```
# ===== 1 =====

library(leaps)
source("world.R")
x<-world[,3:11]
x<-scale(x,center=TRUE,scale=TRUE)
erg<-princomp(x, cor=TRUE, scores=TRUE)
summary(erg) # we see that we get 9 principal components,
# and the first one describes almost 60% of the variance of our data;
# the second one describes ~13% and all subsequent less and less
plot(erg) # histogramm: x-axis~component, y-axis~the amount of
# variance it describes
erg$loadings # the first principal component does not take variable
# Women into account, and takes ~0.28 to ~0.42 of all the other variables
# for the direction (LifeexpF and LifeexpM have most influence)
# the second component ignores LifeexpF, LifeexpM, InfMort and
# BabyUnderw; Women has the most influence on the direction

sc<-erg$scores
plot(sc[,1:2]) # we see that Comp.1 has very large variance compared to Comp.2
# (we could also read that from summary)
plot(sc[,c(1,3)]) # although this plot shows bigger variety on Comp.3,
# the standard deviation of Comp.3 is smaller than that of Comp.2 because
# the points vary around 0, and in the previous plot they were mostly above 0
y<-world[,1]
plot(sc[,1:2], col=y) # Europe (Continent 1) is concentrated in the lower
# left corner (black), South America is on the left half (green) and Africa
# on the right half (red) of the plot; Asia has a big variance on both axes
plot(sc[,c(1,3)], col=y) # the situation on the x-axis is the same as on the
# previous plot, and on the y-axis (Component 3) all the variables are pretty
# dispersed (except Europe)

# we conclude that Europe has little variety in regard to these variables
# (when we look at components 1 and 2, which represent ~73% of all of the variance)


library(robustbase)
robust_covmat<-covMcd(x)
erg_rob<-princomp(x, cor=TRUE, scores=TRUE, covmat=robust_covmat)

summary(erg)
summary(erg_rob) # comparing these two, we see that robust Component 1 describes
# a little more variance (~63%) than the ordinary (~60%), robust Component 2 as well
# (~16% compared to 13%); the other robust Components describe respectively less
# variance than their ordinary analogues;
# we conclude that the use of the first two components of the version with robust
# covmat brings better results than in the part a)
par(mfrow=c(1,2))
plot(erg)
plot(erg_rob)
# => better results with robust covmat
par(mfrow=c(1,1))

erg$loadings
erg_rob$loadings # comp.1 ignores variable GiveBirth, and comp.2 ignores
# BabyUnderw (so comp.2 ignores only one variable, and above it ignored 4)

sc_rob<-erg_rob$scores
plot(sc_rob[,1:2])
```

```
plot(sc_rob[,c(1,3)])
# same explanation as above, only in this case relatively many points are
# concentrated on the left and down(1.plot)/up(2.plot)

plot(sc_rob[,1:2], col=y)
plot(sc_rob[,c(1,3)], col=y)
# again, Europe doesn't show much variety,...

# Question! (scale)



# ===== 2 =====

library(pls)
data(yarn)
dat<-yarn$NIR
matplot(t(dat), type="l") # we have to transpose the data first
library(pcaPP)

res<-PCAgrid(dat,k=5,method="sd") # sparse robust PC using Grid search algorithm;
# we give input parameter k - number of principal components
summary(res)
res2<-PCAgrid(dat,k=5,method="mad")
summary(res2)
res3<-PCAgrid(dat,k=5,method="qn")
summary(res3)
# changes: the first component shows describes more and the second component
# less variance as we go from "sd" over "mad" to "qn"

res$loadings
matplot(res$loadings,type="l")
matplot(res2$loadings,type="l")
matplot(res3$loadings,type="l")
```