

Introduction to Machine Learning

Lab 3

Rasmus Holm

2016-11-24

Contents

Assignment 1	2
1	2
2	2
3	2
4	2
5	2
Assignment 2	3
1	3
2	5
3	6
4	8
Appendix	9
Code for Assignment 1	9
Code for Assignment 2	9

Assignment 1

1

2

3

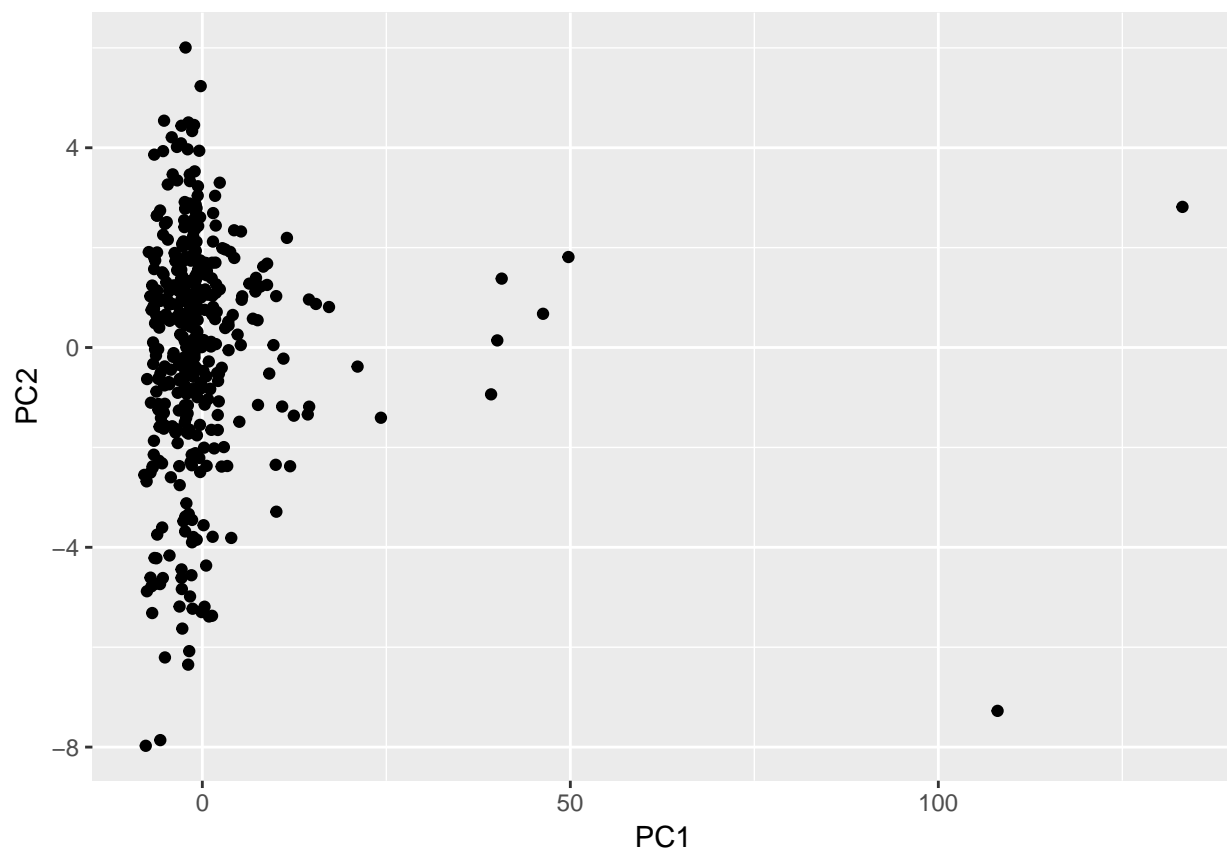
4

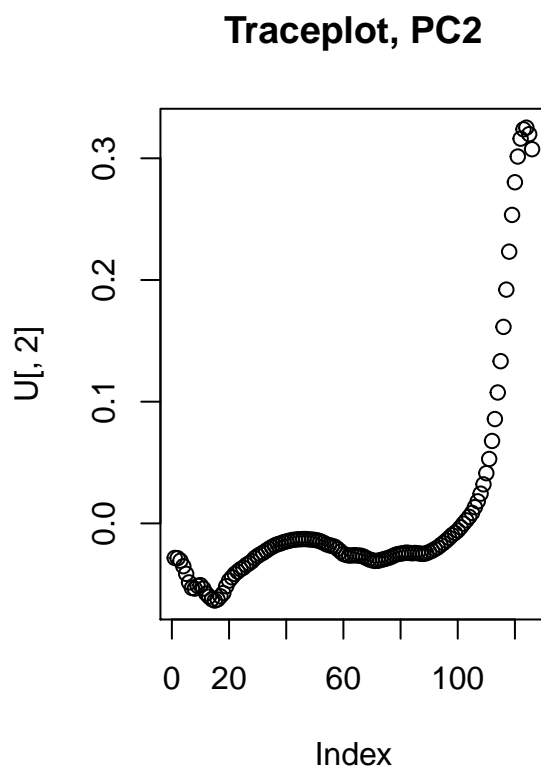
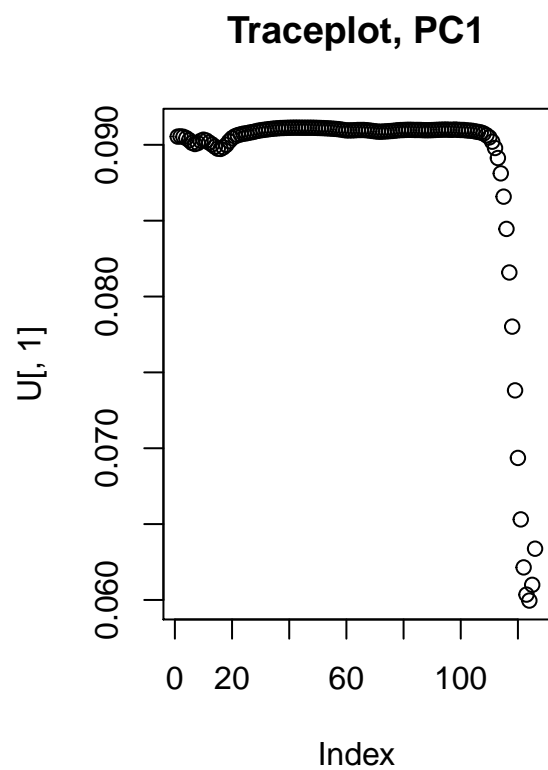
5

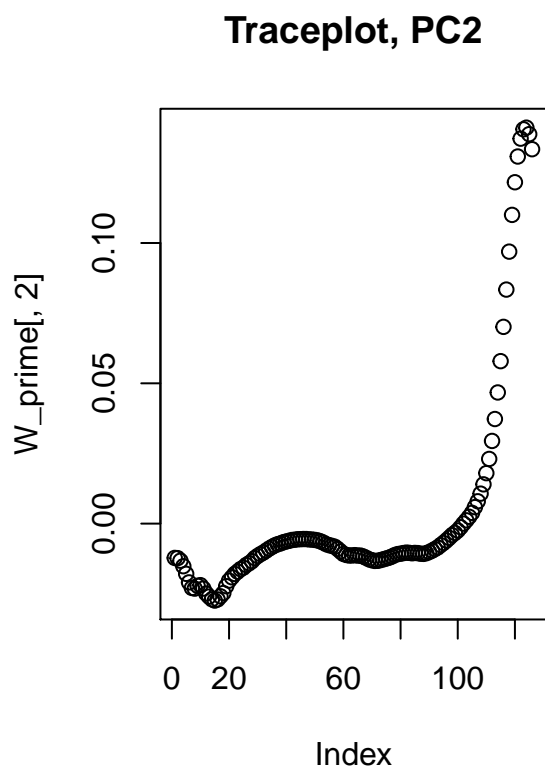
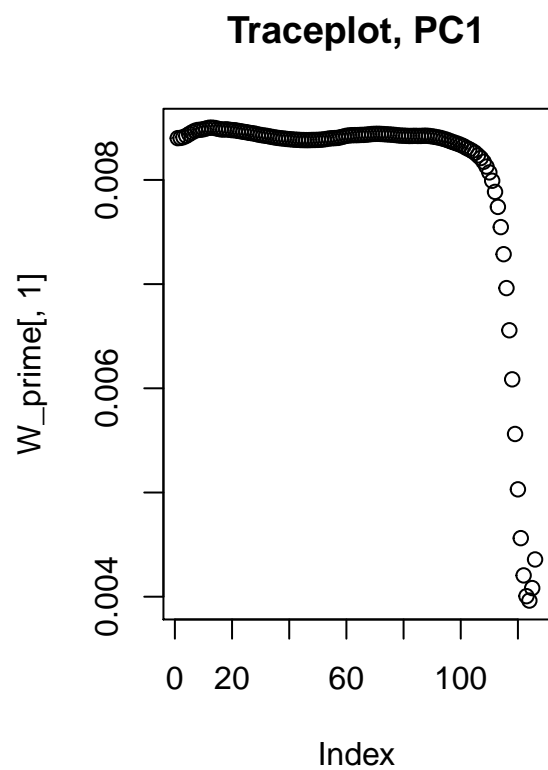
Assignment 2

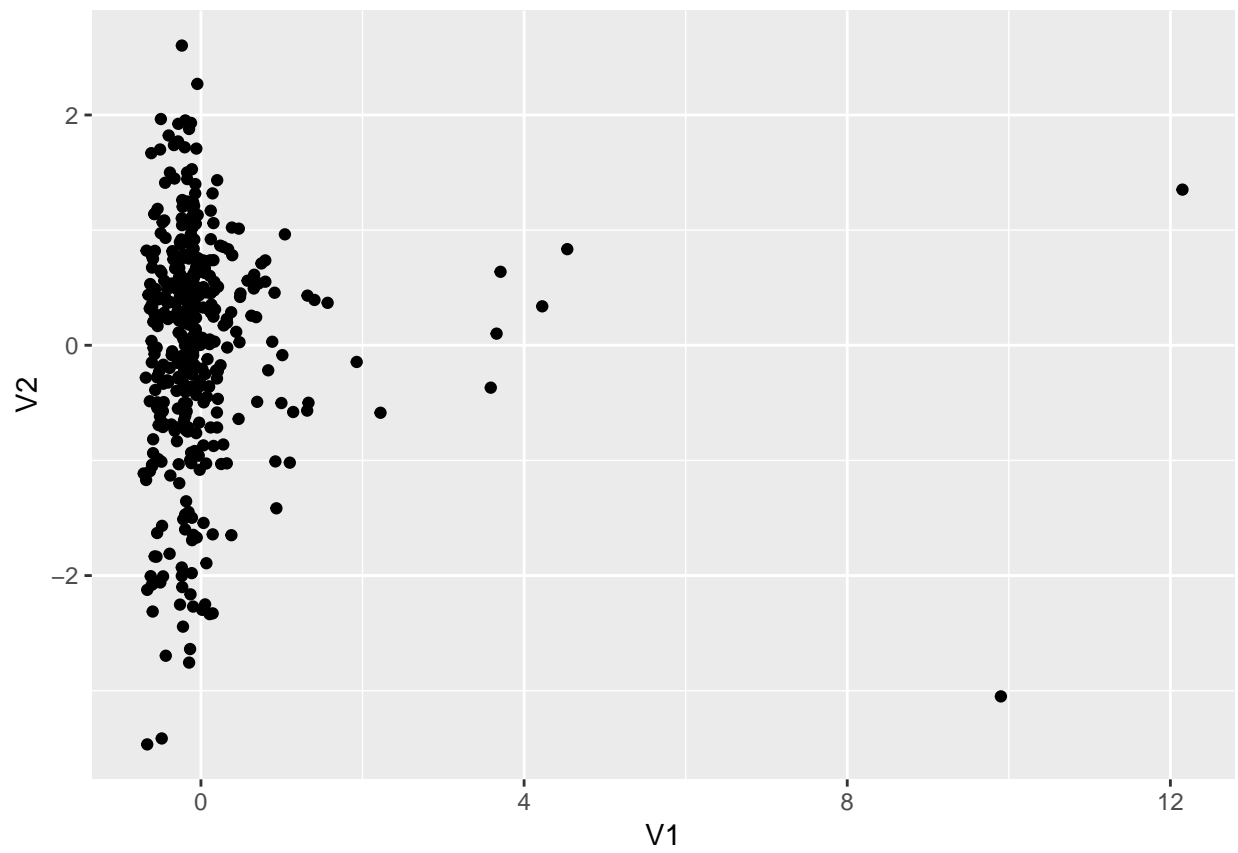
1

```
#> [1] "95.385" "4.229" "0.186" "0.084" "0.072" "0.021" "0.007"
#> [8] "0.003" "0.003" "0.002" "0.001" "0.001" "0.001" "0.001"
#> [15] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [22] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [29] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [36] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [43] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [50] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [57] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [64] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [71] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [78] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [85] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [92] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [99] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [106] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [113] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [120] "0.000" "0.000" "0.000" "0.000" "0.000" "0.000" "0.000"
#> [1] "0.954" "0.996" "0.998" "0.999" "1.000" "1.000" "1.000" "1.000"
#> [9] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [17] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [25] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [33] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [41] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [49] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [57] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [65] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [73] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [81] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [89] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [97] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [105] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [113] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
#> [121] "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000" "1.000"
```

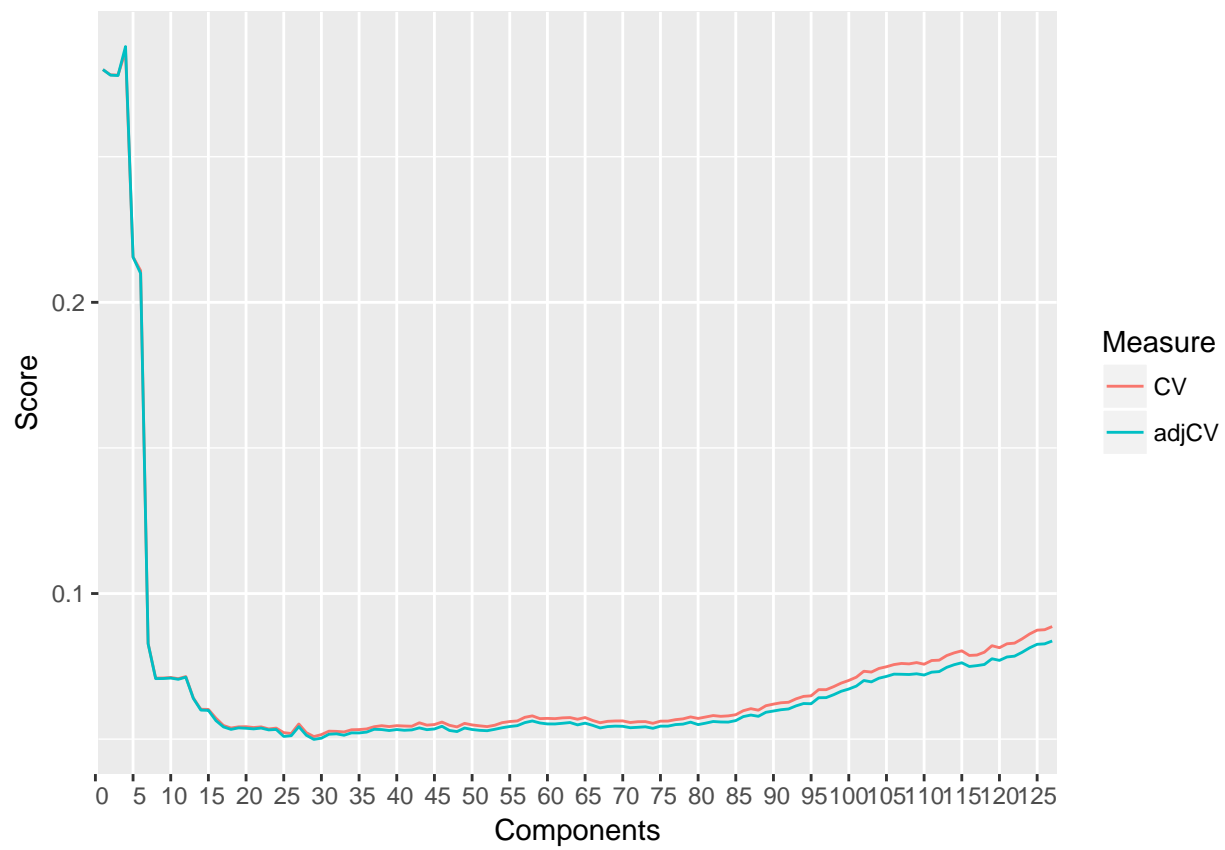








4



Appendix

Code for Assignment 1

Code for Assignment 2

```
library(ggplot2)
library(fastICA)
library(pcaMethods)
library(pls)
library(reshape2)

data <- read.csv2("../data/NIRSpectra.csv")

X <- scale(data[, -ncol(data)])
y <- data[, ncol(data)]

pca <- prcomp(X)

## Eigenvalues
lambda <- pca$sdev^2
variances <- lambda / sum(lambda)

var99_comp_count <- which.max(cumsum(variances * 100) > 99)
components <- as.data.frame(pca$x[, 1:var99_comp_count])

U <- pca$rotation
par(mfrow=c(1, 2))
plot(U[, 1], main="Traceplot, PC1")
plot(U[, 2], main="Traceplot, PC2")

set.seed(12345)
ica <- fastICA(X, var99_comp_count, alg.typ = "parallel", fun = "logcosh", alpha = 1,
               method = "R", row.norm = FALSE, maxit = 200, tol = 1e-06, verbose = FALSE)

W_prime <- ica$K %*% ica$W
components <- as.data.frame(ica$S)
par(mfrow=c(1, 2))
plot(W_prime[, 1], main="Traceplot, PC1")
plot(W_prime[, 2], main="Traceplot, PC2")
ggplot(components) +
  geom_point(aes(x=V1, y=V2))

set.seed(12345)
pcrfit <- pcr(Viscosity ~ ., data=data, scale=TRUE)
cvpcrfit <- crossval(pcrfit, segments=10, segment.type="random")
cv_scores <- t(matrix(MSEP(cvpcrfit)$val, nrow=2))
plot_data <- data.frame(cbind(1:ncol(data), cv_scores))
colnames(plot_data) <- c("Components", "CV", "adjCV")
plot_data <- melt(plot_data, id="Components", variable.name="Measure", value.name="Score")
```

```
xlimits <- seq(0, ncol(data), by=5)

ggplot(plot_data) +
  geom_line(aes(x=Components, y=Score, color=Measure)) +
  scale_x_discrete(limits=xlimits)
```