Homework 6

Rolando Santos

2023-12-09

Question 1: We're going to use the mtcars dataset that can be found in the R package "datasets". Import the dataset by running "library(datasets); data(mtcars)".

```r
library(datasets)
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

a. Fit a logistic regression model with the variable am as the response and mpg and hp as predictors. What are the estimated regression coefficients from this model? How do we interpret them here?

```r
lmod <- glm(am ~ mpg + hp, family = binomial, data = mtcars)
summary(lmod)
```

```
##
## Call:
## glm(formula = am ~ mpg + hp, family = binomial, data = mtcars)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -33.60517   15.07672  -2.229   0.0258 *
## mpg           1.25961    0.56747   2.220   0.0264 *
## hp            0.05504    0.02692   2.045   0.0409 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 43.230  on 31  degrees of freedom
## Residual deviance: 19.233  on 29  degrees of freedom
## AIC: 25.233
```

```
##
## Number of Fisher Scoring iterations: 7
```

One unit increase of `mpg` will increase the odds of a car being automatic by a factor of $e^{1.25961} = 3.524$.

One unit incrase of `hp` will increase the odds of a car being automatic by a factor of $e^{0.05504} = 1.056$.

Both coefficients are significant at the 5% level.

b. What is the predicted probability that a car is automatic if it has hp = 180 and mpg = 20?

```r
intercept <- -33.60517
mpg_coef <- 1.25961
hp_coef <- 0.05504

p <- intercept + mpg_coef * 20 + hp_coef * 180
round(1/(1+exp(-p)), 3)
```

```
## [1] 0.817
```

```r
round(predict(lmod, list(mpg = 20, hp = 180), type = "response"), 3)
```

```
##     1
## 0.817
```

Using r `predict()` and manually calculating the predicted probability, we can see that with `mpg` = 20 and `hp` = 180 the probability of a car being automatic is 0.817.

c. Randomly split the data into a 80% train set and a 20% test set. Fit a logistic model on the training set and predict on the test set. What is the prediction accuracy of transmission type on the test set? (Hint: if the probability of being 1 is greater than 0.5 then set the transmission type equal to 1, otherwise, set it to 0)

```r
set.seed(2023)
index.train <- sample(1:dim(mtcars)[1], 0.8 * dim(mtcars)[1])
data.train <- mtcars[index.train,]
data.test <- mtcars[-index.train,]

lmod <- glm(am ~ mpg + hp, data = data.train, family = binomial)
p.pred <- predict(lmod, data.test, type='response')

y.pred <- ifelse(p.pred > 0.5, 1, 0)
y.truth <- data.test$am
acc.test <- mean(y.pred==y.truth)
acc.test
```

```
## [1] 0.8571429
```

The prediction accuracy for our model is 85.7%.

d. Show the confusion matrix. Calculate the true positive rate, true negative rate, and precision.

```r
table(y.pred, y.truth)
```

```
##       y.truth
## y.pred 0 1
##      0 4 0
##      1 1 2
```

```r
# True Positive
TP <- intersect(which(y.truth==1), which(y.pred==1))
# True Negative
TN <- intersect(which(y.truth==0), which(y.pred==0))
# False Positive
FP <- which(y.truth[which(y.pred==1)]==0)
# False Negative
FN <- which(y.truth[which(y.pred==0)]==1)

# Precision
prec <- length(TP) / (length(TP) + length(FP))
prec
```

```
## [1] 0.6666667
```

```r
# True Positive Rate
TPR <- length(TP) / (length(TP) + length(FN))
TPR
```

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

```r
# True Negative Rate
TNR <- length(TN) / (length(TN) + length(FP))
TNR
```

```
## [1] 0.8
```

Question 2: Use seatpos data to conduct the following analysis. Make sure you understand the meaning of each variable in this dataset.

```r
seatpos <- read.csv("seatpos.csv")
head(seatpos)
```

```
##   Age Weight HtShoes    Ht Seated  Arm Thigh  Leg hipcenter
## 1  46    180   187.2 184.9   95.2 36.1  45.3 41.3  -206.300
## 2  31    175   167.5 165.5   83.8 32.9  36.5 35.9  -178.210
## 3  23    100   153.6 152.2   82.9 26.0  36.6 31.0   -71.673
## 4  19    185   190.3 187.4   97.3 37.4  44.1 41.0  -257.720
## 5  23    159   178.0 174.1   93.9 29.5  40.1 36.9  -173.230
## 6  47    170   178.7 177.0   92.4 36.0  43.2 37.4  -185.150
```

a. Use hipcenter as response and all other variables as predictors to fit a linear model. How you interpret this model? What is the issue of this model?

```r
model <- lm(hipcenter ~ ., data = seatpos)
summary(model)
```

```
##
## Call:
## lm(formula = hipcenter ~ ., data = seatpos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -73.827 -22.833  -3.678  25.017  62.337
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 436.43213  166.57162   2.620   0.0138 *
## Age           0.77572    0.57033   1.360   0.1843
## Weight        0.02631    0.33097   0.080   0.9372
## HtShoes      -2.69241    9.75304  -0.276   0.7845
## Ht            0.60134   10.12987   0.059   0.9531
## Seated        0.53375    3.76189   0.142   0.8882
## Arm          -1.32807    3.90020  -0.341   0.7359
## Thigh        -1.14312    2.66002  -0.430   0.6706
## Leg          -6.43905    4.71386  -1.366   0.1824
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.72 on 29 degrees of freedom
## Multiple R-squared:  0.6866, Adjusted R-squared:  0.6001
## F-statistic:  7.94 on 8 and 29 DF,  p-value: 1.306e-05
```

One glaring issue that we notice is that none of our coefficients are significant at any level.

b. Use cor function to check the correlation of all predictors. What predictors are highly correlated? Is there any relation between correlations and model fitting in (a)?

```r
round(cor(seatpos), 2)
```

```
##             Age Weight HtShoes    Ht Seated   Arm Thigh   Leg hipcenter
## Age        1.00   0.08   -0.08 -0.09  -0.17  0.36  0.09 -0.04      0.21
## Weight     0.08   1.00    0.83  0.83   0.78  0.70  0.57  0.78     -0.64
## HtShoes   -0.08   0.83    1.00  1.00   0.93  0.75  0.72  0.91     -0.80
## Ht        -0.09   0.83    1.00  1.00   0.93  0.75  0.73  0.91     -0.80
## Seated    -0.17   0.78    0.93  0.93   1.00  0.63  0.61  0.81     -0.73
## Arm        0.36   0.70    0.75  0.75   0.63  1.00  0.67  0.75     -0.59
## Thigh      0.09   0.57    0.72  0.73   0.61  0.67  1.00  0.65     -0.59
## Leg       -0.04   0.78    0.91  0.91   0.81  0.75  0.65  1.00     -0.79
## hipcenter  0.21  -0.64   -0.80 -0.80  -0.73 -0.59 -0.59 -0.79      1.00
```

We see that `Ht` and `HtShoes` are perfectly correlated. `Weight` and `Ht` (and `HtShoes`), `Weight` and `Seated`, `Weight` and `Leg`, `Seated` and `Ht`, `Leg` and `Ht` are all highly positively correlated. `Ht` and `Hipcenter` are highly negatively correlated. Outside of `Age`, `Ht` appears to be the most highly correlated variable against all other variables.

c. Conduct a PCA transformation on all predictors. How much variance the first two PCs have?

```r
pr.out <- prcomp(seatpos[,1:8], scale = TRUE)
summary(pr.out)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5    PC6     PC7
## Standard deviation     2.3818 1.1121 0.68099 0.49088 0.44070 0.3731 0.22438
## Proportion of Variance 0.7091 0.1546 0.05797 0.03012 0.02428 0.0174 0.00629
## Cumulative Proportion  0.7091 0.8638 0.92171 0.95183 0.97611 0.9935 0.99980
##                            PC8
## Standard deviation     0.03985
## Proportion of Variance 0.00020
## Cumulative Proportion  1.00000
```

The proportion of variance for the first two principal components is 0.7091 for PC1 and 0.1546 for PC2. The variance lowers significantly for all other principal components.

d. Show the linear combination coefficients in the first two PCs. Based on those coefficients, what interpretation can you make for the first two PCs?

```r
phi <- pr.out$rotation[, 1:2]
phi
```

```
##                  PC1        PC2
## Age     -0.007219379 -0.8763467
## Weight  -0.366979122 -0.0448877
## HtShoes -0.411460536  0.1055831
## Ht      -0.412057421  0.1119799
## Seated  -0.381270226  0.2178995
## Arm     -0.348771387 -0.3742641
## Thigh   -0.327523319 -0.1251793
## Leg     -0.389747512  0.0555930
```

In PC1 we can see that all coefficients aside from `Age` are weighted nearly the same, whereas in PC2 `Age` is weighted highly compared to the other coefficients which are weighted differently than PC1's coefficients with `Arm` being the exception. Another thing to note is that all of PC1's coefficients are negative whereas PC2's coefficients are a mix of positive and negative values.

e. Conduct a PCA regression of hipcenter vs. first two PCs. How do you interpret this model result? Compare this model with the regular linear regression in (a) and give your insight.

```
Z <- pr.out$x
seatpos.pca <- data.frame(Z[, 1:2], hipcenter = seatpos$hipcenter)
model.pca <- lm(hipcenter ~ ., seatpos.pca)
summary(model.pca)
```

```
##
## Call:
## lm(formula = hipcenter ~ ., data = seatpos.pca)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -84.643 -25.582  -0.743  24.887  61.798
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -164.885      5.772 -28.568  < 2e-16 ***
## PC1           19.701      2.456   8.022 1.93e-09 ***
## PC2          -11.321      5.259  -2.153   0.0383 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.58 on 35 degrees of freedom
## Multiple R-squared:  0.6634, Adjusted R-squared:  0.6442
## F-statistic:  34.5 on 2 and 35 DF,  p-value: 5.292e-09
```

In this model we notice an increase in adjusted r-squared and both coefficients PC1 and PC2 are significant.

Question 3: Take the fat data, and use the percentage of body fat, siri, as the response and the other variables, except brozek and density, as potential predictors. Remove every tenth observation from the data for use as the test set (1, 11, 21, . . .). Use the remaining data as the training data building the following models, predict on the test set, and calculate the prediction RMSE on the test set.

```
fat <- read.csv("fat.csv")
train <- fat[-seq(1, nrow(fat), 10), ]
test <- fat[seq(1, nrow(fat), 10), ]
train <- train[, !names(train) %in% c('brozek', 'density')]
test <- test[, !names(test) %in% c('brozek', 'density')]
RMSEs <- c()
head(train)
```

```
##   siri age weight height adipos  free neck chest abdom   hip thigh knee ankle
## 2  6.1  22 173.25  72.25   23.4 161.3 38.5  93.6  83.0  98.7  58.7 37.3  23.4
## 3 25.3  22 154.00  66.25   24.7 116.0 34.0  95.8  87.9  99.2  59.6 38.9  24.0
## 4 10.4  26 184.75  72.25   24.9 164.7 37.4 101.8  86.4 101.2  60.1 37.3  22.8
## 5 28.7  24 184.25  71.25   25.6 133.1 34.4  97.3 100.0 101.9  63.2 42.2  24.0
## 6 20.9  24 210.25  74.75   26.5 167.0 39.0 104.5  94.4 107.8  66.0 42.0  25.6
## 7 19.2  26 181.00  69.75   26.2 146.6 36.4 105.1  90.7 100.3  58.4 38.3  22.9
##   biceps forearm wrist
## 2   30.5    28.9  18.2
```

```
## 3   28.8    25.2  16.6
## 4   32.4    29.4  18.2
## 5   32.2    27.7  17.7
## 6   35.7    30.6  18.8
## 7   31.9    27.8  17.7
```

a. Linear regression with all predictors.

```r
lm.model <- lm(siri ~ ., data = train)
yhat <- predict(lm.model, test)
RMSEs[1] <- sqrt(mean((yhat - test$siri)^2))
RMSEs[1]
```

```
## [1] 1.946023
```

b. Linear regression with variables selected using backward AIC (hint: consider step function).

```r
step(lm.model, direction = "backward")
```

```
## Start:  AIC=186.31
## siri ~ age + weight + height + adipos + free + neck + chest +
##     abdom + hip + thigh + knee + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - hip      1       0.0  447.4 184.32
## - neck     1       0.2  447.5 184.39
## - knee     1       0.2  447.5 184.39
## - age      1       0.3  447.6 184.45
## - wrist    1       1.4  448.7 185.02
## - height   1       1.6  449.0 185.13
## - ankle    1       2.9  450.2 185.76
## <none>                  447.3 186.31
## - biceps   1      10.7  458.1 189.66
## - abdom    1      16.1  463.5 192.31
## - forearm  1      18.5  465.8 193.47
## - chest    1      23.3  470.6 195.76
## - thigh    1      25.4  472.7 196.78
## - adipos   1      42.1  489.4 204.62
## - weight   1     576.0 1023.4 371.33
## - free     1    3385.3 3832.6 669.75
##
## Step:  AIC=184.32
## siri ~ age + weight + height + adipos + free + neck + chest +
##     abdom + thigh + knee + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - neck     1       0.2  447.5 182.39
## - knee     1       0.2  447.5 182.39
## - age      1       0.3  447.7 182.47
## - wrist    1       1.4  448.8 183.03
```

```
## - height   1       1.7  449.1 183.19
## - ankle    1       3.0  450.4 183.83
## <none>                  447.4 184.32
## - biceps   1      10.8  458.2 187.72
## - abdom    1      16.4  463.7 190.44
## - forearm  1      18.8  466.2 191.63
## - chest    1      24.8  472.1 194.50
## - thigh    1      27.1  474.4 195.59
## - adipos   1      43.6  491.0 203.34
## - weight   1     683.5 1130.8 391.90
## - free     1    3415.7 3863.0 669.54
##
## Step:  AIC=182.39
## siri ~ age + weight + height + adipos + free + chest + abdom +
##     thigh + knee + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - knee     1       0.2  447.7 180.50
## - age      1       0.2  447.8 180.52
## - wrist    1       1.3  448.8 181.03
## - height   1       1.7  449.2 181.23
## - ankle    1       3.3  450.8 182.07
## <none>                  447.5 182.39
## - biceps   1      10.7  458.2 185.74
## - abdom    1      16.4  463.9 188.54
## - forearm  1      18.7  466.2 189.66
## - chest    1      24.7  472.2 192.55
## - thigh    1      26.9  474.4 193.60
## - adipos   1      45.7  493.2 202.38
## - weight   1     688.4 1135.9 390.90
## - free     1    3464.1 3911.6 670.37
##
## Step:  AIC=180.5
## siri ~ age + weight + height + adipos + free + chest + abdom +
##     thigh + ankle + biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - age      1       0.4  448.1 178.68
## - wrist    1       1.3  449.1 179.17
## - height   1       1.6  449.3 179.30
## - ankle    1       4.0  451.7 180.49
## <none>                  447.7 180.50
## - biceps   1      10.6  458.3 183.76
## - abdom    1      16.6  464.3 186.72
## - forearm  1      19.1  466.8 187.94
## - chest    1      24.7  472.4 190.62
## - thigh    1      32.1  479.8 194.15
## - adipos   1      48.9  496.6 201.94
## - weight   1     731.7 1179.4 397.41
## - free     1    3464.0 3911.7 668.37
##
## Step:  AIC=178.68
## siri ~ weight + height + adipos + free + chest + abdom + thigh +
##     ankle + biceps + forearm + wrist
```

```
##
##           Df Sum of Sq    RSS    AIC
## - height   1       1.4  449.5 177.41
## - wrist    1       2.4  450.5 177.89
## - ankle    1       3.9  452.0 178.63
## <none>                  448.1 178.68
## - biceps   1      10.8  458.9 182.08
## - forearm  1      18.7  466.8 185.94
## - abdom    1      20.1  468.2 186.59
## - chest    1      25.1  473.2 188.99
## - thigh    1      33.4  481.5 192.95
## - adipos   1      49.4  497.5 200.31
## - weight   1     738.0 1186.1 396.68
## - free     1    3491.5 3939.6 667.97
##
## Step:  AIC=177.41
## siri ~ weight + adipos + free + chest + abdom + thigh + ankle +
##     biceps + forearm + wrist
##
##           Df Sum of Sq    RSS    AIC
## - wrist    1       2.6  452.1 176.72
## - ankle    1       3.9  453.5 177.38
## <none>                  449.5 177.41
## - biceps   1      11.2  460.7 180.98
## - forearm  1      19.0  468.6 184.79
## - abdom    1      20.4  469.9 185.44
## - chest    1      25.3  474.9 187.81
## - thigh    1      32.1  481.6 190.99
## - adipos   1      79.2  528.7 212.09
## - weight   1     847.9 1297.4 414.96
## - free     1    3492.9 3942.4 666.14
##
## Step:  AIC=176.72
## siri ~ weight + adipos + free + chest + abdom + thigh + ankle +
##     biceps + forearm
##
##           Df Sum of Sq    RSS    AIC
## <none>                  452.1 176.72
## - ankle    1       6.1  458.2 177.74
## - biceps   1      12.9  465.1 181.09
## - forearm  1      22.1  474.2 185.50
## - abdom    1      23.4  475.5 186.12
## - chest    1      25.3  477.4 187.01
## - thigh    1      29.5  481.7 189.02
## - adipos   1      79.2  531.3 211.20
## - weight   1     847.4 1299.6 413.33
## - free     1    3709.0 4161.1 676.34
```

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

```
## (Intercept)      weight      adipos        free       chest       abdom
##     -2.9190      0.3925     -0.5277     -0.5698      0.1246      0.1179
##       thigh       ankle      biceps     forearm
##      0.1561      0.1475      0.1490      0.2146
```

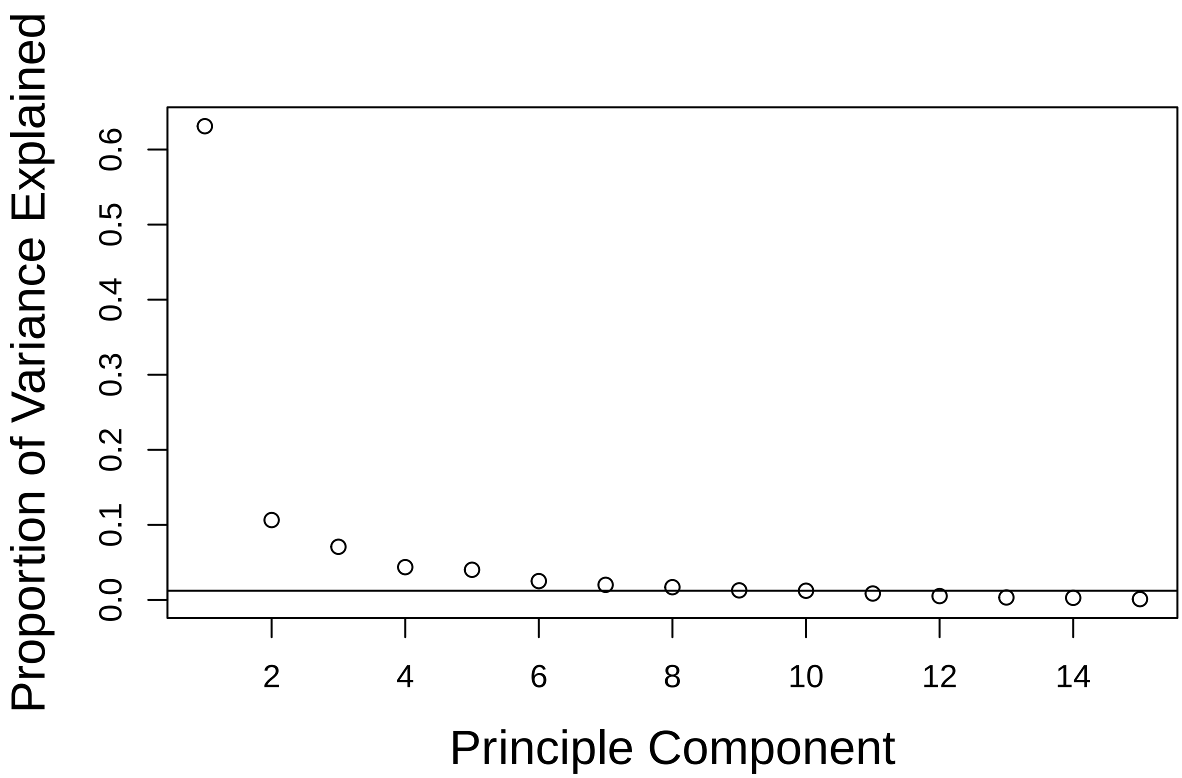
```
aic.model <- lm(siri ~ weight + adipos + free + chest + abdom + thigh + ankle +
    biceps + forearm, data = train)
yhat <- predict(aic.model, test)
RMSEs[2] <- sqrt(mean((yhat - test$siri)^2))
RMSEs[2]
```

```
## [1] 1.98911
```

c. Principal component regression. Use the first 7 PCs.

```
pr.out <- prcomp(train[,2:16], scale = TRUE)
phi <- pr.out$rotation

PVE.matrix <- summary(pr.out)$importance
PVE <- PVE.matrix[2,]
plot(PVE, xlab='Principle Component', ylab='Proportion of Variance Explained', cex.lab=1.5)
abline(a=PVE[10], b=0)
text(x=20, y=0.04, labels='Elbow point 10 PCs', cex=1.5)
```




```
pca.train <- data.frame(pr.out$x[, 1:7], siri = train$siri)
pca.test <- predict(pr.out, test)
```

```r
pca.test <- data.frame(pca.test[, 1:7], siri = test$siri)
pcr.model <- lm(siri ~ ., data = pca.train)
yhat <- predict(pcr.model, pca.test)
RMSEs[3] <- sqrt(mean((yhat - pca.test$siri)^2))
RMSEs[3]
```

```
## [1] 4.003633
```

d. Ridge regression. Use cross-validation on the training set to select best penalty

```r
require(MASS)
```

```
## Loading required package: MASS
```

```r
rg.model <- lm.ridge(siri ~ ., data = train, lambda = seq(0, 5e-8, len=21))
which.min(rg.model$GCV)
```

```
## 5.00e-08
##       21
```

```r
yhat <- cbind(1,as.matrix(test[,-1])) %*% coef(rg.model)[21,]
RMSEs[4] <- sqrt(mean((yhat - test$siri)^2))
RMSEs[4]
```

```
## [1] 1.946023
```

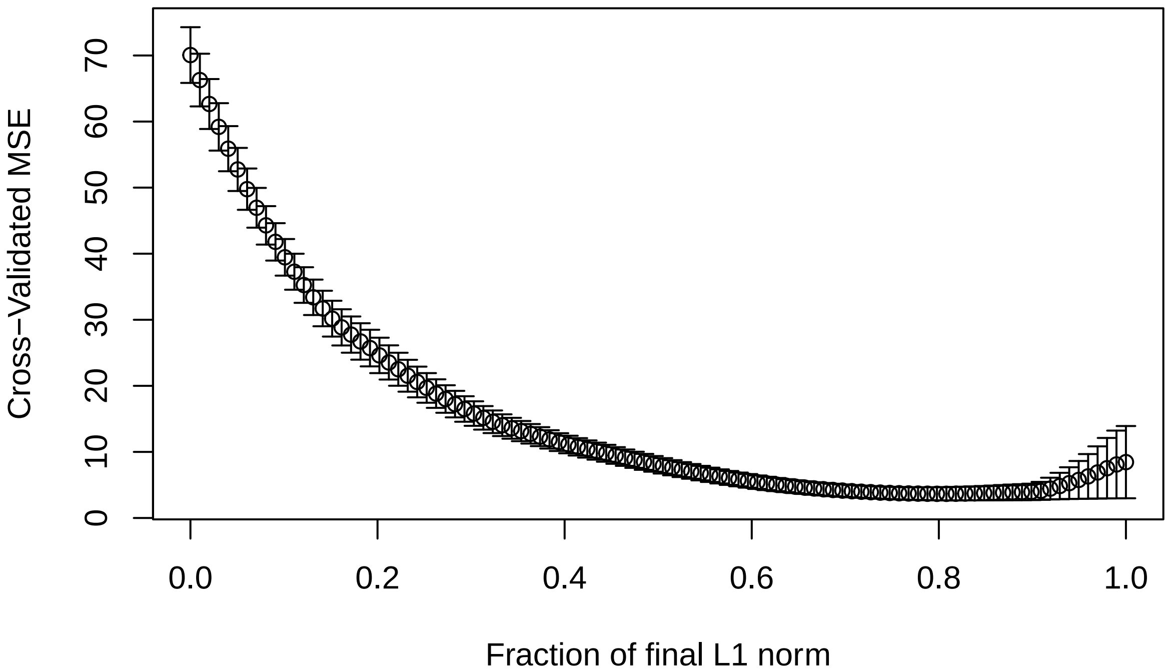
e. Lasso. Use cross-validation on the training set to select best penalty.

```r
require(lars)
```

```
## Loading required package: lars
```

```
## Loaded lars 1.3
```

```r
las.model <- lars(as.matrix(train[,2:16]),train$siri)
set.seed(2022)
cvout <- cv.lars(as.matrix(train[,2:16]),train$siri)
```




```
cvout$index[which.min(cvout$cv)]
```

```
## [1] 0.8080808
```

```
yhat <- predict(las.model, as.matrix(test[,2:16]), s=0.8080808,mode="fraction")
RMSEs[5] <- sqrt(mean((yhat$fit - test$siri)^2))
RMSEs[5]
```

```
## [1] 1.935332
```

f. Compare all the RMSEs. Are you surprised on the model performance comparison? Give you speculation about why you see such result.

```
library(knitr)
table <- data.frame(
  c("Linear", "AIC", "PCA", "Ridge", "Lasso"), RMSEs
)
kable(table, format = "markdown", col.names = c("Regression", "RMSE"))
```

Regression	RMSE
Linear	1.946023
AIC	1.989110
PCA	4.003633
Ridge	1.946023
Lasso	1.935332

The most suprising model result is the AIC model, typically you would expect with a reduction of insignificant variables we would attain a better model but that does not appear to be the case. The PCA model having a higher RMSE makes sense because although we have a moderate amount of variables, it wouldn't be considered high dimensional data.

```
round(cor(train[, 2:16]), 2)
```

```
##            age weight height adipos  free neck chest abdom   hip thigh knee
## age       1.00  -0.02  -0.18   0.12 -0.23 0.12  0.18  0.22 -0.05 -0.19 0.01
## weight   -0.02   1.00   0.30   0.89  0.80 0.82  0.89  0.89  0.94  0.87 0.86
## height   -0.18   0.30   1.00  -0.03  0.48 0.25  0.13  0.08  0.16  0.14 0.26
## adipos    0.12   0.89  -0.03   1.00  0.54 0.77  0.91  0.92  0.88  0.81 0.73
## free     -0.23   0.80   0.48   0.54  1.00 0.68  0.59  0.49  0.71  0.68 0.71
## neck      0.12   0.82   0.25   0.77  0.68 1.00  0.78  0.75  0.73  0.69 0.67
## chest     0.18   0.89   0.13   0.91  0.59 0.78  1.00  0.91  0.83  0.72 0.72
## abdom     0.22   0.89   0.08   0.92  0.49 0.75  0.91  1.00  0.87  0.76 0.74
## hip      -0.05   0.94   0.16   0.88  0.71 0.73  0.83  0.87  1.00  0.90 0.84
## thigh    -0.19   0.87   0.14   0.81  0.68 0.69  0.72  0.76  0.90  1.00 0.81
## knee      0.01   0.86   0.26   0.73  0.71 0.67  0.72  0.74  0.84  0.81 1.00
## ankle    -0.10   0.66   0.25   0.54  0.60 0.49  0.51  0.48  0.59  0.59 0.67
## biceps   -0.04   0.81   0.21   0.75  0.65 0.73  0.73  0.69  0.75  0.76 0.69
## forearm -0.09   0.62   0.22   0.54  0.54 0.60  0.56  0.48  0.53  0.55 0.56
## wrist     0.20   0.72   0.33   0.60  0.69 0.74  0.64  0.60  0.62  0.55 0.67
##          ankle biceps forearm wrist
## age      -0.10  -0.04   -0.09  0.20
## weight    0.66   0.81    0.62  0.72
## height    0.25   0.21    0.22  0.33
## adipos    0.54   0.75    0.54  0.60
## free      0.60   0.65    0.54  0.69
## neck      0.49   0.73    0.60  0.74
## chest     0.51   0.73    0.56  0.64
## abdom     0.48   0.69    0.48  0.60
## hip       0.59   0.75    0.53  0.62
## thigh     0.59   0.76    0.55  0.55
## knee      0.67   0.69    0.56  0.67
## ankle     1.00   0.52    0.46  0.60
## biceps    0.52   1.00    0.67  0.64
## forearm   0.46   0.67    1.00  0.56
## wrist     0.60   0.64    0.56  1.00
```

We can still see in our correlation matrix that we do have quite a few correlated variables too, so PCA isn't a bad option for this model, but if we are only looking at RMSE it performs the worst.

Both Ridge and LASSO performed well, however it appears that Ridge performed exactly the same as our normal linear regression model. Which means out original model didn't suffer from any kind of overfitting (relative to ridge). Our LASSO model performed the best, which could mean that the L1 penalty introduced in LASSO made more of a difference.