Ridge Regression & Lasso

Claudius Taylor

7/1/2019

Predict the number of applications received using the other variables in the College data set.

```
data("College", package = "ISLR")
x = model.matrix(Apps~., College)[,-1]
y = College$Apps
```

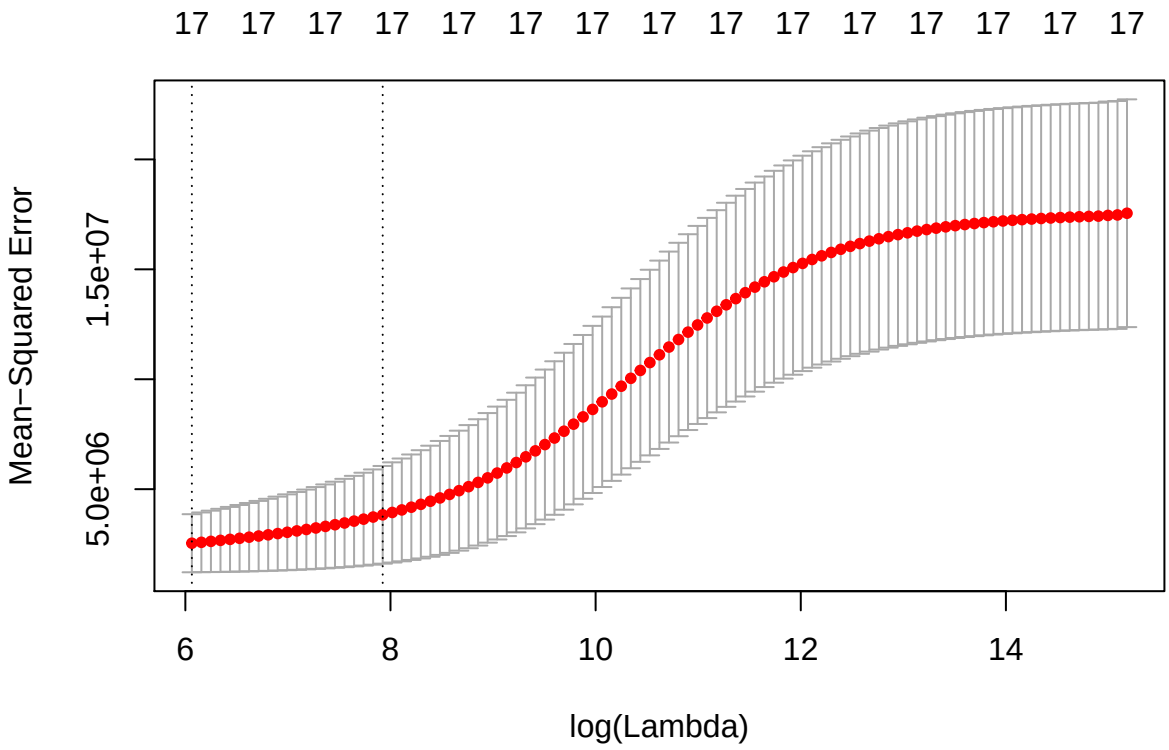
```
grid = 10^seq(10, -2, length = 100)
ridge.mod = glmnet(x, y, alpha = 0, lambda = grid)
```

(a) Split the data into a training and a test set

```
set.seed(10)
train = sample(1:nrow(x), nrow(x)/2)
test = (-train)
y.test = y[test]
```

(b) Fit a ridge regression model on the training set, with λ chosen by cross-validation. Report the test error obtained.

```
set.seed(11)
cross.val.output = cv.glmnet(x[train, ], y[train], alpha = 0)
plot(cross.val.output)
```




```
bestlamda = cross.val.output$lambda.min
bestlamda
```

```
## [1] 429.864
```

test error obtained for ridge

```
ridge.pred = predict(ridge.mod, s = bestlamda, newx = x[test, ])
mean((ridge.pred - y.test)^2)
```

```
## [1] 884281.3
```

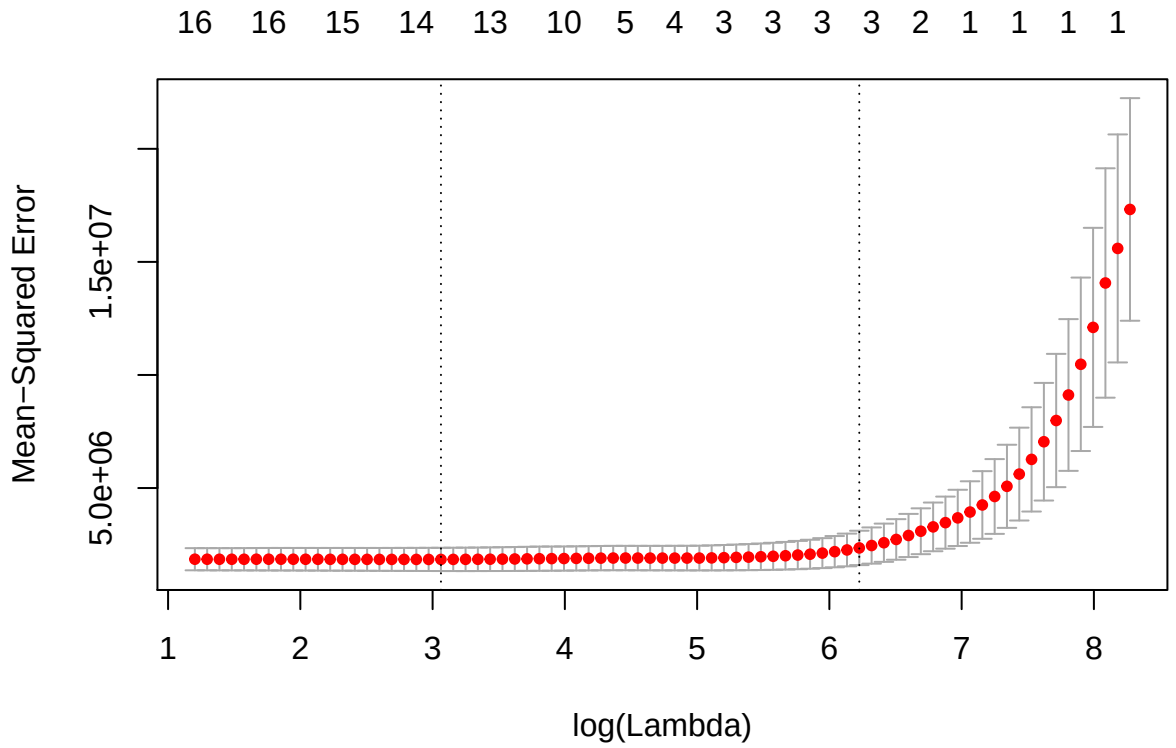
Refit the ridge model obtained on the full data set, using the value of the λ chosen by cross-validation and examine the coefficient estimates

```
out = glmnet(x,y, alpha = 0)
predict(out, type = 'coefficient', s = bestlamda)[1:18,]
```

```
##   (Intercept)    PrivateYes        Accept        Enroll     Top10perc
## -1.550866e+03 -5.304238e+02  9.579367e-01  4.917803e-01  2.435812e+01
##     Top25perc   F.Undergrad   P.Undergrad      Outstate    Room.Board
##  1.427479e+00  7.982443e-02  2.485755e-02 -1.938471e-02  2.004349e-01
##         Books      Personal           PhD      Terminal     S.F.Ratio
##  1.406329e-01 -9.419936e-03 -3.567770e+00 -4.620777e+00  1.269716e+01
##   perc.alumni        Expend     Grad.Rate
## -9.038903e+00  7.479540e-02  1.144520e+01
```

(c) Fit a lasso model on the training set, with λ chosen by cross-validation. Report the test error obtained.

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




```
bestlamda = cv.out.l$lambda.min
```

test error obtained for lasso

```
lasso.pred = predict(lasso.mod, s = bestlamda, newx = x[test, ])
mean((lasso.pred - y.test)^2)
```

```
## [1] 935388.9
```

Refit the lasso model obtained on the full data set, using the value of the λ chosen by cross-validation and observe the number of no-zero coefficient estimates

```
out = glmnet(x,y, alpha = 1, lambda = grid)
predict(out, type = 'coefficient', s = bestlamda)[1:18,]
```

```
##   (Intercept)    PrivateYes        Accept        Enroll     Top10perc
## -6.026111e+02 -4.242059e+02  1.456502e+00 -2.040322e-01  3.383949e+01
##     Top25perc   F.Undergrad   P.Undergrad      Outstate    Room.Board
## -2.530427e+00  0.000000e+00  2.091227e-02 -5.813391e-02  1.248869e-01
##          Books      Personal           PhD      Terminal     S.F.Ratio
##  0.000000e+00  1.363744e-03 -5.608209e+00 -3.306533e+00  4.484244e+00
##   perc.alumni        Expend     Grad.Rate
## -9.491136e-01  6.967778e-02  5.185590e+00
```

Conclusion:

The Lasso has a slight advantage over the Ridge regression in that the resulting coefficient are somewhat sparse. Here we see that 2 of the 17 coefficient estimates are exactly zero. So the Lasso model with λ chosen by the cross-validation contains only 15 variables.