:(

Contents

1 LAB 1:

2 LAB 2:

3 LAB 3: PRINCIPAL COMPONENT ANALYSIS

3.1 PCA SCHEME

To perform principal component analysis, a schematic approach to follow is:

1. INITIAL IMPORT

- import dataset
- separate numeric and categorical data, perform next steps on **numeric part**
- visual **boxplot** exploration -> **scaling** if ranges vary too much to avoid masking

2. PERFORM PCA

- use **princomp** command
- barplot the percentage of **var explained** by each principal component
- plot the **loadings** of the PCs and try to give an interpretation

3. ADDITIONAL EXPLORATION

- plot the transformed data (scores) in the first 2/3 principal components
- **projection** on space generated by k (or first k-th) principal component(s)
- biplot

3.2 IMPORTANT FUNCTIONS

```
scale(d_numeric)
princomp(d_numeric, scores=T)
boxplot(scale(x, center=T, scale=F), col='gold')
Boxplot(..., id.method='y') #same as boxplot but shows outliers
biplot(pca, scale=0, cex=0.7)
```

3.3 CODE

3.3.1 Import and visual exploration

Import of a numerical dataset with only 4 numerical columns.

PACKAGES USED: `library(car)`

```
dataset <- read.table(here::here('dataset','dataset_pca.txt'), header=T)

dim(dataset)
dimnames(dataset)

var.names <- c("I Comp.","II Comp.","III Comp.","IV Comp.")
dimnames(dataset)[[2]] <- var.names

# Scatter plot
pairs(dataset, col=rainbow(dim(dataset)[1]), pch=16, main='Scatter plot')
```

```
M <- sapply(dataset,mean)
S <- cov(dataset)
round(S,digits = 2)
R <- cor(dataset)
round(R,digits = 2)

# Boxplot
x11()
boxplot(dataset, las=1, col='red', main='Boxplot',grid=T)
```

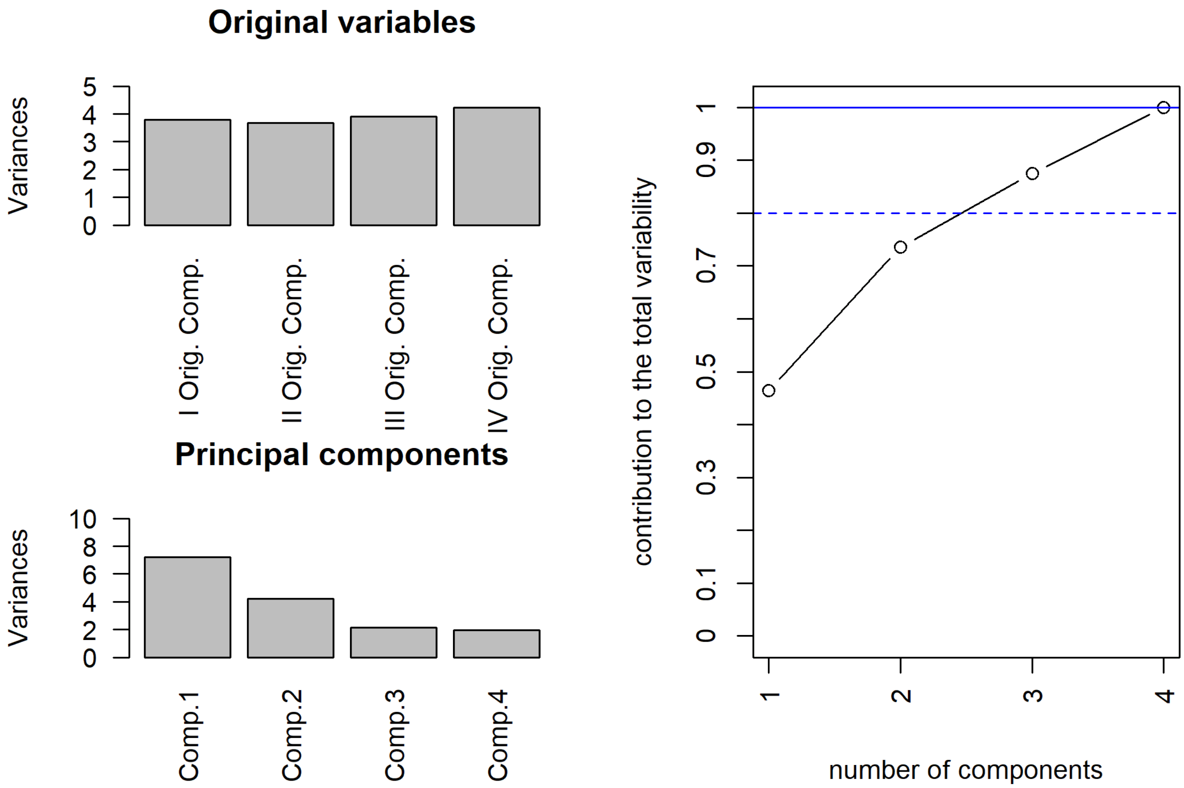
```
# Boxplot with outliers (requires CAR)
x11()
Boxplot(dataset, id.method="y")
```

```
# Matplot + boxplot
x11()
matplot(t(dataset), type='l', axes=F)
box()
boxplot(dataset, add=T, boxwex=0.1, col='red')
```

```
# If variability changes too much with variables
dataset <- scale(dataset)
```

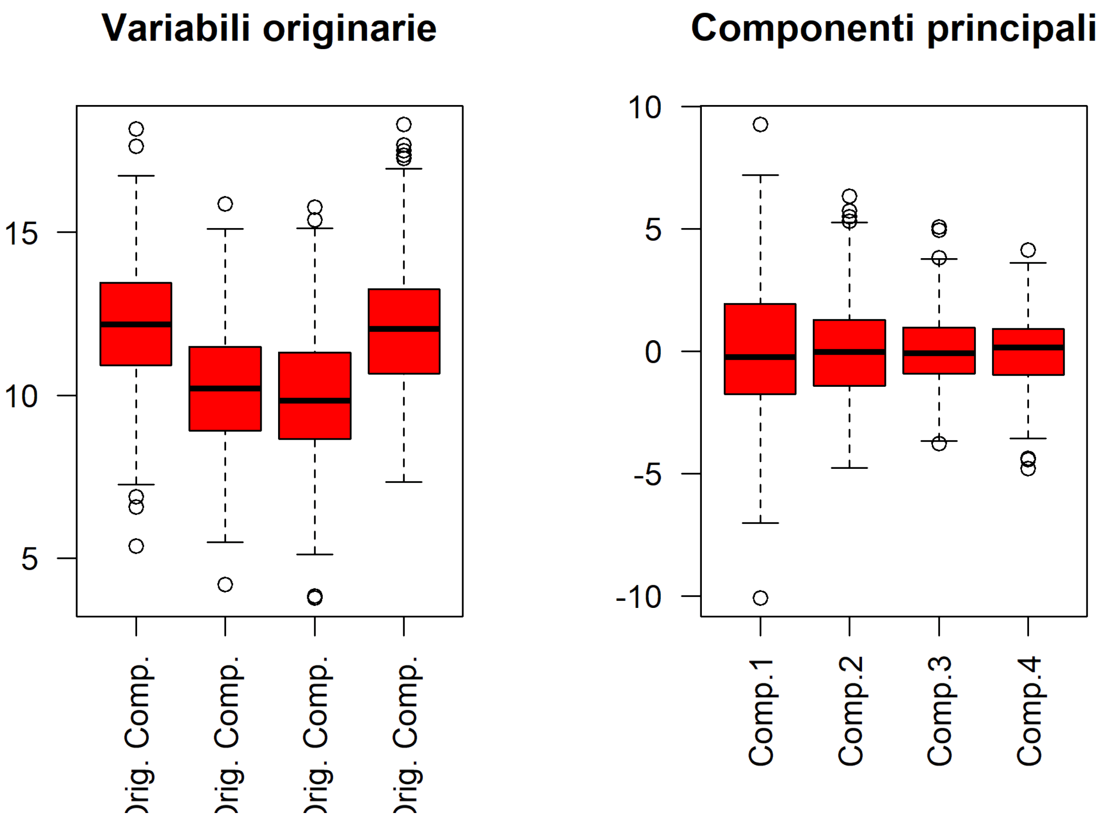
3.3.2 PCA and variability explained plot

```
pca <- princomp(dataset, scores=T)
pca
summary(pca)


# Plot original vs pca vs var explained
# Set ylim in barplots consistently with yout data
x11()
layout(matrix(c(2,3,1,3),2,byrow=T))
barplot(pca$sdev^2, las=2, main='Principal components', ylim=c(0,10), ylab='Variances')
barplot(sapply(dataset,sd)^2, las=2, main='Original variables', ylim=c(0,5), ylab='Variances')
plot(cumsum(pca$sdev^2)/sum(pca$sdev^2), type='b', axes=F, xlab='number of components',
     ylab='contribution to the total variability', ylim=c(0,1))
abline(h=1, col='blue')
abline(h=0.8, lty=2, col='blue')
box()
axis(2,at=0:10/10,labels=0:10/10)
axis(1,at=1:ncol(dataset),labels=1:ncol(dataset),las=2)
```



3.3.3 Scores plot

```
# Scores
scores <- pca$scores

x11()
layout(matrix(c(1,2),1,2))
boxplot(dataset, las=2, col='red', main='Variabili originarie')
boxplot(scores, las=2, col='red', main='Componenti principali')
```

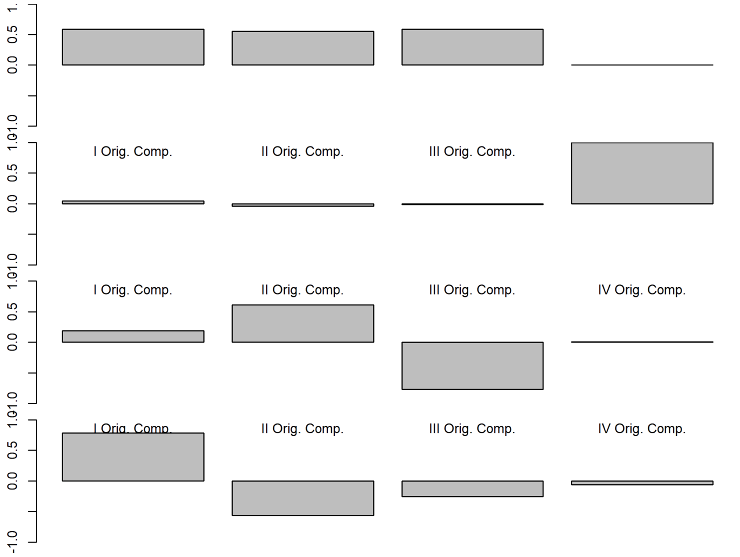


```
x11()
Boxplot(dataset, id.method="y",las=2, col='red', main='Variabili originarie')
```

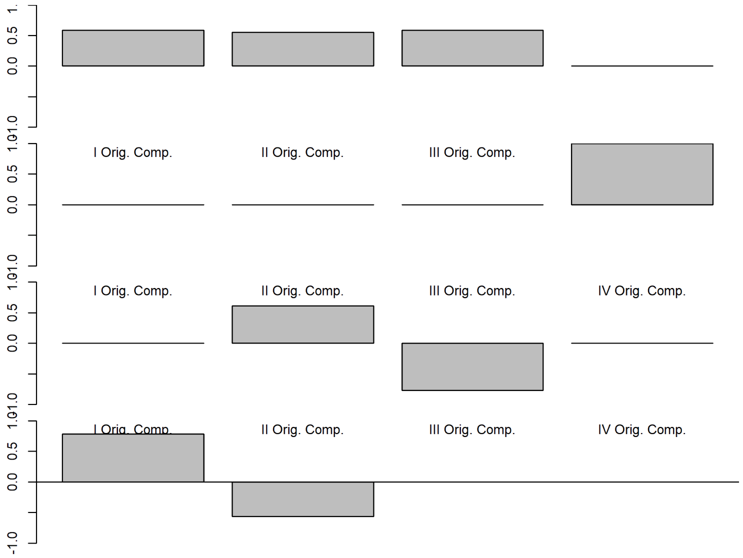
```
Boxplot(scores, id.method="y",las=2, col='red', main='Componenti principali')
```

3.3.4 Loadings interpretation plot

```r
# Loadings
load <- pca$loadings
a = 4 # number of principal components to be interpreted, change accordingly

x11()
par(mar = c(1,4,0,2), mfrow = c(a,1))
for(i in 1:a)
  barplot(load[,i], ylim = c(-1, 1))
```



```r
# filter the most significant loadings
x11()
par(mar = c(1,4,0,2), mfrow = c(a,1))
for(i in 1:a) barplot(ifelse(abs(load[,i]) < 0.3, 0, load[,i]) , ylim = c(-1, 1));abline(h=0)
```

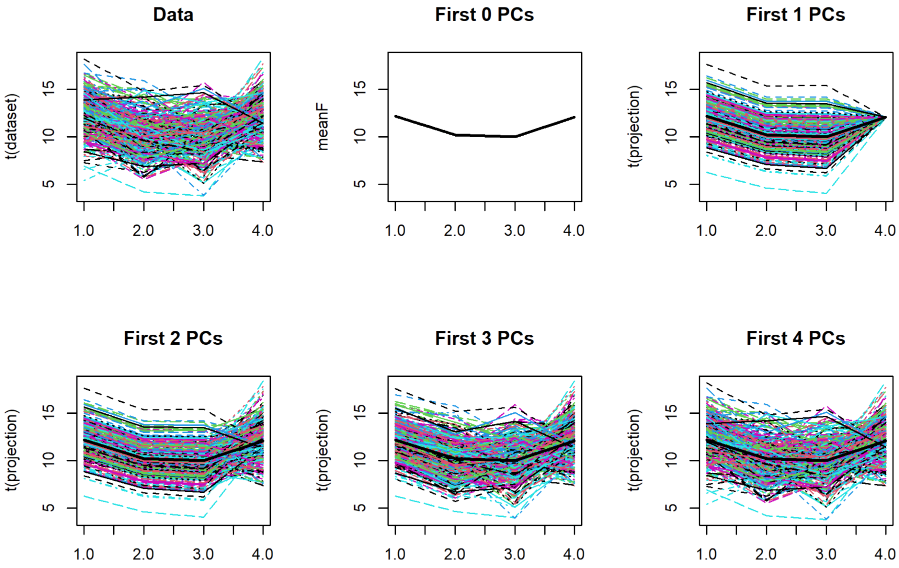


3.3.5 Biplot and projection on space generated by (first) k-(th) PC

```
# Biplot
x11()
biplot(pca, scale=0, cex=.7)
```

```
# Projection on the space generated by the k-th principal component
x11(width=21, height=7)
par(mfrow=c(2,3))
matplot(t(dataset), type='l', main = 'Data', ylim=range(dataset))

meanF <- colMeans(dataset)
matplot(meanF, type='l', main = '0 PC', lwd=2, ylim=range(dataset))
for(i in 1:a)
{
  projection <- matrix(meanF, dim(dataset)[[1]], dim(dataset)[[2]], byrow=T) + scores[,i] %*% t(load[,i
  matplot(t(projection), type='l', main = paste(i, 'PC'), ylim=range(dataset))
  matplot(meanF, type='l', lwd=2, add=T)
}
```

```
# Projection on the space generated by the first k principal components
x11(width=21, height=7)
par(mfrow=c(2,3))
matplot(t(dataset), type='l', main = 'Data', ylim=range(dataset))
meanF <- colMeans(dataset)
matplot(meanF, type='l', main = 'First 0 PCs', lwd=2, ylim=range(dataset))
projection <- matrix(meanF, dim(dataset)[[1]], dim(dataset)[[2]], byrow=T)
for(i in 1:a)
{
  projection <- projection + scores[,i] %*% t(load[,i])
  matplot(t(projection), type='l', main = paste('First', i, 'PCs'), ylim=range(dataset))
  matplot(meanF, type='l', lwd=2, add=T)
}
```



4 LAB 4:

5 LAB 5:

6 LAB 6:

7 LAB 7:

8 LAB 8:

9 LAB 9:

10 LAB 10: