Assignment 3: Principle component and factor analysis

Group 12

Dávid Hrabovszki (davhr856), Laura Julia Melis (lauju103), Spyridon Dimitriadis (spydi472), Vasileia Kampouraki (vaska979)

08/12/2019

Question 1: Principal components, including interpretation of them.

1.a.

The sample correlation matrix (R) is:

```
##             100m    200m    400m    800m   1500m   3000m marathon
## 100m     1.00000 0.94109 0.87078 0.80918 0.78155 0.72788  0.66896
## 200m     0.94109 1.00000 0.90881 0.81983 0.80133 0.73185  0.67995
## 400m     0.87078 0.90881 1.00000 0.80579 0.71980 0.67380  0.67694
## 800m     0.80918 0.81983 0.80579 1.00000 0.90505 0.86657  0.85399
## 1500m    0.78155 0.80133 0.71980 0.90505 1.00000 0.97338  0.79056
## 3000m    0.72788 0.73185 0.67380 0.86657 0.97338 1.00000  0.79873
## marathon 0.66896 0.67995 0.67694 0.85399 0.79056 0.79873  1.00000
```

The eigenvalues of R are:

```
##                PC1     PC2     PC3     PC4     PC5     PC6    PC7
## Eigenvalue 5.80762 0.62869 0.27933 0.12455 0.09097 0.05452 0.0143
```

And its eigenvectors:

```
##         e1       e2       e3       e4       e5       e6       e7
## 1 -0.37777 -0.40718 -0.14058  0.58706 -0.16707  0.53970  0.08894
## 2 -0.38321 -0.41363 -0.10078  0.19408  0.09350 -0.74493 -0.26566
## 3 -0.36804 -0.45935  0.23703 -0.64543  0.32727  0.24009  0.12660
## 4 -0.39478  0.16125  0.14754 -0.29521 -0.81905 -0.01651 -0.19521
## 5 -0.38926  0.30909 -0.42199 -0.06669  0.02613 -0.18899  0.73077
## 6 -0.37609  0.42319 -0.40606 -0.08016  0.35170  0.24050 -0.57151
## 7 -0.35520  0.38922  0.74106  0.32108  0.24701 -0.04827  0.08208
```

1.b.

First two principal components

In general, the i-th principal component is given by

$$\hat{y}_i = \hat{e}_i' z = \hat{e}_{i1}' z_1 + \hat{e}_{i2}' z_2 + \cdots + \hat{e}_{ip}' z_p \quad i = 1, 2, \ldots, p.$$

where z_i are the standardized variables.

So given the eigenvectors e_1 and e_2 obtained in part (a), the first two principal components for the standardized variables are:

$$\hat{y}_1 = \hat{e}_1' z = -0.37777 z_1 - 0.38321 z_2 - 0.36804 z_3 - 0.39478 z_4 - 0.38926 z_5 - 0.37609 z_6 - 0.35520 z_7.$$
$$\hat{y}_2 = \hat{e}_2' z = -0.40718 z_1 - 0.41363 z_2 - 0.45935 z_3 + 0.16125 z_4 + 0.30909 z_5 + 0.42319 z_6 + 0.38922 z_7.$$

Correlations of the standardized variables with the components

The correlation coefficients between the components Y_i and the standarized variables z_i are given by

$$r_{\hat{y}_i, z_k} = \frac{\hat{e}_{ik}\sqrt{\hat{\lambda}_i}}{\sqrt{s_{kk}}}, \quad i, k = 1, 2, \ldots, p$$

Note: The covariance matrix S of standardized data z is the correlation matrix R.

In these data we have the following coefficients:

```
##         e1       e2       e3      e4       e5       e6       e7
## 1 -0.91038 -0.98125 -0.33878 1.41476 -0.40262  1.30062  0.21434
## 2 -0.30385 -0.32797 -0.07991 0.15388  0.07414 -0.59066 -0.21064
```

Cumulative percentage of the total (standardized) sample variance.

The percentage of the total sample variance due to the k-th principal component is given by

$$\left(\frac{\hat{\lambda}_k}{\hat{\lambda}_1 + \hat{\lambda}_2 + \cdots + \hat{\lambda}_p} \right) \cdot 100\%, \quad k = 1, 2, \ldots, p$$

So, the total sample variance in the first two components are:

```
##                   PC1       PC2
## percentage 82.96606  8.981335
## cumulative 82.96606 91.947398
```

The percentage of the total standarized sample variance explained by the first two principal components is 91.95%. Then, as the majority of the total sample variance is attributed to these first two components, and in our case, if we remove the other variables, we won't lose much information.

1.c.

The projections in PC1 are more or less the same (around 0.3) which means that all the variables contribute almost the same. So, PC1 might measure the athletic excellence of a given nation because all the variables contribute the same.

Regarding the projections in PC2, the first three elements (100m, 200m and 400m) of the PC2 have the smallest values, so they contribute more to the second principal component. Because of this, we can interpret it as it captures how a nation performes in short distance runtypes.

1.d.

In order to rank the countries based on the first principal component ($\hat{y}_1$) first we need the standarize our observations:

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{s_{jj}}} \quad i = 1, 2, \ldots, n \quad j = 1, 2, \ldots, p$$

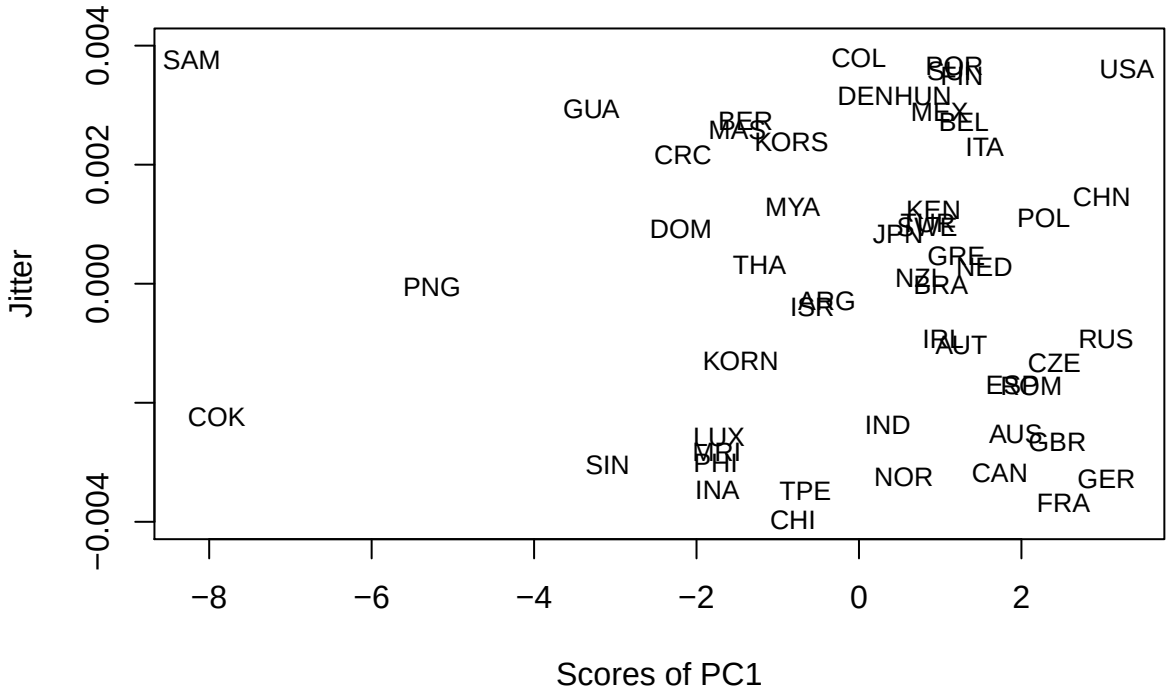
And then, we only have to compute the following formula, replacing the z_j values with the standarized observations:

$$\hat{y}_1 = \hat{e}_1' z = -0.37777 z_1 - 0.38321 z_2 - 0.36804 z_3 - 0.39478 z_4 - 0.38926 z_5 - 0.37609 z_6 - 0.35520 z_7.$$

As a result, the 5 countries with highest scores are:

```
##      USA      GER      RUS      CHN      FRA
## 3.299149 3.047517 3.042948 2.989467 2.518346
```

This ranking corresponds with the group of countries that perform better in all runtypes.



So as we expected the ranking of the first Principal Component corresponds to the athletic excellence for the various countries.

NOTE: In the above plot the x-axis represent the scores of the PC1 and the y-axis is just the jitter of this points. ("Jittering" is adding a bit of random noise to scatterplots, to better see the information contained in the data, because there is a lot of overplotting.)

Question 2: Factor analysis.

- Covariance matrix S:

```
##             100m     200m     400m    800m   1500m    3000m  marathon
## 100m     0.15532  0.34456  0.89130 0.02770 0.08389  0.23388   4.33418
## 200m     0.34456  0.86309  2.19284 0.06617 0.20276  0.55435  10.38499
## 400m     0.89130  2.19284  6.74546 0.18181 0.50918  1.42682  28.90373
## 800m     0.02770  0.06617  0.18181 0.00755 0.02141  0.06138   1.21965
## 1500m    0.08389  0.20276  0.50918 0.02141 0.07418  0.21616   3.53984
## 3000m    0.23388  0.55435  1.42682 0.06138 0.21616  0.66476  10.70609
## marathon 4.33418 10.38499 28.90373 1.21965 3.53984 10.70609 270.27015
```

- Correlation matrix R:

```
##             100m    200m    400m    800m   1500m   3000m marathon
## 100m     1.00000 0.94109 0.87078 0.80918 0.78155 0.72788  0.66896
## 200m     0.94109 1.00000 0.90881 0.81983 0.80133 0.73185  0.67995
## 400m     0.87078 0.90881 1.00000 0.80579 0.71980 0.67380  0.67694
## 800m     0.80918 0.81983 0.80579 1.00000 0.90505 0.86657  0.85399
## 1500m    0.78155 0.80133 0.71980 0.90505 1.00000 0.97338  0.79056
## 3000m    0.72788 0.73185 0.67380 0.86657 0.97338 1.00000  0.79873
## marathon 0.66896 0.67995 0.67694 0.85399 0.79056 0.79873  1.00000
```

What does it mean that the parameter rotation of factanal() is set to "varimax" by default (equivalently rotate of principal())?

The rotated coefficients scaled by the square root of the communalities. i.e. $\tilde{l}^*_{ij} = \hat{l}^*_{ij}/\hat{h}_i$ Scaling the rotated coefficients has the effect of giving variables with small communalities relatively more weight in the determination of simple structure. (principal() has the rotate option of "varimax") In our analysis we use varimax rotation.

Principal Component method

- Principal component analysis on S with varimax rotation

```
## principal(r = df1, nfactors = 2, rotate = "varimax", covar = T,
##     method = "Bartlett")
```

```
##
## Loadings:
##          RC1    RC2
## 100m      0.173  0.307
## 200m      0.404  0.765
## 400m      1.038  2.376
## 800m
## 1500m     0.179  0.142
## 3000m     0.561  0.371
## marathon 15.537  5.375
##
##                    RC1    RC2
## SS loadings    243.005 35.375
## Proportion Var  34.715  5.054
## Cumulative Var  34.715 39.768
```

Looking at the covariance matrix we can see that marathon has significantly much bigger variance (270.270150) compared to the other variables. Note: our data are not scaled. The first three (100m,200m,400m) are in

seconds while the last four (800m,1500m,3000m and marathon) are in minutes. We use $structure to take the unstandardized loadings and the 800m loadings are not visible and that's because they are very small. After conducting our PCA analysis, we see that marathon expains the biggest part of the total variance and that was expected as it has the biggest variance.

- Principal component analysis on R with varimax rotation

```
## principal(r = df1, nfactors = 2, rotate = "varimax", covar = F)
```

```
##
## Loadings:
##          RC1   RC2
## 100m     0.431 0.865
## 200m     0.437 0.877
## 400m     0.385 0.878
## 800m     0.773 0.569
## 1500m    0.845 0.475
## 3000m    0.885 0.388
## marathon 0.830 0.373
##
##                  RC1   RC2
## SS loadings    3.309 3.128
## Proportion Var 0.473 0.447
## Cumulative Var 0.473 0.919
```

Looking at the results of the PCA analysis using the correlation matrix, the first principal component focuses on the long distances(800m,1500m,3000m and marathon) whereas the second one focuses on the smaller distances(100m,200m and 400m) and this can be seen from the loadings.

NOTE: Principal component method: the correlation matrix R (which is also the covariance matrix of the standardized data) is used instead of S, to avoid problems related to measurements being in different scales. Department of Mathematics, Uppsala University

That is why using correlation matrix R instead of the covarianve matrix S we are adressing the problem with the different scales (e.g. marathon value), thus we have better analysis using R matrix.

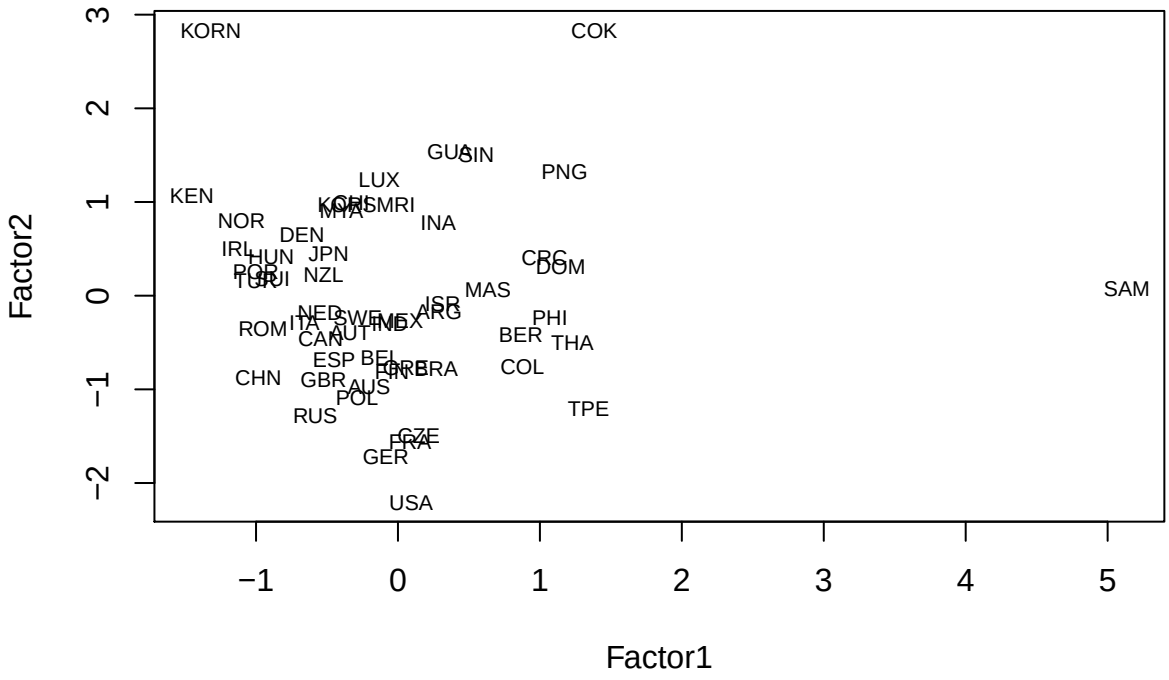
Factor analysis with ML method on S using varimax rotation

```
##
## Call:
## factanal(factors = 2, covmat = S, n.obs = nrow(df1), rotation = "varimax",     method = "mle")
##
## Uniquenesses:
##     100m     200m     400m     800m    1500m    3000m marathon
##    0.094    0.024    0.152    0.144    0.016    0.028    0.338
##
## Loadings:
##          Factor1 Factor2
## 100m     0.461   0.833
## 200m     0.455   0.877
## 400m     0.401   0.829
## 800m     0.732   0.566
## 1500m    0.882   0.454
## 3000m    0.918   0.361
## marathon 0.693   0.427
##
##                Factor1 Factor2
```

```
## SS loadings      3.216   2.987
## Proportion Var   0.459   0.427
## Cumulative Var   0.459   0.886
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 31.43 on 8 degrees of freedom.
## The p-value is 0.000118
```

From the loadings of the 2 factors we observe that the Factor1 focuses on the long distance runtypes (800m, 1500m, 3000m, marathon) and Factor2 on sort distance (100m, 200m, 400m). We recall that sort distance runtypes are in seconds and long distance runtypes in minutes. Moreover we see that both factors have almost the same Proportion Variance.

Factor scores and check for outliers in the data



Plotting the loadings of the two factors we can observe that SAM is an outlier for factor1 and KORN and COK are outliers for factor2.

Repeat the analysis with Sample correlation matrix R.

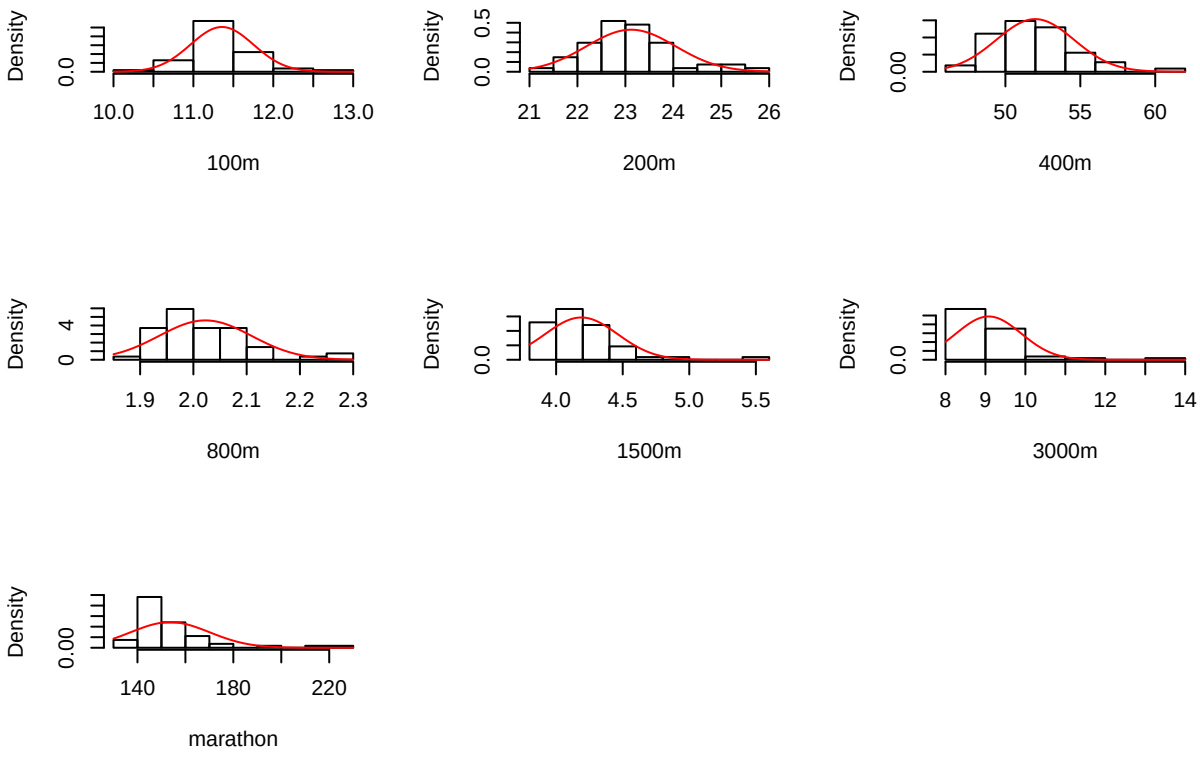
We observe that after using the correlation matrix we get exactly the same results as using the covariance matrix.

There is a simple relationship between FA solution for S (covariance matrix) and R (correlation matrix) using Maximum Likelihood estimate. If θ is the MLE of θ, then $h(\hat{\theta})$ is the MLE of $h(\hat{\theta})$. University of Minnesota

```
##
## Call:
## factanal(factors = 2, covmat = R, n.obs = nrow(df1), rotation = "varimax",     method = "mle")
##
## Uniquenesses:
##     100m     200m     400m     800m    1500m    3000m marathon
##    0.094    0.024    0.152    0.144    0.016    0.028    0.338
##
## Loadings:
##          Factor1 Factor2
## 100m     0.461   0.833
## 200m     0.455   0.877
## 400m     0.401   0.829
## 800m     0.732   0.566
## 1500m    0.882   0.454
## 3000m    0.918   0.361
## marathon 0.693   0.427
##
##                Factor1 Factor2
## SS loadings      3.216   2.987
## Proportion Var   0.459   0.427
## Cumulative Var   0.459   0.886
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 31.43 on 8 degrees of freedom.
## The p-value is 0.000118
```

Check the adequacy of your model, the data should be approximatelly multivariate normally distributed.

```
##             Test        Statistic              p value Result
## 1 Mardia Skewness 357.524576271262 1.97530106980739e-35     NO
## 2 Mardia Kurtosis 11.1569164817406                    0     NO
## 3            MVN             <NA>                 <NA>     NO
```



From the "Mardia" test (see reference) and from the Histograms we observe that the data does not follow a multivariate normal distribution. Thus we can say that the outcome of the Factor Analysis is not reliable.

Mardia KV. Measures of multivariate skewness and kurtosis with applications. Biometrika. 1970;57:519–530.[Google Scholar]

Appendix

```r
knitr::opts_chunk$set(echo = TRUE, message = F, warning = F, error = F)
# NEEDED LIBRARIES
library(psych)
library(MVN)

RNGversion('3.5.1')
##################--QUESTION 1--##################
# Importing and modifying data file
df = read.table("T1-9.dat")
colnames(df) = c("country","100m", "200m", "400m", "800m", "1500m", "3000m" , "marathon")
df1 = df[,-1]  # without the column of the countries
# A1. Sample correlation matrix:
R <- cor(df1)
round(R, 5)
# A2. Eigenvalues and eigenvectors:
eigen_decomposition <- eigen(R)

eigenval <- t(as.matrix(eigen_decomposition$values))
colnames(eigenval) <- c("PC1","PC2","PC3","PC4","PC5","PC6","PC7")
rownames(eigenval) <- "Eigenvalue"
round(eigenval,5)
eigenvector <- as.data.frame(eigen_decomposition$vector)
colnames(eigenvector) <- c("e1","e2","e3","e4","e5","e6","e7")
round(eigenvector,5)
# B.2. The correlation coefficients between Y and z.
Sstd <- cov(scale(df1)) #because the S of standardized data is the R

RYK= eigenvector * matrix(sqrt(eigenval),byrow=T,nrow=dim(eigenvector)[2])/
  matrix(sqrt(diag(Sstd)),byrow=F,nrow=dim(eigenvector)[2])

round(RYK[1:2,], 5)
# B.3. Cumulative percentage of the total sample variance.
percentage <- vector()
cumulative <- vector()
for(i in 1:2){
  percentage[i] <- (eigenval[i]/ sum(eigenval))*100
  cumulative[i] <- sum(percentage)
}

total_var <- rbind(percentage, cumulative)
colnames(total_var) <- c("PC1", "PC2")
total_var
# B.5. Ranking of nations based on the PC1 score.
# First we need to standarize the observations.
dfstd <- scale(df1)

# Now we can calculate the scores.
ranking <- matrix(0,nrow=nrow(dfstd), ncol=1)
for(i in 1:nrow(dfstd)){
  ranking[i,1] <- sum(eigenvector[,1]* dfstd[i,])
}
```

```r
names(ranking) = df$country
head(sort(ranking, decreasing=TRUE),5)
plot(sort(ranking, decreasing=TRUE), jitter(rep(0, length(ranking)),0.2) ,
     pch=" ", xlab="Scores of PC1", ylab="Jitter")
text(sort(ranking, decreasing=TRUE),  jitter(rep(0, length(ranking)),0.2),
     labels = names(sort(ranking, decreasing=TRUE)), cex=0.8)
df1 = df[,-1]
colnames(df1) = colnames(df[,-1])
rownames(df1) = df[,1]

S <- cov(df1)
round(S,5)
R = cor(df1)
round(R,5)
library(psych)
#covariance matrix
fit1 <-  principal(df1,nfactors = 2,rotate = "varimax",covar = T,method = "Bartlett")
fit1$Call
fit1$Structure


#correlation matrix
fit2 <- principal(df1,nfactors = 2 ,rotate = "varimax",covar = F)
fit2$Call
fit2$Structure
FS = factanal(covmat = S, factors = 2, method = "mle",
              n.obs = nrow(df1), rotation="varimax")
FS
Fscrs = factanal(df1, factors = 2, method = "mle", scores = "Bartlett")
#Bartlett's weighted least-squares scores.

plot(Fscrs$scores, type="n") # set up plot
text(Fscrs$scores,labels=rownames(Fscrs$scores),cex=.7) # add variable names
FR = factanal(covmat = R, factors = 2, method = "mle",
              n.obs = nrow(df1), rotation="varimax")
FR
##  https://cran.r-project.org/web/packages/MVN/vignettes/MVN.pdf
# "mardia", "hz", "royston", "dh", "energy"

library(MVN)
result <- mvn(data = df1, mvnTest = "mardia")
result$multivariateNormality

#result <- mvn(data = df1, mvnTest = "royston", univariatePlot = "qqplot")
result <- mvn(data = df1, mvnTest = "royston", univariatePlot = "histogram")
```