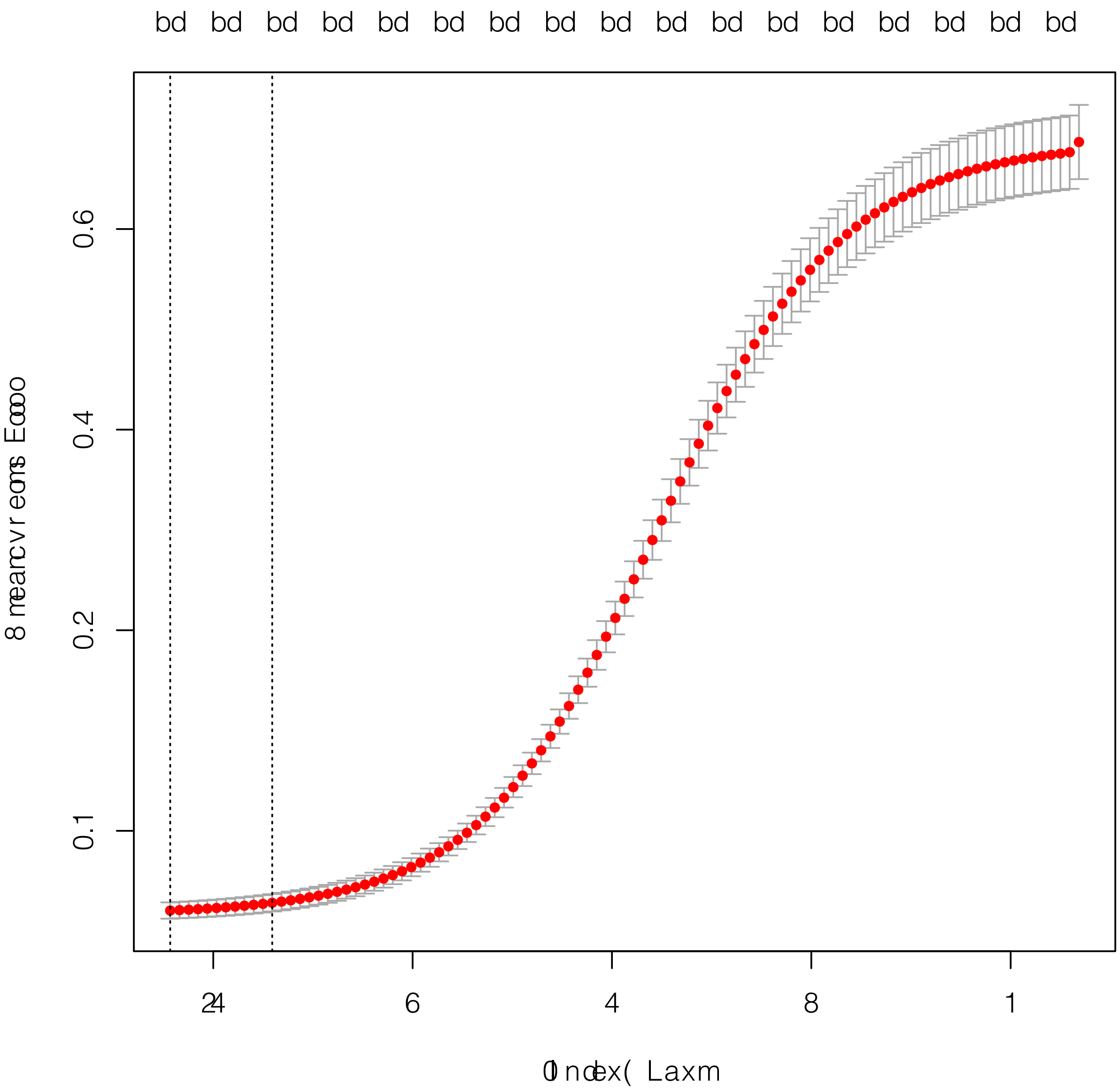
In [12]:

```
cv.lasso = cv.glmnet(model.matrix(crim ~ ., Boston),
                     Boston$crim, alpha = 1, type.measure = "mse")
plot(cv.lasso)
```



Based on the Cross Validation of the Lasso, the lower the value of the $log(\lambda)$, the lower the Mean-Saquared Error. The increase of errors is similar to an exponential function.

```
In [13]: cv.ridge = cv.glmnet(model.matrix(crim ~ ., Boston),
                              Boston$crim, alpha = 0, type.measure = "mse")
         plot(cv.ridge)
```



Based on the Cross Validation of the Ridge Regression, the lower the value of the $log(\lambda)$, the lower the Mean-Saqared Error. The increase of errors is similar to an log function.

Part B

Based off the the the graphs, it's easy to infer which the minimun Mean Sqared Error occurs for Best Subset Selection, lasso, and Ridge Regression. After looking at the Mean Sqare errors generated by various examples of each model, the best model, based of MSE, would be Best Subset Selection with 9 variables.

```
In [15]: min(mean.cv.errors)
         cv.lasso$cvm[cv.lasso$lambda == cv.lasso$lambda.min]
         cv.ridge$cvm[cv.ridge$lambda == cv.ridge$lambda.min]
```

```
Out[15]: 0.113525982496003

Out[15]: 0.11462334668151

Out[15]: 0.120528283513175
```