NCSU ST 503 HW 10

Probems 11.1, 11.2, 11.3, and 11.4 Faraway, Julian J. Linear Models with R, Second Edition Chapman & Hall / CRC Press.

Bruce Campbell

05 November, 2017

11.1 seatpos PCR analysis

Using the seatpos data, perform a PCR analysis with hipcenter as the response and HtShoes, Ht, Seated, Arm, Thigh and Leg as predictors. Select an appropriate number of components and give an interpretation to those you choose. Add Age and Weight as predictors and repeat the analysis. Use both models to predict the response for predictors taking these values:

$$(HtShoes, Ht, Seated, Arm, Thigh, Leg, Age, Weight) = (181.080, 178.560, 91.440, 35.640, 40.950, 38.790, 6$$

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6
## Standard deviation     2.2240 0.7082 0.58575 0.39551 0.22554 0.04149
## Proportion of Variance 0.8244 0.0836 0.05718 0.02607 0.00848 0.00029
## Cumulative Proportion  0.8244 0.9080 0.96516 0.99124 0.99971 1.00000
```

We see that the first three PCA cpmonents account for 96.5% of the variance and the proportion of the variance in the third component is 0.8%. We could choose to fit a regression model with the first two or three principal components. First we investigate the loadings on the first two principal components to see if we can discern any patterns that will allow for interpretation. Based on that we can decide how many components to put in the model.

Table 1: First Principal Component

	first.pc.loadings
HtShoes	-0.441
Ht	-0.442
Seated	-0.408
Arm	-0.374
Thigh	-0.359
Leg	-0.418

Table 2: Second Principal Component

	first.pc.loadings
HtShoes	-0.201
Ht	-0.186
Seated	-0.464
Arm	0.485
Thigh	0.673
Leg	-0.149

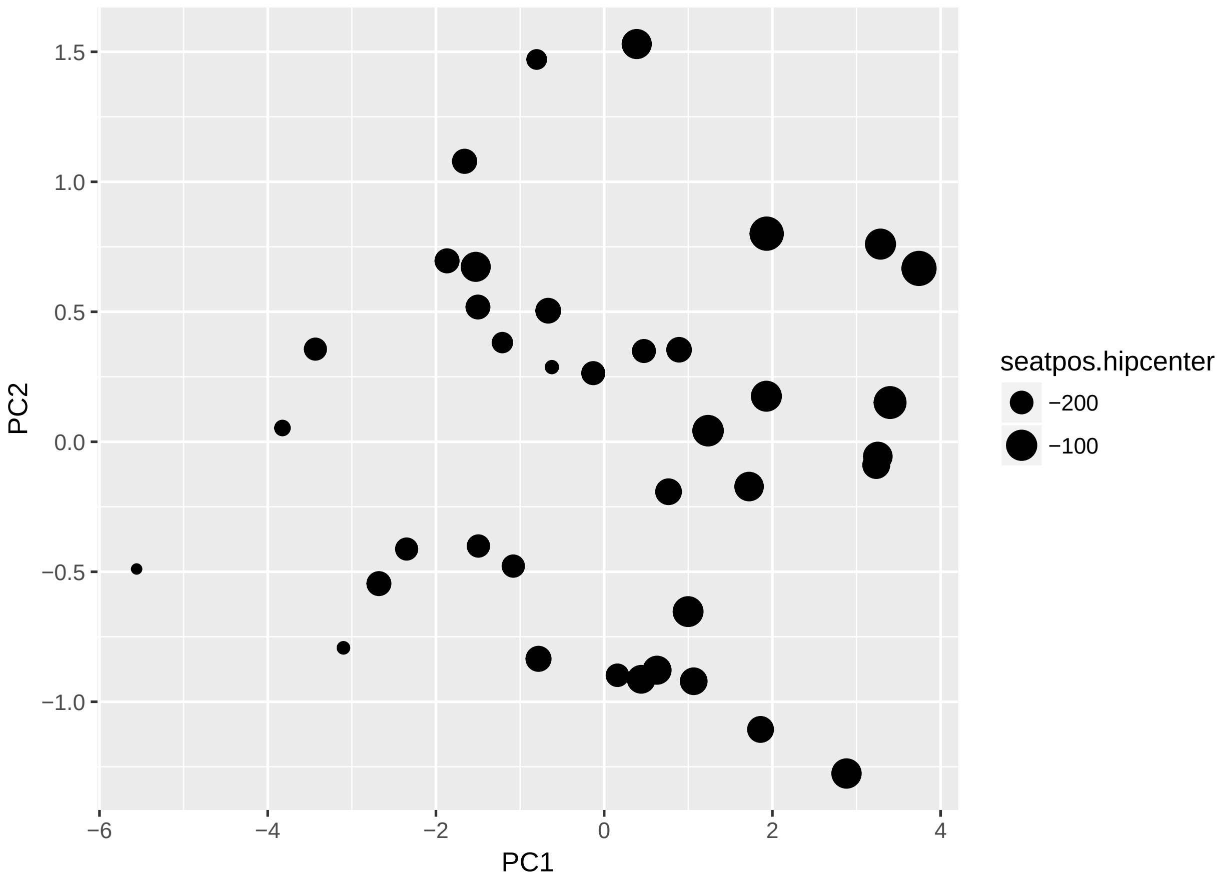
We see that the first component is an average size measure while the second is a contrast measure between $\{Arm, Thigh\}$ and $\{HtShoes, Ht, Seated, Leg\}$.

Table 3: Third Principal Component

	first.pc.loadings
HtShoes	0.065
Ht	0.082
Seated	0.189
Arm	-0.707
Thigh	0.627
Leg	-0.245

The third principal component is a contrast between $\{Arm, Leg\}$ and $\{HtShoes, Ht, Seated, Thigh\}$ We leave this out of the regression model.

Here's a bubble plot of the first 2 components sized by the response.



Now we perform the PCR on the first 2 components.

```
##
## Call:
## lm(formula = seatpos$hipcenter ~ pca.seatpos$x[, 1:2])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -93.076 -28.678   3.274  23.196  72.607
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -164.885      5.949 -27.715  < 2e-16 ***
## pca.seatpos$x[, 1:2]PC1   21.261      2.711   7.843 3.23e-09 ***
## pca.seatpos$x[, 1:2]PC2    9.939      8.513   1.168    0.251
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.67 on 35 degrees of freedom
## Multiple R-squared:  0.6424, Adjusted R-squared:  0.622
## F-statistic: 31.44 on 2 and 35 DF,  p-value: 1.53e-08
```

Full Model

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

Table 4: First Principal Component

	first.pc.loadings
HtShoes	-0.411
Ht	-0.412
Seated	-0.381
Arm	-0.349
Thigh	-0.328
Leg	-0.39
Age	-0.007
Weight	-0.367

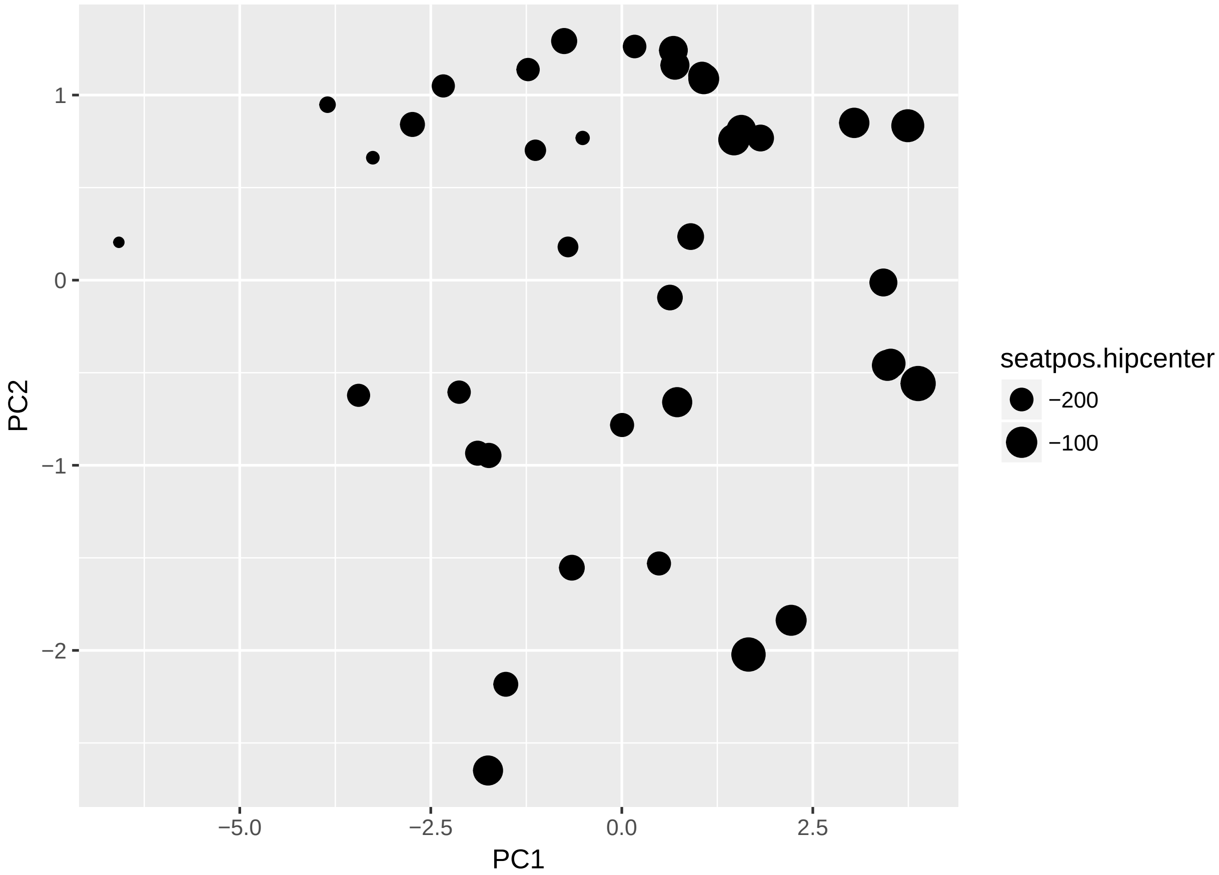
Table 5: Second Principal Component

	first.pc.loadings
HtShoes	0.106
Ht	0.112
Seated	0.218
Arm	-0.374
Thigh	-0.125
Leg	0.056
Age	-0.876
Weight	-0.045

Table 6: Third Principal Component

	first.pc.loadings
HtShoes	0.034
Ht	0.011
Seated	0.171

	first.pc.loadings
Arm	-0.017
Thigh	-0.862
Leg	0.117
Age	0.164
Weight	0.43



```
##
## Call:
## lm(formula = seatpos$hipcenter ~ pca.seatpos.full$x[, 1:2])
##
## 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 ***
## pca.seatpos.full$x[, 1:2]PC1   19.701      2.456   8.022 1.93e-09 ***
## pca.seatpos.full$x[, 1:2]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
```

We tried three PC's but did not achieve significant results for the this component's coefficient and we dropped that term from the model. The First PC had the same interpretation while second PC added Age and Weight to the Arm ,Thigh part of the contrast $\{Arm, Thigh\}$ and $\{HtShoes, Ht, Seated, Leg\}$ from our first model. Thus the second PC can be interpreted as a contrast between $\{Arm, Thigh, Age, Weight\}$ and $\{HtShoes, Ht, Seated, Leg\}$.

To do the prediction we need to scale (we used scaling) and project the test point onto the the first two PCA. We were also careful when creating the prediction data element to order the variables as they were in the rotation matrix. We had some trouble with the predict function so we went ahead and calculated the predicted value manually. First we scaled, then rotated, then took the first 2 components to calculate $\hat{\beta} \cdot\ x_0$

```
DFTest <- data.frame( HtShoes=181.080, Ht=178.560, Seated=91.440, Arm=35.640, Thigh=40.9

x <- as.matrix(DFTest)

x <- (x-mean.df.full) / sd.df.full

R <- pca.seatpos.full$rotation

x.r <- R %*% t(x)

pred.manual.comp  <- lm.pcr.full$coefficients["(Intercept)"] +  lm.pcr.full$coefficients
names(pred.manual.comp) <- "predicted.hipcenter"

pander(data.frame(pred.manual.comp=pred.manual.comp), caption = "Predicted hipcenter for
```

Table 7: Predicted hipcenter for full data element

	pred.manual.comp
predicted.hipcenter	-222.8

Now we calculate the predicted hipcenter for the reduced data in a similar fashion- i.e. no $Age, Weight$.

Table 8: Predicted hipcenter for model with no Age, Weight

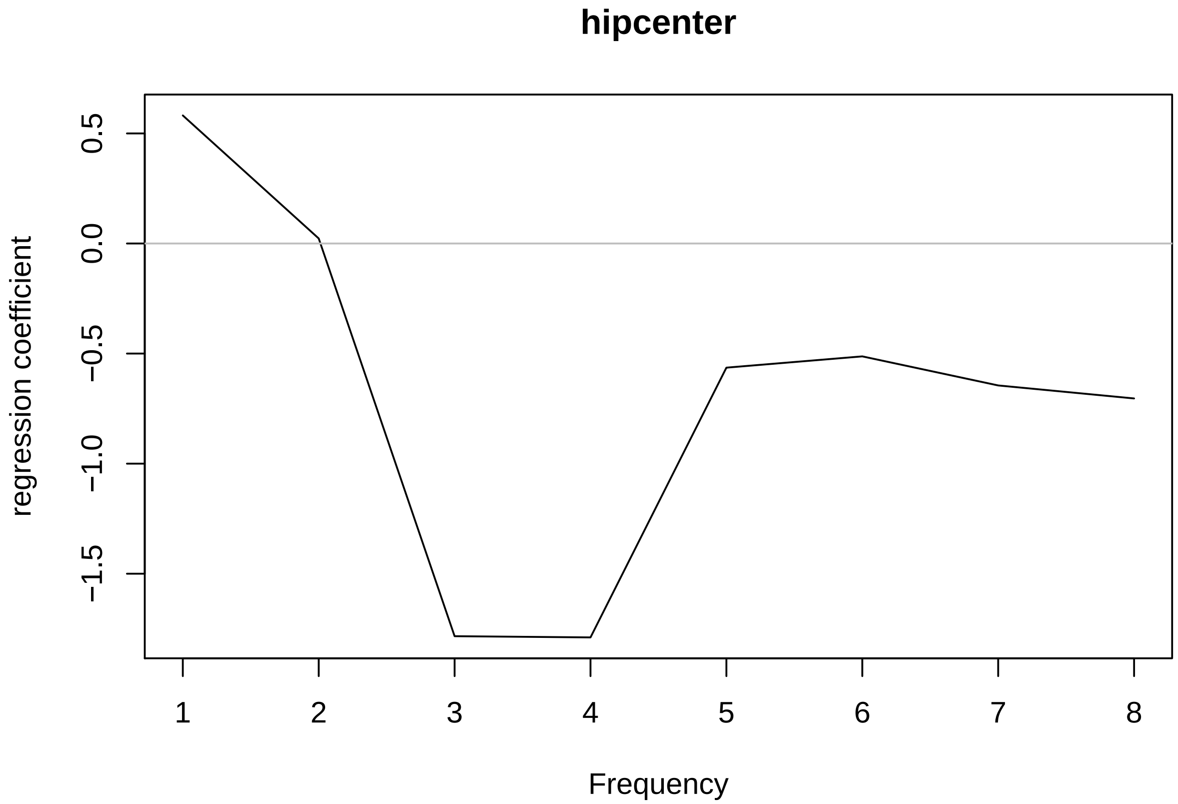
	pred.manual.comp
predicted.hipcenter	-178

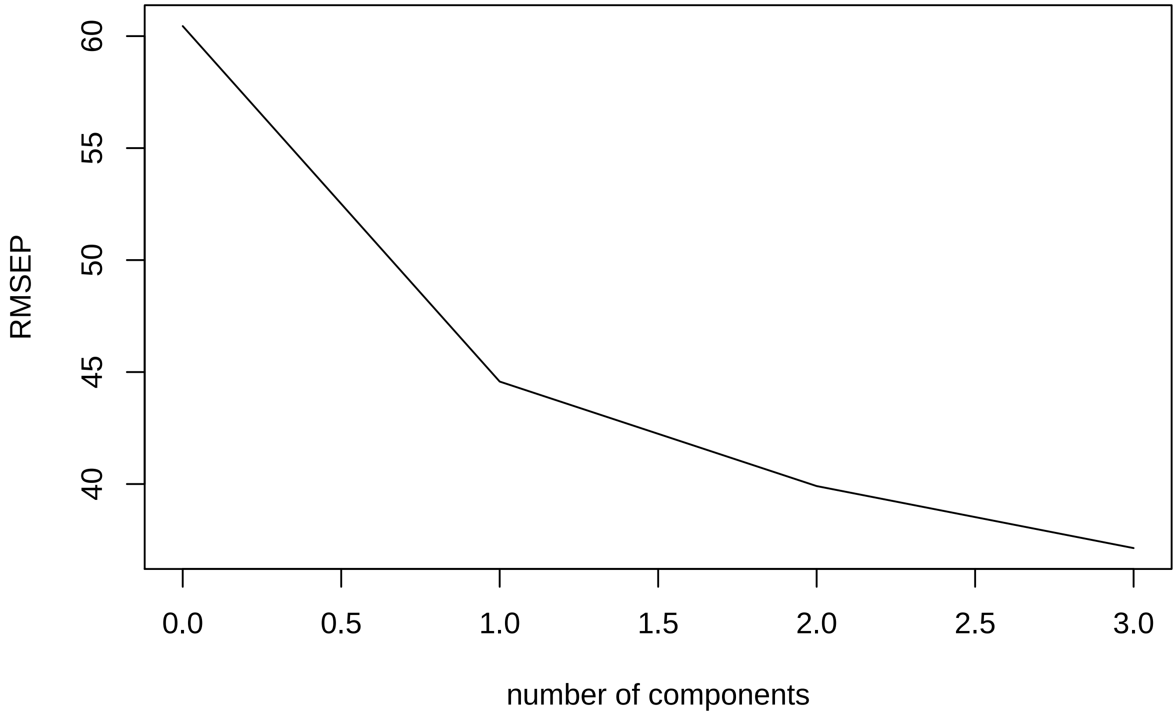
We get a markedly different result in this case.

11.2 PLS analysis with seatpos data

Fit a PLS model to the seatpos data with hipcenter as the response and all other variables as predictors. Take care to select an appropriate number of components. Use the model to predict the response at the values of the predictors specified in the first question.

Based on our PCA modelling and some experimenting we choose to go with three components.





Now we predict the response for the test data.

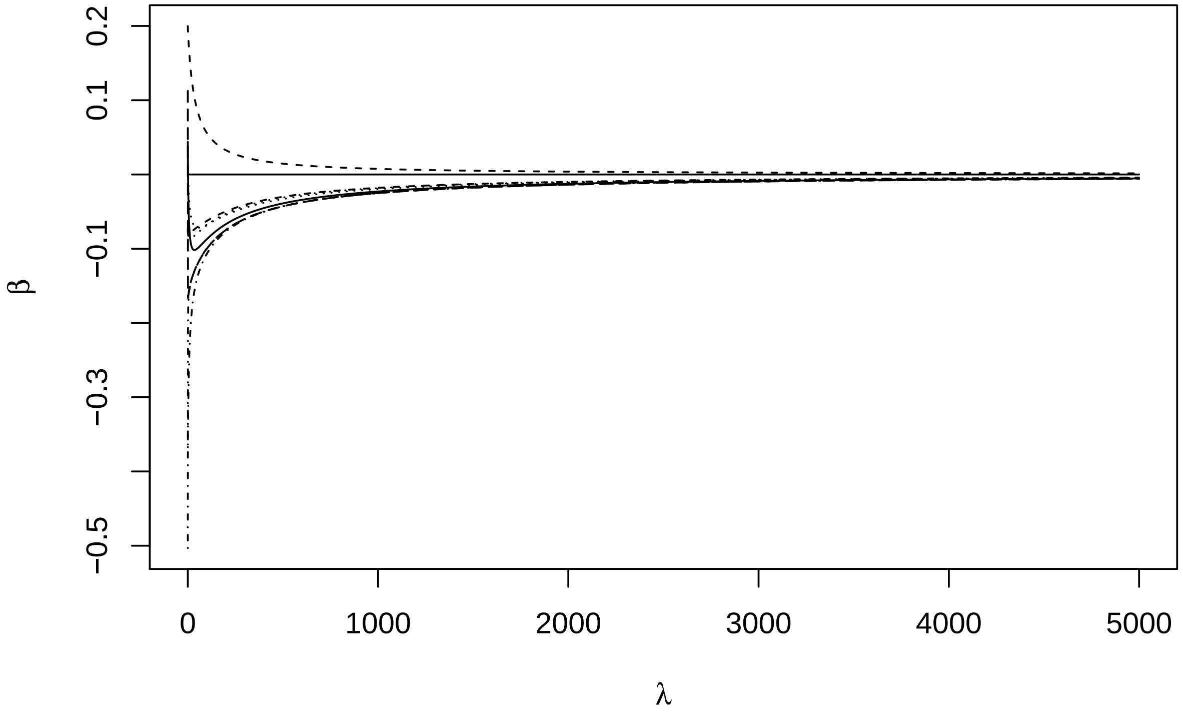
Table 9: PLS predicted hipcenter

hipcenter.3.comps
-185.8

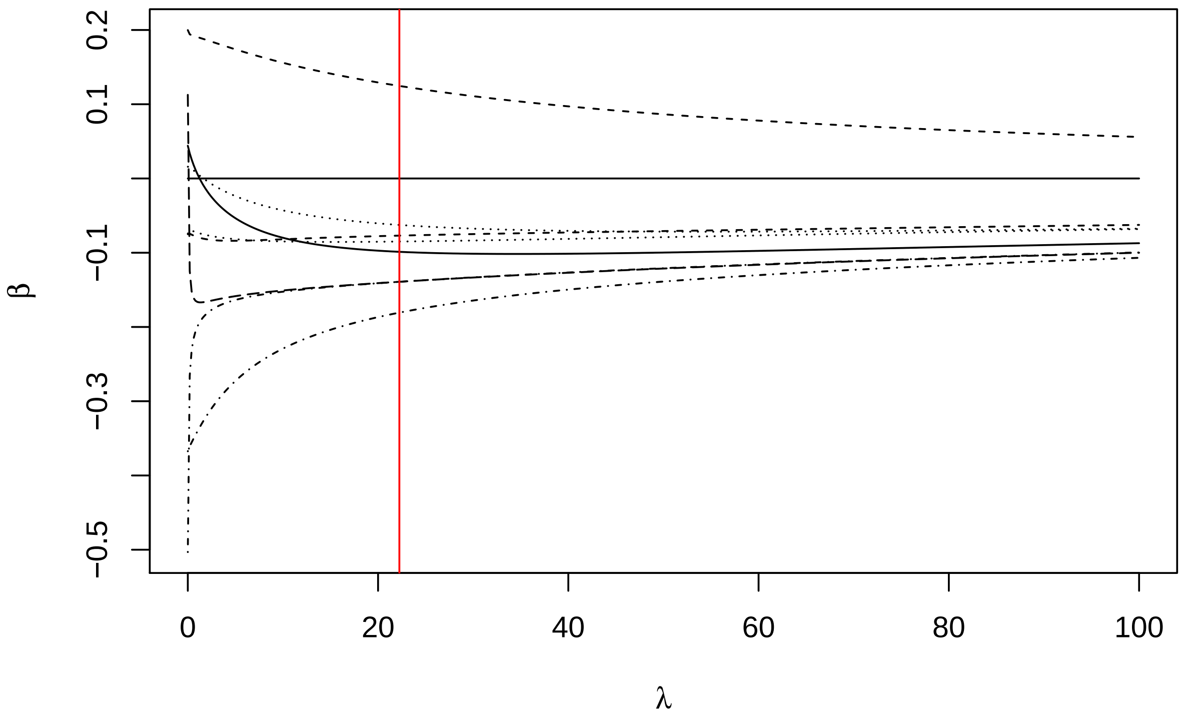
11.3 Ridge regression with seatpos data

Fit a ridge regression model to the seatpos data with hipcenter as the response and all other variables as predictors. Take care to select an appropriate amount of shrinkage. Use the model to predict the response at the values of the predictors specified in the first question.

First we make a few plots to see what the range of λ should be.



Now we fit 500 models in the range $\lambda \in (0, 100)$, find the minimum error model via cross validation, and plot the location of the λ that minimizes the cross validation error on the coefficient plot.



Here we predict the response for the ridge model with predictor values

$HtShoes = 181.080$, $Ht = 178.560$, $Seated = 91.440$, $Arm = 35.640$ $Thigh = 40.950;$, $Leg = 38.7$, $Age =$

We scaled the data before fitting the ridge model. We display the code below for applying the scaling to the predictors, predicting the fit from the optimal model determined by cross validation, and then undoing the scaling on the predicted response.

```
DFTest <- data.frame( HtShoes=181.080, Ht=178.560, Seated=91.440, Arm=35.640, Thigh=40.9

mean.pred <- c(mean.df.full["HtShoes"], mean.df.full["Ht"], mean.df.full["Seated"], mean

sd.pred <- c(sd.df.full["HtShoes"], sd.df.full["Ht"], sd.df.full["Seated"], sd.df.full["

x <- as.matrix(DFTest)

x <- (x-mean.pred) / sd.pred

ypred <- cbind(1,as.matrix(x)) %*% coef(ridge.fit)[112,]
```

```
pred.manual.comp <- ypred*sd(seatpos$hipcenter) +mean(seatpos$hipcenter)

pander(data.frame(pred.manual.comp=pred.manual.comp), caption = "ridge Regression predic
```

Table 10: ridge Regression predicted hipcenter

pred.manual.comp
-223.3

11.4 fat Analysis

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 a test sample. Use the remaining data as a training sample building the following models: Use the models you find to predict the response in the test sample. Make a report on the performances of the models.

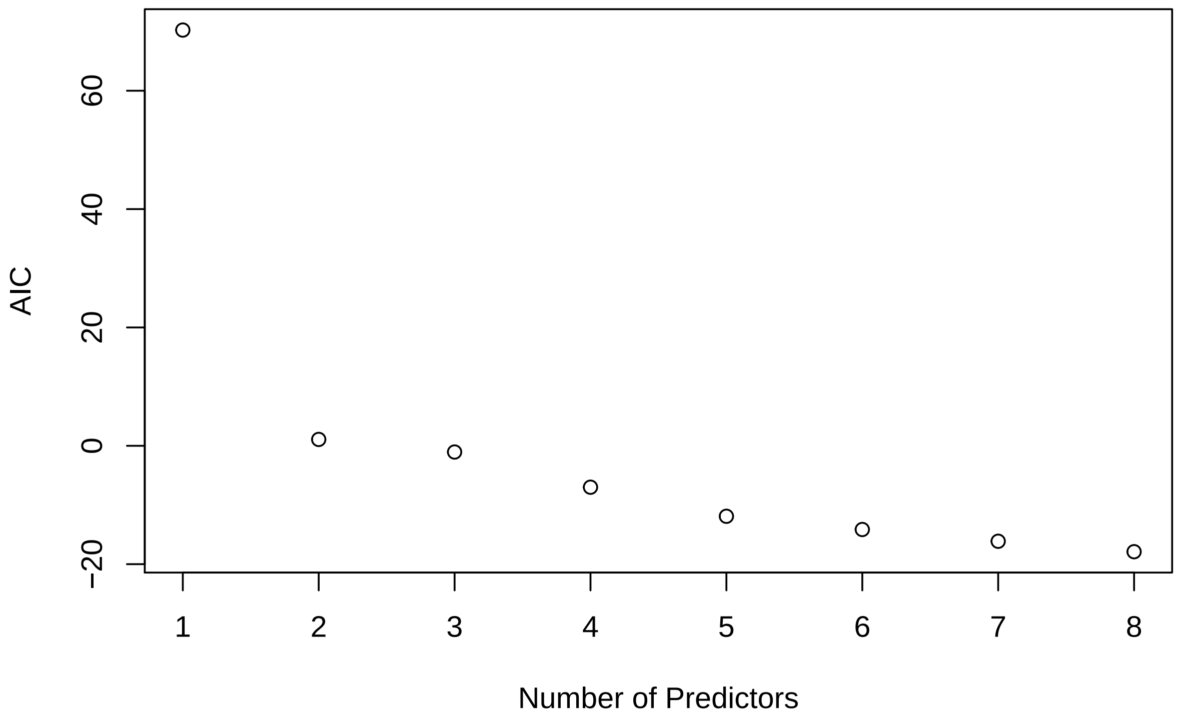
(a) Linear regression with all predictors

```
##
## Call:
## lm(formula = siri ~ ., data = df.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.91300 -0.33943  0.06558  0.28091  0.74021
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -150.28386   53.69480  -2.799   0.0208 *
## age           -0.01585    0.03507  -0.452   0.6619
## weight         0.03936    0.16746   0.235   0.8194
## height         2.16312    0.72677   2.976   0.0155 *
## adipos         2.21152    0.95912   2.306   0.0466 *
## free          -0.56551    0.04668 -12.115  7.1e-07 ***
## neck          -0.09268    0.17480  -0.530   0.6088
## chest          0.14160    0.11043   1.282   0.2318
## abdom          0.10218    0.07521   1.359   0.2074
## hip           -0.03646    0.13089  -0.279   0.7869
## thigh         -0.01546    0.13606  -0.114   0.9120
## knee          -0.14962    0.25385  -0.589   0.5701
## ankle          0.10486    0.37143   0.282   0.7841
## biceps         0.47382    0.16961   2.794   0.0209 *
## forearm       -0.17431    0.24492  -0.712   0.4947
## wrist          0.75940    0.51383   1.478   0.1736
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7632 on 9 degrees of freedom
## Multiple R-squared:  0.9953, Adjusted R-squared:  0.9876
## F-statistic: 128.2 on 15 and 9 DF,  p-value: 1.278e-08
```

(b) Linear regression with variables selected using AIC

We plot the AIC for the models here and compare to the exhaustive search for reference.

```
##   (Intercept)   age weight height adipos  free  neck chest abdom   hip
## 1        TRUE FALSE  FALSE  FALSE   TRUE FALSE FALSE FALSE FALSE FALSE
## 2        TRUE FALSE   TRUE  FALSE  FALSE  TRUE FALSE FALSE FALSE FALSE
## 3        TRUE FALSE   TRUE  FALSE  FALSE  TRUE FALSE FALSE FALSE FALSE
## 4        TRUE FALSE  FALSE   TRUE   TRUE  TRUE FALSE FALSE FALSE FALSE
## 5        TRUE FALSE  FALSE   TRUE   TRUE  TRUE FALSE FALSE  TRUE FALSE
## 6        TRUE FALSE  FALSE   TRUE   TRUE  TRUE FALSE  TRUE  TRUE FALSE
## 7        TRUE FALSE  FALSE   TRUE   TRUE  TRUE FALSE  TRUE FALSE FALSE
## 8        TRUE FALSE  FALSE   TRUE   TRUE  TRUE FALSE  TRUE  TRUE FALSE
##   thigh  knee ankle biceps forearm wrist
## 1 FALSE FALSE FALSE  FALSE   FALSE FALSE
## 2 FALSE FALSE FALSE  FALSE   FALSE FALSE
## 3 FALSE FALSE FALSE  FALSE   FALSE  TRUE
## 4 FALSE FALSE FALSE   TRUE   FALSE FALSE
## 5 FALSE FALSE FALSE   TRUE   FALSE FALSE
## 6 FALSE FALSE FALSE   TRUE   FALSE FALSE
## 7 FALSE FALSE FALSE   TRUE    TRUE  TRUE
## 8 FALSE FALSE FALSE   TRUE    TRUE  TRUE
```



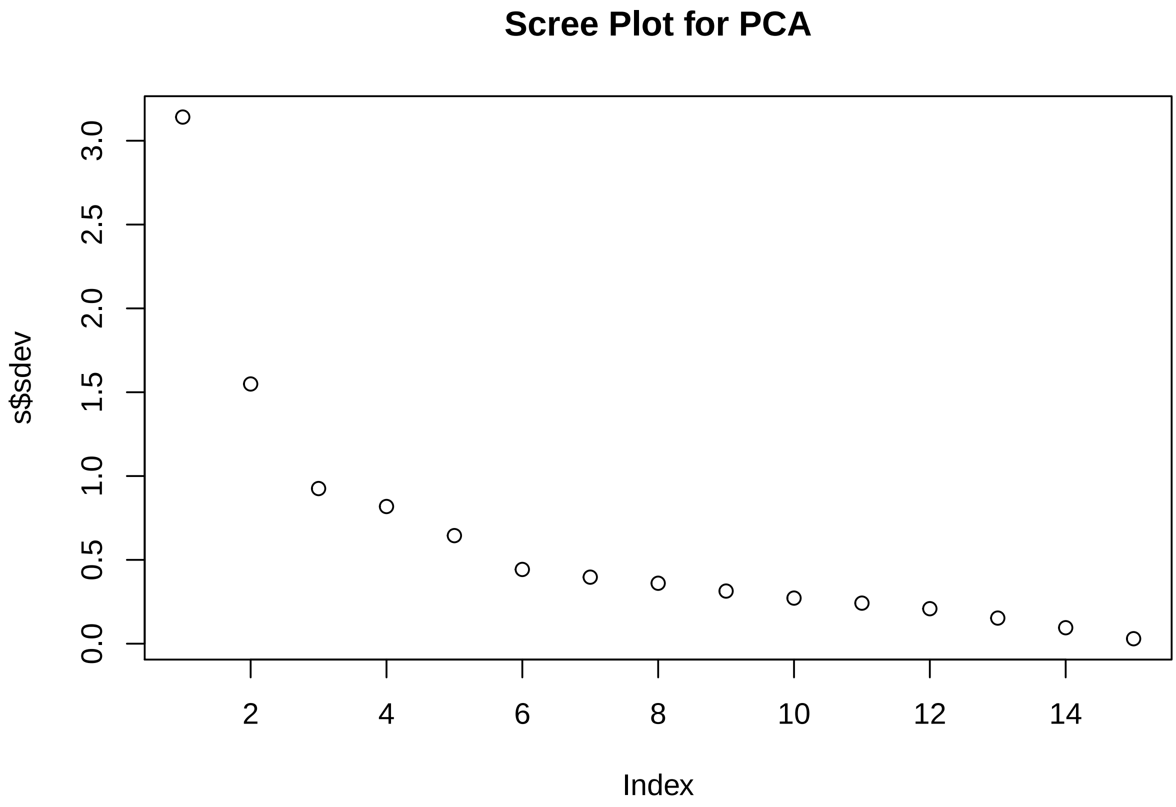
We see that the model with the lowest AIC has 8 predictors. The best 8 predictor model being $height + adipos + free + chest + abdom + biceps + forearm + wrist$

Now we fit the model and calculate the MSPE

```
##
## Call:
## lm(formula = siri ~ height + adipos + free + chest + abdom +
##     biceps + forearm + wrist, data = df.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00148 -0.44689 -0.06075  0.46347  0.82056
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -164.89479    9.41916 -17.506 7.38e-12 ***
## height         2.30717    0.17517  13.171 5.28e-10 ***
## adipos         2.41012    0.25311   9.522 5.41e-08 ***
## free          -0.56865    0.02602 -21.855 2.43e-13 ***
## chest          0.15903    0.07381   2.155 0.046771 *
## abdom          0.05946    0.03686   1.613 0.126249
## biceps         0.50955    0.10308   4.943 0.000147 ***
## forearm       -0.26744    0.16192  -1.652 0.118090    
## wrist          0.67669    0.28418   2.381 0.030019 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.6097 on 16 degrees of freedom
## Multiple R-squared:  0.9947, Adjusted R-squared:  0.9921 
## F-statistic: 376.3 on 8 and 16 DF,  p-value: < 2.2e-16
```

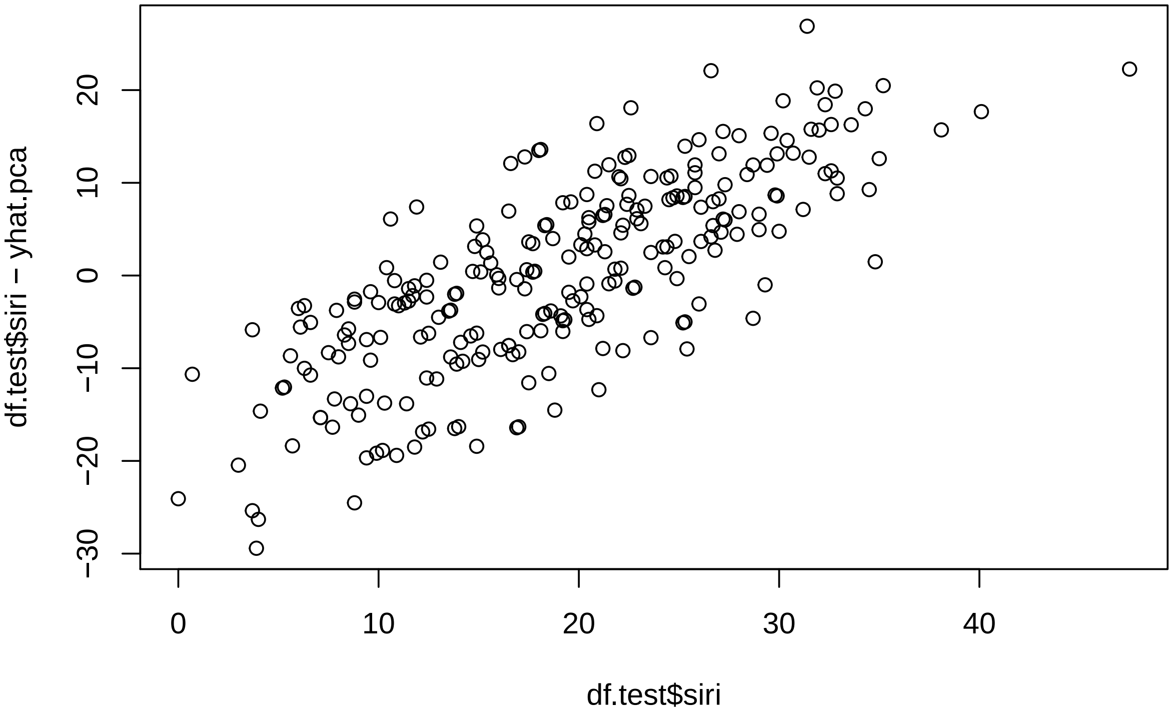
(c) Principal component regression

```
## Importance of components:
##                           PC1    PC2     PC3     PC4    PC5     PC6    PC7
## Standard deviation     3.1413 1.5491 0.92537 0.81824 0.6446 0.44277 0.3969
## Proportion of Variance 0.6578 0.1600 0.05709 0.04463 0.0277 0.01307 0.0105
## Cumulative Proportion  0.6578 0.8178 0.87491 0.91954 0.9473 0.96032 0.9708
##                            PC8     PC9    PC10   PC11   PC12    PC13
## Standard deviation     0.36048 0.31381 0.27192 0.2419 0.2087 0.15269
## Proportion of Variance 0.00866 0.00657 0.00493 0.0039 0.0029 0.00155
## Cumulative Proportion  0.97948 0.98605 0.99098 0.9949 0.9978 0.99934
##                           PC14    PC15
## Standard deviation     0.09532 0.02965
## Proportion of Variance 0.00061 0.00006
## Cumulative Proportion  0.99994 1.00000
```



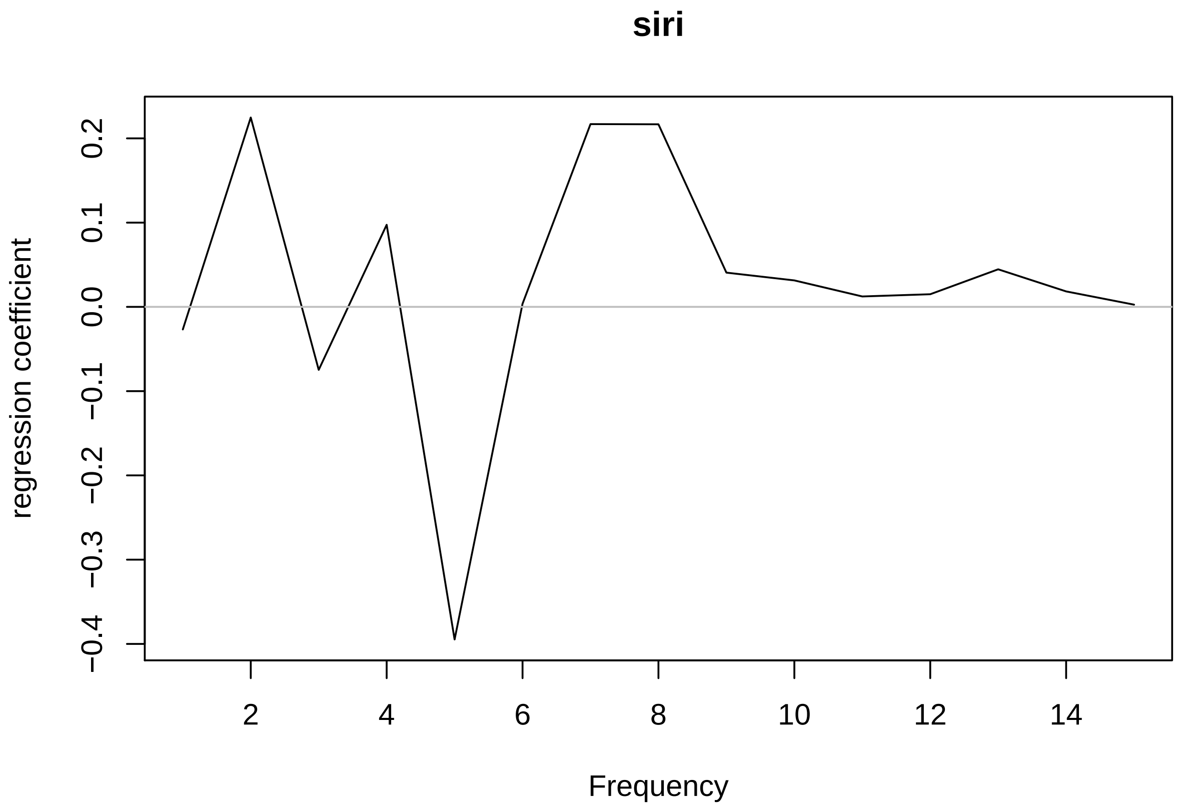
Based on the scree plot we choose 5 principal components for our model.

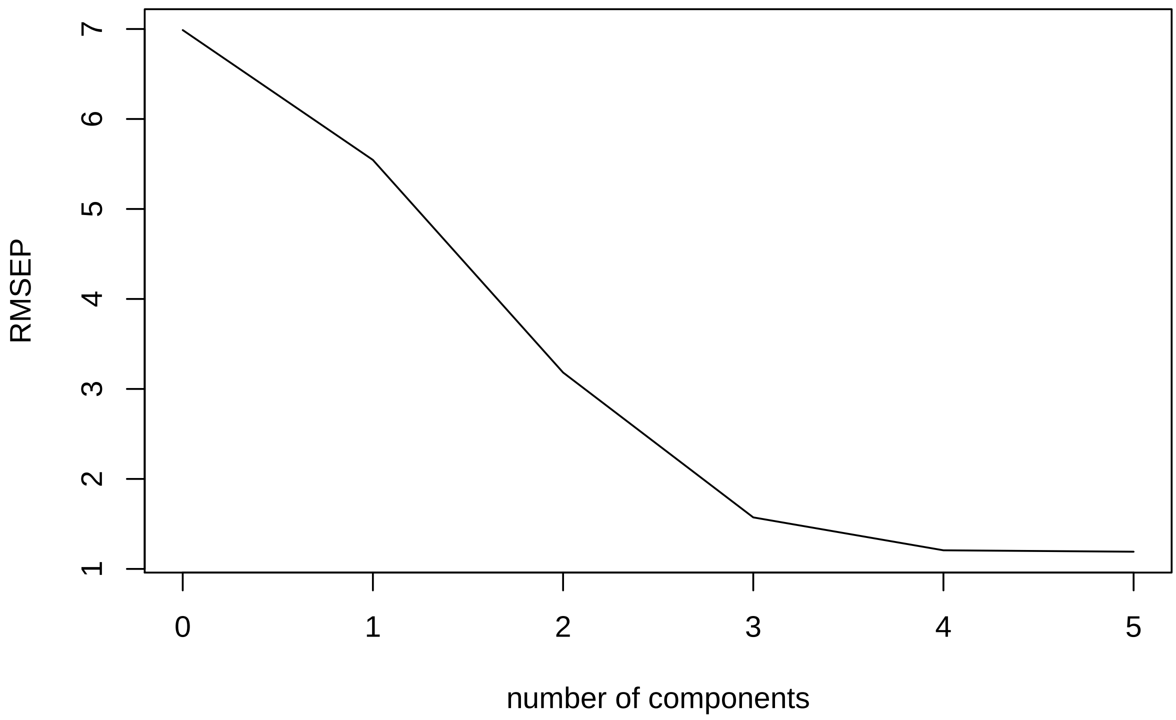
```
##
## Call:
## lm(formula = df.train$siri ~ pca.results$x[, 1:15])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.91300 -0.33943  0.06558  0.28091  0.74021
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                18.88400    0.15263 123.724 7.48e-16 ***
## pca.results$x[, 1:15]PC1  -1.09230    0.04959 -22.026 3.87e-09 ***
## pca.results$x[, 1:15]PC2  -3.20817    0.10056 -31.903 1.43e-10 ***
## pca.results$x[, 1:15]PC3  -0.82145    0.16834  -4.880 0.000872 ***
## pca.results$x[, 1:15]PC4   1.00858    0.19038   5.298 0.000495 ***
## pca.results$x[, 1:15]PC5  -2.88314    0.24166 -11.931 8.09e-07 ***
## pca.results$x[, 1:15]PC6   1.44809    0.35182   4.116 0.002614 **
## pca.results$x[, 1:15]PC7   0.32943    0.39249   0.839 0.423018
## pca.results$x[, 1:15]PC8   0.92046    0.43213   2.130 0.062017 .
## pca.results$x[, 1:15]PC9   0.21670    0.49640   0.437 0.672725
## pca.results$x[, 1:15]PC10 -5.95196    0.57288 -10.390 2.60e-06 ***
## pca.results$x[, 1:15]PC11  3.50815    0.64389   5.448 0.000407 ***
## pca.results$x[, 1:15]PC12 -2.72292    0.74635  -3.648 0.005331 **
## pca.results$x[, 1:15]PC13 -5.80915    1.02024  -5.694 0.000297 ***
## pca.results$x[, 1:15]PC14 -7.18207    1.63433  -4.394 0.001734 **
## pca.results$x[, 1:15]PC15  4.67824    5.25399   0.890 0.396420
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7632 on 9 degrees of freedom
## Multiple R-squared:  0.9953, Adjusted R-squared:  0.9876
## F-statistic: 128.2 on 15 and 9 DF,  p-value: 1.278e-08
```



Our MSPE is high and the residuals are showing a strong linear association with the response - even if we include all components as we did above. This is definitely a problem with the code we've written. We'll revisit this if there is time.

(d) Partial least squares

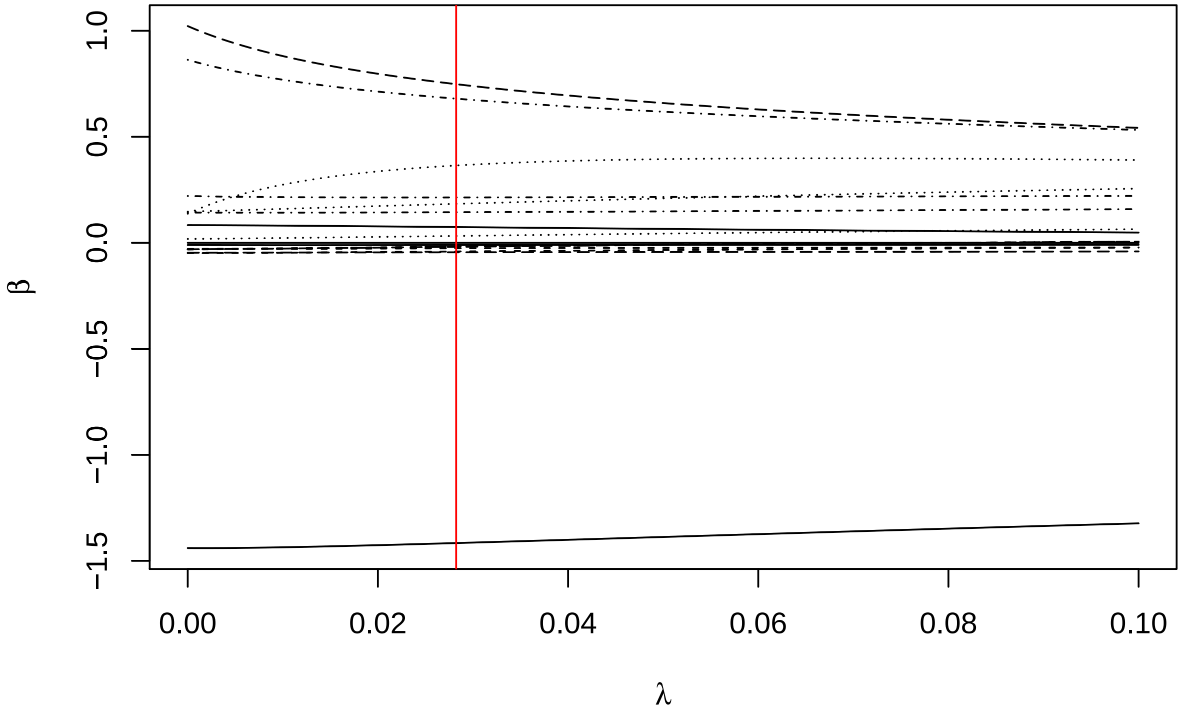




Now we find the MSPE for the test set.

(e) Ridge regression

```
## 0.0282282282
##          283
```



Here we predict the test set response for the ridge model. We scaled the data before fitting the ridge model. We display the code below for applying the scaling to the predictors, predicting the fit from the optimal model determined by cross validation, and then undoing the scaling on the predicted response.

```
df <- df.test[ , -which(names(df.test) %in% c("siri"))]
x <- as.matrix(df)


df.train.predict <- df.train[ , -which(names(df.train) %in% c("siri"))]
mean.pred.train <-apply(df.train.predict,2,mean  )
sd.pred.train <-apply(df.train.predict,2,sd)

x<- scale(x,center = mean.pred.train,scale = sd.pred.train)

yhat.ridge.scaled <- cbind(1,as.matrix(x)) %*% coef(ridge.fit)[283,]

yhat.ridge <- yhat.ridge.scaled* sd(df.train$siri) +mean(df.train$siri)

mspe.lm.ridge <- mean((df.test$siri - yhat.ridge) ^ 2)
```

Performance Results

Table 11: MSPE Results

mspe.lm	mspe.lm.regsubsets	mspe.lm.pca	mspe.lm.pls	mspe.lm.ridge
34.37	38.08	110	12.15	22.66

The PLS and ridge models performed the best. We are concerned about the PCA results and will debug this further. We did not get good results with the PCA using 3 components, when we allowed all components we did not see a reduction in the MSPE to the full linear model. This is why we're concerned. Using the full set of PCA components is just a rotation of the data, so we'd expect similar MSPE results.