Multivariate Statistical Analysis - Exercise Session 3

24.01.2022

Problem 1: Principal component analysis

First we read the data.

```
decat <- read.table("DECATHLON.txt", header = TRUE, sep = "\t", row.names = 1)
```

Data includes results of 48 decathletes (“kymmenottelija” in Finnish). We remove variables `Points`, `Height` and `Weight` from the analysis.

```
decat <- decat[, -c(1, 12, 13)]
head(decat)
```

```
##          R100m Long_jump Shot_put High_jump R400m Hurdles Discus_throw
## Skowrone   853       931      725       857   838     903          772
## Hedmark    853       853      814       769   833     914          855
## Le_Roy     879       951      799       779   838     881          819
## Zeilbaue   826       931      793       865   875     891          729
## Zigert     879       840      924       857   788     892          866
## Bennett    905       859      647       779   938     859          651
##          Pole_vault Javelin R1500m
## Skowrone        981     818    528
## Hedmark         884     975    438
## Le_Roy         1028     758    408
## Zeilbaue        909     774    543
## Zigert          920     671    497
## Bennett        1028     794    661
```

```
dim(decat)
```

```
## [1] 48 10
```

Next let us visualize correlation matrix. Correlation matrix can be visualized, for example, with heat map. Figure 1 show one way for making heatmap of correlation matrix.

```
#install.packages("corrgram") # Install package corrgram
library(corrgram)
corrgram(decat, upper.panel = NULL)
```

Package `corrgram` gives various different options for visualizing correlation matrix. In Figure 2 we use pie charts instead of colored panels.

```
colors <- c("blue4", "blue3", "blue2", "blue1", "blue", "red", "red1",
            "red2","red3", "red4")
corrgram(decat, lower.panel = panel.pie, diag.panel = panel.minmax,
         upper.panel = NULL, col.regions = colorRampPalette(colors))
```

One can also use base R for plotting heatmap as in Figure 3.

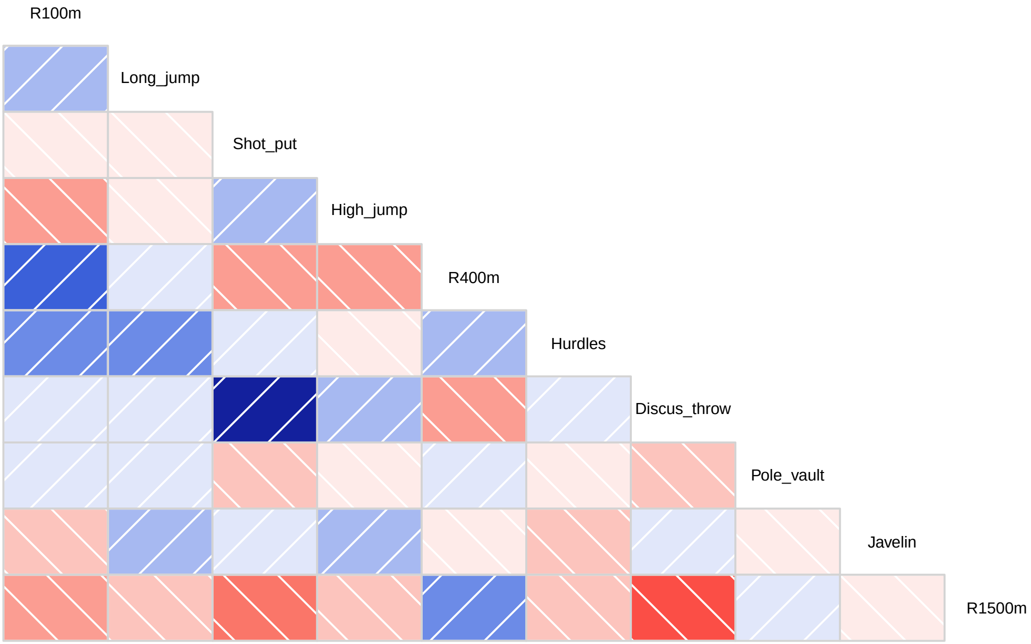

Figure 1: Heatmap of correlation matrix with `corrgram` package using colored panels.

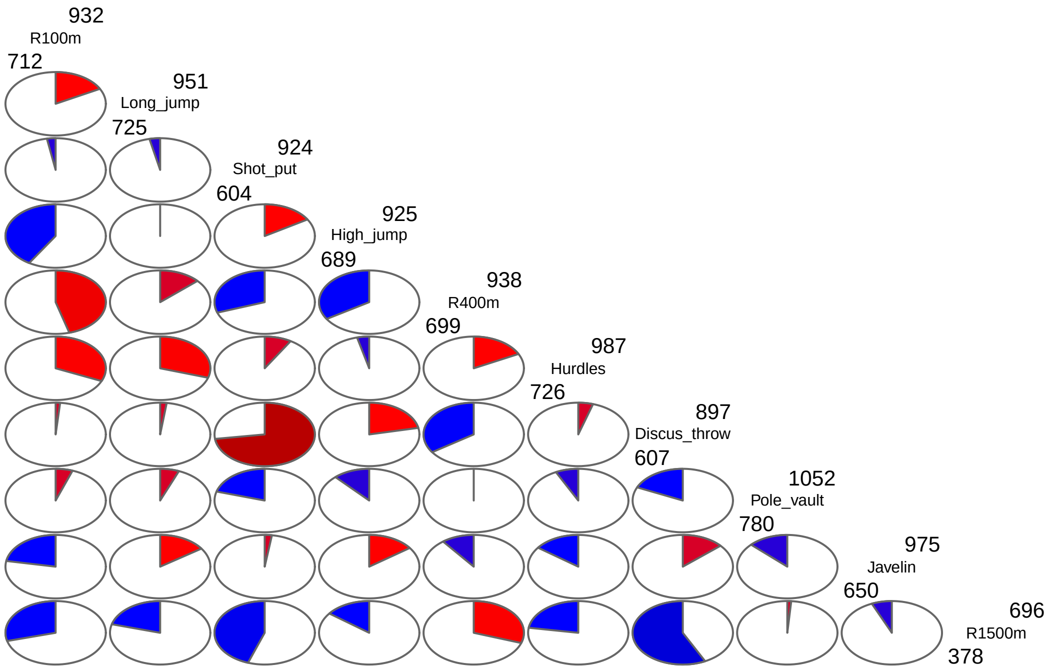

Figure 2: Heatmap of correlation matrix with `corrgram` package using pie charts.

```
heatmap(cor(decat), Rowv = NA, Colv = NA, symm = T,
        col = colorRampPalette(c('red','white','blue'))(50))
```

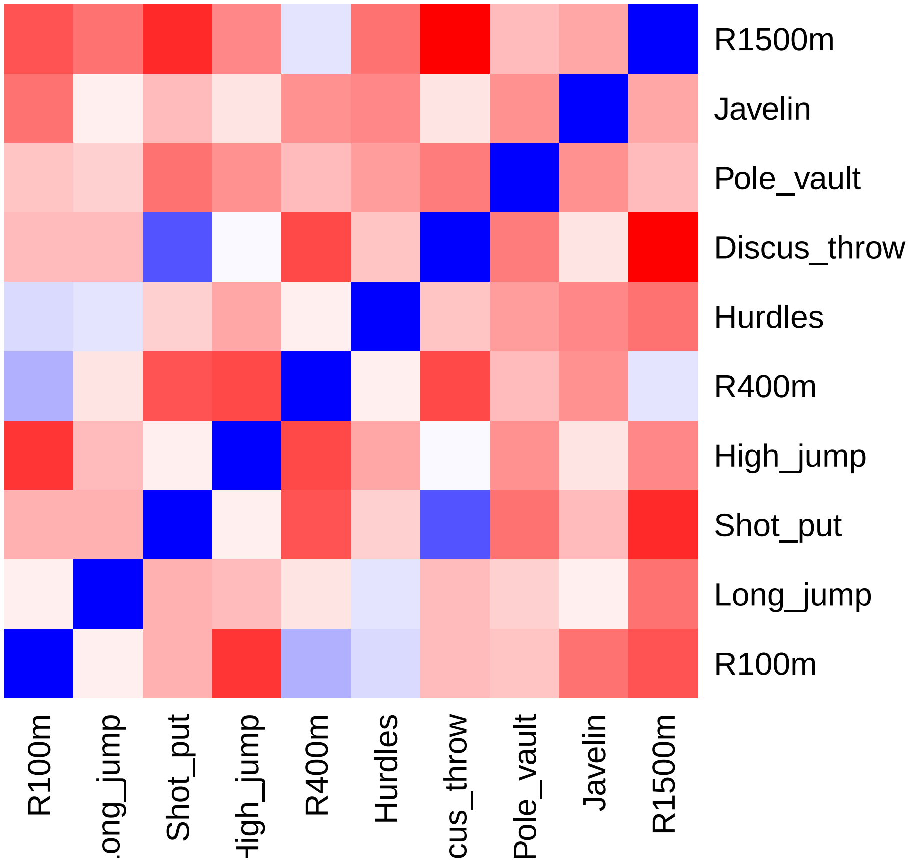

Figure 3: Heatmap of correlation matrix with base R.

a) How much variation is explained by k principal components?

First, we perform PCA with correlation matrix.

```
decat_pca <- princomp(decat, cor = TRUE)
```

Summary tells how much variation is explained by k principal components.

```
summary(decat_pca)
```

```
## Importance of components:
##                           Comp.1    Comp.2   Comp.3    Comp.4     Comp.5
## Standard deviation     1.6130891 1.4169733 1.098463 1.0329820 0.96516226
## Proportion of Variance 0.2602056 0.2007813 0.120662 0.1067052 0.09315382
## Cumulative Proportion  0.2602056 0.4609870 0.581649 0.6883542 0.78150800
##                            Comp.6     Comp.7     Comp.8     Comp.9    Comp.10
## Standard deviation     0.77116160 0.75437360 0.73395022 0.49482809 0.48745515
## Proportion of Variance 0.05946902 0.05690795 0.05386829 0.02448548 0.02376125
## Cumulative Proportion  0.84097702 0.89788497 0.95175326 0.97623875 1.00000000
```

Figure 4 shows a visualization about how much variation is explained by k principal components.

```
vars <- decat_pca$sdev^2
var_prop <- vars / sum(vars)
var_prop_cum <- cumsum(var_prop)

plot(var_prop_cum, type = "b", pch = 21, lty = 3, bg = "skyblue", cex = 1.5,
     ylim = c(0, 1), xlab = "Principal component",
     ylab = "Cumulative proportion of variance explained",
     xaxt = "n", yaxt = "n")
axis(1, at = 1:10)
axis(2, at = 0:10 / 10, las = 2)
```

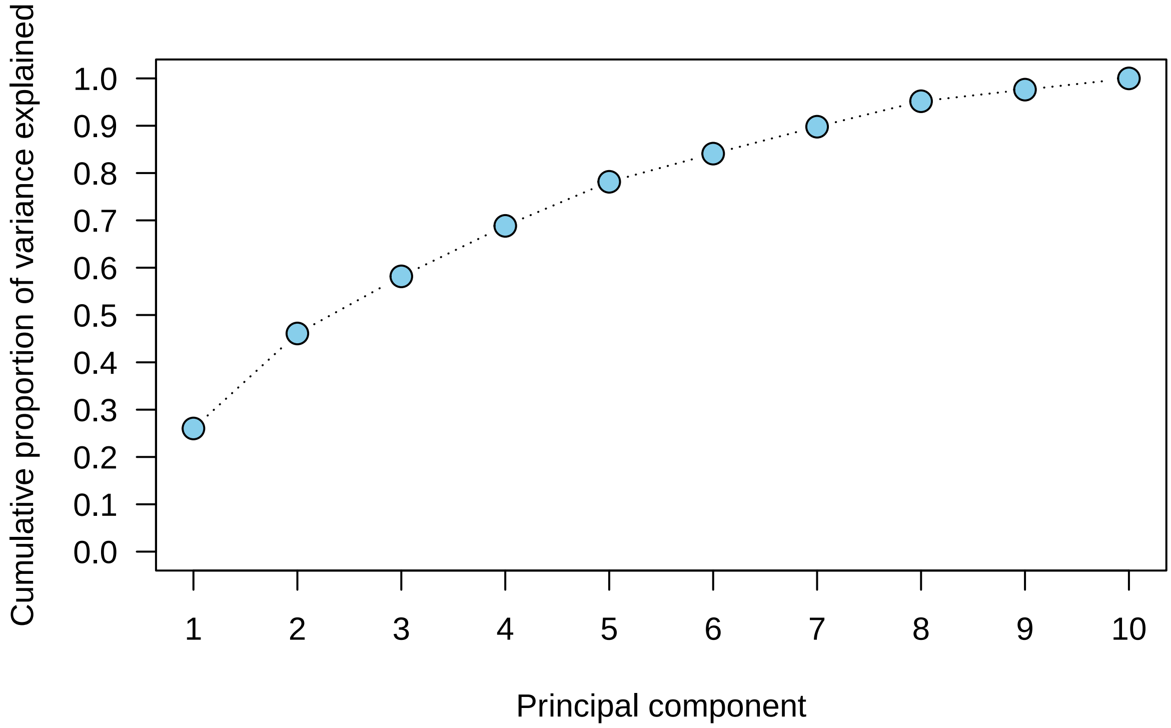

Figure 4: Cumulative proportion of variance explained by k principal components.

b) Interpreting principal components

```
pc12 <- decat_pca$scores[, 1:2]
load12 <- decat_pca$loadings[, 1:2]
pc_axis <- c(-max(abs(pc12)), max(abs(pc12)))
ld_axis <- c(-0.8, 0.8)

plot(pc12, xlim = pc_axis, ylim = pc_axis, pch = 21, bg = 8, cex = 1.25,
     xlab = paste0("PC 1 (", round(100 * var_prop[1], 2), "%)"),
     ylab = paste0("PC 2 (", round(100 * var_prop[2], 2), "%)"))
par(new = T)
```

```
plot(load12, axes = F, type = "n", xlab = "", ylab = "", xlim = ld_axis,
     ylim = ld_axis)
axis(3, col = 2)
axis(4, col = 2)
arrows(0, 0, load12[, 1], load12[, 2], length = 0.1, col = 2)
text(load12[, 1], load12[, 2], rownames(load12), pos = 3)
abline(h = 0, lty = 3)
abline(v = 0, lty = 3)
```

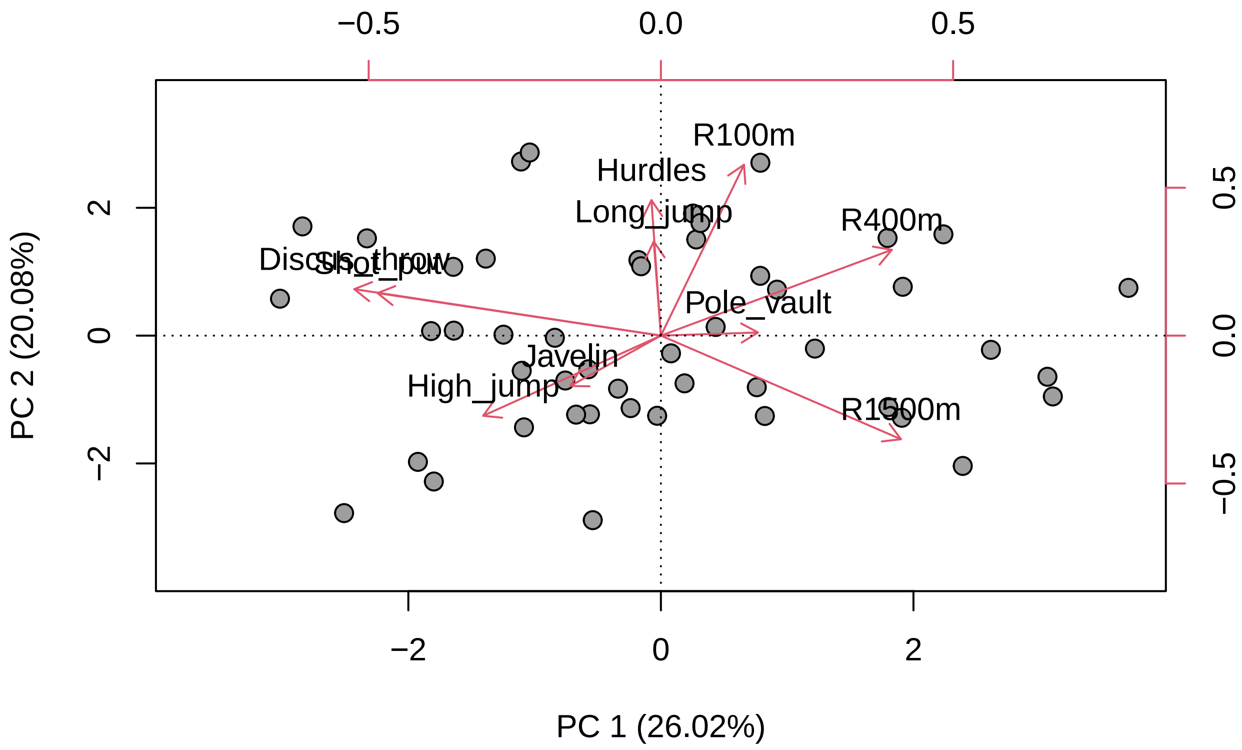

Figure 5: Biplot of scores and loadings.

By looking at loadings one can interpret principal components

```
round(decat_pca$loadings[, 1:4], 2)
```

```
##              Comp.1 Comp.2 Comp.3 Comp.4
## R100m          0.14   0.58   0.15   0.03
## Long_jump     -0.01   0.32  -0.65  -0.21
## Shot_put      -0.48   0.14   0.24   0.13
## High_jump     -0.30  -0.27  -0.27  -0.07
## R400m          0.40   0.29  -0.08   0.32
## Hurdles       -0.02   0.46  -0.19   0.07
## Discus_throw  -0.52   0.16   0.14   0.05
## Pole_vault     0.17   0.01   0.08  -0.86
## Javelin       -0.16  -0.17  -0.60   0.15
## R1500m         0.41  -0.35   0.00   0.25
```

For example, possible interpretation for the first component is strength. Interpretations for first four principal components are same as in Session 2.

c) PCA and outlier

Now, let's add outlier to the original data.

```r
s <- 9
decat[49, ] <- c(rep(1200, s), rep(18000, 10 - s))
rownames(decat)[49] <- "outlier"
```

Figure 6 visualizes contaminated data.

```r
pairs(decat, gap = 0, upper.panel = NULL)
```

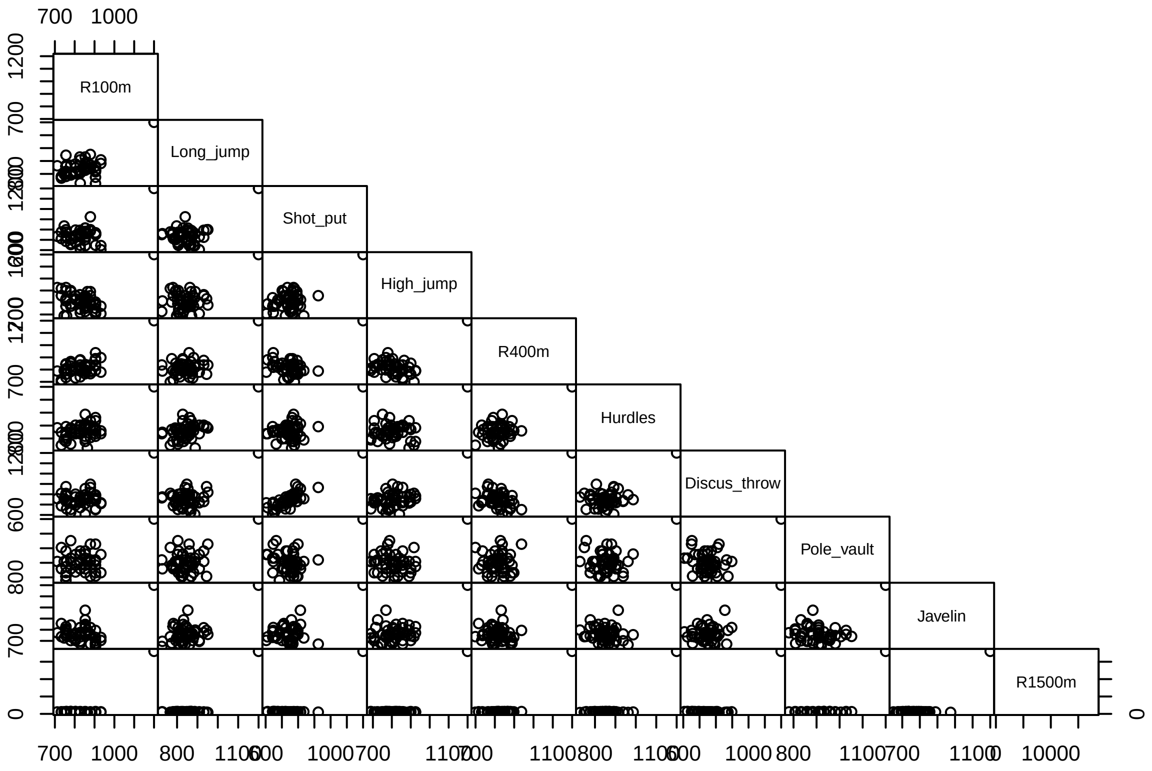

Figure 6: Pairwise scatter plots of contaminated data.

Now let's perform PCA for contaminated data

```r
decat_pca <- princomp(decat, cor = TRUE)
vars <- decat_pca$sdev^2
var_prop <- vars / sum(vars)
```

Figure 7 shows that first principal component detects outlier the best. Also, it can be seen that PCA is quite nonrobust method. That is, outliers have significant effect to the results of PCA.

```r
pc12 <- decat_pca$scores[, 1:2]
load12 <- decat_pca$loadings[, 1:2]
```

```
pc_axis <- c(-max(abs(pc12)), max(abs(pc12)))
ld_axis <- c(-0.8, 0.8)

plot(pc12, xlim = pc_axis, ylim = pc_axis, pch = 21, bg = c(rep(8,48), 1),
     cex = 1.25, xlab = paste0("PC 1 (", round(100 * var_prop[1], 2), "%)"),
     ylab = paste0("PC 2 (", round(100 * var_prop[2], 2), "%)"))
par(new = T)
plot(load12, axes = F, type = "n", xlab = "", ylab = "", xlim = ld_axis,
     ylim = ld_axis)
axis(3, col = 2)
axis(4, col = 2)
arrows(0, 0, load12[, 1], load12[, 2], length = 0.1, col = 2)
text(load12[, 1], load12[, 2], rownames(load12), pos = 3)
abline(h = 0, lty = 3)
abline(v = 0, lty = 3)
```

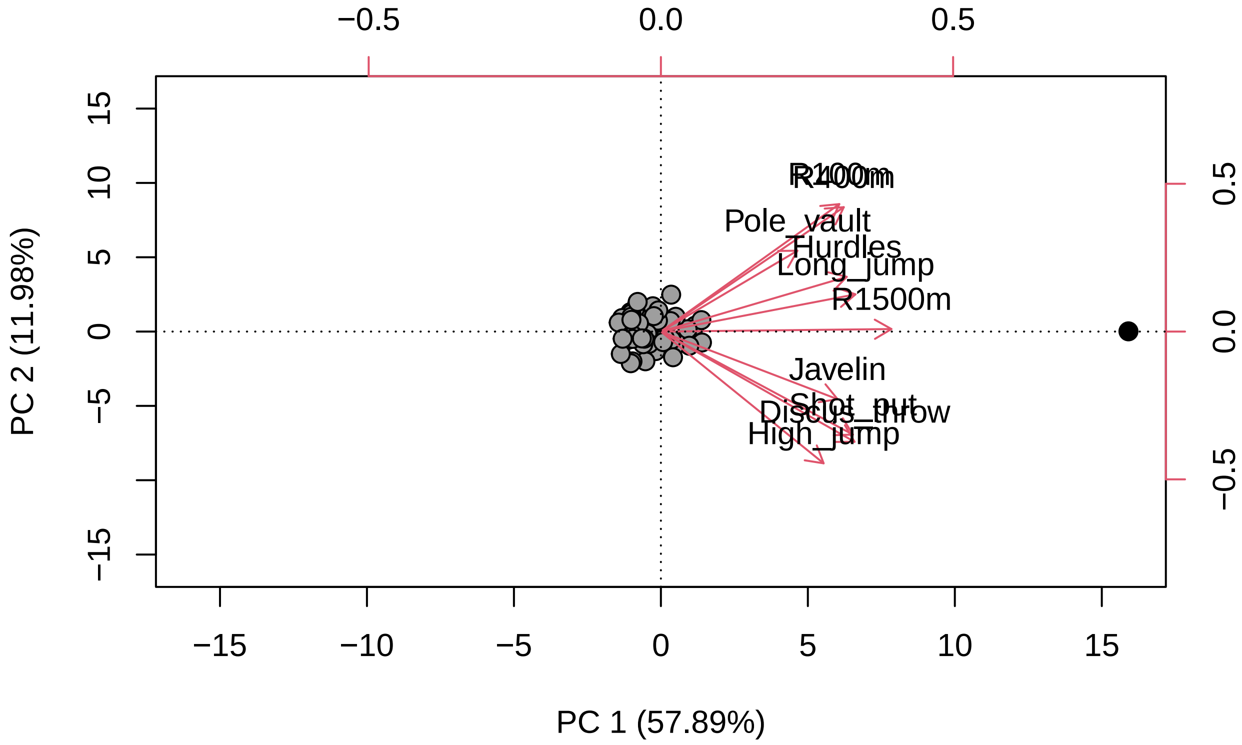

Figure 7: Biplot of scores and loadings for contaminated data with respect to 1st and 2nd components. Score of the outlier is colored as black.

Figure 8 shows that outlier is not as well detected by 2nd or 3rd principal components.

```
pc23 <- decat_pca$scores[, 2:3]
load23 <- decat_pca$loadings[, 2:3]
pc_axis <- c(-max(abs(pc23)), max(abs(pc23)))
ld_axis <- c(-0.8, 0.8)
```

```
plot(pc23, xlim = pc_axis, ylim = pc_axis, pch = 21, bg = c(rep(8,48), 1),
     cex = 1.25, xlab = paste0("PC 2 (", round(100 * var_prop[2], 2), "%)"),
     ylab = paste0("PC 3 (", round(100 * var_prop[3], 2), "%)"))
text(pc23["outlier", 1], pc23["outlier", 2], labels = "outlier", pos = 2)
par(new = T)
plot(load23, axes = F, type = "n", xlab = "", ylab = "", xlim = ld_axis,
     ylim = ld_axis)
axis(3, col = 2)
axis(4, col = 2)
arrows(0, 0, load23[, 1], load23[, 2], length = 0.1, col = 2)
text(load23[, 1], load23[, 2], rownames(load23), pos = 3)
abline(h = 0, lty = 3)
abline(v = 0, lty = 3)
```

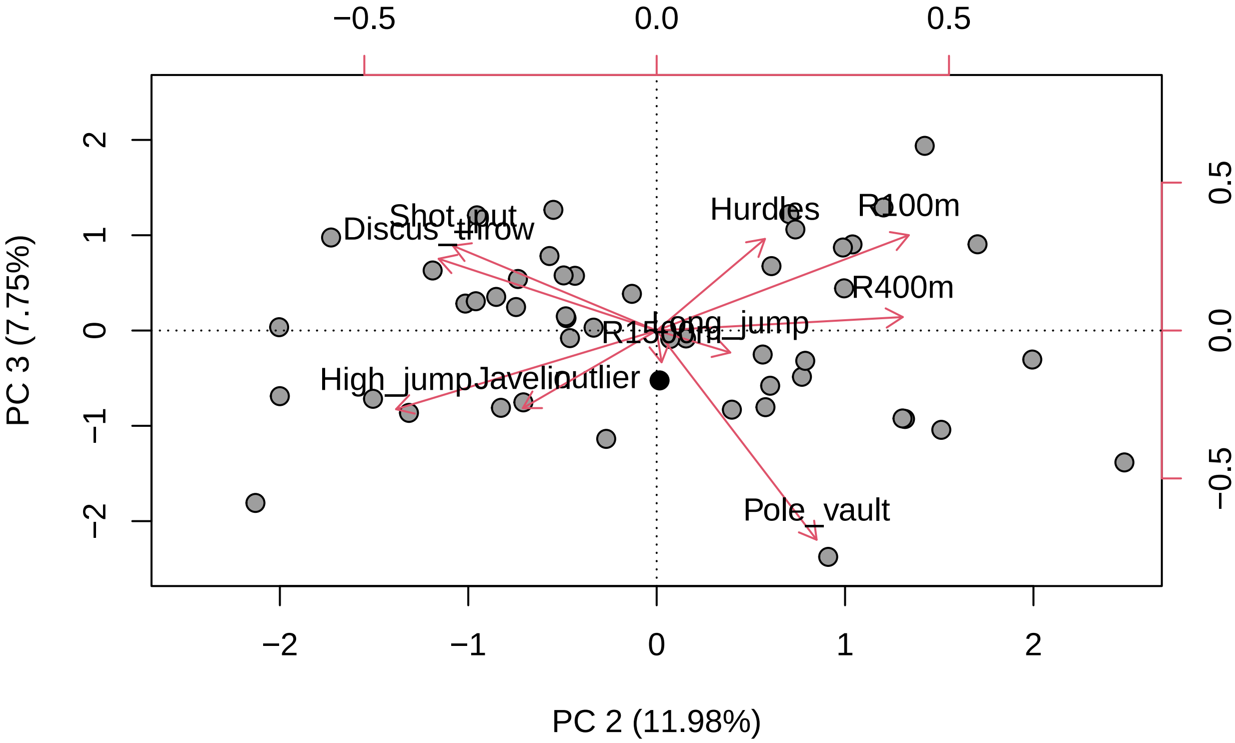

Figure 8: Biplot of scores and loadings for contaminated data with respect to 2nd and 3rd components. Score of the outlier is colored as black.

Problem 2: Affine equivariance

a)

In the following, let X denote a $n \times p$ data matrix of n i.i.d. p-variate observations $x_1, x_2, \dots, x_n$ from some continuous distribution with a finite covariance matrix Σ. Furthermore, consider the transformation,

$$y_i = Ax_i + b,$$

where A is a nonsingular $p \times p$ matrix A and b is a p-variate location vector b.

Proposition 1. *Sample mean $T(\cdot)$ is affine equivariant. In other words, if you transform your data $X \to Y$ such that*

$$y_i = Ax_i + b,$$

then

$$T(Y) = AT(X) + b,$$

for all nonsingular $p \times p$ matrices A and for all p-vectors b.

Proof of Proposition 1. Remember that

$$T(X) = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

Then

$$T(Y) = \frac{1}{n}\sum_{i=1}^{n}(Ax_i + b) = A\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right) + \frac{1}{n}nb = AT(X) + b.$$

□

b)

Proposition 2. *Sample covariance matrix $S(\cdot)$ is affine equivariant. In other words, if you transform your data $X \to Y$ such that*

$$y_i = Ax_i + b,$$

then

$$S(Y) = AS(X)A^T,$$

for all nonsingular $p \times p$ matrices A and for all p-vectors b.

Proof of Proposition 2. Remember that

$$T(X) = \frac{1}{n-1}\sum_{i=1}^{n}\left(x_i - T(X)\right)\left(x_i - T(X)\right)^T.$$

Then

$$\begin{aligned}
S(Y) &= \frac{1}{n-1}\sum_{i=1}^{n}\left(Ax_i + b - T(Y)\right)\left(Ax_i + b - T(Y)\right)^T \\
&= \frac{1}{n-1}\sum_{i=1}^{n}\left(Ax_i + b - (AT(X) + b)\right)\left(Ax_i + b - (AT(X) + b)\right)^T \\
&= \frac{1}{n-1}\sum_{i=1}^{n}\left(A(x_i - T(X))\right)\left(A(x_i - T(X))\right)^T \\
&= A\left(\frac{1}{n-1}\sum_{i=1}^{n}\left(x_i - T(X)\right)\left(x_i - T(X)\right)^T\right)A^T \\
&= AS(X)A^T.
\end{aligned}$$

□