Lab 6

IMC 490: Machine Learning for IMC

5/3/2017

In this lab, we'll be going over the following topics:

- ridge regression
- lasso regression
- stepwise selection

In this lab, we will build a model to predict the salary of a baseball player based in his batting statistics.

As always, we should explore and clean the data before analyzing it.

- Check how many `NA` values are in `Hitters$Salary`
- Fix this problem with dplyr
- Check the correlations across numeric variables. You will need to use dplyr to remove the categorical variables.

```
require(ISLR)
data(Hitters)
str(Hitters)
```

```
## 'data.frame':    322 obs. of  20 variables:
##  $ AtBat    : int  293 315 479 496 321 594 185 298 323 401 ...
##  $ Hits     : int  66 81 130 141 87 169 37 73 81 92 ...
##  $ HmRun    : int  1 7 18 20 10 4 1 0 6 17 ...
##  $ Runs     : int  30 24 66 65 39 74 23 24 26 49 ...
##  $ RBI      : int  29 38 72 78 42 51 8 24 32 66 ...
##  $ Walks    : int  14 39 76 37 30 35 21 7 8 65 ...
##  $ Years    : int  1 14 3 11 2 11 2 3 2 13 ...
##  $ CAtBat   : int  293 3449 1624 5628 396 4408 214 509 341 5206 ...
##  $ CHits    : int  66 835 457 1575 101 1133 42 108 86 1332 ...
##  $ CHmRun   : int  1 69 63 225 12 19 1 0 6 253 ...
##  $ CRuns    : int  30 321 224 828 48 501 30 41 32 784 ...
##  $ CRBI     : int  29 414 266 838 46 336 9 37 34 890 ...
##  $ CWalks   : int  14 375 263 354 33 194 24 12 8 866 ...
##  $ League   : Factor w/ 2 levels "A","N": 1 2 1 2 2 1 2 1 2 1 ...
##  $ Division : Factor w/ 2 levels "E","W": 1 2 2 1 1 2 1 2 2 1 ...
##  $ PutOuts  : int  446 632 880 200 805 282 76 121 143 0 ...
##  $ Assists  : int  33 43 82 11 40 421 127 283 290 0 ...
##  $ Errors   : int  20 10 14 3 4 25 7 9 19 0 ...
##  $ Salary   : num  NA 475 480 500 91.5 750 70 100 75 1100 ...
##  $ NewLeague: Factor w/ 2 levels "A","N": 1 2 1 2 2 1 1 1 2 1 ...
```

```r
fit = lm(Salary ~ ., Hitters)
summary(fit)
```

```
##
## Call:
## lm(formula = Salary ~ ., data = Hitters)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -907.62 -178.35  -31.11  139.09 1877.04
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  163.10359   90.77854   1.797 0.073622 .
## AtBat         -1.97987    0.63398  -3.123 0.002008 **
## Hits           7.50077    2.37753   3.155 0.001808 **
## HmRun          4.33088    6.20145   0.698 0.485616
## Runs          -2.37621    2.98076  -0.797 0.426122
## RBI           -1.04496    2.60088  -0.402 0.688204
## Walks          6.23129    1.82850   3.408 0.000766 ***
## Years         -3.48905   12.41219  -0.281 0.778874
## CAtBat        -0.17134    0.13524  -1.267 0.206380
## CHits          0.13399    0.67455   0.199 0.842713
## CHmRun        -0.17286    1.61724  -0.107 0.914967
## CRuns          1.45430    0.75046   1.938 0.053795 .
## CRBI           0.80771    0.69262   1.166 0.244691
## CWalks        -0.81157    0.32808  -2.474 0.014057 *
## LeagueN       62.59942   79.26140   0.790 0.430424
## DivisionW   -116.84925   40.36695  -2.895 0.004141 **
## PutOuts        0.28189    0.07744   3.640 0.000333 ***
## Assists        0.37107    0.22120   1.678 0.094723 .
## Errors        -3.36076    4.39163  -0.765 0.444857
## NewLeagueN   -24.76233   79.00263  -0.313 0.754218
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 315.6 on 243 degrees of freedom
## Multiple R-squared:  0.5461, Adjusted R-squared:  0.5106
## F-statistic: 15.39 on 19 and 243 DF,  p-value: < 2.2e-16
```

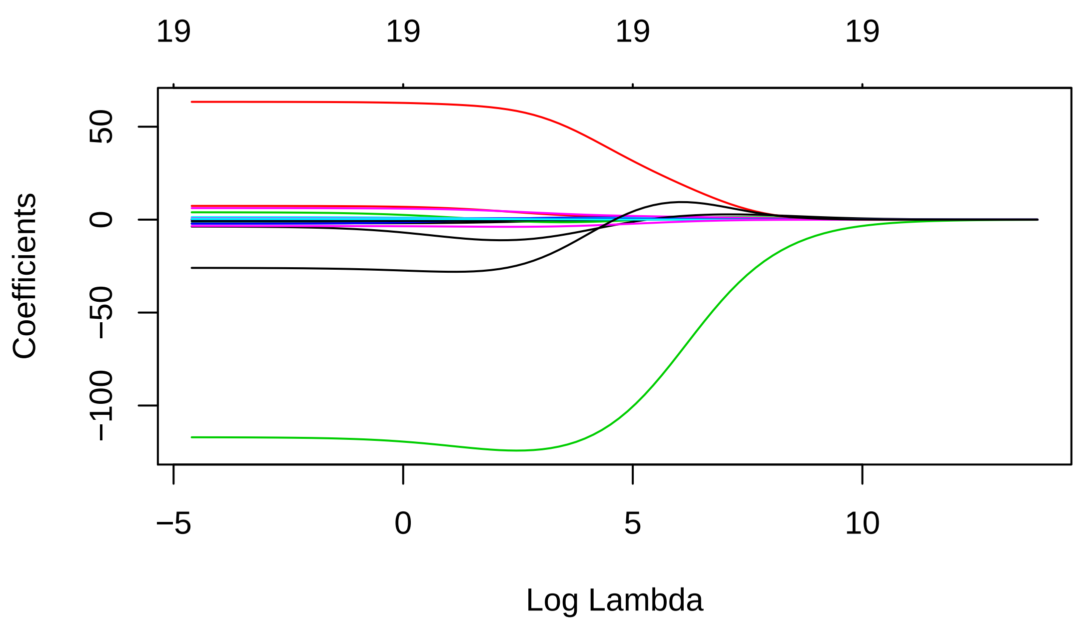
Ridge (L2)

We have three regularization options to deal with our correlated predictors.

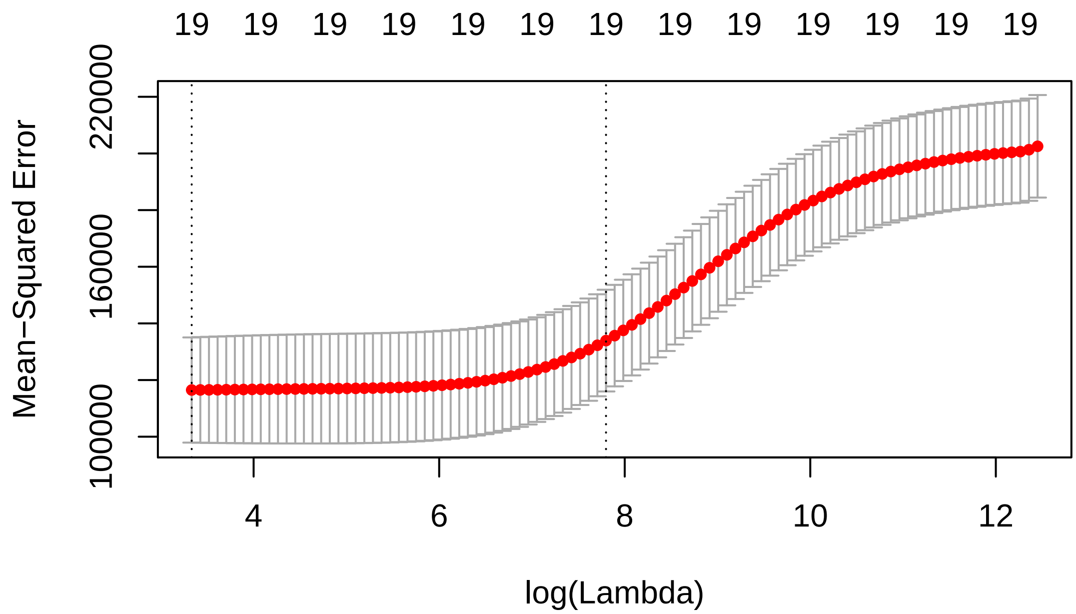
1. Ridge: L2 regularization - penalize the squared Euclidian length of the slope vector.
2. Lasso: L1 regularization - penalize the Manhattan distance of the slope vector.
3. Elastic Net: L1 and L2 regularization - penalize upon a linear combination of Ridge and Lasso penalties.

- What is the ideal amount of shrinkage (lambda) for Ridge regression? How can we choose an ideal lambda?
- How do the coefficients change with L2 regularization?

```
require(glmnet)

# using model.matrix is important, as it expands the categorical variables
x = model.matrix(Salary ~ ., Hitters)[ ,-1]
y = Hitters$Salary

# create a grid of lambdas
grid = 10 ^ seq(-2, 6, length=100)

# fit ridge regression
fit_ridge = glmnet(x, y,
                    lambda = grid,
                    alpha = 0)
plot(fit_ridge, xvar = "lambda")
```



```r
# perform cross validation to determine a good lambda
fit_ridge_cv = cv.glmnet(x, y, alpha = 0, nfolds = 5)
plot(fit_ridge_cv)
```



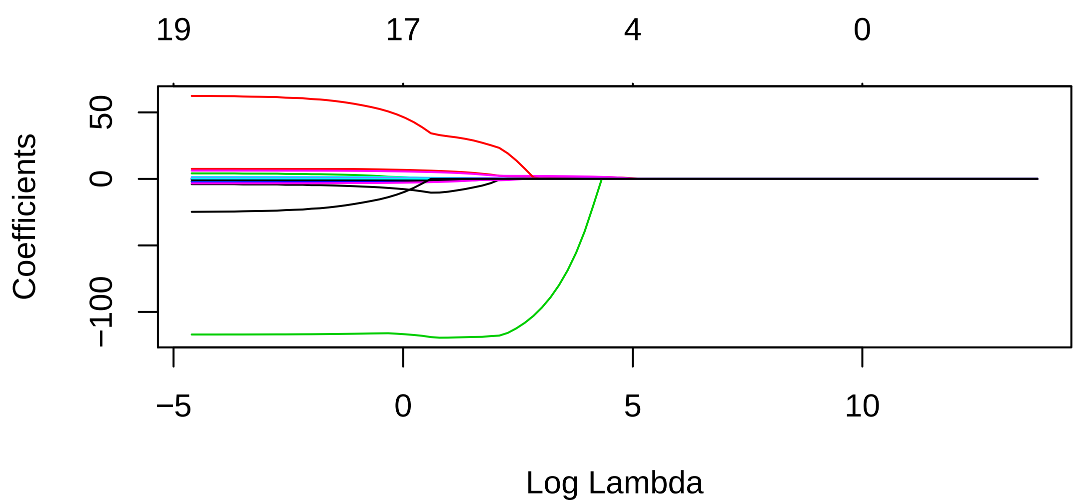

```r
# perform ridge on the ideal lambda value we found
fit_ridge_final = glmnet(x, y, lambda = exp(8), alpha = 0)
coef(fit_ridge_final)
```

```
## 20 x 1 sparse Matrix of class "dgCMatrix"
##                        s0
## (Intercept) 229.162120205
## AtBat         0.085998660
## Hits          0.349878481
## HmRun         1.137725825
## Runs          0.564836752
## RBI           0.565967055
## Walks         0.728894848
## Years         2.383927132
## CAtBat        0.007245018
## CHits         0.027787042
## CHmRun        0.206603632
## CRuns         0.055724626
## CRBI          0.057618155
## CWalks        0.056225361
## LeagueN       2.782734692
## DivisionW   -20.078273857
## PutOuts       0.048752514
## Assists       0.006999616
## Errors       -0.125363721
## NewLeagueN    2.602954981
```

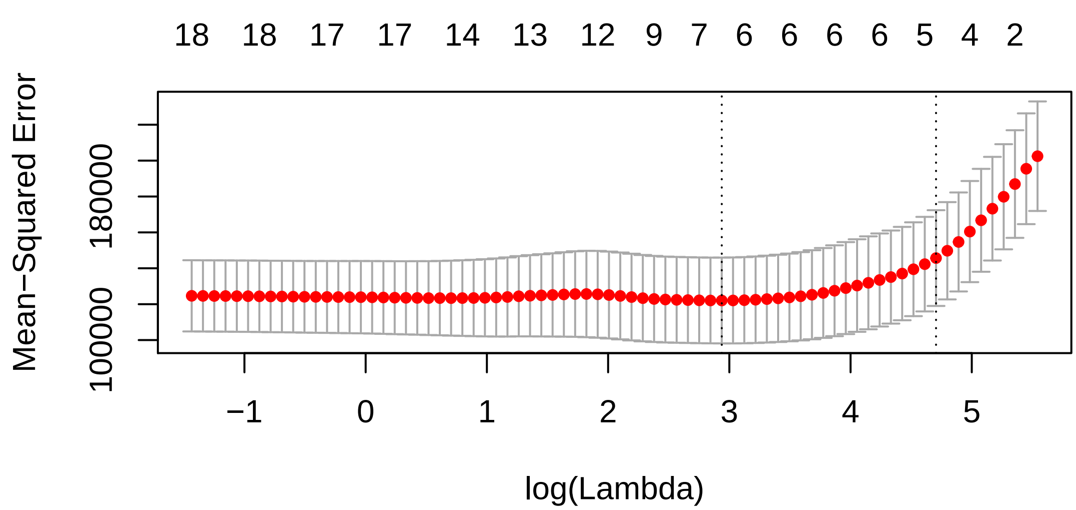
Lasso (L1)

- What is the ideal amount of shrinkage (lambda) for Ridge regression?
- How do the coefficients change with L2 regularization? (To see the traces more clearly, reduce the range of the lambda grid so the "action" is mostly in the plot.)

```r
require(glmnet)

fit_lasso = glmnet(x, y,
                   lambda = grid,
                   alpha = 1)
plot(fit_lasso, xvar = "lambda")
```




```r
fit_lasso_cv = cv.glmnet(x, y, alpha = 1)
plot(fit_lasso_cv)
```



Stepwise Selection

```r
step(fit, direction = "backward")
```

```r
fit_2 = lm(formula = Salary ~ AtBat + Hits + Walks + CAtBat +
             CRuns + CRBI + CWalks + Division + PutOuts + Assists, data = Hitters)
summary(fit_2)
```

```
##
## Call:
## lm(formula = Salary ~ AtBat + Hits + Walks + CAtBat + CRuns +
##     CRBI + CWalks + Division + PutOuts + Assists, data = Hitters)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -939.11 -176.87  -34.08  130.90 1910.55
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  162.53544   66.90784   2.429 0.015830 *
## AtBat         -2.16865    0.53630  -4.044 7.00e-05 ***
## Hits           6.91802    1.64665   4.201 3.69e-05 ***
## Walks          5.77322    1.58483   3.643 0.000327 ***
## CAtBat        -0.13008    0.05550  -2.344 0.019858 *
## CRuns          1.40825    0.39040   3.607 0.000373 ***
## CRBI           0.77431    0.20961   3.694 0.000271 ***
## CWalks        -0.83083    0.26359  -3.152 0.001818 **
## DivisionW   -112.38006   39.21438  -2.866 0.004511 **
## PutOuts        0.29737    0.07444   3.995 8.50e-05 ***
## Assists        0.28317    0.15766   1.796 0.073673 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 311.8 on 252 degrees of freedom
## Multiple R-squared:  0.5405, Adjusted R-squared:  0.5223
## F-statistic: 29.64 on 10 and 252 DF,  p-value: < 2.2e-16
```