Exercise 3 - Advanced Methods for Regression and Classification

12433732 - Stefan Merdian

2024-11-03

```
load("building.RData")
require(pls)
```

```
## Lade nötiges Paket: pls

##
## Attache Paket: 'pls'

## Das folgende Objekt ist maskiert 'package:stats':
##
##     loadings
```

```
library(pls)
```

```
set.seed(1)

sample <- sample(c(TRUE, FALSE), nrow(df), replace=TRUE, prob=c(0.7,0.3))
train_data <- df[sample, ]
test_data <- df[!sample, ]
```

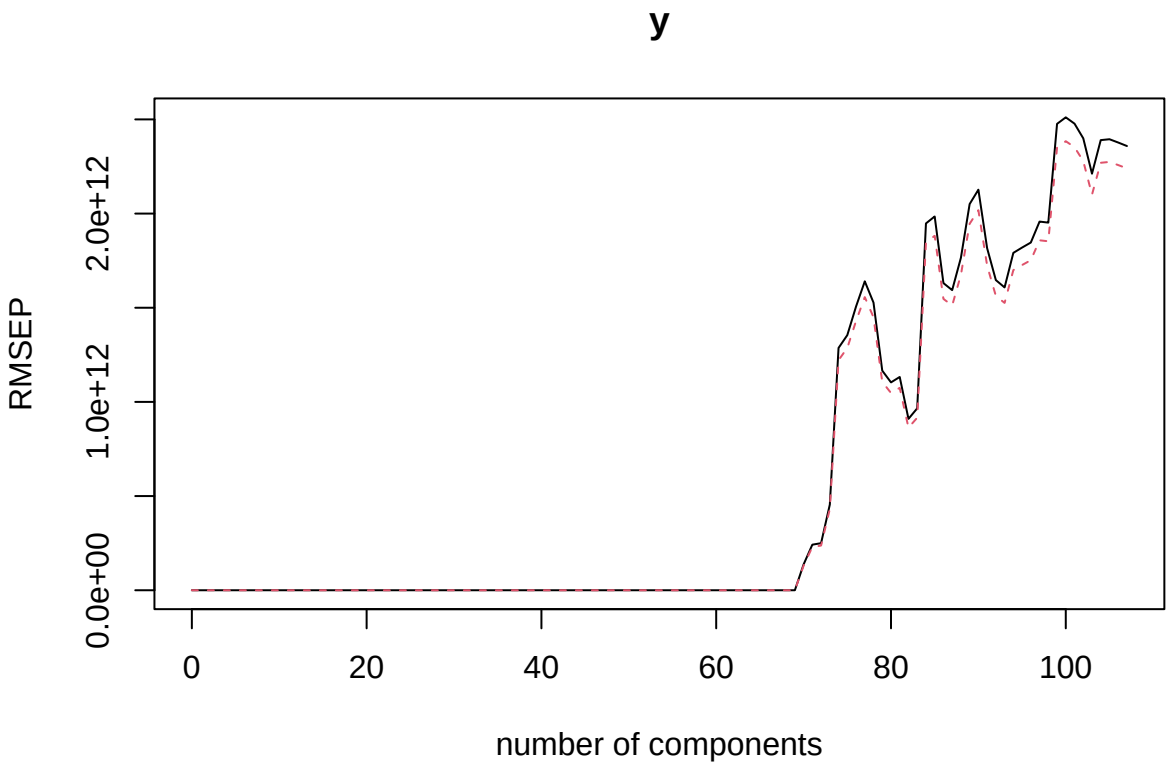
1) Principal component regression (PCR):

a)

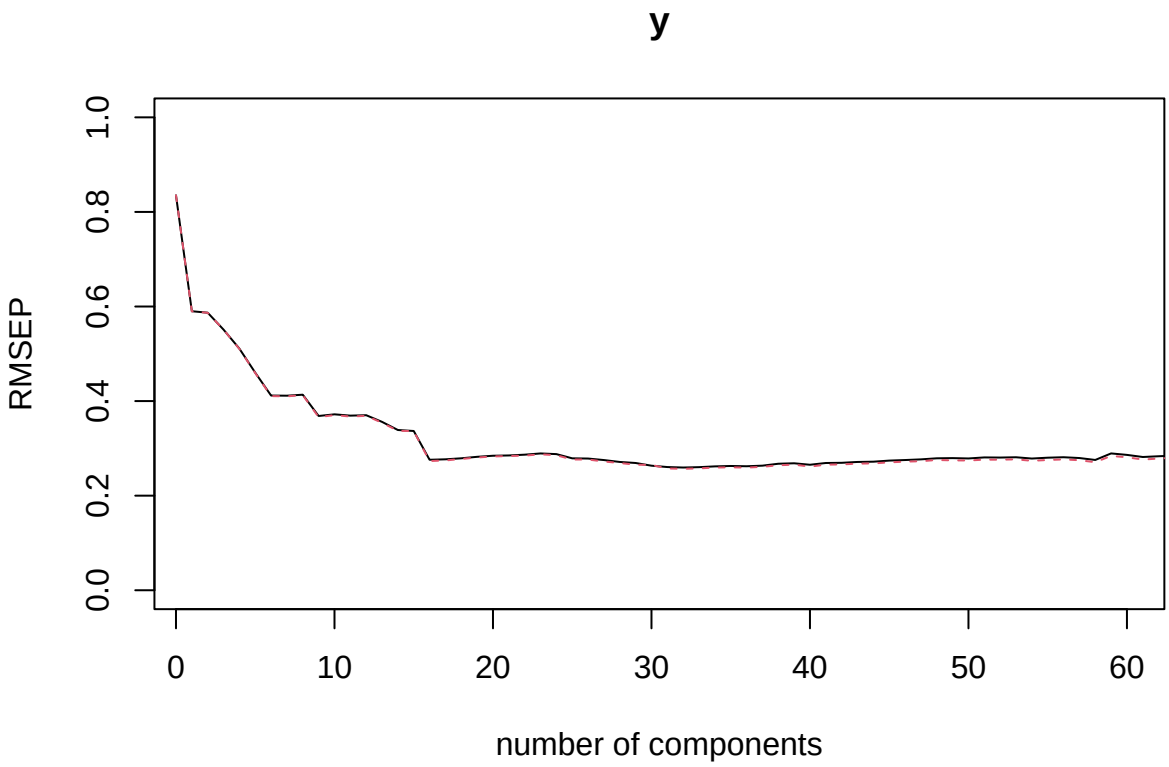
```
pcr_fit = pcr(y~., data = train_data, scale = TRUE, validation = "CV",segments = 10)
```

b)

```
validationplot(pcr_fit)
```




```
validationplot(pcr_fit,xlim=c(1,60), ylim = c(0, 1))
```




```
cv_results <- pcr_fit$validation
rmse_values <- RMSEP(pcr_fit, estimate = "CV")
optimal_number <- which.min(cv_results$PRESS)
optimal_rmse <- rmse_values$val[optimal_number]
cat('\n RMSE  ', optimal_rmse)
```

```
##
##  RMSE   0.2604981
```

The optimal number of components is likely between 30 and 40, as this range shows the lowest RMSE values. The RMSE values are very similar from around 30 to 60 components, so we will select fewer components to keep the model simpler.

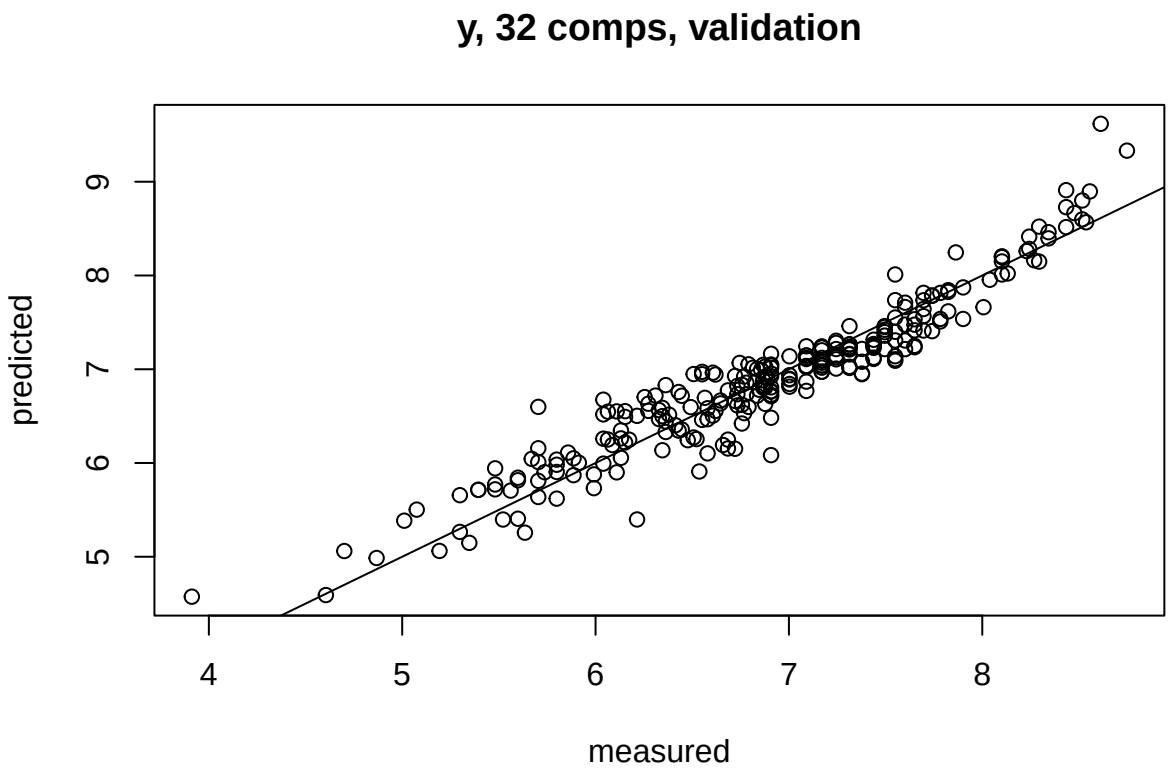
c)

```
cv_results <- pcr_fit$validation

optimal_number <- which.min(cv_results$PRESS)
optimal_number
```

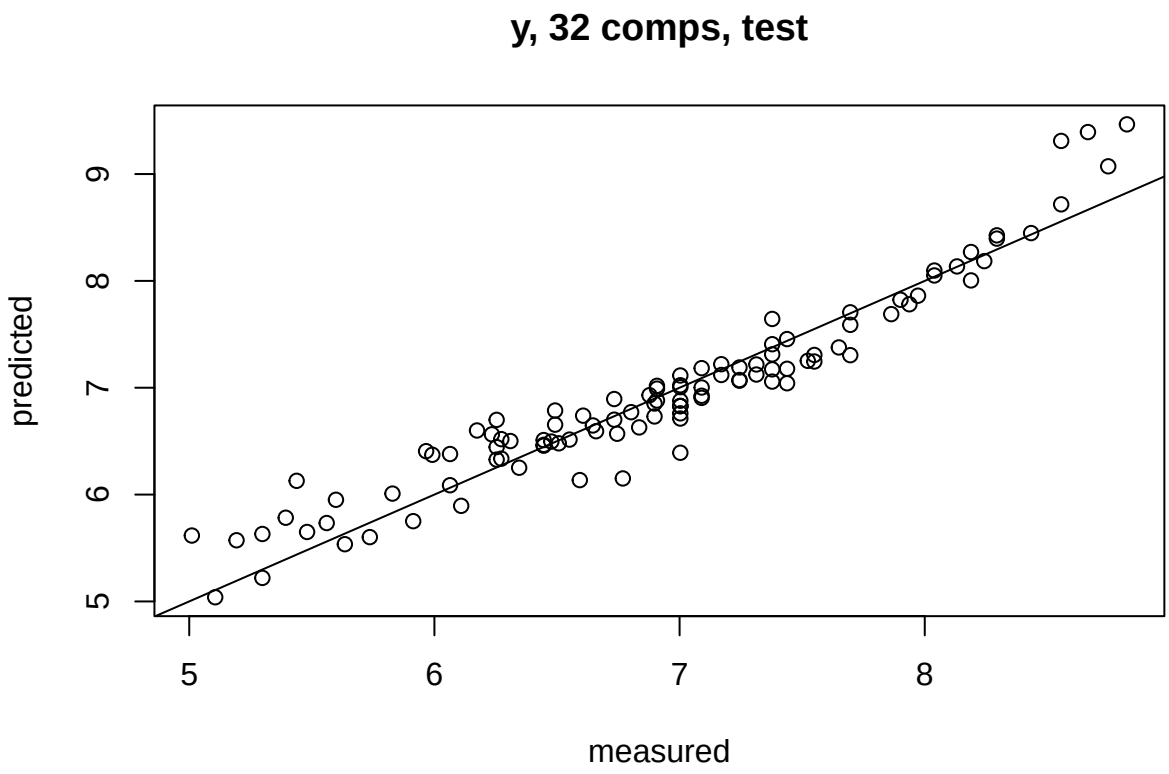
```
## [1] 32
```

```r
predplot(pcr_fit,ncomp = optimal_number)
abline(0,1)
```



d)

```r
predplot(pcr_fit,ncomp = optimal_number, newdata = test_data)
abline(0,1)
```




```
pred.pcr_test <- predict(pcr_fit,newdata=test_data,ncomp=optimal_number)
rmse <- sqrt(mean((test_data$y - pred.pcr_test)^2))
cat('\n RMSE for the test data: ', rmse)
```

```
##
##  RMSE for the test data:  0.2582652
```

2)

a)

```
install.packages("pls")
```

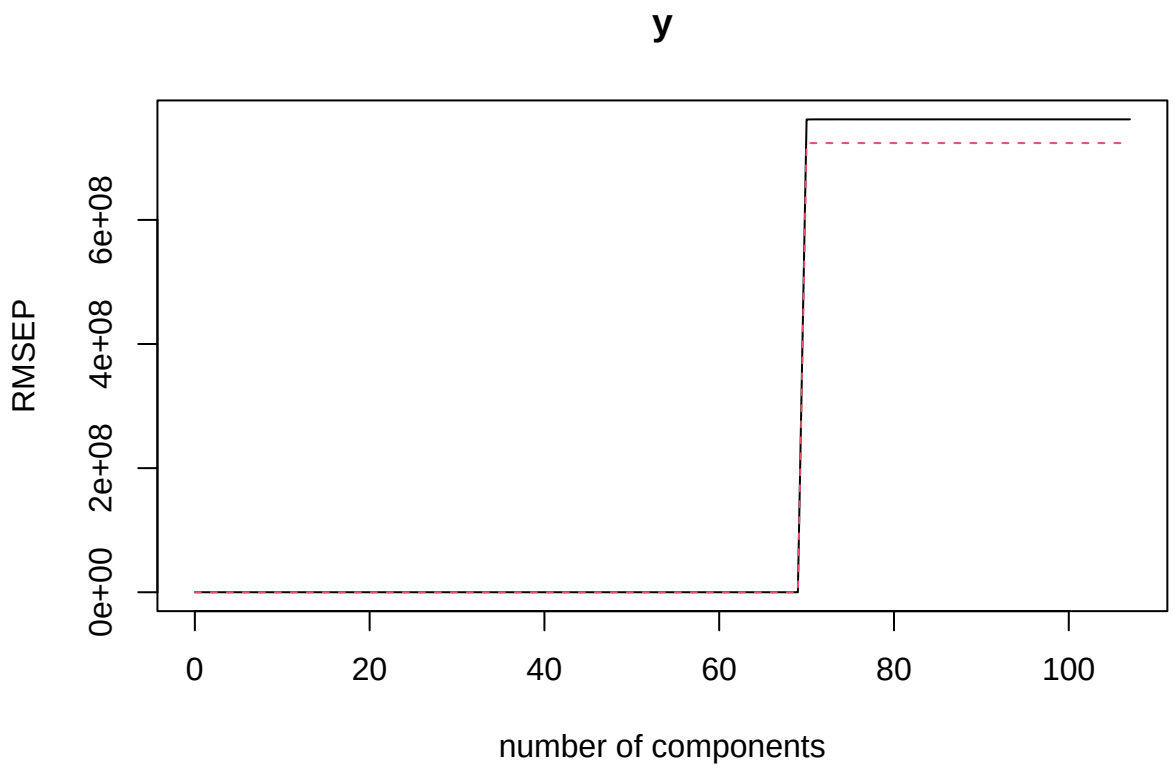
```
## Warning: Paket 'pls' wird gerade benutzt und deshab nicht installiert
```

```
library(pls)
```

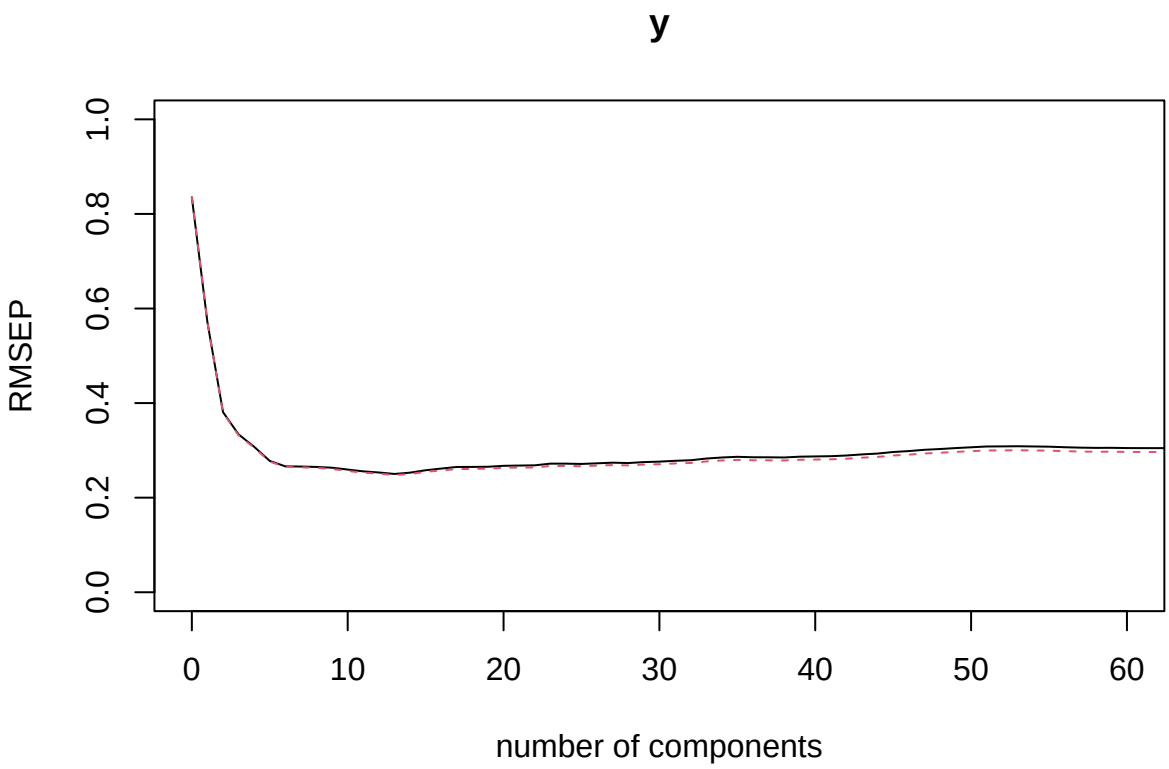
```
pls_fit <- plsr(y~., data = train_data, scale=TRUE, validation="CV", segments = 10)
```

b)

```
validationplot(pls_fit)
```




```
validationplot(pls_fit,xlim=c(0,60), ylim = c(0, 1))
```



At first glance, the PLS model appears similar to the PCR model. However, on closer inspection, we see that the PLS model achieves a much lower RMSE with fewer components compared to PCR. Although the lowest RMSE values for both models are similar, the PLS model accomplishes this with a simpler structure, using fewer components to achieve the same low RMSE.

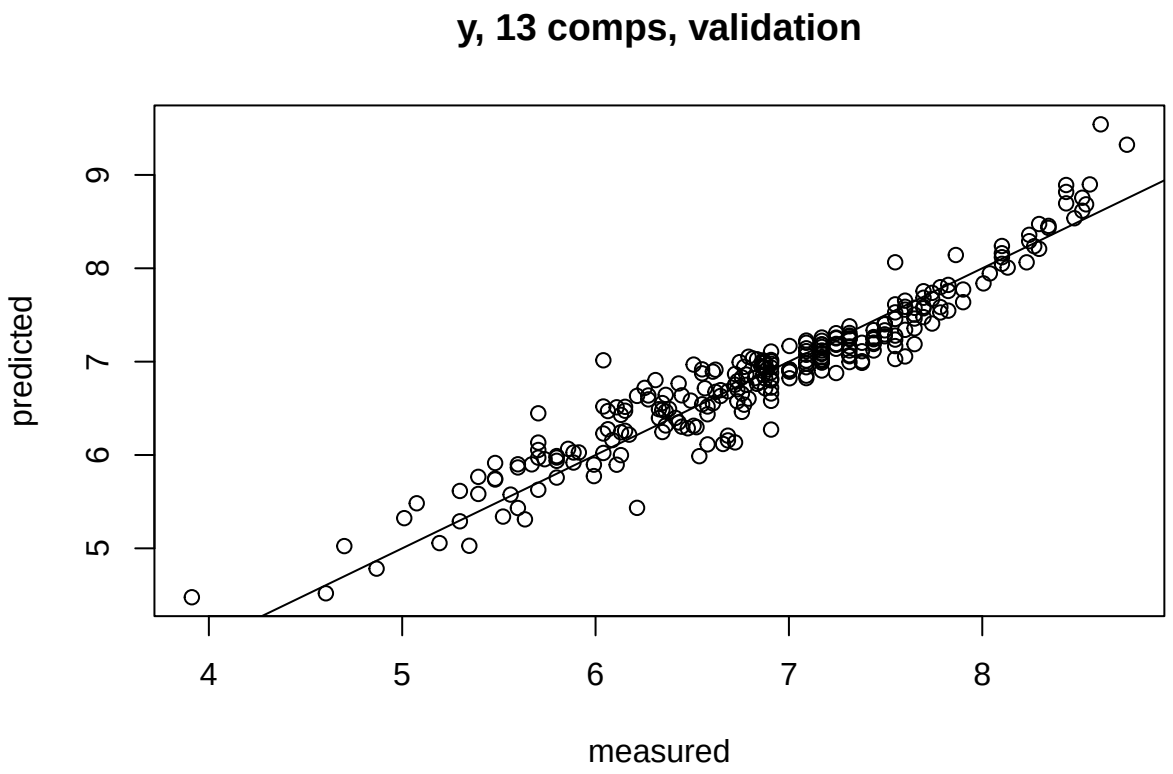
c)

```
cv_results <- pls_fit$validation

optimal_number_pls <- which.min(cv_results$PRESS)
optimal_number_pls
```

```
## [1] 13
```

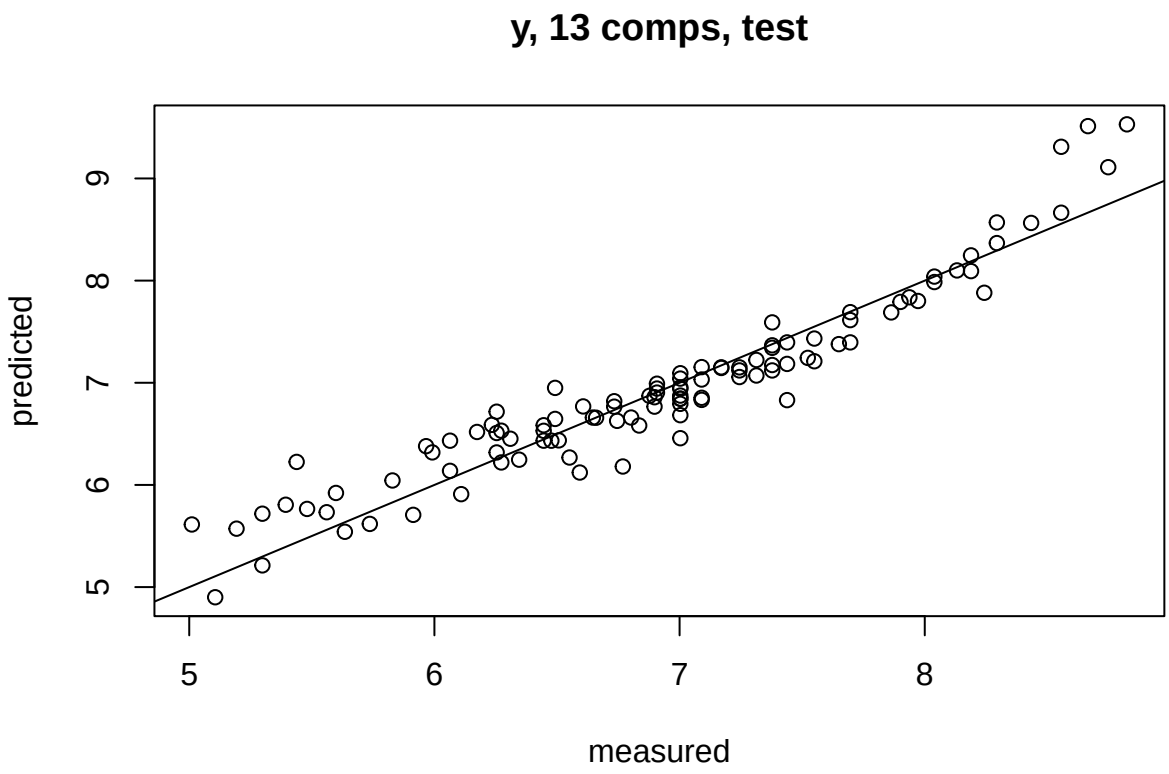
```
predplot(pls_fit,ncomp = optimal_number_pls)
abline(0,1)
```



The scatter plot for the training data shows a comparable distribution, indicating that both models represent the data patterns effectively.

d)

```
predplot(pls_fit,ncomp = optimal_number_pls, newdata = test_data)
abline(0,1)
```




```
pred.pls_test <- predict(pls_fit,newdata=test_data,ncomp=optimal_number_pls)
rmse <- sqrt(mean((test_data$y - pred.pls_test)^2))
cat('\n RMSE for the test data: ', rmse)
```

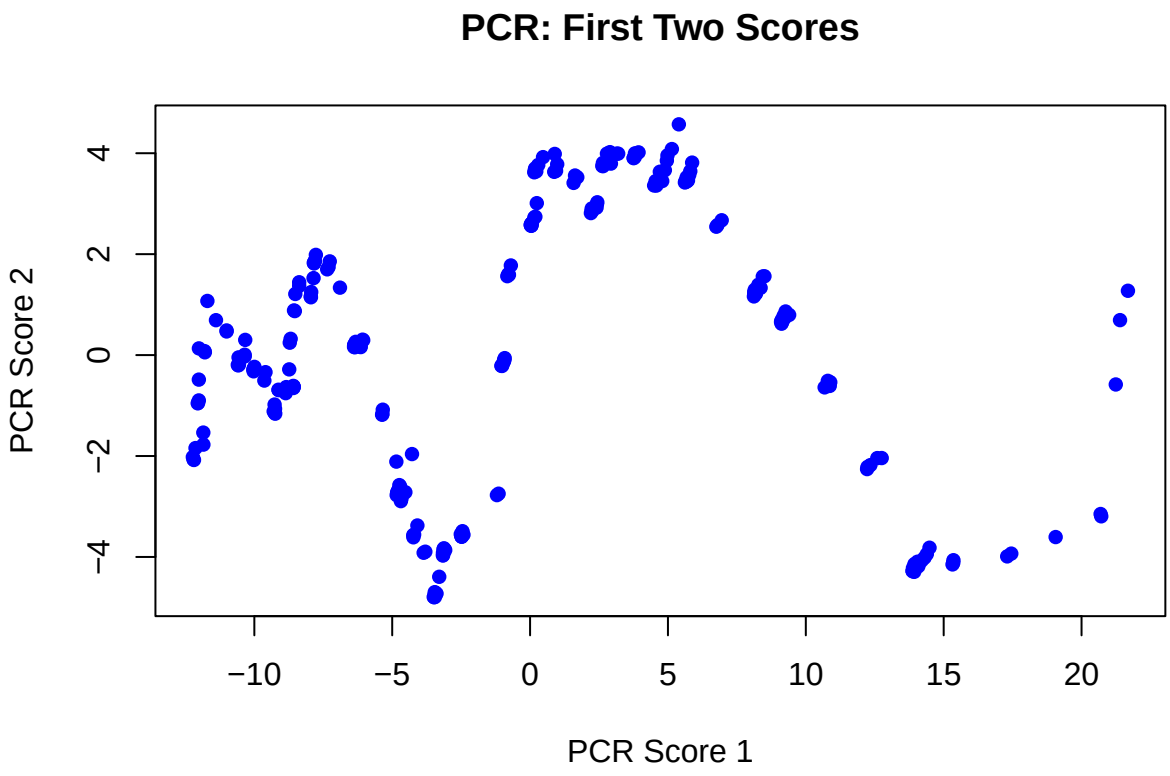
```
##
##  RMSE for the test data:  0.2749538
```

While both models (PLS and PCR) produce similar predictions on the test data—as shown by comparable scatter plots—the PCR model achieves slightly better accuracy in terms of test RMSE:

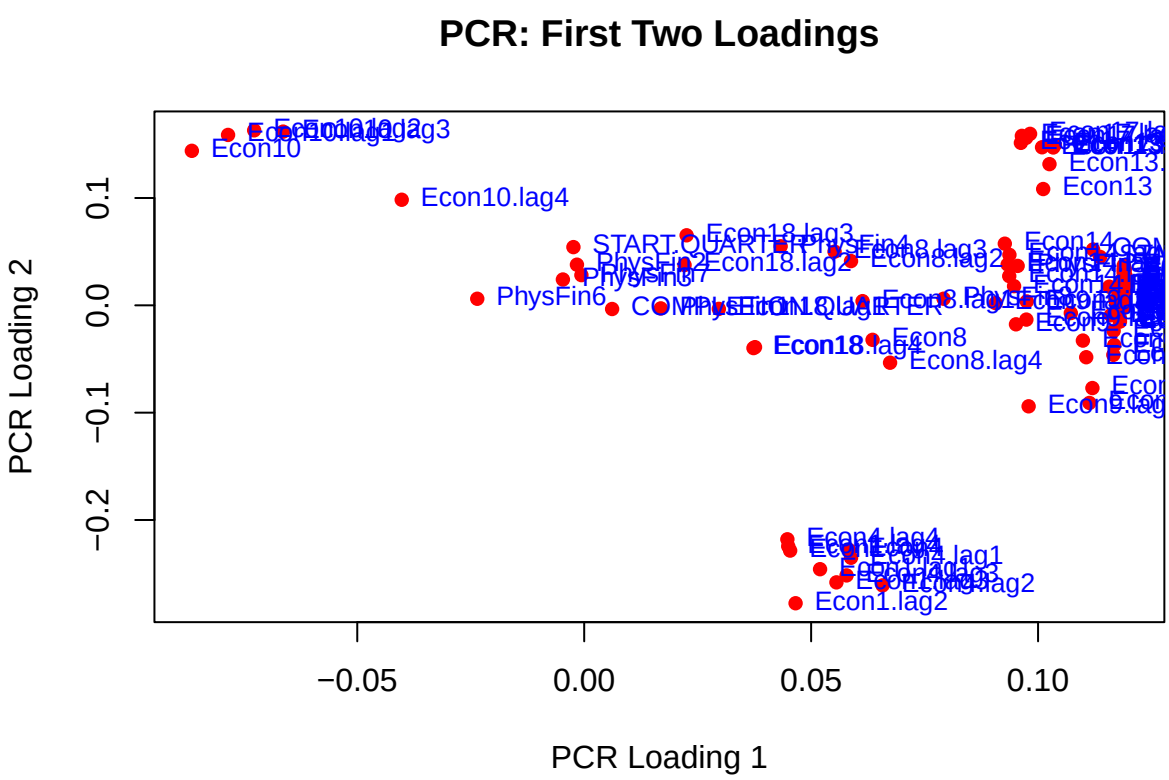
- PLS Test RMSE: 0.2749538
- PCR Test RMSE: 0.2582652

3)

```
pcr_scores <- pcr_fit$scores
plot(pcr_scores[, 1], pcr_scores[, 2],
     xlab = "PCR Score 1", ylab = "PCR Score 2",
     main = "PCR: First Two Scores",
     col = "blue", pch = 16)
```




```
pcr_loadings <- pcr_fit$loadings
plot(pcr_loadings[, 1], pcr_loadings[, 2],
     xlab = "PCR Loading 1", ylab = "PCR Loading 2",
     main = "PCR: First Two Loadings",
     col = "red", pch = 16)
text(pcr_loadings[, 1], pcr_loadings[, 2], labels = rownames(pcr_loadings),
     pos = 4, cex = 0.8, col = "blue")
```



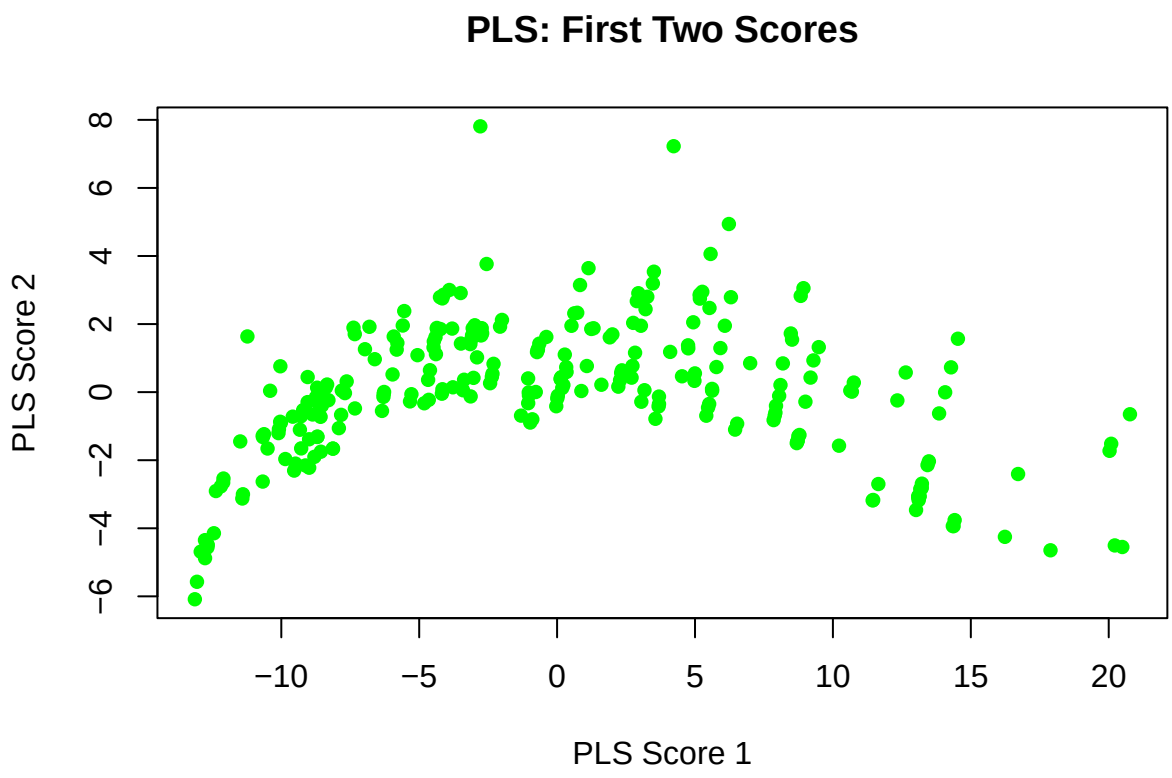
The score-scatterplot shows the first two scores (the first two principal components) of the data in Principal Component Regression. The points in the plot represent the projections of the data along the first and second principal components.

- The curved structure of the points in the plot indicates that the data have a clear, non-linear relationship along the first two principal components. This structure suggests that these components capture the primary variability in the data.
- The distribution along the curve implies that the data may contain a non-linear pattern, which is well represented by these two principal components.

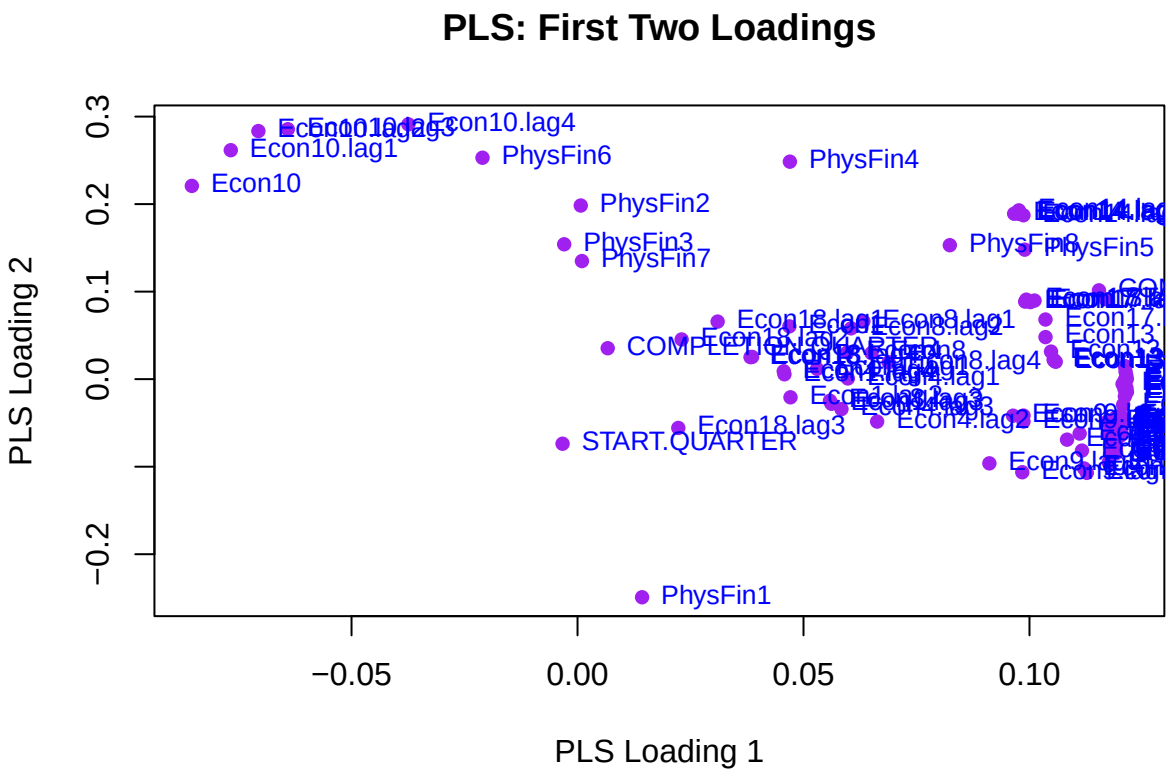
The second plot, we see the first two loadings for Principal Component Regression. The loadings come from the matrix V, which represents the weights or coefficients indicating how strongly each original variable contributes to each principal component.

- The loadings show the extent to which each original variable (column in the original matrix X) projects onto the first and second principal components.
- The points in the scatterplot represent the weight of each variable along the first two principal components. Variables with high positive or negative loadings have a stronger influence on the respective principal component.

```
pls_scores <- pls_fit$scores
plot(pls_scores[, 1], pls_scores[, 2],
     xlab = "PLS Score 1", ylab = "PLS Score 2",
     main = "PLS: First Two Scores",
     col = "green", pch = 16)
```




```
# PLS Loadings Scatterplot (First Two Components)
pls_loadings <- pls_fit$loadings
plot(pls_loadings[, 1], pls_loadings[, 2],
     xlab = "PLS Loading 1", ylab = "PLS Loading 2",
     main = "PLS: First Two Loadings",
     col = "purple", pch = 16)
text(pls_loadings[, 1], pls_loadings[, 2], labels = rownames(pls_loadings),
     pos = 4, cex = 0.8, col = "blue")
```



The first plot, we see the first two scores from the T matrix in Partial Least Squares regression.

- The T matrix contains the PLS scores, which are linear combinations of the original variables X created to capture both high variance and a strong correlation with the target variable y.
- The scatterplot of the first two PLS scores (the first two columns of the T matrix) shows how the data is distributed along the PLS components.
- The distribution and pattern in this plot provide insights into how well these two PLS components capture the relationship with the target variable y.

The scond plot shows the first two loading vectors from the W matrix in Partial Least Squares regression.

- The W matrix contains the PLS loadings, which represent the weights assigned to each original variable to construct the PLS components. These loadings indicate the contribution of each variable to the first two PLS components.
- The points in this plot, represent how strongly each original variable influences the first and second PLS components.
- Variables with loadings farther from zero have a stronger influence on the respective component, while those closer to zero contribute less.