Lab 2: Ridge Regression and the Lasso

Jonathan Bryan

June 2, 2018

6.6.1 Ridge Regression

```
library(ISLR)
sum(is.na(Hitters))
```

```
## [1] 59
```

```
Hitters = na.omit(Hitters)
sum(is.na(Hitters))
```

```
## [1] 0
```

```
x = model.matrix(Salary ~., Hitters)[,-1]
y = Hitters$Salary
```

```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 3.4.4
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

```
## Loaded glmnet 2.0-16
```

```
grid = 10^seq(10,-2,length=100)
ridge.mod = glmnet(x,y,alpha=0, lambda=grid)
```

```
#Coefficent data when lambda = 11,498
dim(coef(ridge.mod))
```

```
## [1]  20 100
```

```
ridge.mod$lambda[50]
```

```
## [1] 11497.57
```

```
round(coef(ridge.mod)[,50],3)
```

```
## (Intercept)       AtBat        Hits       HmRun        Runs         RBI
##     407.356       0.037       0.138       0.525       0.231       0.240
##       Walks       Years      CAtBat       CHits      CHmRun       CRuns
##       0.290       1.108       0.003       0.012       0.088       0.023
##        CRBI      CWalks     LeagueN   DivisionW     PutOuts     Assists
##       0.024       0.025       0.085      -6.215       0.016       0.003
##      Errors  NewLeagueN
##      -0.021       0.301
```

```
sqrt(sum(coef(ridge.mod)))
```

```
## [1] 172.7644
```

```
#Coefficent data when lambda = 705
ridge.mod$lambda[60]
```

```
## [1] 705.4802
```

```
round(coef(ridge.mod)[,60],3)
```

```
## (Intercept)       AtBat        Hits       HmRun        Runs         RBI
##      54.325       0.112       0.656       1.180       0.938       0.847
##       Walks       Years      CAtBat       CHits      CHmRun       CRuns
##       1.320       2.596       0.011       0.047       0.338       0.094
##        CRBI      CWalks     LeagueN   DivisionW     PutOuts     Assists
##       0.098       0.072      13.684     -54.659       0.119       0.016
##      Errors  NewLeagueN
##      -0.704       8.612
```

```
sqrt(sum(coef(ridge.mod)[-1,60]^2))
```

```
## [1] 57.11001
```

```
round(predict(ridge.mod ,s=50, type ="coefficients")[1:20 ,],3)
```

```
## (Intercept)       AtBat        Hits       HmRun        Runs         RBI
##      48.766      -0.358       1.969      -1.278       1.146       0.804
##       Walks       Years      CAtBat       CHits      CHmRun       CRuns
##       2.716      -6.218       0.005       0.106       0.624       0.221
##        CRBI      CWalks     LeagueN   DivisionW     PutOuts     Assists
##       0.219      -0.150      45.926    -118.201       0.250       0.122
##      Errors  NewLeagueN
##      -3.279      -9.497
```

```
set.seed(1)
train = sample(1:nrow(x), nrow(x)/2, replace = FALSE)
test = (-train)
y.test = y[test]
```

```
#predicting Ridge (lambda = 4) on test set for test MSE
ridge.mod = glmnet(x[train,], y[train], alpha=0, lambda = grid, thresh=1e-12)
ridge.pred = predict(ridge.mod, s=4, newx=x[test,])
mean((ridge.pred-y.test)^2)
```

```
## [1] 101036.8
```

```
#null test MSE using intercepts using mean of training observations
mean((mean(y[train])-y.test)^2)
```

```
## [1] 193253.1
```

```
#null test MSE using large lambda for Ridge regresion
ridge.pred=predict (ridge.mod ,s=1e10 ,newx=x[test ,])
mean((ridge.pred -y.test)^2)
```

```
## [1] 193253.1
```

```
#comparing lambda = 4 Ridge regression to OLS
ridge.pred = predict(ridge.mod, s=0, y=y, x=x, newx=x[test,], exact=TRUE)
mean((ridge.pred-y.test)^2)
```

```
## [1] 68503.97
```

```
lm(y~x, subset=train)
```

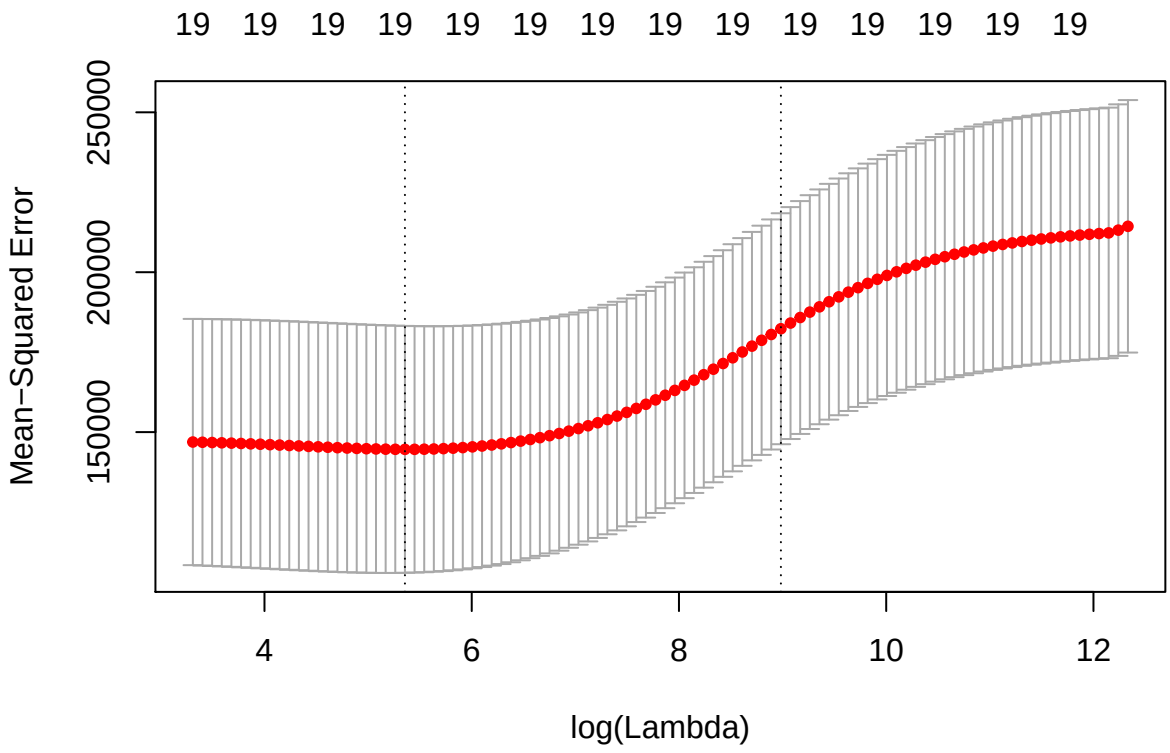
```
##
```

```
## Call:
## lm(formula = y ~ x, subset = train)
##
## Coefficients:
## (Intercept)       xAtBat        xHits       xHmRun        xRuns
##   299.42849     -2.54027      8.36682     11.64512     -9.09923
##        xRBI       xWalks       xYears      xCAtBat       xCHits
##     2.44105      9.23440    -22.93673     -0.18154     -0.11598
##     xCHmRun       xCRuns        xCRBI      xCWalks     xLeagueN
##    -1.33888      3.32838      0.07536     -1.07841     59.76065
##  xDivisionW     xPutOuts     xAssists      xErrors  xNewLeagueN
##   -98.86233      0.34087      0.34165     -0.64207     -0.67442
```

```
predict(ridge.mod, s=0, y=y, x=x, exact=T, type="coefficients")[1:20]
```

```
##  [1]  163.0999892   -1.9797728    7.5002874    4.3303888   -2.3758215
##  [6]   -1.0448385    6.2311036   -3.4871752   -0.1713926    0.1342504
## [11]   -0.1724802    1.4541450    0.8075644   -0.8115061   62.6023306
## [16] -116.8489695    0.2818909    0.3710871   -3.3609423  -24.7651585
```

```
#optimal lambda search through cross validation
set.seed(1)
cv.out = cv.glmnet(x[train,], y[train], alpha = 0)
plot(cv.out)
```




```
bestlam=cv.out$lambda.min
bestlam
```

```
## [1] 211.7416
```

```
#MSE of lambda = 212
ridge.pred= predict(ridge.mod, s=bestlam, newx=x[test,])
mean((ridge.pred-y.test)^2)
```

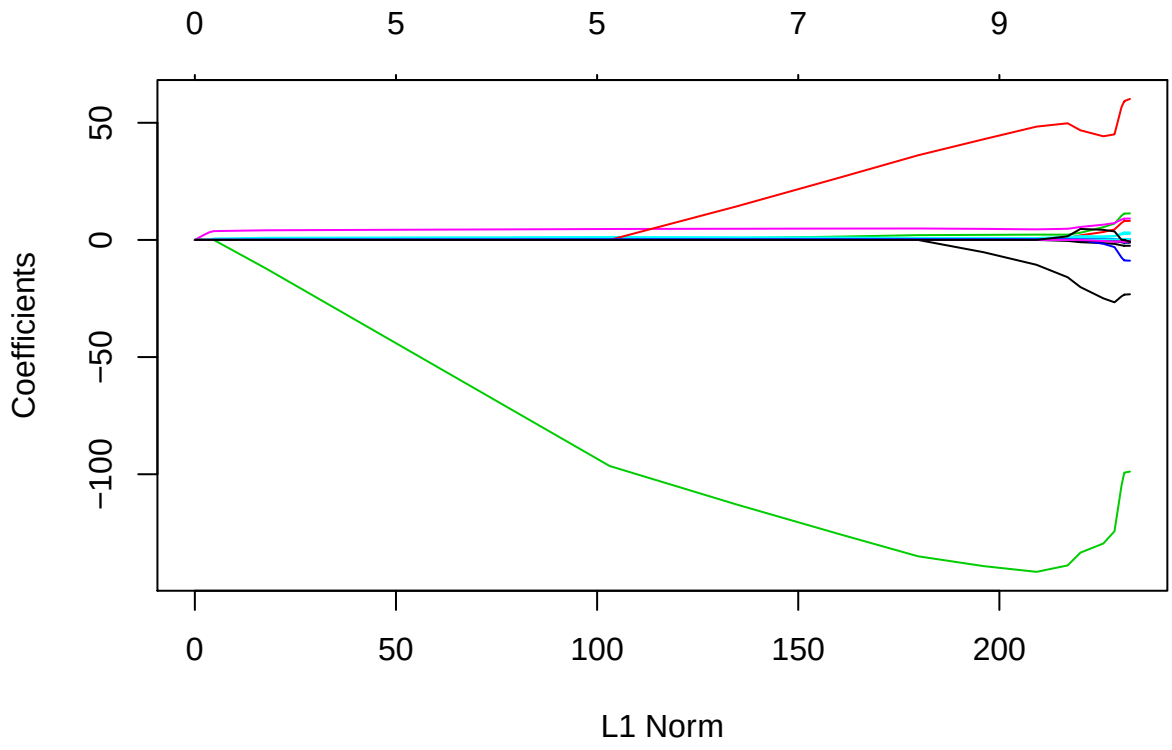
```
## [1] 96015.51
```

```
out = glmnet(x,y,alpha=0)
predict(out,type="coefficients", s=bestlam)[1:20,]
```

```
##  (Intercept)        AtBat         Hits        HmRun         Runs
##   9.88487157   0.03143991   1.00882875   0.13927624   1.11320781
##          RBI        Walks        Years       CAtBat        CHits
##   0.87318990   1.80410229   0.13074383   0.01113978   0.06489843
##       CHmRun        CRuns         CRBI       CWalks      LeagueN
##   0.45158546   0.12900049   0.13737712   0.02908572  27.18227527
##    DivisionW      PutOuts      Assists       Errors   NewLeagueN
## -91.63411282   0.19149252   0.04254536  -1.81244470   7.21208394
```

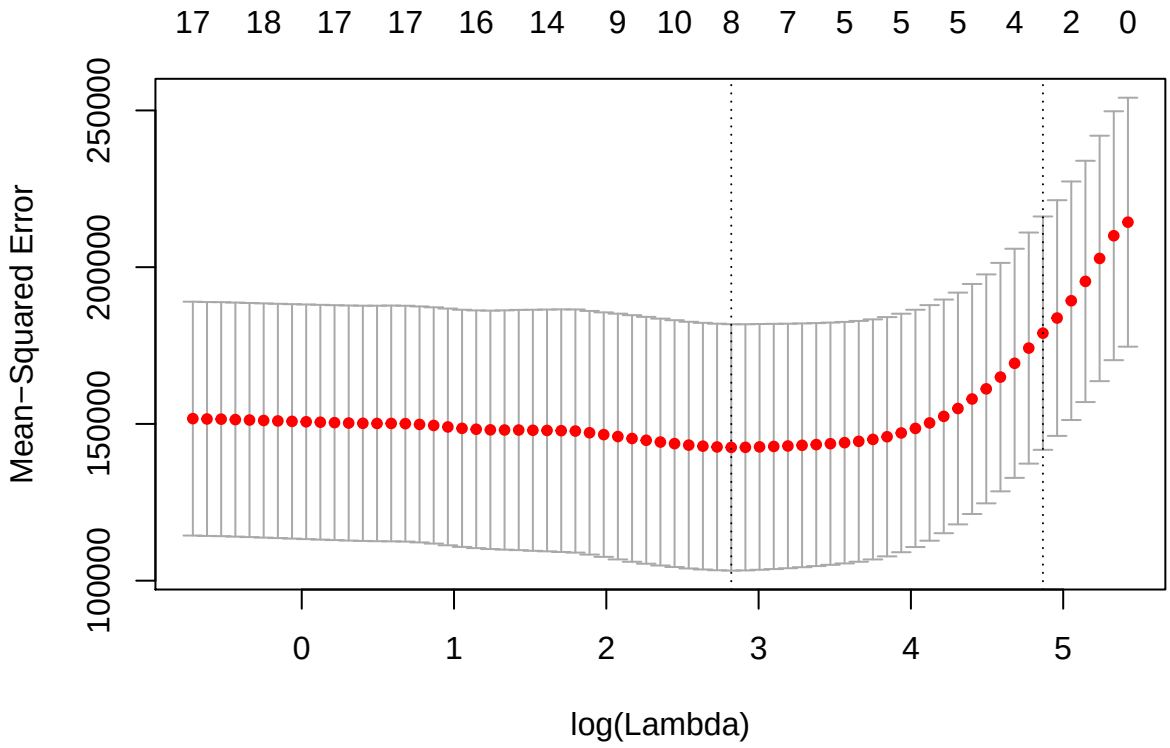
6.6.2 The Lasso

```
lasso.mod = glmnet(x[train,], y[train],alpha=1, lambda=grid)
plot(lasso.mod)
```




```
set.seed(1)
cv.out=cv.glmnet(x[train,], y[train],alpha=1)
```

```
plot(cv.out)
```




```
bestlam = cv.out$lambda.min
lasso.pred = predict(lasso.mod,s=bestlam,newx= x[test,])
mean((lasso.pred - y.test)^2)
```

```
## [1] 100743.4
```

```
out=glmnet(x,y,alpha=1,lambda=grid)
lasso.coef = predict(out, type="coefficients", s=bestlam)[1:20,]
lasso.coef
```

```
##  (Intercept)        AtBat         Hits        HmRun         Runs
##   18.5394844    0.0000000    1.8735390    0.0000000    0.0000000
##          RBI        Walks        Years       CAtBat        CHits
##    0.0000000    2.2178444    0.0000000    0.0000000    0.0000000
##       CHmRun        CRuns         CRBI       CWalks      LeagueN
##    0.0000000    0.2071252    0.4130132    0.0000000    3.2666677
##    DivisionW      PutOuts      Assists       Errors   NewLeagueN
## -103.4845458    0.2204284    0.0000000    0.0000000    0.0000000
```

```
lasso.coef[lasso.coef!=0]
```

```
##  (Intercept)         Hits        Walks        CRuns         CRBI
##   18.5394844    1.8735390    2.2178444    0.2071252    0.4130132
##      LeagueN    DivisionW      PutOuts
##    3.2666677 -103.4845458    0.2204284
```