

Lab6 Jin Kweon_3032235207

Jin Kweon

10/9/2017

```
hitters <- Hitters
str(Hitters, vec.len = 1)

## 'data.frame': 322 obs. of 20 variables:
## $ AtBat : int 293 315 ...
## $ Hits : int 66 81 ...
## $ HmRun : int 1 7 ...
## $ Runs : int 30 24 ...
## $ RBI : int 29 38 ...
## $ Walks : int 14 39 ...
## $ Years : int 1 14 ...
## $ CAtBat : int 293 3449 ...
## $ CHits : int 66 835 ...
## $ CHmRun : int 1 69 ...
## $ CRuns : int 30 321 ...
## $ CRBI : int 29 414 ...
## $ CWalks : int 14 375 ...
## $ League : Factor w/ 2 levels "A","N": 1 2 ...
## $ Division : Factor w/ 2 levels "E","W": 1 2 ...
## $ PutOuts : int 446 632 ...
## $ Assists : int 33 43 ...
## $ Errors : int 20 10 ...
## $ Salary : num NA 475 ...
## $ NewLeague: Factor w/ 2 levels "A","N": 1 2 ...

pcr_fit <- pcr(Salary ~., data = Hitters, scale = T, validation = "none")
names(pcr_fit)

## [1] "coefficients" "scores" "loadings" "Yloadings"
## [5] "projection" "Xmeans" "Ymeans" "fitted.values"
## [9] "residuals" "Xvar" "Xtotvar" "fit.time"
## [13] "na.action" "ncomp" "method" "scale"
## [17] "call" "terms" "model"

hittersnew <- na.omit(hitters)

model <- model.matrix(Salary ~., data = Hitters) #They automatically take out the NA.

X <- scale(model, T, T)
X <- X[,-1]
head(X)

## AtBat Hits HmRun Runs RBI
## -Alan Ashby -0.6017532 -0.5945419 -0.5275454 -1.2038163 -0.52106946
## -Alvin Davis 0.5115664 0.4913228 0.7285771 0.4406748 0.79254856
## -Andre Dawson 0.6269715 0.7350884 0.9569630 0.4015202 1.02436351
## -Andres Galarraga -0.5610220 -0.4615789 -0.1849665 -0.6164981 -0.36652617
## -Alfredo Griffin 1.2922478 1.3555825 -0.8701243 0.7539112 -0.01880375
## -Al Newman -1.4842626 -1.5696041 -1.2127031 -1.2429709 -1.68014419
```

	Walks	Years	CAtBat	CHits
## -Alan Ashby	-0.09734151	1.3952334	0.3461306	0.1740416
## -Alvin Davis	1.60631004	-0.8994853	-0.4520036	-0.4091121
## -Andre Dawson	-0.18943079	0.7694010	1.2990809	1.3156652
## -Andres Galarraga	-0.51174324	-1.1080961	-0.9890495	-0.9583256
## -Alfredo Griffin	-0.28152006	0.7694010	0.7655337	0.6337765
## -Al Newman	-0.92614497	-1.1080961	-1.0686443	-1.0493469

	CHmRun	CRuns	CRBI	CWalks
## -Alan Ashby	-0.002914243	-0.1214393	0.25847281	0.43450593
## -Alvin Davis	-0.075909091	-0.4143150	-0.19921055	0.01035326
## -Andre Dawson	1.894951816	1.4093644	1.56967378	0.35497730
## -Andres Galarraga	-0.696365303	-0.9457182	-0.87955068	-0.86067453
## -Alfredo Griffin	-0.611204646	0.4220413	0.01726131	-0.25095507
## -Al Newman	-0.830189192	-1.0000663	-0.99397151	-0.89475822

	LeagueN	DivisionW	PutOuts	Assists	Errors
## -Alan Ashby	1.0567429	0.9792988	1.21917406	-0.52219572	0.2129461
## -Alvin Davis	-0.9427059	0.9792988	2.10509535	-0.25337958	0.8184036
## -Andre Dawson	1.0567429	-1.0172561	-0.32404367	-0.74276281	-0.8466046
## -Andres Galarraga	1.0567429	-1.0172561	1.83717561	-0.54287389	-0.6952402
## -Alfredo Griffin	-0.9427059	0.9792988	-0.03111808	2.08325298	2.4834118
## -Al Newman	1.0567429	-1.0172561	-0.76700431	0.05679288	-0.2411471

	NewLeagueN
## -Alan Ashby	1.0730066
## -Alvin Davis	-0.9284171
## -Andre Dawson	1.0730066
## -Andres Galarraga	1.0730066
## -Alfredo Griffin	-0.9284171
## -Al Newman	-0.9284171

```
y <- hittersnew$Salary
```

```
svdx <- svd(X)
```

```
z <- X %*% svdx$v
```

```
head(z)
```

	[,1]	[,2]	[,3]	[,4]	[,5]
## -Alan Ashby	-0.009630358	-1.8669625	-1.2627377	-0.9337009	-1.1075240
## -Alvin Davis	0.410650757	2.4247988	0.9074630	-0.2637096	-1.2296868
## -Andre Dawson	3.460224766	-0.8243753	-0.5544124	-1.6136499	0.8558560
## -Andres Galarraga	-2.553449083	0.2305443	-0.5186536	-2.1721095	0.8187399
## -Alfredo Griffin	1.025746581	1.5705427	-1.3288484	3.4873546	-0.9815556
## -Al Newman	-3.973081710	-1.5044104	0.1551832	0.3691364	1.2070332

	[,6]	[,7]	[,8]	[,9]
## -Alan Ashby	-1.2096657	0.06502176	-0.09806158	-0.2519428
## -Alvin Davis	-1.8231407	-0.35920809	-1.19710045	-0.3711125
## -Andre Dawson	1.0267547	0.99748363	0.84264033	0.1970036
## -Andres Galarraga	-1.4888575	0.27561549	0.20669943	0.4141660
## -Alfredo Griffin	-0.5126979	-0.14018880	0.98782374	-0.6606737
## -Al Newman	-0.0334499	-0.01298256	-0.53928684	0.1216059

	[,10]	[,11]	[,12]	[,13]
## -Alan Ashby	-0.63588901	0.45846720	-0.61468829	0.47717836
## -Alvin Davis	0.01518571	0.17590012	-0.08037596	-0.40271546
## -Andre Dawson	0.76902393	0.05553923	-0.10473855	-0.02156399
## -Andres Galarraga	0.13658153	-0.08521228	0.14593911	-0.14190178

```

## -Alfredo Griffin -0.21715059 0.11104723 -0.19088017 0.30185066
## -Al Newman 0.41118223 -1.19642599 -0.53785673 -0.10581557
## [,14] [,15] [,16] [,17]
## -Alan Ashby 0.38766763 -0.06347540 -0.14451398 -0.08006351
## -Alvin Davis -0.01934883 -0.06502464 -0.16075777 -0.04425909
## -Andre Dawson -0.37548104 -0.23384538 -0.01150363 0.21786749
## -Andres Galarraga 0.02891979 -0.04237783 -0.05294216 0.04435846
## -Alfredo Griffin -0.09059747 -0.04372765 -0.01227672 0.05049665
## -Al Newman -0.23839489 -0.19274030 0.05733473 -0.05243208
## [,18] [,19]
## -Alan Ashby -0.03806717 -0.019213448
## -Alvin Davis -0.01480600 -0.003388059
## -Andre Dawson -0.03775314 -0.066374190
## -Andres Galarraga -0.01034065 -0.013555820
## -Alfredo Griffin -0.12579630 -0.070900042
## -Al Newman 0.02399045 0.025016886

head(pcr_fit$scores)

## [1] -0.009630358 0.410650757 3.460224766 -2.553449083 1.025746581
## [6] -3.973081710

z1 <- z[,1]
coef <- lm(y ~ z1)$coefficients

yhat <- coef[2] * z1

head(yhat)

## -Alan Ashby -Alvin Davis -Andre Dawson -Andres Galarraga
## -1.026321 43.763623 368.760971 -272.124624
## -Alfredo Griffin -Al Newman
## 109.315241 -423.416849

head(pcr_fit$fitted.values[,1])

## -Alan Ashby -Alvin Davis -Andre Dawson -Andres Galarraga
## 534.8996 579.6895 904.6869 263.8013
## -Alfredo Griffin -Al Newman
## 645.2411 112.5090

head(yhat + mean(y)) #cuz we do not have an intercept when getting yhat.

## -Alan Ashby -Alvin Davis -Andre Dawson -Andres Galarraga
## 534.8996 579.6895 904.6869 263.8013
## -Alfredo Griffin -Al Newman
## 645.2411 112.5090

# coefall <- lm(y ~ z)$coefficients
# coefall <- as.matrix(coefall)[-1,]

coefall <- solve(crossprod(z, z)) %*% t(z) %*% y
yhatnew <- z %*% coefall

head(yhatnew)

## [,1]
## -Alan Ashby -173.78982

```

```
## -Alvin Davis      176.76932
## -Andre Dawson    635.38523
## -Andres Galarraa  20.86163
## -Alfredo Griffin -42.67439
## -Al Newman       -288.54068

head(pcr_fit$fitted.values[, , 19])

##      -Alan Ashby      -Alvin Davis      -Andre Dawson -Andres Galarraa
##      362.1361         712.6952         1171.3111         556.7875
## -Alfredo Griffin      -Al Newman
##      493.2515         247.3852

head(yhatnew + mean(y)) #cuz we do not have an intercept when getting yhat.

##      [,1]
## -Alan Ashby      362.1361
## -Alvin Davis     712.6952
## -Andre Dawson    1171.3111
## -Andres Galarraa 556.7875
## -Alfredo Griffin 493.2515
## -Al Newman       247.3852

head(yhat)

##      -Alan Ashby      -Alvin Davis      -Andre Dawson -Andres Galarraa
##      -1.026321        43.763623        368.760971        -272.124624
## -Alfredo Griffin      -Al Newman
##      109.315241        -423.416849

yhat2 <- X %*% (coef[2] * svdx$v[, 1])
head(yhat2)

##      [,1]
## -Alan Ashby      -1.026321
## -Alvin Davis     43.763623
## -Andre Dawson    368.760971
## -Andres Galarraa -272.124624
## -Alfredo Griffin 109.315241
## -Al Newman       -423.416849

head(coef[2] * svdx$v[, 1])

## [1] 21.13208 20.87321 21.77988 21.13706 25.06280 22.26530

head(pcr_fit$coefficients[, , 1])

##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 21.13208 20.87321 21.77988 21.13706 25.06280 22.26530

for(i in 1:19){
  print(head(svdx$v[, 1:i] %*% solve(diag(svdx$d)[1:i, 1:i]) %*% t(svdx$u[, 1:i]) %*% y))
}

##      [,1]
## [1,] 21.13208
## [2,] 20.87321
## [3,] 21.77988
## [4,] 21.13706
```

```

## [5,] 25.06280
## [6,] 22.26530
##      [,1]
## [1,] 29.43897
## [2,] 29.03913
## [3,] 26.91261
## [4,] 29.31272
## [5,] 31.87073
## [6,] 27.23505
##      [,1]
## [1,] 31.59617
## [2,] 30.84112
## [3,] 21.65053
## [4,] 28.89488
## [5,] 30.09179
## [6,] 28.34585
##      [,1]
## [1,] 30.41158
## [2,] 30.17476
## [3,] 30.38956
## [4,] 30.74553
## [5,] 35.24208
## [6,] 33.18599
##      [,1]
## [1,] 28.76604
## [2,] 30.44702
## [3,] 25.84450
## [4,] 33.00088
## [5,] 33.81997
## [6,] 35.08779
##      [,1]
## [1,] 24.36304
## [2,] 25.32142
## [3,] 16.51782
## [4,] 24.48354
## [5,] 26.85981
## [6,] 33.87370
##      [,1]
## [1,] 27.005477
## [2,] 28.531195
## [3,]  4.031036
## [4,] 29.464202
## [5,] 18.974255
## [6,] 47.658639
##      [,1]
## [1,] 31.2842125
## [2,] 34.6956830
## [3,]  0.4426343
## [4,] 31.2802648
## [5,] 19.0548531
## [6,] 37.7734537
##      [,1]
## [1,] 30.923182
## [2,] 32.870155

```

```

## [3,] 4.218567
## [4,] 26.206161
## [5,] 22.941132
## [6,] 37.138179
##      [,1]
## [1,] 45.461722
## [2,] 45.896623
## [3,] -30.835695
## [4,] 29.425451
## [5,] 5.745737
## [6,] 25.063737
##      [,1]
## [1,] 42.525374
## [2,] 45.461187
## [3,] -24.735724
## [4,] 38.843973
## [5,] -1.597974
## [6,] 19.742034
##      [,1]
## [1,] 44.694896
## [2,] 48.006327
## [3,] -31.357920
## [4,] 32.077755
## [5,] 5.716194
## [6,] 23.638927
##      [,1]
## [1,] 41.804151
## [2,] 48.266973
## [3,] -39.242208
## [4,] 4.424083
## [5,] 37.644227
## [6,] 26.614529
##      [,1]
## [1,] -43.09181
## [2,] -10.11856
## [3,] -38.86714
## [4,] 78.32428
## [5,] 39.71934
## [6,] 92.57465
##      [,1]
## [1,] -82.658167
## [2,] -8.376493
## [3,] -65.281885
## [4,] 113.124806
## [5,] 74.012144
## [6,] 79.454906
##      [,1]
## [1,] -298.79956
## [2,] 296.63763
## [3,] 19.59726
## [4,] 19.72850
## [5,] -26.52000
## [6,] 122.88437
##      [,1]

```

```
## [1,] -347.251377
## [2,] 354.832778
## [3,] 3.010332
## [4,] -29.281173
## [5,] 10.577355
## [6,] 137.571853
##      [,1]
## [1,] -287.16387
## [2,] 330.31827
## [3,] 35.85694
## [4,] -55.75452
## [5,] -25.43236
## [6,] 133.82752
##      [,1]
## [1,] -291.64955
## [2,] 338.47458
## [3,] 37.92601
## [4,] -60.68796
## [5,] -27.04645
## [6,] 135.33143

# It does not work... Cuz, it did not count for the iteration before.
# z2 <- z[,2]
# coef <- lm(y ~ z2)$coefficients
# coef[2] * svdx$v[,2]

for (i in 1:19){
  print(head(pcr_fit$coefficients[,i]))
}
```

```
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 21.13208 20.87321 21.77988 21.13706 25.06280 22.26530
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 29.43897 29.03913 26.91261 29.31272 31.87073 27.23505
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 31.59617 30.84112 21.65053 28.89488 30.09179 28.34585
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 30.41158 30.17476 30.38956 30.74553 35.24208 33.18599
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 28.76604 30.44702 25.84450 33.00088 33.81997 35.08779
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 24.36304 25.32142 16.51782 24.48354 26.85981 33.87370
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 27.005477 28.531195 4.031036 29.464202 18.974255 47.658639
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 31.2842125 34.6956830 0.4426343 31.2802648 19.0548531 37.7734537
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 30.923182 32.870155 4.218567 26.206161 22.941132 37.138179
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 45.461722 45.896623 -30.835695 29.425451 5.745737 25.063737
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 42.525374 45.461187 -24.735724 38.843973 -1.597974 19.742034
##      AtBat      Hits      HmRun      Runs      RBI      Walks
## 44.694896 48.006327 -31.357920 32.077755 5.716194 23.638927
##      AtBat      Hits      HmRun      Runs      RBI      Walks
```

```
## 41.804151 48.266973 -39.242208 4.424083 37.644227 26.614529
## AtBat Hits HmRun Runs RBI Walks
## -43.09181 -10.11856 -38.86714 78.32428 39.71934 92.57465
## AtBat Hits HmRun Runs RBI Walks
## -82.658167 -8.376493 -65.281885 113.124806 74.012144 79.454906
## AtBat Hits HmRun Runs RBI Walks
## -298.79956 296.63763 19.59726 19.72850 -26.52000 122.88437
## AtBat Hits HmRun Runs RBI Walks
## -347.251377 354.832778 3.010332 -29.281173 10.577355 137.571853
## AtBat Hits HmRun Runs RBI Walks
## -287.16387 330.31827 35.85694 -55.75452 -25.43236 133.82752
## AtBat Hits HmRun Runs RBI Walks
## -291.64955 338.47458 37.92601 -60.68796 -27.04645 135.33143
```

Q. Do I need to standardize y for every deflation? ==> do not standardize y to get the right answer Q. Why do I get wrong coefficients...???

```
pls_fit <- plsrf(formula =Salary ~., data = Hitters, scale = T, validation = "none")

X0 <- X
y <- hittersnew$Salary
#y <- scale(y, T, T)
y0 <- as.matrix(y)

residuals <- matrix(0, 263, 19)
rank <- rankMatrix(X)[1]
Weights <- matrix(0, 19, 19)
Scores <- matrix(0, 263, 19)
loadings <- matrix(0, 19, 19)
coefficients <- c()

x <- c()
z <- c()

Weights[,1] <- crossprod(X0, y0)
Weights[,1] <- scale(Weights[,1], center = F, scale = sqrt(crossprod(Weights[,1], Weights[,1])))
Scores[,1] <- (X0 %*% Weights[,1]) / as.numeric(crossprod(Weights[,1], Weights[,1]))
loadings[,1] <- crossprod(X0, Scores[,1]) / as.numeric(crossprod(Scores[,1], Scores[,1]))

head(Weights[,1], 10)

## [1] 0.2256137 0.2507049 0.1960424 0.2399514 0.2568671 0.2536725 0.2289776
## [8] 0.3006891 0.3137047 0.3000006

head(pls_fit$loading.weights[,1], 10)

## AtBat Hits HmRun Runs RBI Walks Years
## 0.2256137 0.2507049 0.1960424 0.2399514 0.2568671 0.2536725 0.2289776
## CAtBat CHits CHmRun
## 0.3006891 0.3137047 0.3000006

head(Scores[,1], 10)

## [1] -0.1090169 0.6670947 3.4717021 -2.1298594 0.9770842 -4.0036686
```



```
## [7] -3.6684969 -3.4262063 3.5184199 3.2931917
```

```
head(pls_fit$scores[,1], 10)
```

```
##      -Alan Ashby      -Alvin Davis      -Andre Dawson -Andres Galarrraga
##      -0.1090169      0.6670947      3.4717021      -2.1298594
## -Alfredo Griffin      -Al Newman  -Argenis Salazar      -Andres Thomas
##      0.9770842      -4.0036686      -3.6684969      -3.4262063
## -Andre Thornton      -Alan Trammell
##      3.5184199      3.2931917
```

```
head(loadings[,1], 10)
```

```
## [1] 0.2256185 0.2231972 0.2179161 0.2249696 0.2566359 0.2292001 0.2660024
## [8] 0.3198516 0.3211356 0.3112691
```

```
head(pls_fit$loadings[,1], 10)
```

```
##      AtBat      Hits      HmRun      Runs      RBI      Walks      Years
## 0.2256185 0.2231972 0.2179161 0.2249696 0.2566359 0.2292001 0.2660024
##      CAtBat      CHits      CHmRun
## 0.3198516 0.3211356 0.3112691
```

#They are the same.

```
plsrs <- function(x0, y0){
  Weights[,1] <- crossprod(X0, y0)
  Weights[,1] <- scale(Weights[,1], center = F, scale = sqrt(crossprod(Weights[,1], Weights[,1])))

  Scores[,1] <- (X0 %*% Weights[,1]) / as.numeric(crossprod(Weights[,1], Weights[,1]))

  loadings[,1] <- crossprod(X0, Scores[,1]) / as.numeric(crossprod(Scores[,1], Scores[,1]))

  newX <- X0 - tcrossprod(Scores[,1], loadings[,1])

  coefficients[1] <- as.numeric(crossprod(y0, Scores[,1]) / crossprod(Scores[,1], Scores[,1]))

  newY <- y0 - coefficients[1] * Scores[,1] #deflation

  residuals[,1] <- newY - coefficients[1] * Scores[,1]

  for(i in 2:rankMatrix(x0)){
    Weights[,i] <- crossprod(newX, newY) #Weight
    Weights[,i] <- scale(Weights[,i], center = F, scale = sqrt(crossprod(Weights[,i], Weights[,i]))) #Wei

    Scores[,i] <- (newX %*% Weights[,i]) / as.numeric(crossprod(Weights[,i], Weights[,i])) #scores

    loadings[,i] <- crossprod(newX, Scores[,i]) / as.numeric(crossprod(Scores[,i], Scores[,i])) #loadings

    newX <- newX - tcrossprod(Scores[,i], loadings[,i])

    coefficients[i] <- as.numeric(crossprod(newY, Scores[,i]) / crossprod(Scores[,i], Scores[,i])) #coeff

    newY <- newY - coefficients[i] * Scores[,i] #deflation
    yhat <- 0
    for (k in 1:i){
```

```

    yhat <- yhat + (coefficients[k] * Scores[,k]) #residuals
  }

residuals[,i] <- newY - yhat
}

print("The first 10 rows of Weights will be:")
print(head(Weights, 10))
print("The first 10 rows of Scores will be:")
print(head(Scores, 10))
print("The first 10 rows of loadings will be:")
print(head(loadings, 10))
print("The first 10 elements of coefficients will be:")
print(head(coefficients, 10))
print("The first 10 rows of residuals will be:")
print(head(residuals, 10))
}

plsrs(X0, y0)

```

```

## [1] "The first 10 rows of Weights will be:"
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.2256137 -3.672182e-05 -0.46626308 -0.13465184 -0.254496553
## [2,] 0.2507049  2.088446e-01 -0.19752726  0.32998021  0.200091635
## [3,] 0.1960424 -1.660701e-01 -0.34608301 -0.07308201 -0.064993692
## [4,] 0.2399514  1.137448e-01 -0.30093413  0.14155898 -0.024967001
## [5,] 0.2568671  1.755372e-03 -0.30869665  0.06504328  0.023557558
## [6,] 0.2536725  1.857999e-01 -0.13119431  0.06940144  0.043809652
## [7,] 0.2289776 -2.811005e-01  0.08518189 -0.34375601 -0.194162154
## [8,] 0.3006891 -1.454858e-01  0.15144446 -0.11723994  0.003763243
## [9,] 0.3137047 -5.641651e-02  0.24475622  0.06422926  0.181986489
## [10,] 0.3000006 -8.555283e-02  0.18870338  0.10163180  0.137804452
##           [,6]      [,7]      [,8]      [,9]      [,10]
## [1,] -0.30790899 -0.40858271 -0.22032972  0.15655915 -0.06007446
## [2,]  0.01903897  0.06626299  0.18260953  0.53369055  0.19950085
## [3,]  0.49714619  0.20531354  0.03713366 -0.14325442  0.34482004
## [4,]  0.04572172 -0.06483011 -0.03006665  0.07105787 -0.37424085
## [5,]  0.24721569  0.01428430 -0.11927870 -0.17064998 -0.04920678
## [6,]  0.12055865  0.14784985  0.20799044 -0.16599159 -0.31453071
## [7,] -0.09118564  0.28129374  0.30111464  0.27204738  0.11596855
## [8,] -0.16537989 -0.04073454 -0.05586171 -0.14094251 -0.40205331
## [9,] -0.05212182  0.12183252  0.09712127  0.01385025 -0.23830077
## [10,] 0.09991064 -0.28803514 -0.37447232 -0.17964453  0.43889543
##           [,11]      [,12]      [,13]      [,14]      [,15]
## [1,] -0.002828448  0.239506363  0.15459643  0.17141362 -0.06059125
## [2,]  0.046688250  0.216741244 -0.32158889 -0.37829496 -0.11773563
## [3,] -0.186704978 -0.162790736  0.06728287 -0.11833307  0.09961507
## [4,] -0.033374106 -0.054656958 -0.26222521  0.25863829  0.49429782
## [5,] -0.305769551 -0.002373214  0.36577679 -0.06129290 -0.27385644
## [6,]  0.637388984 -0.377850677  0.08861766  0.02885383 -0.09481860
## [7,] -0.142762557 -0.119947622 -0.06705776  0.11319280  0.37265410
## [8,] -0.327861699 -0.157681307 -0.27192933 -0.30472943 -0.24413047

```

```

## [9,] -0.206613572 0.090297286 0.04724941 -0.03895325 -0.16175167
## [10,] 0.195082242 -0.092045796 -0.32780485 0.10187476 -0.01224271
##      [,16]      [,17]      [,18]      [,19]
## [1,] -0.026259627 0.06619698 0.426148749 -0.0968775652
## [2,] 0.128431500 -0.06699386 0.050124424 0.0991193317
## [3,] -0.079223999 0.49737588 0.176148188 -0.0063330452
## [4,] -0.142495693 0.13397140 -0.493024677 -0.0233427500
## [5,] 0.134217331 -0.55888172 -0.299720515 0.0245822505
## [6,] -0.009687937 -0.11857311 0.300296402 -0.0008270587
## [7,] -0.071961576 -0.45805612 0.182293662 -0.0261788570
## [8,] -0.236525698 0.10939588 0.121564483 0.4434313704
## [9,] -0.023559130 0.16550575 0.040508716 -0.7790639128
## [10,] -0.373421071 -0.24061802 0.005599505 -0.1333772116
## [1] "The first 10 rows of Scores will be:"
##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] -0.1090169 -0.08794742 1.1146654 -1.4059430 -0.61582496 -1.22859824
## [2,] 0.6670947 0.87856868 -1.0205639 0.9638650 0.03072175 0.14967378
## [3,] 3.4717021 0.52704957 1.2975660 -0.3869003 0.62786341 2.03068091
## [4,] -2.1298594 2.45419359 2.0763635 0.2077588 -0.10789413 0.58373496
## [5,] 0.9770842 -0.79366161 -2.1394733 0.4122068 0.84145421 -2.30384249
## [6,] -4.0036686 0.14999902 1.6439292 0.7910763 0.53093510 0.54853774
## [7,] -3.6684969 -1.34396286 -0.5804047 0.9551784 0.76302415 -0.41322568
## [8,] -3.4262063 -0.30266807 -1.0076750 -1.1485030 0.72441476 -0.41489008
## [9,] 3.5184199 -1.37455643 1.0514306 0.3407655 -0.87143317 0.09455316
## [10,] 3.2931917 0.17158739 -1.7482730 0.4770480 0.11008841 -1.63989398
##      [,7]      [,8]      [,9]      [,10]      [,11]
## [1,] -0.984619457 0.64156564 -0.22195650 0.38347060 0.28742746
## [2,] -0.511238332 1.15767249 -0.98266390 -0.30583914 0.30919351
## [3,] 0.706641597 -0.59686880 0.24451290 0.38002932 -0.34387371
## [4,] -0.315855788 0.06988085 -0.34315249 0.38480904 -0.02053905
## [5,] -0.146716353 0.39959882 0.49134279 0.06739601 -0.52183712
## [6,] 0.756747087 -0.35992566 -0.55821210 -0.55171421 -0.22781146
## [7,] -0.474399908 -0.53129145 -0.23283972 0.30082294 0.06183782
## [8,] 0.075271810 -0.24618897 -0.68422361 0.46068158 0.14036373
## [9,] -0.001731483 -1.31457258 -0.77503437 -0.36137202 0.82731327
## [10,] 1.138930684 0.23128242 0.05196033 -0.04911558 -0.16631295
##      [,12]      [,13]      [,14]      [,15]      [,16]
## [1,] 0.180051332 -0.13397459 -0.2374611592 -0.104270272 0.15340983
## [2,] 0.084203521 0.07231222 -0.0299120180 -0.388166812 0.03761008
## [3,] 0.238938003 -0.02598121 0.1957092074 0.072683555 -0.20918617
## [4,] -0.059119575 0.11883350 0.0050751054 -0.089518955 -0.06658609
## [5,] 0.325715928 -0.27465970 -0.1291336970 -0.003456517 -0.13717213
## [6,] -0.007211084 0.38778500 -0.3070951793 0.504319696 -0.04911759
## [7,] -0.139010256 0.15625338 0.0717692602 -0.082818017 0.05750562
## [8,] 0.421403902 0.14371869 0.1140581211 -0.067249272 0.03439537
## [9,] -0.035608749 0.16959386 0.0007108516 -0.222808866 0.22863526
## [10,] -0.384380590 -0.18693375 -0.0384366613 0.446093667 -0.06294017
##      [,17]      [,18]      [,19]
## [1,] -0.45492154 0.21553840 0.0194824008
## [2,] 0.02304197 0.09910835 0.0033005036
## [3,] -0.31372447 0.15648633 0.0678146723
## [4,] 0.08567642 -0.02875137 0.0139405891
## [5,] -0.40312457 0.01374031 0.0737582301
## [6,] 0.08166719 0.23972507 -0.0261787430

```

```

## [7,] 0.04828525 -0.11450067 -0.0009680118
## [8,] 0.20992113 -0.05468436 0.0038133401
## [9,] 0.05686659 0.02860831 0.0214171857
## [10,] 0.53227418 -0.32439064 -0.0279899050
## [1] "The first 10 rows of loadings will be:"
##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] 0.2256185 0.34657361 -0.3955083 0.09853735 0.10887572 -0.09246890
## [2,] 0.2231972 0.35568227 -0.3709201 0.14664095 0.17762319 -0.01590060
## [3,] 0.2179161 0.09120089 -0.3076810 -0.01352982 -0.65169025 0.38888716
## [4,] 0.2249696 0.33745299 -0.3753183 0.16443565 -0.07892452 0.07990575
## [5,] 0.2566359 0.23123408 -0.3428746 0.04345805 -0.26818880 0.23968377
## [6,] 0.2292001 0.28332703 -0.1676623 0.02925969 -0.09846508 0.04259944
## [7,] 0.2660024 -0.34442292 0.2658135 -0.16584980 -0.08655135 -0.23950799
## [8,] 0.3198516 -0.25806652 0.2130499 -0.12068811 0.19893283 -0.14390113
## [9,] 0.3211356 -0.23836322 0.2110060 -0.10252067 0.24349695 -0.11636245
## [10,] 0.3112691 -0.22583102 0.1352995 -0.02463517 0.01989703 0.25178764
##      [,7]      [,8]      [,9]     [,10]     [,11]
## [1,] -0.198855540 -0.322786722 0.19385746 -0.056580491 -0.155223801
## [2,] -0.107559127 -0.166653559 0.40982685 0.141827030 -0.091221899
## [3,] 0.169966806 0.130883640 -0.35734216 0.575455992 -0.083122963
## [4,] -0.036210302 -0.076569053 0.30341205 -0.333013948 0.001403536
## [5,] 0.127823152 -0.007600237 -0.14009907 0.328509190 -0.304259500
## [6,] -0.050131812 0.316620308 0.02929047 -1.101894906 0.877811354
## [7,] -0.005330873 0.123078679 0.20004622 0.292322593 -0.066441160
## [8,] 0.012439026 0.036375291 0.10867954 0.002952999 -0.227530759
## [9,] 0.029384851 0.088057256 0.16180358 0.016928194 -0.264068775
## [10,] 0.068417088 -0.256907561 -0.45214066 0.197911072 0.253650004
##      [,12]     [,13]     [,14]     [,15]     [,16]
## [1,] 0.15813615 0.003754557 0.22759799 -0.04922531 -0.11260794
## [2,] 0.38600618 0.011306019 -0.26912238 -0.17332456 0.21581927
## [3,] -0.19820437 0.171414512 -0.21070301 0.13390554 -0.72800844
## [4,] 0.08336253 -0.489823721 -0.19970863 0.55597414 -0.31724997
## [5,] -0.19489599 0.419713793 0.19264561 -0.33194964 0.86323090
## [6,] -0.42449365 0.063226644 0.11677615 -0.09062537 0.14498058
## [7,] -0.08465247 -0.166666031 -0.23235770 0.40380119 0.52553359
## [8,] -0.01455416 -0.003771184 -0.07835487 -0.14175520 -0.37922330
## [9,] 0.06542805 0.081527792 0.11103402 -0.15155458 -0.23944728
## [10,] 0.08049085 -0.417453383 0.11322704 0.14938490 -0.05955537
##      [,17]     [,18]     [,19]
## [1,] -0.11725032 0.43407764 -0.0968775652
## [2,] -0.08857127 0.04201206 0.0991193317
## [3,] 0.42154811 0.17666651 -0.0063330452
## [4,] 0.34620726 -0.49111420 -0.0233427500
## [5,] -0.42985889 -0.30173243 0.0245822505
## [6,] -0.24784385 0.30036409 -0.0008270587
## [7,] -0.53652937 0.18443626 -0.0261788570
## [8,] 0.05706515 0.08527211 0.4434313704
## [9,] 0.14806767 0.10427074 -0.7790639128
## [10,] -0.24302848 0.01651568 -0.1333772116
## [1] "The first 10 elements of coefficients will be:"
## [1] 110.99529 65.58384 34.73057 49.48602 99.57464 52.46386 59.49169
## [8] 84.66053 37.62463 57.15701
## [1] "The first 10 rows of residuals will be:"
##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]

```

```
## [1,] 499.2007 510.7366 433.31065 572.45971 695.10081 824.01482
## [2,] 331.9113 216.6714 287.56097 192.16528 186.04706 170.34213
## [3,] -270.6851 -339.8170 -429.94741 -391.65509 -516.69364 -729.76836
## [4,] 564.3087 242.3978 98.17127 77.60896 99.09599 37.84601
## [5,] 533.0965 637.1993 785.80951 745.01256 577.43755 819.17450
## [6,] 958.7767 939.1017 824.91247 746.61802 640.88268 583.32586
## [7,] 914.3717 1090.6562 1130.97180 1036.43583 884.48012 927.83895
## [8,] 835.5855 875.2858 945.28001 1058.94970 914.68301 958.21649
## [9,] 318.9440 499.2413 426.20777 392.48151 566.02680 556.10555
## [10,] -213.9145 -236.4212 -114.98420 -162.19862 -184.12265 -12.05231
##      [,7]      [,8]      [,9]     [,10]     [,11]     [,12]
## [1,] 941.1682 832.53759 849.23966 805.40359 777.44018 760.34714
## [2,] 231.1710 35.15266 109.09740 144.05910 113.97809 105.98430
## [3,] -813.8470 -712.78451 -731.18392 -774.62660 -741.17161 -763.85502
## [4,] 75.4276 63.59530 89.41727 45.42821 47.42642 53.03889
## [5,] 836.6313 768.97081 731.99762 724.29331 775.06212 744.14051
## [6,] 493.2855 554.22853 596.23358 659.30225 681.46571 682.15029
## [7,] 984.2846 1074.24348 1091.76450 1057.37622 1051.36011 1064.55694
## [8,] 949.2604 990.94537 1042.43270 989.77033 976.11454 936.10887
## [9,] 556.3116 778.89640 837.21717 878.52706 798.03891 801.41940
## [10,] -147.5661 -186.72711 -190.63709 -185.02249 -168.84214 -132.35125
##      [,13]     [,14]     [,15]     [,16]     [,17]     [,18]
## [1,] 777.65946 823.43008 832.47184 809.95604 830.43623 824.52847
## [2,] 96.64004 102.40558 136.06532 130.54533 129.50800 126.79150
## [3,] -760.49770 -798.22063 -804.52336 -773.82133 -759.69772 -763.98690
## [4,] 37.68312 36.70489 44.46749 54.24026 50.38318 51.17124
## [5,] 779.63230 804.52281 804.82254 824.95515 843.10348 842.72687
## [6,] 632.04034 691.23292 647.50103 654.70996 651.03337 644.46267
## [7,] 1044.36573 1030.53221 1037.71375 1029.27371 1027.09995 1030.23833
## [8,] 917.53740 895.55271 901.38421 896.33603 886.88556 888.38442
## [9,] 779.50432 779.36731 798.68809 765.13154 762.57146 761.78732
## [10,] -108.19549 -100.78683 -139.46967 -130.23201 -154.19455 -145.30322
##      [,19]
## [1,] 822.57963
## [2,] 126.46135
## [3,] -770.77046
## [4,] 49.77675
## [5,] 835.34877
## [6,] 647.08135
## [7,] 1030.33516
## [8,] 888.00297
## [9,] 759.64494
## [10,] -142.50337
```

#Check the answers

```
head(pls_fit$loading.weights[,1:19], 10)
```

```
##      Comp 1      Comp 2      Comp 3      Comp 4      Comp 5
## AtBat 0.2256137 -3.672182e-05 -0.46626308 -0.13465184 -0.254496553
## Hits 0.2507049 2.088446e-01 -0.19752726 0.32998021 0.200091635
## HmRun 0.1960424 -1.660701e-01 -0.34608301 -0.07308201 -0.064993692
## Runs 0.2399514 1.137448e-01 -0.30093413 0.14155898 -0.024967001
## RBI 0.2568671 1.755372e-03 -0.30869665 0.06504328 0.023557558
## Walks 0.2536725 1.857999e-01 -0.13119431 0.06940144 0.043809652
## Years 0.2289776 -2.811005e-01 0.08518189 -0.34375601 -0.194162154
```

```
## CatBat 0.3006891 -1.454858e-01 0.15144446 -0.11723994 0.003763243
## CHits 0.3137047 -5.641651e-02 0.24475622 0.06422926 0.181986489
## CHmRun 0.3000006 -8.555283e-02 0.18870338 0.10163180 0.137804452
##          Comp 6      Comp 7      Comp 8      Comp 9      Comp 10
## AtBat -0.30790899 -0.40858271 -0.22032972 0.15655915 -0.06007446
## Hits 0.01903897 0.06626299 0.18260953 0.53369055 0.19950085
## HmRun 0.49714619 0.20531354 0.03713366 -0.14325442 0.34482004
## Runs 0.04572172 -0.06483011 -0.03006665 0.07105787 -0.37424085
## RBI 0.24721569 0.01428430 -0.11927870 -0.17064998 -0.04920678
## Walks 0.12055865 0.14784985 0.20799044 -0.16599159 -0.31453071
## Years -0.09118564 0.28129374 0.30111464 0.27204738 0.11596855
## CatBat -0.16537989 -0.04073454 -0.05586171 -0.14094251 -0.40205331
## CHits -0.05212182 0.12183252 0.09712127 0.01385025 -0.23830077
## CHmRun 0.09991064 -0.28803514 -0.37447232 -0.17964453 0.43889543
##          Comp 11      Comp 12      Comp 13      Comp 14      Comp 15
## AtBat -0.002828448 0.239506363 0.15459643 0.17141362 -0.06059125
## Hits 0.046688250 0.216741244 -0.32158889 -0.37829496 -0.11773563
## HmRun -0.186704978 -0.162790736 0.06728287 -0.11833307 0.09961507
## Runs -0.033374106 -0.054656958 -0.26222521 0.25863829 0.49429782
## RBI -0.305769551 -0.002373214 0.36577679 -0.06129290 -0.27385644
## Walks 0.637388984 -0.377850677 0.08861766 0.02885383 -0.09481860
## Years -0.142762557 -0.119947622 -0.06705776 0.11319280 0.37265410
## CatBat -0.327861699 -0.157681307 -0.27192933 -0.30472943 -0.24413047
## CHits -0.206613572 0.090297286 0.04724941 -0.03895325 -0.16175167
## CHmRun 0.195082242 -0.092045796 -0.32780485 0.10187476 -0.01224271
##          Comp 16      Comp 17      Comp 18      Comp 19
## AtBat -0.026259627 0.06619698 0.426148749 -0.0968775652
## Hits 0.128431500 -0.06699386 0.050124424 0.0991193317
## HmRun -0.079223999 0.49737588 0.176148188 -0.0063330452
## Runs -0.142495693 0.13397140 -0.493024677 -0.0233427500
## RBI 0.134217331 -0.55888172 -0.299720515 0.0245822505
## Walks -0.009687937 -0.11857311 0.300296402 -0.0008270587
## Years -0.071961576 -0.45805612 0.182293662 -0.0261788570
## CatBat -0.236525698 0.10939588 0.121564483 0.4434313704
## CHits -0.023559130 0.16550575 0.040508716 -0.7790639128
## CHmRun -0.373421071 -0.24061802 0.005599505 -0.1333772116
```

```
head(pls_fit$scores[,1:19], 10)
```

```
##          Comp 1      Comp 2      Comp 3      Comp 4      Comp 5
## -Alan Ashby -0.1090169 -0.08794742 1.1146654 -1.4059430 -0.61582496
## -Alvin Davis 0.6670947 0.87856868 -1.0205639 0.9638650 0.03072175
## -Andre Dawson 3.4717021 0.52704957 1.2975660 -0.3869003 0.62786341
## -Andres Galarrraga -2.1298594 2.45419359 2.0763635 0.2077588 -0.10789413
## -Alfredo Griffin 0.9770842 -0.79366161 -2.1394733 0.4122068 0.84145421
## -Al Newman -4.0036686 0.14999902 1.6439292 0.7910763 0.53093510
## -Argenis Salazar -3.6684969 -1.34396286 -0.5804047 0.9551784 0.76302415
## -Andres Thomas -3.4262063 -0.30266807 -1.0076750 -1.1485030 0.72441476
## -Andre Thornton 3.5184199 -1.37455643 1.0514306 0.3407655 -0.87143317
## -Alan Trammell 3.2931917 0.17158739 -1.7482730 0.4770480 0.11008841
##          Comp 6      Comp 7      Comp 8      Comp 9
## -Alan Ashby -1.22859824 -0.984619457 0.64156564 -0.22195650
## -Alvin Davis 0.14967378 -0.511238332 1.15767249 -0.98266390
## -Andre Dawson 2.03068091 0.706641597 -0.59686880 0.24451290
## -Andres Galarrraga 0.58373496 -0.315855788 0.06988085 -0.34315249
```

```
## -Alfredo Griffin -2.30384249 -0.146716353 0.39959882 0.49134279
## -Al Newman 0.54853774 0.756747087 -0.35992566 -0.55821210
## -Argenis Salazar -0.41322568 -0.474399908 -0.53129145 -0.23283972
## -Andres Thomas -0.41489008 0.075271810 -0.24618897 -0.68422361
## -Andre Thornton 0.09455316 -0.001731483 -1.31457258 -0.77503437
## -Alan Trammell -1.63989398 1.138930684 0.23128242 0.05196033
## Comp 10 Comp 11 Comp 12 Comp 13
## -Alan Ashby 0.38347060 0.28742746 0.180051332 -0.13397459
## -Alvin Davis -0.30583914 0.30919351 0.084203521 0.07231222
## -Andre Dawson 0.38002932 -0.34387371 0.238938003 -0.02598121
## -Andres Galarraga 0.38480904 -0.02053905 -0.059119575 0.11883350
## -Alfredo Griffin 0.06739601 -0.52183712 0.325715928 -0.27465970
## -Al Newman -0.55171421 -0.22781146 -0.007211084 0.38778500
## -Argenis Salazar 0.30082294 0.06183782 -0.139010256 0.15625338
## -Andres Thomas 0.46068158 0.14036373 0.421403902 0.14371869
## -Andre Thornton -0.36137202 0.82731327 -0.035608749 0.16959386
## -Alan Trammell -0.04911558 -0.16631295 -0.384380590 -0.18693375
## Comp 14 Comp 15 Comp 16 Comp 17
## -Alan Ashby -0.2374611592 -0.104270272 0.15340983 -0.45492154
## -Alvin Davis -0.0299120180 -0.388166812 0.03761008 0.02304197
## -Andre Dawson 0.1957092074 0.072683555 -0.20918617 -0.31372447
## -Andres Galarraga 0.0050751054 -0.089518955 -0.06658609 0.08567642
## -Alfredo Griffin -0.1291336970 -0.003456517 -0.13717213 -0.40312457
## -Al Newman -0.3070951793 0.504319696 -0.04911759 0.08166719
## -Argenis Salazar 0.0717692602 -0.082818017 0.05750562 0.04828525
## -Andres Thomas 0.1140581211 -0.067249272 0.03439537 0.20992113
## -Andre Thornton 0.0007108516 -0.222808866 0.22863526 0.05686659
## -Alan Trammell -0.0384366613 0.446093667 -0.06294017 0.53227418
## Comp 18 Comp 19
## -Alan Ashby 0.21553840 0.0194824008
## -Alvin Davis 0.09910835 0.0033005036
## -Andre Dawson 0.15648633 0.0678146723
## -Andres Galarraga -0.02875137 0.0139405891
## -Alfredo Griffin 0.01374031 0.0737582301
## -Al Newman 0.23972507 -0.0261787430
## -Argenis Salazar -0.11450067 -0.0009680118
## -Andres Thomas -0.05468436 0.0038133401
## -Andre Thornton 0.02860831 0.0214171857
## -Alan Trammell -0.32439064 -0.0279899050
```

```
head(pls_fit$loadings[,1:19], 10)
```

```
## Comp 1 Comp 2 Comp 3 Comp 4 Comp 5
## AtBat 0.2256185 0.34657361 -0.3955083 0.09853735 0.10887572
## Hits 0.2231972 0.35568227 -0.3709201 0.14664095 0.17762319
## HmRun 0.2179161 0.09120089 -0.3076810 -0.01352982 -0.65169025
## Runs 0.2249696 0.33745299 -0.3753183 0.16443565 -0.07892452
## RBI 0.2566359 0.23123408 -0.3428746 0.04345805 -0.26818880
## Walks 0.2292001 0.28332703 -0.1676623 0.02925969 -0.09846508
## Years 0.2660024 -0.34442292 0.2658135 -0.16584980 -0.08655135
## CAtBat 0.3198516 -0.25806652 0.2130499 -0.12068811 0.19893283
## CHits 0.3211356 -0.23836322 0.2110060 -0.10252067 0.24349695
## CHmRun 0.3112691 -0.22583102 0.1352995 -0.02463517 0.01989703
## Comp 6 Comp 7 Comp 8 Comp 9 Comp 10
## AtBat -0.09246890 -0.198855540 -0.322786722 0.19385746 -0.056580491
```

```
## Hits      -0.01590060 -0.107559127 -0.166653559  0.40982685  0.141827030
## HmRun      0.38888716  0.169966806  0.130883640 -0.35734216  0.575455992
## Runs       0.07990575 -0.036210302 -0.076569053  0.30341205 -0.333013948
## RBI        0.23968377  0.127823152 -0.007600237 -0.14009907  0.328509190
## Walks      0.04259944 -0.050131812  0.316620308  0.02929047 -1.101894906
## Years     -0.23950799 -0.005330873  0.123078679  0.20004622  0.292322593
## CAtBat    -0.14390113  0.012439026  0.036375291  0.10867954  0.002952999
## CHits     -0.11636245  0.029384851  0.088057256  0.16180358  0.016928194
## CHmRun     0.25178764  0.068417088 -0.256907561 -0.45214066  0.197911072
##           Comp 11      Comp 12      Comp 13      Comp 14      Comp 15
## AtBat    -0.155223801  0.15813615  0.003754557  0.22759799 -0.04922531
## Hits     -0.091221899  0.38600618  0.011306019 -0.26912238 -0.17332456
## HmRun    -0.083122963 -0.19820437  0.171414512 -0.21070301  0.13390554
## Runs      0.001403536  0.08336253 -0.489823721 -0.19970863  0.55597414
## RBI      -0.304259500 -0.19489599  0.419713793  0.19264561 -0.33194964
## Walks     0.877811354 -0.42449365  0.063226644  0.11677615 -0.09062537
## Years    -0.066441160 -0.08465247 -0.166666031 -0.23235770  0.40380119
## CAtBat   -0.227530759 -0.01455416 -0.003771184 -0.07835487 -0.14175520
## CHits    -0.264068775  0.06542805  0.081527792  0.11103402 -0.15155458
## CHmRun   0.253650004  0.08049085 -0.417453383  0.11322704  0.14938490
##           Comp 16      Comp 17      Comp 18      Comp 19
## AtBat    -0.11260794 -0.11725032  0.43407764 -0.0968775652
## Hits      0.21581927 -0.08857127  0.04201206  0.0991193317
## HmRun    -0.72800844  0.42154811  0.17666651 -0.0063330452
## Runs     -0.31724997  0.34620726 -0.49111420 -0.0233427500
## RBI       0.86323090 -0.42985889 -0.30173243  0.0245822505
## Walks     0.14498058 -0.24784385  0.30036409 -0.0008270587
## Years     0.52553359 -0.53652937  0.18443626 -0.0261788570
## CAtBat   -0.37922330  0.05706515  0.08527211  0.4434313704
## CHits    -0.23944728  0.14806767  0.10427074 -0.7790639128
## CHmRun   -0.05955537 -0.24302848  0.01651568 -0.1333772116
```

```
head(pls_fit$coefficients, 10)
```

```
## [1] 25.04206 27.82707 21.75978 26.63347 28.51104 28.15645 25.41544
## [8] 33.37508 34.81975 33.29865
```