HW 4

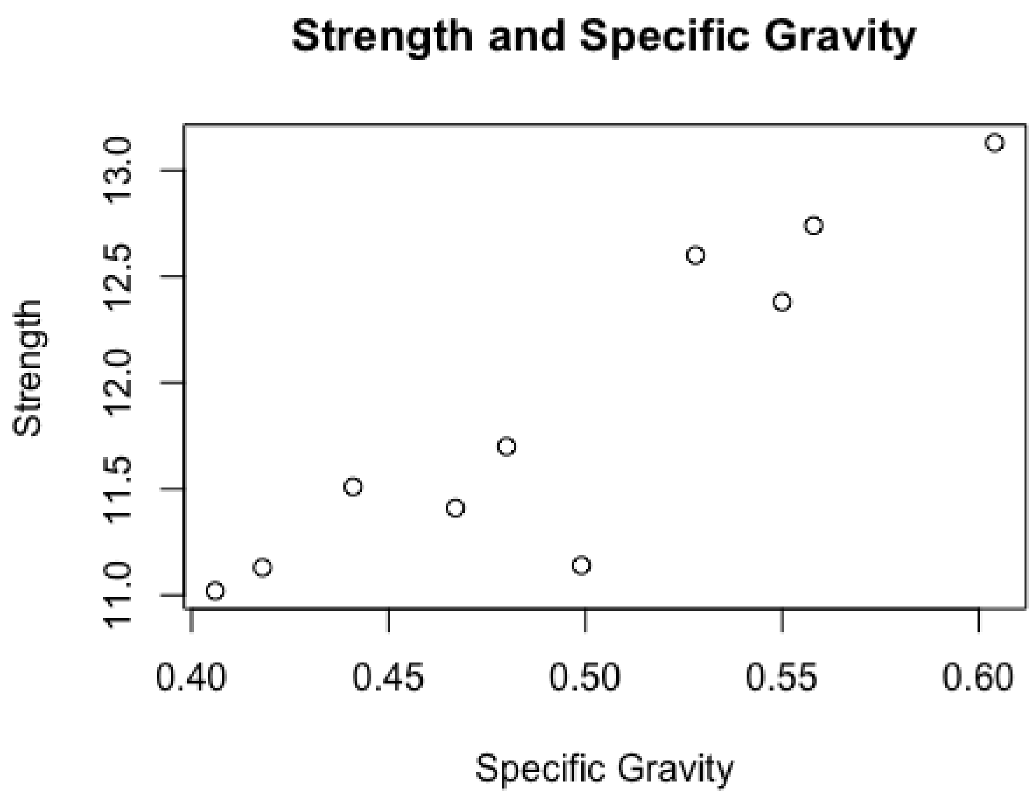
Si Chen

10/31/2018

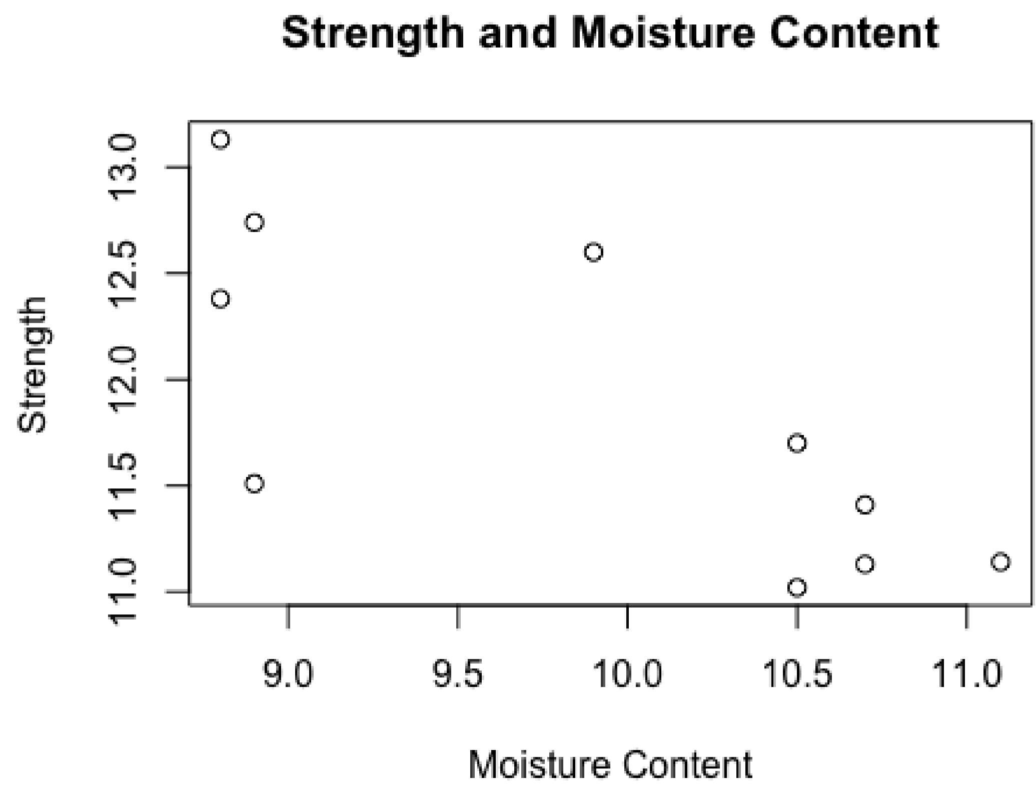
4.8

a)

```
SpecificGravity <-
c(0.499,0.558,0.604,0.441,0.550,0.528,0.418,0.480,0.406,0.467)
MoistureContent <- c(11.1,8.9,8.8,8.9,8.8,9.9,10.7,10.5,10.5,10.7)
Strength <- c(11.14,12.74,13.13,11.51,12.38,12.60,11.13,11.70,11.02,11.41)
plot(SpecificGravity, Strength, main = "Strength and Specific Gravity", xlab
= "Specific Gravity", ylab = "Strength")
```




```
plot(MoistureContent, Strength, main = "Strength and Moisture Content", xlab
= "Moisture Content", ylab = "Strength")
```



Moisture(8.9,11.51), i.e., observation No. 4 seems to be influential.

b)

```
X= as.matrix(data.frame(c(rep(1,10)),SpecificGravity,MoistureContent))
H <- X %*% solve(t(X) %*% X) %*% t(X)
H[4,4]

## [1] 0.6043904

H[4,4] > 2 * (2 + 1) / 10

## [1] TRUE
```

Observation No. 4 is influential.

c)

```
fit_strength <- lm(Strength ~ SpecificGravity + MoistureContent)
cooks.distance(fit_strength)[4]

##         4
## 0.4756415

cooks.distance(fit_strength)[4] > qf(0.2,3,7)
```

```
##    4
## TRUE
```

Observation No. 4 is influential.

d)

```r
fit_str <- lm(Strength ~ SpecificGravity + MoistureContent)
summary(fit_str)
```

```
##
## Call:
## lm(formula = Strength ~ SpecificGravity + MoistureContent)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44422 -0.12780  0.05365  0.10521  0.44985
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      10.3015     1.8965   5.432 0.000975 ***
## SpecificGravity   8.4947     1.7850   4.759 0.002062 **
## MoistureContent  -0.2663     0.1237  -2.152 0.068394 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2754 on 7 degrees of freedom
## Multiple R-squared:    0.9,  Adjusted R-squared:  0.8714
## F-statistic:  31.5 on 2 and 7 DF,  p-value: 0.0003163
```

```r
SpecificGravity_new <- SpecificGravity[c(-4)]
MoistureContent_new <- MoistureContent[c(-4)]
Strength_new <- Strength[c(-4)]
fit_strnew <- lm(Strength_new ~ SpecificGravity_new + MoistureContent_new)
summary(fit_strnew)
```

```
##
## Call:
## lm(formula = Strength_new ~ SpecificGravity_new + MoistureContent_new)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33339 -0.05037  0.01127  0.05615  0.46579
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          12.4107     2.9071   4.269  0.00527 **
## SpecificGravity_new   6.7992     2.5166   2.702  0.03549 *
## MoistureContent_new  -0.3905     0.1794  -2.177  0.07237 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.277 on 6 degrees of freedom
## Multiple R-squared:  0.9108, Adjusted R-squared:  0.8811
## F-statistic: 30.65 on 2 and 6 DF,  p-value: 0.0007089
```

The R-squared improves and the fit changes.

4.10

a)

```
x_1 <- c(rep(8,3),rep(0,3),rep(2,3),rep(0,3))
x_2 <- c(rep(1,3),rep(0,3),rep(7,3),rep(0,3))
x_3 <- c(rep(1,3),rep(9,3),rep(0,6))
x_4 <- c(1,rep(0,2),rep(1,6),rep(10,3))
predictor1 <- data.frame(x_1,x_2,x_3,x_4)
cor(predictor1)
```

```
##             x_1         x_2        x_3        x_4
## x_1  1.00000000  0.05230658 -0.3433818 -0.4976109
## x_2  0.05230658  1.00000000 -0.4315953 -0.3706964
## x_3 -0.34338179 -0.43159531  1.0000000 -0.3551214
## x_4 -0.49761095 -0.37069641 -0.3551214  1.0000000
```

All absolute values of correlations in the correlation matrix are less than 0.5 and thus there is not an indication of multicollinearity.

b)

```
solve(cor(predictor1))
```

```
##          x_1      x_2      x_3      x_4
## x_1 178.2874 166.7955 213.6104 226.4059
## x_2 166.7955 158.0460 201.1317 213.0125
## x_3 213.6104 201.1317 257.9074 272.4421
## x_4 226.4059 213.0125 272.4421 289.3750
```

VIF_1 is 178.29, VIF_2 is 158.05, VIF_3 is 257.91, VIF_4 is 289.38 (maximum)

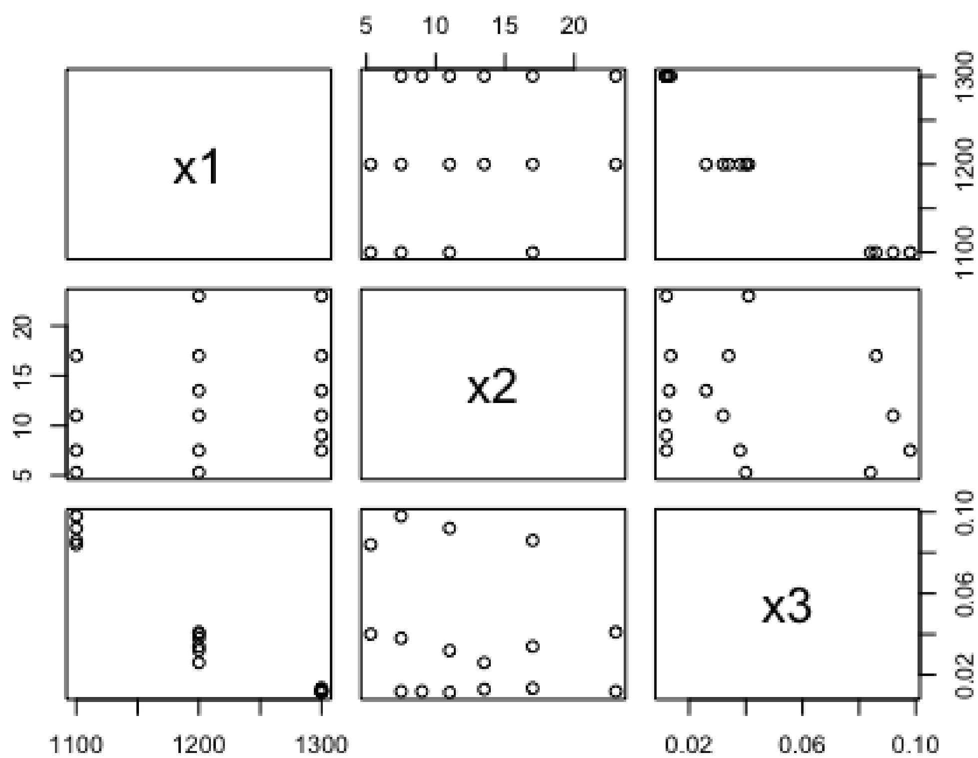
VIF's indicate serious multicollinearity

4.11

a)

```
x1 <- c(rep(1300,6),rep(1200,6),rep(1100,4))
x2 <-
c(7.5,9.0,11.0,13.5,17.0,23.0,5.3,7.5,11.0,13.5,17.0,23.0,5.3,7.5,11.0,17.0)
x3 <-
c(0.0120,0.0120,0.0115,0.0130,0.0135,0.0120,0.0400,0.0380,0.0320,0.0260,0.034
```

```
0,0.0410,0.0840,0.0980,0.0920,0.0860)
y <-
c(49.0,50.2,50.5,48.5,47.5,44.5,28.0,31.5,34.5,35.0,38.0,38.5,15.0,17.0,20.5,
29.5)
predictor <- data.frame(x1,x2,x3)
plot(predictor)
```




```
cor(predictor)

##             x1         x2         x3
## x1  1.0000000  0.2236278 -0.9582041
## x2  0.2236278  1.0000000 -0.2402310
## x3 -0.9582041 -0.2402310  1.0000000
```

Corr(x1,x3) = -0.958, they are largely negative-correlated.

b)

```
x12 <- x1 * x2
x23 <- x2 * x3
x13 <- x1 * x3
x11 <- x1 * x1
x22 <- x2 * x2
x33 <- x3 * x3
```

```
predictor_all <- data.frame(x1,x2,x3,x12,x13,x23,x11,x22,x33)
VIF <- solve(cor(predictor_all))
VIF
```

```
##                 x1          x2          x3          x12          x13
## x1   2856748.965   8897.3985  2390899.263  -3218.1250 -2013929.675
## x2      8897.398  10956.1361    14456.797 -10321.5214    -9999.968
## x3   2390899.263  14456.7971  2017162.536  -9111.3951 -1696804.020
## x12    -3218.125 -10321.5214    -9111.395   9802.9028     5719.427
## x13 -2013929.675   -9999.9677 -1696804.020  5719.4269  1428091.893
## x23    -2991.046   -1593.9702    -3589.211   1488.7481     2689.011
## x11 -2673262.265   -6480.2702 -2235548.668  1300.1341  1883581.393
## x22    -3991.378    -161.8378    -3596.866     83.1922     2911.488
## x33  -185160.442   -1968.4832  -157998.496   1437.6079   132486.496
##               x23          x11          x22          x33
## x1   -2991.04612 -2673262.265 -3991.37822 -185160.4420
## x2   -1593.97025    -6480.270  -161.83776   -1968.4832
## x3   -3589.21100 -2235548.668 -3596.86634 -157998.4958
## x12   1488.74812     1300.134    83.19220    1437.6079
## x13   2689.01109  1883581.393  2911.48790  132486.4956
## x23    240.35938     2527.182    31.55158     413.6923
## x11   2527.18248  2501944.625  3681.63234  172870.4048
## x22     31.55158     3681.632    65.73359     352.2762
## x33    413.69229   172870.405   352.27622   12667.0995
```

VIF_1 is 2856748.965, VIF_2 is 10956.1361, VIF_3 is 2017162.536, VIF_{12} is 9802.9028, VIF_{13} is 1428091.893, VIF_{23} is 240.35938, VIF_{11} is 2501944.625, VIF_{22} is 65.73359, VIF_{33} is 12667.0995

All VIFs are larger than 10. There is a clear indication of multicollinearity among predictors.

c)

```
x1_c <- x1 - mean(x1)
x2_c <- x2 - mean(x2)
x3_c <- x3 - mean(x3)
x12_c <- x1_c * x2_c
x23_c <- x2_c * x3_c
x13_c <- x1_c * x3_c
x11_c <- x1_c * x1_c
x22_c <- x2_c * x2_c
x33_c <- x3_c * x3_c
predictor_all_c <-
data.frame(x1_c,x2_c,x3_c,x12_c,x13_c,x23_c,x11_c,x22_c,x33_c)
VIF_c <- solve(cor(predictor_all_c))
VIF_c
```

```
##                 x1_c         x2_c         x3_c        x12_c       x13_c
## x1_c     375.2477589  -3.07020571   503.120135   0.69112832  1416.40577
## x2_c      -3.0702057   1.74063104    -3.920902   0.01146391   -28.18176
```

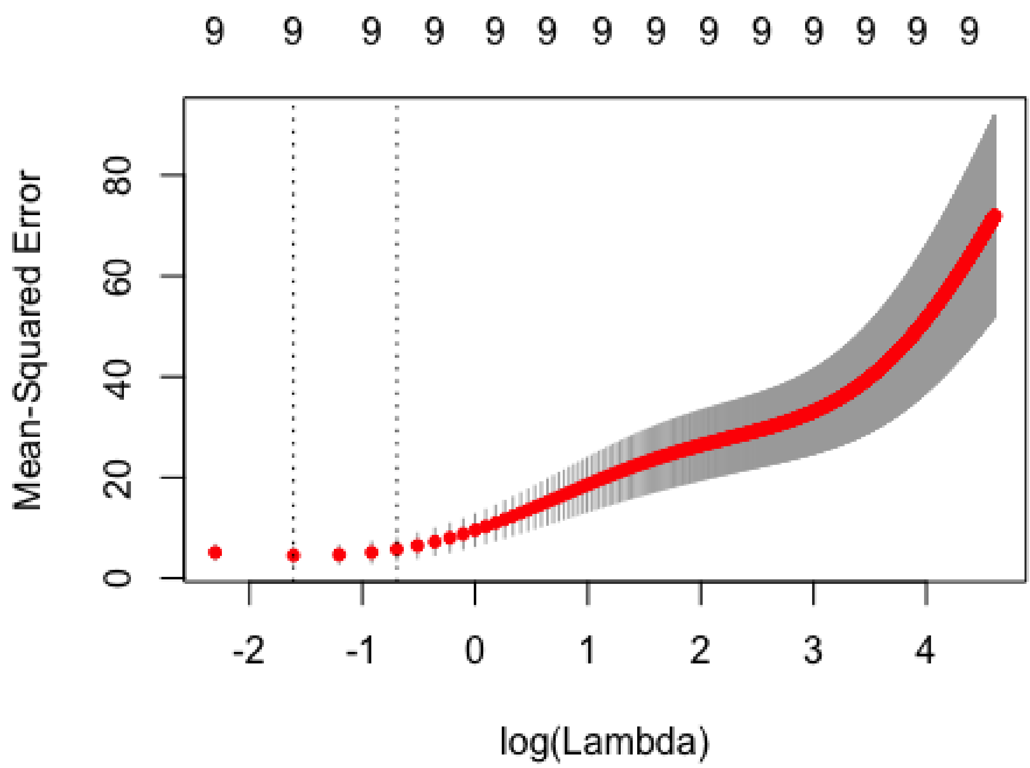
```
## x3_c   503.1201353  -3.92090230  680.280039 -1.79605218 1926.77853
## x12_c    0.6911283   0.01146391   -1.796052 31.03705864   21.81725
## x13_c 1416.4057660 -28.18175565 1926.778533 21.81724905 6563.34519
## x23_c    7.9156857  -0.60988977    7.554270 32.24389120   70.16822
## x11_c  727.5163656 -14.51238927  995.470812  1.94171767 3389.25261
## x22_c    1.6472929  -1.41404985    1.816817  1.02705081   43.30579
## x33_c  560.5223955 -13.25439701  755.819382 24.44486240 2714.19083
##            x23_c       x11_c     x22_c      x33_c
## x1_c   7.9156857  727.516366  1.647293  560.52240
## x2_c  -0.6098898  -14.512389 -1.414050  -13.25440
## x3_c   7.5542701  995.470812  1.816817  755.81938
## x12_c 32.2438912    1.941718  1.027051   24.44486
## x13_c 70.1682154 3389.252609 43.305788 2714.19083
## x23_c 35.6112865   25.818731  2.664592   48.11992
## x11_c 25.8187311 1762.575365 21.439854 1386.56384
## x22_c  2.6645924   21.439854  3.164318   23.35683
## x33_c 48.1199242 1386.563839 23.356833 1156.76628
```

The centering makes the multicollinearity problem less severe.

5.4

install.packages(“glmnet”)

```
library(glmnet)

## Loading required package: Matrix

## Loading required package: foreach

## Loaded glmnet 2.0-16

ridgecv = cv.glmnet(as.matrix(predictor_all), y, lambda =
seq(0,100,0.1),alpha = 0)

## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3
observations
## per fold

plot(ridgecv)
```



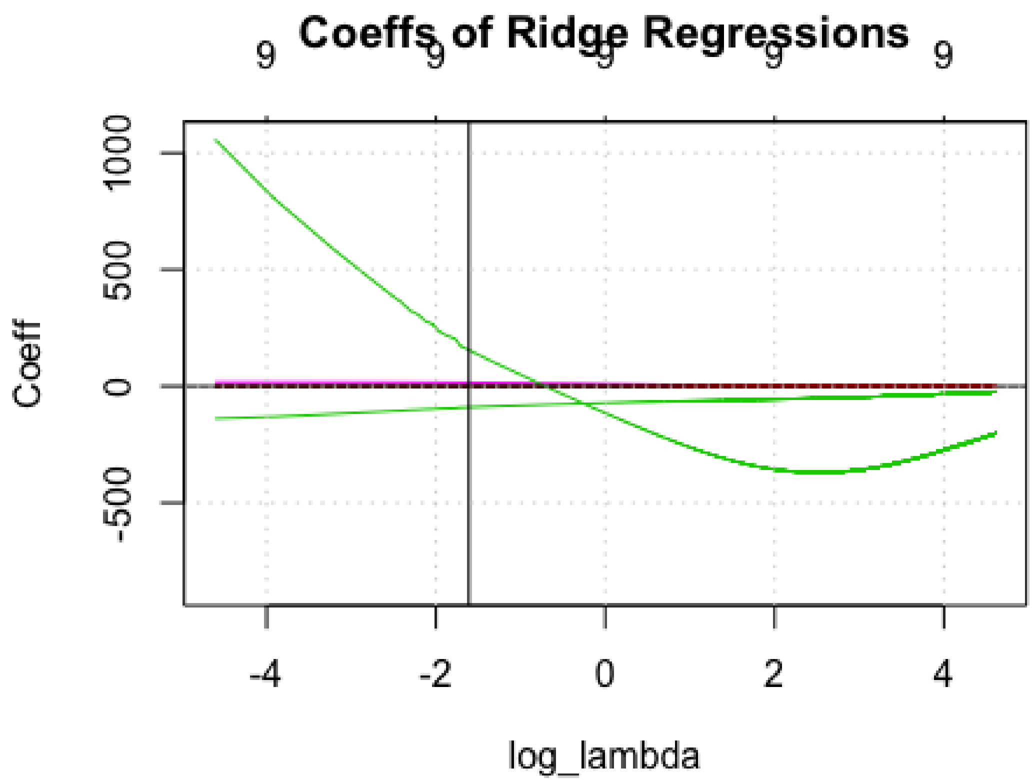

```r
small.lambda.index <- which(ridgecv$lambda == ridgecv$lambda.min)
small.lambda.betas <- coef(ridgecv$glmnet.fit)[,small.lambda.index]
print(small.lambda.betas)
```

```
##   (Intercept)            x1            x2            x3           x12
## -8.300143e+01  6.484227e-02  2.563943e-01 -9.277340e+01  3.260840e-05
##           x13           x23           x11           x22           x33
## -1.034145e-01  1.641786e+01  2.780714e-05 -2.013840e-02  1.788287e+02
```

```r
lambdaridge = ridgecv$lambda.min
print(lambdaridge)
```

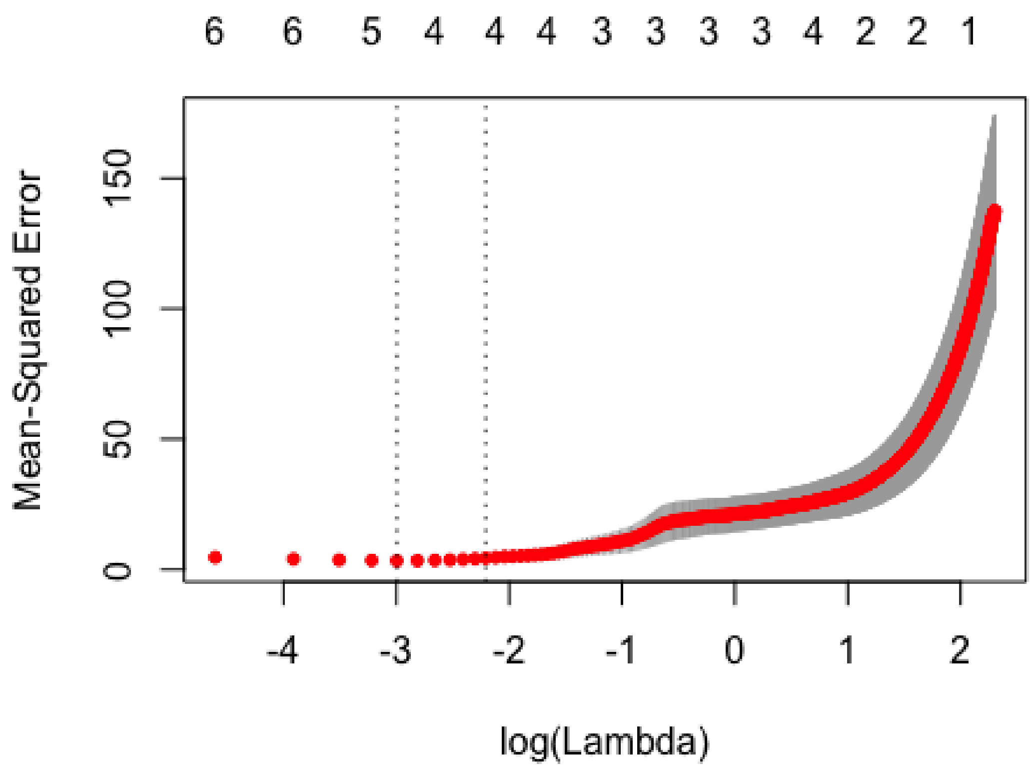
```
## [1] 0.2
```

```r
ridgefit = glmnet(as.matrix(predictor_all), y, alpha = 0, lambda =
seq(0,100,0.01))
plot(ridgefit, xvar = "lambda", main = "Coeffs of Ridge Regressions", xlab =
expression("log_lambda"), ylab = "Coeff")
abline(h = 0); abline(v = log(ridgecv$lambda.min))
grid()
```

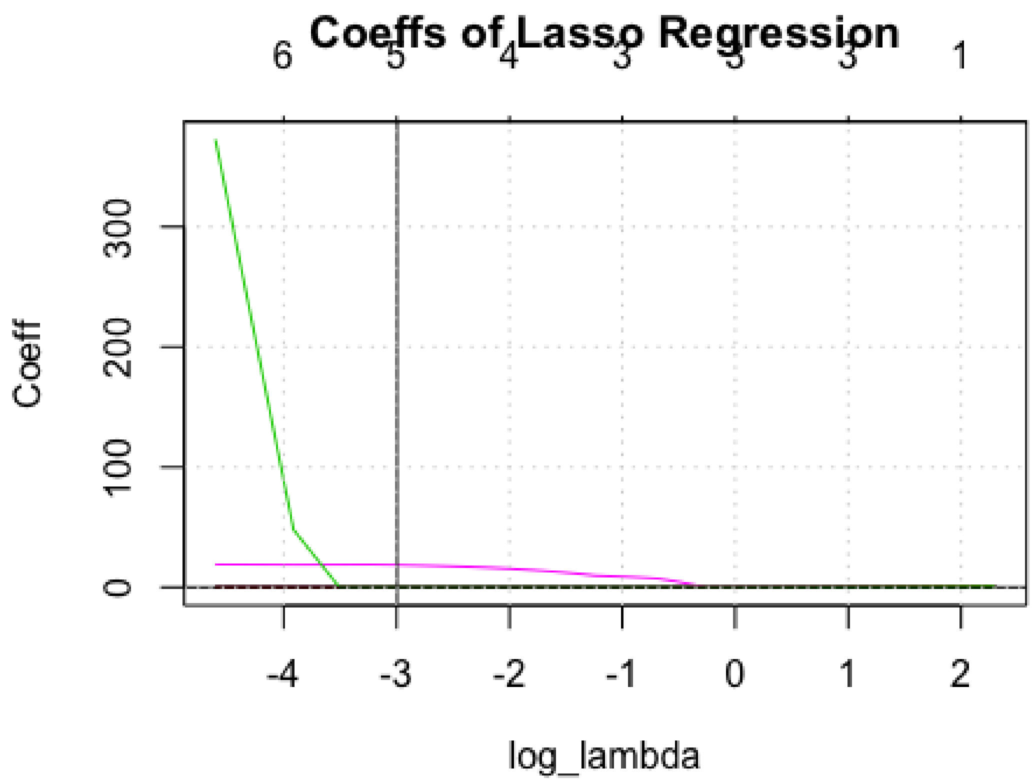


5.5

```
lassocv = cv.glmnet(as.matrix(predictor_all), y, alpha = 1, lambda =
seq(0,10,0.01))

## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3
observations
## per fold

plot(lassocv)
```




```r
lambdalasso = lassocv$lambda.min
print(lambdalasso)

## [1] 0.05

lassofit = glmnet(as.matrix(predictor_all), y, alpha = 1, lambda =
seq(0,10,0.01))
plot(lassofit, xvar = "lambda", label = TRUE, main = "Coeffs of Lasso
Regression", xlab = expression("log_lambda"), ylab = "Coeff")
abline(h = 0)
abline(v = log(lassocv$lambda.min))
grid()
```




```
small.lambda.index <- which(lassocv$lambda == lassocv$lambda.min)
small.lambda.betas <- coef(lassocv$glmnet.fit)[,small.lambda.index]
print(small.lambda.betas)
```

```
##   (Intercept)            x1            x2            x3           x12
## -4.523077e+01  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
##           x13           x23           x11           x22           x33
## -1.874581e-01  1.862793e+01  5.678371e-05 -1.270771e-02  0.000000e+00
```

$\beta_1, \beta_2, \beta_3, \beta_{12}, \beta_{33}$ are set to zero.