Problem Set 3

API222: Big Data & Machine Learning

Brendan Hellweg

10/21/2021

Problem 1

Problem 1A

PLS: Yes

PCR: Yes

Lasso: Yes

Ridge: Yes

Linear Regression: No

Logistic Regression: No

KNN: No

Problem 1B

The purpose of scaling when performing shrinkage methods is to ensure that variables are removed (as in Lasso) or reduced (as in Ridge) depending on their relevance to the model rather than the manner in which data is collected/reported. If, for instance, salaries and age are both in a model, a shrinkage model might otherwise erroneously remove the age because the standard range is somewhere from `30 < x < 70` while the standard range for salaries in a dataset might be `60,000 < x < 90,000`. Shrinkage scales each variable by their variance such that no variable is erroneously removed from the model or excessively reduced due to its manner of reporting.

Problem 1C

The purpose of scaling when performing dimension reduction is because we do not want the variance or values of any particular predictor to exert excess influence on the novel predictors based on how it is reported in the dataset. Dimension reduction produces new components from multiple predictors, so we would not want one to be overweighted or underweighted by how it is reported. It is an arbitrary decision that we define binary variables, for instance, as zero or one. We could just as easily define them as zero and one million. In the latter case, their variance would be excessively influential in the dimension reduction model, and that single variable would overwhelm the other factors. This scaling is important for dimension reduction in particular because it uses the variance of individual predictors as part of the dimension reduction process, and undue influence from one predictor could throw off the whole model.

Problem 2

No, for PCR, the first stage is unsupervised, meaning that values of `Y` are not necessarily related to the model design. Instead, the first principle component is found to maximize variance so that it summarizes all variance among `X`'s. In the second stage, PCR finds weights for `Z2` such that it is independent of `Z1`, so each offers new information.

Problem 3

Yes, for the first factor loading of this PLS Regression, we would expect PLS to weigh `X2` more highly. The reason for this is that PLS puts more weight on predictors most related to `Y` in the first factor stage. The second component will weigh `X1` more highly.

Problem 4

True, although this is not uniformly the case. Because PLS is a supervised model, it will tend to have lower bias compared to PCR. PCR tends to have lower variance and higher bias. The bias/variance trade-off strikes again.

Problem 5

Ridge and Lasso will outperform PCR when there are fewer dimensions and a greater sample size, while dimension reduction models like PCR usually perform better in high dimensional datasets, especially when `n<<p`. This is because RSS breaks down when the number of variables exceeds the number of observations without dimension reduction.

Problem 6

Problem 6a

In this case, the data we're working with is quasi-categorical (that is, we know that someone who graduates 12th grade is in a different category of educational attainment as someone who does not), so we expect that there will be a real change in the trendline at that point. As a result, we would use a step function to create a distinct cutoff at this threshold and then use the separate bins to measure changes in outcome and educational attainment without influence from trends in other bins.

Problem 6b

For regression splines, we need degrees of freedom to equal the number of cut points `K` and the number of parameters. So, for a cubic spline, there would be four parameters plus the number of cut points in the dataset.

Problem 6c

If the smoothing parameter `lambda = 0`, the function will not smooth at all and it will be very rough. When `lambda` approaches infinity, the function will become increasingly smooth until it is the smoothest shape of all, a straight line.

Problem 6d

Natural splines deal with the tricky problem of what to do at the edges of the data range, which is typically more sparse than the middle parts. As a result, polynomial regressions often have a much higher error range and take on weird shapes at the edges because there is less guidance on where to go. For a natural spline, the edges of the range are linear, so it does a better job of approximating the trend-line.

Data Questions

Problem 1

```
c("There are ", ncol(star)-1," predictors.") %>%
str_c(.,collapse = '')
```

```
## [1] "There are 39 predictors."
```

```
c("There are ", nrow(star)-nrow(star %>% drop_na()),
  " observations with missing values, totaling ",
  sum(is.na(star)),
      " missing values.") %>%
str_c(.,collapse = '')
```

```
## [1] "There are 9230 observations with missing values, totaling 189871 missing values."
```

```
star %<>% na.omit()
star[33:40] <- lapply(star[33:40],as.numeric)

c("There are ", ncol(select_if(star,is.factor)),"categorical variables (before we create dummies).") %
str_c(.,collapse = '')
```

```
## [1] "There are 26 categorical variables (before we create dummies)."
```

```
star%<>%dummy_cols(remove_first_dummy = F)%<>%select_if(.,is.numeric)

c("There are ", ncol(select_if(star,function(col) {var(col)<0.05})),"variables with variance less than
str_c(.,collapse = '')
```

```
## [1] "There are 22 variables with variance less than 0.05."
```

```
lowvar.rm <- function(df){
  df[, sapply(df, var) > 0.05]
}
star%<>%lowvar.rm()
```

Problem 2

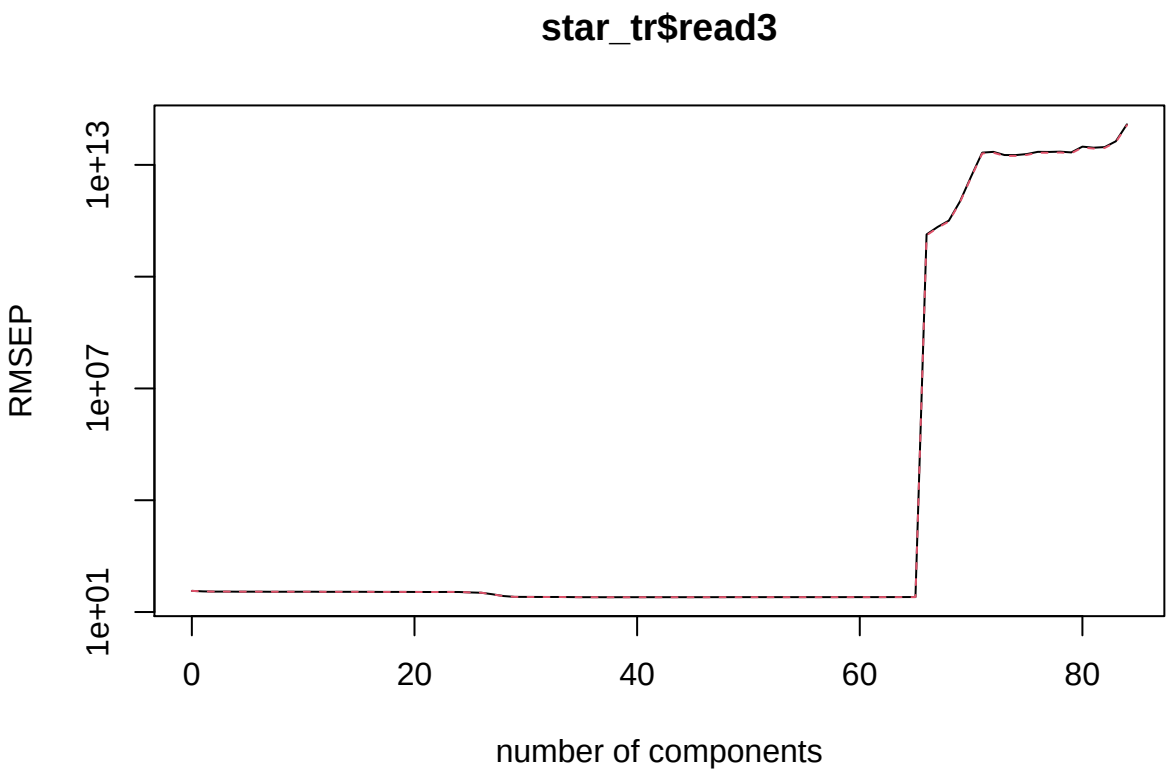
```
pcr_model <- pcr(star_tr$read3~.,data = star_tr,scale = T,validation = "CV")
summary(pcr_model)
```

```
## Data:    X dimension: 1894 84
##  Y dimension: 1894 1
## Fit method: svdpc
## Number of components considered: 84
##
```

```
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           36.89    35.81     35.3    35.30    35.13    35.13    35.16
## adjCV        36.89    35.81     35.3    35.29    35.12    35.13    35.15
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       34.98    34.99    35.02     35.03     35.01     34.81     34.84
## adjCV    34.98    34.98    35.02     35.02     35.00     34.79     34.82
##        14 comps  15 comps  16 comps  17 comps  18 comps  19 comps  20 comps
## CV        34.83     34.86     34.79     34.62     34.61     34.64     34.57
## adjCV     34.81     34.85     34.78     34.57     34.60     34.63     34.57
##        21 comps  22 comps  23 comps  24 comps  25 comps  26 comps  27 comps
## CV        34.43     34.43     34.58     34.45     33.48     32.92     29.96
## adjCV     34.42     34.42     34.60     34.55     33.53     32.96     29.81
##        28 comps  29 comps  30 comps  31 comps  32 comps  33 comps  34 comps
## CV        26.82     25.50     25.44     25.36     25.33     25.30     25.14
## adjCV     26.74     25.46     25.40     25.33     25.29     25.29     25.11
##        35 comps  36 comps  37 comps  38 comps  39 comps  40 comps  41 comps
## CV        25.00     25.00     24.95     24.95     24.98     24.98     24.98
## adjCV     24.97     24.98     24.92     24.93     24.95     24.95     24.95
##        42 comps  43 comps  44 comps  45 comps  46 comps  47 comps  48 comps
## CV        24.98     24.99     25.00     24.98     25.00     25.03     25.05
## adjCV     24.95     24.96     24.97     24.95     24.97     24.99     25.01
##        49 comps  50 comps  51 comps  52 comps  53 comps  54 comps  55 comps
## CV        25.05     25.06     25.06     25.07     25.07     25.07     25.07
## adjCV     25.02     25.02     25.03     25.03     25.03     25.03     25.03
##        56 comps  57 comps  58 comps  59 comps  60 comps  61 comps  62 comps
## CV        25.07     25.07     25.08     25.09     25.09     25.10     25.11
## adjCV     25.04     25.03     25.04     25.05     25.05     25.06     25.07
##        63 comps  64 comps  65 comps   66 comps   67 comps   68 comps   69 comps
## CV        25.17     25.18     25.22  1.358e+11  2.166e+11  3.188e+11  1.042e+12
## adjCV     25.12     25.13     25.17  1.288e+11  2.055e+11  3.025e+11  9.882e+11
##         70 comps   71 comps   72 comps   73 comps   74 comps  75 comps
## CV     4.869e+12  2.129e+13  2.236e+13  1.837e+13  1.832e+13  1.95e+13
## adjCV  4.619e+12  2.020e+13  2.121e+13  1.742e+13  1.738e+13  1.85e+13
##         76 comps   77 comps   78 comps   79 comps   80 comps   81 comps
## CV     2.238e+13  2.223e+13  2.264e+13  2.157e+13  3.089e+13  2.889e+13
## adjCV  2.123e+13  2.109e+13  2.148e+13  2.047e+13  2.930e+13  2.741e+13
##         82 comps   83 comps   84 comps
## CV     3.002e+13  4.292e+13  1.223e+14
## adjCV  2.848e+13  4.072e+13  1.161e+14
##
## TRAINING: % variance explained
##                 1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X                18.146   27.325   34.228   40.545   45.902   50.455    54.34
## star_tr$read3     5.919    8.667    8.795    9.797    9.837    9.841    10.86
##                 8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X                 57.34    60.03     62.69     65.16     67.53     69.76
## star_tr$read3     10.88    10.88     10.91     11.17     12.21     12.31
##                 14 comps  15 comps  16 comps  17 comps  18 comps  19 comps
## X                  71.87     73.78     75.58     77.27     78.91     80.49
## star_tr$read3      12.47     12.50     12.87     13.89     14.04     14.07
##                 20 comps  21 comps  22 comps  23 comps  24 comps  25 comps
## X                  81.96     83.40     84.77     86.07     87.27     88.44
```

```
## star_tr$read3     14.44     15.67     15.87     15.88     16.03     21.04
##                 26 comps  27 comps  28 comps  29 comps  30 comps  31 comps
## X                  89.58     90.63     91.64     92.57     93.36     94.09
## star_tr$read3      24.22     38.38     49.39     53.94     54.21     54.56
##                 32 comps  33 comps  34 comps  35 comps  36 comps  37 comps
## X                  94.76      95.4     96.02     96.56     97.05     97.52
## star_tr$read3      54.79      54.8     55.41     55.92     55.92     56.10
##                 38 comps  39 comps  40 comps  41 comps  42 comps  43 comps
## X                  97.94     98.30     98.64     98.88     99.05     99.18
## star_tr$read3      56.12     56.12     56.14     56.15     56.19     56.19
##                 44 comps  45 comps  46 comps  47 comps  48 comps  49 comps
## X                  99.30     99.40     99.49     99.59     99.67     99.73
## star_tr$read3      56.19     56.22     56.25     56.25     56.26     56.31
##                 50 comps  51 comps  52 comps  53 comps  54 comps  55 comps
## X                  99.78     99.82     99.85     99.88     99.91     99.93
## star_tr$read3      56.32     56.32     56.34     56.38     56.41     56.41
##                 56 comps  57 comps  58 comps  59 comps  60 comps  61 comps
## X                  99.94     99.96     99.97     99.98     99.99     99.99
## star_tr$read3      56.41     56.44     56.46     56.46     56.46     56.46
##                 62 comps  63 comps  64 comps  65 comps  66 comps  67 comps
## X                  99.99    100.00    100.00    100.00    100.00    100.00
## star_tr$read3      56.46     56.48     56.51     56.52     56.52     56.53
##                 68 comps  69 comps  70 comps  71 comps  72 comps  73 comps
## X                 100.00    100.00     100.0    100.00    100.00     100.0
## star_tr$read3      56.54     56.59      56.6     56.67     56.69      56.7
##                 74 comps  75 comps  76 comps  77 comps  78 comps  79 comps
## X                 100.00    100.00    100.00    100.00    100.00    100.00
## star_tr$read3      56.72     56.73     56.74     56.74     56.75     56.75
##                 80 comps  81 comps  82 comps  83 comps  84 comps
## X                 100.00    100.00    100.00     100.0     100.0
## star_tr$read3      56.86     56.86     56.89      56.9      56.9
```

```
validationplot(pcr_model,val.type = "RMSE", log = "y")
```



a) RMSE at 0, 10, and 20 components is 36.89, 35.03, and 34.57, respectively.

b) The improvement from 0 to 10 components is relatively small, representing a bit more than a 5% improvement. However, since this is RMSE, the MSE improvement from 0 to 10 components is significantly larger.

c) The improvement from 10 to 20 components is even smaller, representing about a 1.5% improvement.

Problem 3

```
pcr_test <- predict(pcr_model,star_te,ncomp = 40)
pcr_rmse <- round(sqrt(mean((pcr_test - star_te$read3)^2)),2)
pcr_rmse
```

```
## [1] 25.84
```

a) At roughly 37-38 components we see the lowest RMSE value.

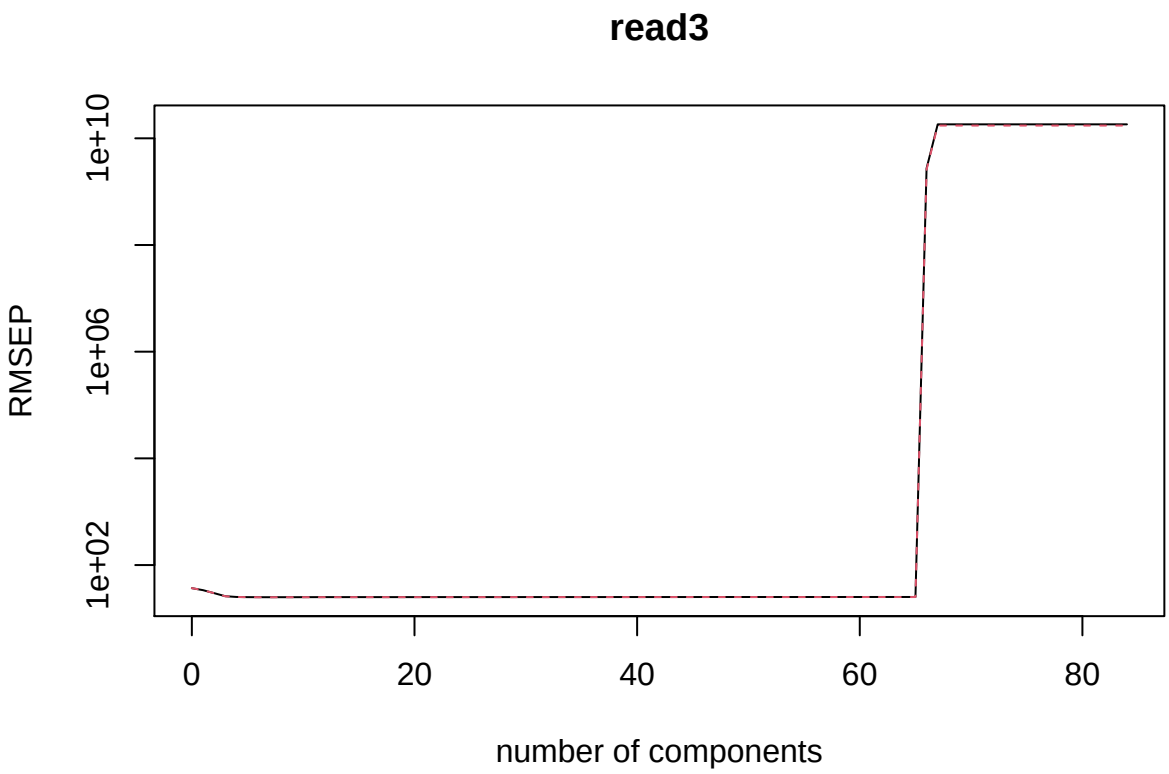
b) The lowest RMSE is 24.95.

c) The test RMSE is 25.84. Not too far off!

Problem 4

```
## Data:    X dimension: 1894 84
##  Y dimension: 1894 1
## Fit method: kernelpls
## Number of components considered: 84
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           36.89    33.87    29.76    26.07    25.25    25.03    24.97
## adjCV        36.89    33.86    29.73    26.00    25.21    25.00    24.94
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       24.96    24.97    24.99     25.00     25.03     25.06     25.05
## adjCV    24.93    24.94    24.96     24.97     25.00     25.02     25.02
##        14 comps  15 comps  16 comps  17 comps  18 comps  19 comps  20 comps
## CV        25.07     25.06     25.07     25.06     25.06     25.07     25.08
## adjCV     25.03     25.03     25.03     25.03     25.03     25.03     25.04
##        21 comps  22 comps  23 comps  24 comps  25 comps  26 comps  27 comps
## CV        25.09     25.08     25.09     25.09     25.09     25.09     25.10
## adjCV     25.05     25.04     25.05     25.05     25.05     25.05     25.06
##        28 comps  29 comps  30 comps  31 comps  32 comps  33 comps  34 comps
## CV        25.10     25.11     25.11     25.12     25.11     25.12     25.13
## adjCV     25.06     25.06     25.07     25.08     25.07     25.08     25.08
##        35 comps  36 comps  37 comps  38 comps  39 comps  40 comps  41 comps
## CV        25.13     25.13     25.14     25.14     25.15     25.15     25.16
## adjCV     25.09     25.09     25.09     25.10     25.10     25.11     25.11
##        42 comps  43 comps  44 comps  45 comps  46 comps  47 comps  48 comps
## CV        25.16     25.16     25.16     25.17     25.17     25.17     25.18
## adjCV     25.11     25.11     25.11     25.12     25.12     25.12     25.13
##        49 comps  50 comps  51 comps  52 comps  53 comps  54 comps  55 comps
## CV        25.17     25.18     25.18     25.18     25.17     25.18     25.18
## adjCV     25.12     25.13     25.13     25.13     25.13     25.13     25.13
##        56 comps  57 comps  58 comps  59 comps  60 comps  61 comps  62 comps
## CV        25.18     25.19     25.20     25.20     25.20     25.20     25.20
## adjCV     25.13     25.14     25.15     25.15     25.15     25.15     25.15
##        63 comps  64 comps  65 comps   66 comps   67 comps   68 comps   69 comps
## CV        25.21     25.21     25.21  2.734e+09  1.827e+10  1.827e+10  1.827e+10
## adjCV     25.16     25.16     25.16  2.594e+09  1.733e+10  1.733e+10  1.733e+10
##         70 comps   71 comps   72 comps   73 comps   74 comps   75 comps
## CV     1.827e+10  1.827e+10  1.827e+10  1.827e+10  1.827e+10  1.827e+10
## adjCV  1.733e+10  1.733e+10  1.733e+10  1.733e+10  1.733e+10  1.733e+10
##         76 comps   77 comps   78 comps   79 comps   80 comps   81 comps
## CV     1.827e+10  1.827e+10  1.827e+10  1.827e+10  1.827e+10  1.827e+10
## adjCV  1.733e+10  1.733e+10  1.733e+10  1.733e+10  1.733e+10  1.733e+10
##         82 comps   83 comps   84 comps
## CV     1.827e+10  1.827e+10  1.827e+10
## adjCV  1.733e+10  1.733e+10  1.733e+10
##
## TRAINING: % variance explained
##        1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X        16.47    23.43    28.51    33.72    37.74    40.78    45.40    48.58
## read3    16.41    37.15    52.78    55.28    55.92    56.12    56.19    56.24
##        9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
```

```
## X         51.88     55.13     56.67     59.45     61.90     63.97     65.51
## read3     56.27     56.29     56.33     56.36     56.38     56.39     56.41
##         16 comps  17 comps  18 comps  19 comps  20 comps  21 comps  22 comps
## X         66.98     68.39     69.74     71.15     72.45     73.87     74.97
## read3     56.42     56.43     56.44     56.45     56.45     56.46     56.46
##         23 comps  24 comps  25 comps  26 comps  27 comps  28 comps  29 comps
## X         76.25     77.41     79.14     80.35     81.95     82.90     84.00
## read3     56.46     56.47     56.47     56.47     56.47     56.48     56.48
##         30 comps  31 comps  32 comps  33 comps  34 comps  35 comps  36 comps
## X         85.08     86.12     86.98     87.87     88.79     89.42     90.38
## read3     56.48     56.48     56.48     56.48     56.49     56.49     56.49
##         37 comps  38 comps  39 comps  40 comps  41 comps  42 comps  43 comps
## X         91.40     92.14     92.83      93.7     94.56     94.98     95.68
## read3     56.49     56.49     56.50      56.5     56.50     56.51     56.51
##         44 comps  45 comps  46 comps  47 comps  48 comps  49 comps  50 comps
## X         96.54     96.86     97.90     98.30     98.58     98.98     99.42
## read3     56.51     56.51     56.52     56.52     56.52     56.52     56.52
##         51 comps  52 comps  53 comps  54 comps  55 comps  56 comps  57 comps
## X         99.53     99.60     99.66     99.73     99.79     99.82     99.90
## read3     56.52     56.52     56.52     56.52     56.52     56.52     56.52
##         58 comps  59 comps  60 comps  61 comps  62 comps  63 comps  64 comps
## X         99.91     99.93     99.95     99.96     99.97     99.98     99.99
## read3     56.52     56.52     56.53     56.53     56.53     56.53     56.53
##         65 comps  66 comps  67 comps  68 comps  69 comps  70 comps  71 comps
## X        100.00    100.00    100.00    100.01    100.02    100.03    100.04
## read3     56.53     56.53     56.53     56.53     56.53     56.53     56.53
##         72 comps  73 comps  74 comps  75 comps  76 comps  77 comps  78 comps
## X        100.05    100.05    100.06    100.07    100.08    100.09    100.10
## read3     56.53     56.53     56.53     56.53     56.53     56.53     56.53
##         79 comps  80 comps  81 comps  82 comps  83 comps  84 comps
## X        100.11    100.12    100.13    100.14    100.15    100.15
## read3     56.53     56.53     56.53     56.53     56.53     56.53
```




```
## [1] 25.99
```

a) 6 to 7 components produce the best CV RMSE, at `24.93`.

b) The corresponding RMSE is `25.04`.

c) The test RMSE is 25.99 at 16 components. Not too shabby!

Problem 5

```
pcr_rmse < pls_rmse
```

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

```
pcr_rmse
```

```
## [1] 25.84
```

```
pls_rmse
```

```
## [1] 25.99
```

If I were simply looking to produce the model with the lowest RMSE, I would choose the PCR model, as it has a lower RMSE than PLS. However, PLS allows me to use only 6 components at its best level, while the PCR model would require 40 to reach its best performance. If I wanted to use fewer components for computational or modeling reasons, I might sacrifice the marginal performance advantages for the simpler model.