PSTAT 127 Homework 5

Kevin Ayala

3/6/2019

(a) Load the faraway package and take a look at the data description by typing ?fat into the R console. Do you suspect that some regularization may be helpful in fitting this linear model? Explain.

No I do not think that regularization will be somewhat helpful here, we note that there are around 200 observations, thus we do not have to worry about such things such as curse of dimensionality here. Since there are 18 variables (and not like 500 or a million), we can eliminate some by normal means such as finding significant predictors and so on from regular regression.

```
library(faraway)
?fat
head(fat)
```

```
##   brozek siri density age weight height adipos  free neck chest abdom
## 1   12.6 12.3  1.0708  23 154.25  67.75   23.7 134.9 36.2  93.1  85.2
## 2    6.9  6.1  1.0853  22 173.25  72.25   23.4 161.3 38.5  93.6  83.0
## 3   24.6 25.3  1.0414  22 154.00  66.25   24.7 116.0 34.0  95.8  87.9
## 4   10.9 10.4  1.0751  26 184.75  72.25   24.9 164.7 37.4 101.8  86.4
## 5   27.8 28.7  1.0340  24 184.25  71.25   25.6 133.1 34.4  97.3 100.0
## 6   20.6 20.9  1.0502  24 210.25  74.75   26.5 167.0 39.0 104.5  94.4
##     hip thigh knee ankle biceps forearm wrist
## 1  94.5  59.0 37.3  21.9   32.0    27.4  17.1
## 2  98.7  58.7 37.3  23.4   30.5    28.9  18.2
## 3  99.2  59.6 38.9  24.0   28.8    25.2  16.6
## 4 101.2  60.1 37.3  22.8   32.4    29.4  18.2
## 5 101.9  63.2 42.2  24.0   32.2    27.7  17.7
## 6 107.8  66.0 42.0  25.6   35.7    30.6  18.8
```

b) Use the code below to divide your data into two sets - a training set fatTrain used to fit the model and a testing set fatTest.

```
set.seed(123) # ensures that everyone uses the same data split
# will give you a testing data set of 25 men, training set of 227

# Create matrix version of input

test.ind <- sample.int(n = nrow(fat), size = floor(0.1*nrow(fat))) #is the same as exameple
train.ind <- setdiff(1:nrow(fat), test.ind) #is the same as the example
fatTrain <- fat[train.ind,]
fatTest <- fat[test.ind,]
```

c) Using the training data fatTrain, fit the linear model using four methods
d) Ordinary Least Squares with all predictors

ii) Ordinary Least Squares after performing backward stepwise selection using AIC
iii) Ridge regression, using $\lambda = 0.5$
iv) Lasso regression, using $\lambda = 0.1$

```
library(glmnet)
```

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

```
## Loading required package: foreach
## Loaded glmnet 2.0-16
```

```
linear_model <- lm(siri~.-brozek-density, data = fatTrain)
summary(linear_model)
```

```
##
## Call:
## lm(formula = siri ~ . - brozek - density, data = fatTrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5942 -0.6466  0.1729  0.9229  6.4780
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -14.529352   6.458658  -2.250 0.025507 *
## age           0.005613   0.012220   0.459 0.646469
## weight        0.356261   0.022864  15.582  < 2e-16 ***
## height        0.044965   0.040195   1.119 0.264556
## adipos       -0.520058   0.113147  -4.596  7.4e-06 ***
## free         -0.561511   0.014598 -38.465  < 2e-16 ***
## neck          0.033761   0.090155   0.374 0.708427
## chest         0.140050   0.039138   3.578 0.000429 ***
## abdom         0.148341   0.039930   3.715 0.000260 ***
## hip          -0.016505   0.056439  -0.292 0.770236
## thigh         0.191258   0.054616   3.502 0.000564 ***
## knee          0.163204   0.095777   1.704 0.089854 .
## ankle         0.124333   0.080382   1.547 0.123417
## biceps        0.099167   0.064351   1.541 0.124808
## forearm       0.222717   0.071956   3.095 0.002234 **
## wrist         0.152718   0.204993   0.745 0.457108
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.529 on 211 degrees of freedom
## Multiple R-squared:  0.9697, Adjusted R-squared:  0.9675
## F-statistic: 449.8 on 15 and 211 DF,  p-value: < 2.2e-16
```

```
AIC_lin_reg <- step(linear_model, direction = 'backward')
```

```
## Start:  AIC=208.06
## siri ~ (brozek + density + age + weight + height + adipos + free +
##     neck + chest + abdom + hip + thigh + knee + ankle + biceps +
##     forearm + wrist) - brozek - density
##
##           Df Sum of Sq    RSS    AIC
## - hip      1       0.2  493.2 206.15
## - neck     1       0.3  493.3 206.21
## - age      1       0.5  493.5 206.28
## - wrist    1       1.3  494.3 206.65
## - height   1       2.9  495.9 207.40
## <none>                  493.0 208.06
## - biceps   1       5.5  498.6 208.60
## - ankle    1       5.6  498.6 208.62
## - knee     1       6.8  499.8 209.16
## - forearm  1      22.4  515.4 216.14
## - thigh    1      28.7  521.7 218.88
## - chest    1      29.9  522.9 219.43
## - abdom    1      32.2  525.3 220.44
## - adipos   1      49.4  542.4 227.72
## - weight   1     567.3 1060.3 379.89
## - free     1    3457.0 3950.0 678.43
##
## Step:  AIC=206.15
## siri ~ age + weight + height + adipos + free + neck + chest +
##     abdom + thigh + knee + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - neck     1       0.4  493.7 204.35
## - age      1       0.5  493.7 204.39
## - wrist    1       1.3  494.5 204.76
## - height   1       3.2  496.4 205.62
## <none>                  493.2 206.15
## - ankle    1       5.8  499.0 206.80
## - biceps   1       6.0  499.2 206.88
## - knee     1       6.6  499.9 207.19
## - forearm  1      23.0  516.2 214.51
## - thigh    1      29.9  523.1 217.50
## - chest    1      32.2  525.4 218.50
## - abdom    1      32.4  525.6 218.60
## - adipos   1      52.1  545.3 226.92
## - weight   1     673.0 1166.2 399.50
## - free     1    3492.1 3985.3 678.45
##
## Step:  AIC=204.35
## siri ~ age + weight + height + adipos + free + chest + abdom +
##     thigh + knee + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - age      1       0.7  494.3 202.67
## - wrist    1       1.9  495.5 203.21
## - height   1       3.4  497.1 203.93
## <none>                  493.7 204.35
## - ankle    1       5.5  499.2 204.87
## - knee     1       6.2  499.9 205.21
## - biceps   1       6.4  500.1 205.29
## - forearm  1      24.3  518.0 213.27
## - thigh    1      30.1  523.8 215.81
## - chest    1      32.2  525.8 216.68
## - abdom    1      32.6  526.3 216.88
## - adipos   1      51.7  545.3 224.94
## - weight   1     685.6 1179.2 400.02
## - free     1    3534.9 4028.5 678.90
##
## Step:  AIC=202.67
## siri ~ weight + height + adipos + free + chest + abdom + thigh +
##     knee + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - height   1       3.2  497.5 202.11
## - wrist    1       3.5  497.8 202.26
## <none>                  494.3 202.67
## - ankle    1       5.1  499.5 203.01
## - biceps   1       7.0  501.3 203.85
## - knee     1       7.5  501.8 204.07
## - forearm  1      23.6  518.0 211.27
## - thigh    1      31.6  525.9 214.73
## - chest    1      33.5  527.8 215.55
## - abdom    1      38.4  532.8 217.67
## - adipos   1      52.0  546.4 223.39
## - weight   1     693.6 1188.0 399.70
## - free     1    3607.2 4101.6 680.98
##
## Step:  AIC=202.11
## siri ~ weight + adipos + free + chest + abdom + thigh + knee +
##     ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - wrist    1       4.2  501.7 202.03
## <none>                  497.5 202.11
## - ankle    1       5.4  502.9 202.58
## - knee     1       6.5  504.0 203.06
## - biceps   1       7.4  504.9 203.45
## - forearm  1      24.3  521.8 210.96
## - thigh    1      30.0  527.5 213.39
## - chest    1      33.9  531.4 215.08
## - abdom    1      40.3  537.8 217.80
## - adipos   1      91.2  588.7 238.34
## - weight   1     813.8 1311.3 420.12
## - free     1    3615.6 4113.1 679.61
##
## Step:  AIC=202.03
## siri ~ weight + adipos + free + chest + abdom + thigh + knee +
##     ankle + biceps + forearm
##
##           Df Sum of Sq    RSS    AIC
## <none>                  501.7 202.03
## - ankle    1       7.6  509.3 203.42
## - knee     1       8.6  510.3 203.89
## - biceps   1       8.6  510.4 203.90
## - thigh    1      26.0  527.7 211.48
## - forearm  1      28.9  530.6 212.75
## - chest    1      33.5  535.2 214.71
## - abdom    1      43.8  545.5 219.01
## - adipos   1      88.7  590.4 236.99
## - weight   1     814.4 1316.1 418.95
## - free     1    3761.3 4263.0 685.74
```

```r
summary(AIC_lin_reg)
```

```
##
## Call:
## lm(formula = siri ~ weight + adipos + free + chest + abdom +
##     thigh + knee + ankle + biceps + forearm, data = fatTrain)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5886 -0.6057  0.1790  0.9211  6.6879
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.06279    3.86138  -2.606 0.009798 **
## weight        0.36091    0.01927  18.725  < 2e-16 ***
## adipos       -0.57877    0.09365  -6.180 3.15e-09 ***
## free         -0.55627    0.01382 -40.241  < 2e-16 ***
## chest         0.14443    0.03802   3.799 0.000189 ***
## abdom         0.16139    0.03718   4.341 2.18e-05 ***
## thigh         0.15164    0.04536   3.343 0.000978 ***
## knee          0.17285    0.08977   1.926 0.055470 .
## ankle         0.13984    0.07755   1.803 0.072758 .
## biceps        0.12027    0.06235   1.929 0.055040 .
## forearm       0.24222    0.06863   3.529 0.000509 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.524 on 216 degrees of freedom
## Multiple R-squared:  0.9691, Adjusted R-squared:  0.9677
## F-statistic: 678.4 on 10 and 216 DF,  p-value: < 2.2e-16
```

```r
X_train <- as.matrix(fatTrain[4:18]) # removing density and brozek,
Y_train <- fatTrain$siri

X_test <-as.matrix(fatTest[4:18])
Y_test <- fatTest$siri

ridge_model <- glmnet(x=X_train, y=Y_train, alpha = 0, lambda = .5)

lasso_model <- glmnet(x=X_train, y=Y_train, alpha = 1, lambda = .1)
```

Use the fitted models to compute predicted body fat percentages for the test data. Which method has the lowest average squared prediction error on the testing data?

```r
linear_model_pred <- predict(linear_model, newdata = fatTest)
linear_model_error <- sum((Y_test - linear_model_pred)^2)
linear_model_error
```

```
## [1] 48.17286
```

```r
AIC_lin_reg_pred <- predict(AIC_lin_reg, newdata = fatTest)
AIC_model_error <- sum((Y_test - AIC_lin_reg_pred)^2)
AIC_model_error
```

```
## [1] 46.57431
```

```r
ridge_predict <- ridge_model$a0 + X_test%*%ridge_model$beta
ridge_error <- sum((Y_test - ridge_predict)^2)
ridge_error
```

```
## [1] 103.1617
```

```
lasso_predict <- lasso_model$a0 + X_test%*%lasso_model$beta
Lasso_error<- sum((Y_test - lasso_predict)^2)
Lasso_error
```

```
## [1] 50.04396
```

```
#Records Holder
Error_record <- matrix(data = NA, nrow=4, ncol = 1)
rownames(Error_record) <- c("Linear Model", "AIC Linear Model", "Ridge Model", "Lasso Model")
colnames(Error_record) <- "Mean Squared Predicted Errors"
Error_record[1] <-linear_model_error
Error_record[2] <- AIC_model_error
Error_record[3] <- ridge_error
Error_record[4] <- Lasso_error

Error_record <- as.data.frame(Error_record)
Error_record
```

```
##                   Mean Squared Predicted Errors
## Linear Model                          48.17286
## AIC Linear Model                      46.57431
## Ridge Model                          103.16171
## Lasso Model                           50.04396
```

Lowest Average predicted Mean Square Value comes from our AIC model, which by AIC, gives the best predictors.