Multivariate Statistics Lab3

Principal Component and Factor Analysis

Ahmet Hakan Akdeve(ahmak554), Jooyoung Lee(joole336), Weng Hang Wong(wonwo535), Zhixuan Duan(zhidu838)

2019 12 8

Question 1

Correlation matrix R, its eigenvalues and eigenvectors are shown as below.

```
## correlation matrix:
```

```
##               100m      200m      400m      800m     1500m     3000m
## 100m     1.0000000 0.9410886 0.8707802 0.8091758 0.7815510 0.7278784
## 200m     0.9410886 1.0000000 0.9088096 0.8198258 0.8013282 0.7318546
## 400m     0.8707802 0.9088096 1.0000000 0.8057904 0.7197996 0.6737991
## 800m     0.8091758 0.8198258 0.8057904 1.0000000 0.9050509 0.8665732
## 1500m    0.7815510 0.8013282 0.7197996 0.9050509 1.0000000 0.9733801
## 3000m    0.7278784 0.7318546 0.6737991 0.8665732 0.9733801 1.0000000
## Marathon 0.6689597 0.6799537 0.6769384 0.8539900 0.7905565 0.7987302
##           Marathon
## 100m     0.6689597
## 200m     0.6799537
## 400m     0.6769384
## 800m     0.8539900
## 1500m    0.7905565
## 3000m    0.7987302
## Marathon 1.0000000
```

Good.

```
##
##  eigenvalues:
```

```
## [1] 5.80762446 0.62869342 0.27933457 0.12455472 0.09097174 0.05451882
## [7] 0.01430226
```

```
##
##  eigenvectors:
```

```
##            [,1]       [,2]       [,3]        [,4]        [,5]        [,6]
## [1,] -0.3777657 -0.4071756 -0.1405803  0.58706293 -0.16706891 -0.53969730
## [2,] -0.3832103 -0.4136291 -0.1007833  0.19407501  0.09350016  0.74493139
## [3,] -0.3680361 -0.4593531  0.2370255 -0.64543118  0.32727328 -0.24009405
## [4,] -0.3947810  0.1612459  0.1475424 -0.29520804 -0.81905467  0.01650651
## [5,] -0.3892610  0.3090877 -0.4219855 -0.06669044  0.02613100  0.18898771
## [6,] -0.3760945  0.4231899 -0.4060627 -0.08015699  0.35169796 -0.24049968
## [7,] -0.3552031  0.3892153  0.7410610  0.32107640  0.24700821  0.04826992
##             [,7]
## [1,]  0.08893934
## [2,] -0.26565662
## [3,]  0.12660435
## [4,] -0.19521315
## [5,]  0.73076817
## [6,] -0.57150644
## [7,]  0.08208401
```

```
##               100m      200m      400m      800m     1500m     3000m
## 100m     1.0000000 0.9410886 0.8707802 0.8091758 0.7815510 0.7278784
## 200m     0.9410886 1.0000000 0.9088096 0.8198258 0.8013282 0.7318546
## 400m     0.8707802 0.9088096 1.0000000 0.8057904 0.7197996 0.6737991
## 800m     0.8091758 0.8198258 0.8057904 1.0000000 0.9050509 0.8665732
## 1500m    0.7815510 0.8013282 0.7197996 0.9050509 1.0000000 0.9733801
## 3000m    0.7278784 0.7318546 0.6737991 0.8665732 0.9733801 1.0000000
## Marathon 0.6689597 0.6799537 0.6769384 0.8539900 0.7905565 0.7987302
##           Marathon
## 100m     0.6689597
## 200m     0.6799537
## 400m     0.6769384
## 800m     0.8539900
## 1500m    0.7905565
## 3000m    0.7987302
## Marathon 1.0000000
```

Covariance matrix of the standardized data is exactly the same as the correlation matrix of the original data. Therefore, the principal components(PCs) of the standardized can also be found using eigenvalues and eigenvectors of the correlation matrix, R. The first two PCs are below, that z_i means standardized i^{th} variable.

$$Y_1 = -0.378z_1 - 0.383z_2 - 0.368z_3 - 0.395z_4 - 0.389z_5 - 0.376z_6 - 0.355z_7$$

$$Y_2 = -0.407z_1 - 0.414z_2 - 0.459z_3 + 0.161z_4 + 0.309z_5 + 0.423z_6 + 0.389z_7$$

```
## Importance of components:
##                           PC1     PC2    PC3     PC4    PC5     PC6
## Standard deviation     2.4099 0.79290 0.5285 0.35292 0.3016 0.23349
## Proportion of Variance 0.8297 0.08981 0.0399 0.01779 0.0130 0.00779
## Cumulative Proportion  0.8297 0.91947 0.9594 0.97717 0.9902 0.99796
##                            PC7
## Standard deviation     0.11959
## Proportion of Variance 0.00204
## Cumulative Proportion  1.00000
```

Correct.

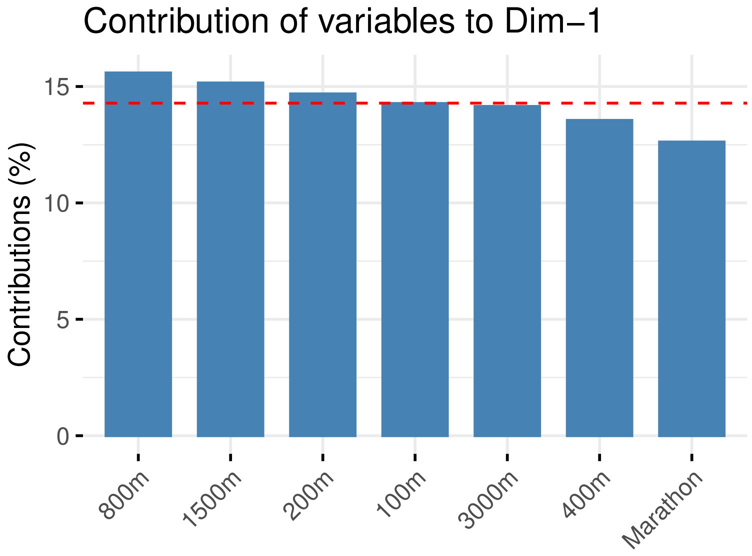
The proportion of variance of each principal components are above shown. The cummulative percentage of the total sample variance explained by the first two PCs are 91.95%.

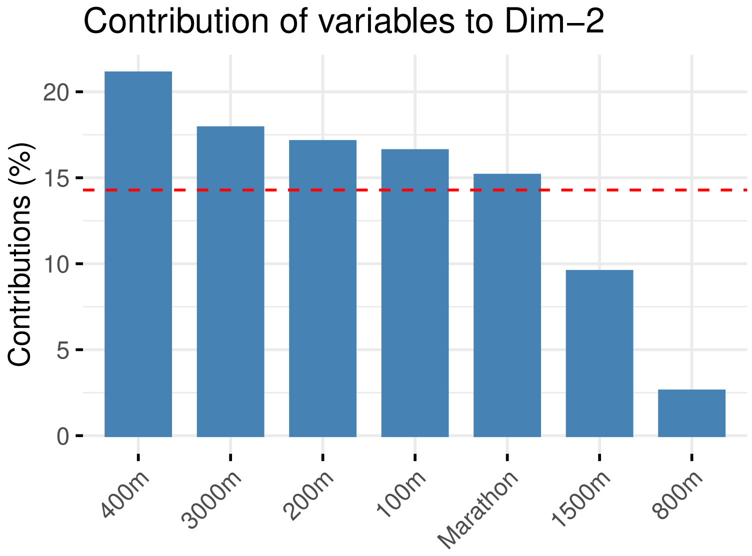
```
## The correlation of the standardized vaiables vs components:
##
```

```
##                z1         z2         z3         z4         z5         z6
## r1_yz -0.9103780 -0.9234990 -0.8869307 -0.9513832 -0.9380805 -0.9063506
## r2_yz -0.3228503 -0.3279673 -0.3642220  0.1278522  0.2450762  0.3355481
##               z7
## r1_yz -0.8560043
## r2_yz  0.3086096
```

```
## [1] "cumulative precentage : 0.919473983845877"
```

From the above eigenvalues, we can calculate the two components cumulative precentage of the total sample variance which is 0.919





On the graph related to the first PC, it is possible to find that all variables are almost equally contributing to the variation. However, on the second PC contribution graph, contribution rate differs by the distances. With the second eigenvector shown before, it is possible to predict that the short-distance race and long-distance race affect the strength of the nations in the opposite ways.

```
##    Country   Scores
## 54     USA 3.299149
## 18     GER 3.047517
```

Good.

```
## 45     RUS 3.042948
## 9      CHN 2.989467
## 17     FRA 2.518346
## 19     GBR 2.442706
```

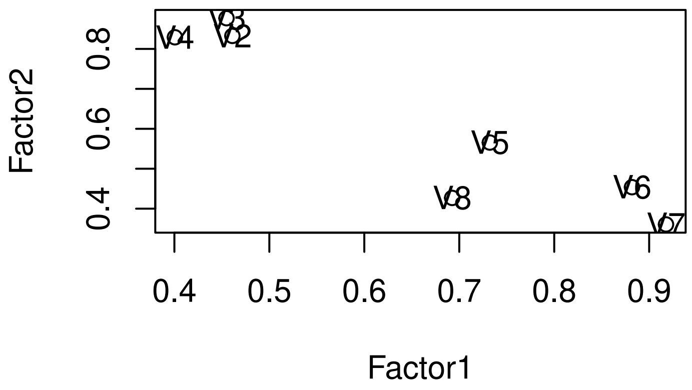
The first PC shows the scores as shown. Intuitive nations of athletic excellence corresponds to the ranking turned out above.

QUESTION 2

Maximum Likelihood

Covariance matrix analysis

```
##
## Call:
## factanal(x = my_data[, -1], factors = 2, covmat = S_matrix, n.obs = nrow(my_data))
##
## Uniquenesses:
##    V2    V3    V4    V5    V6    V7    V8
## 0.094 0.024 0.152 0.144 0.016 0.028 0.338
##
## Loadings:
##    Factor1 Factor2
## V2 0.461   0.833
## V3 0.455   0.877
## V4 0.401   0.829
## V5 0.732   0.566
## V6 0.882   0.454
## V7 0.918   0.361
## V8 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
```

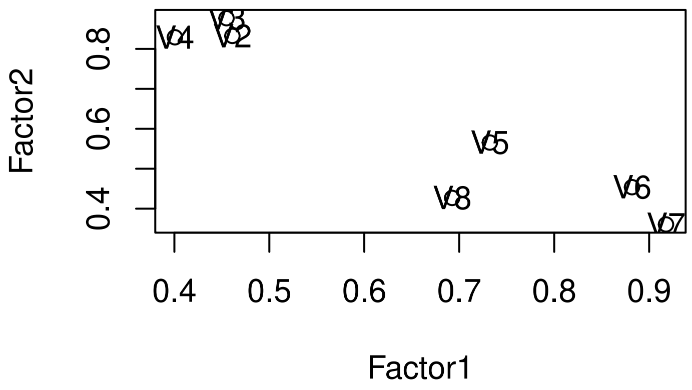


For factor 1: High values for long distance runs and low values for sprint runs. Factor 1 interpretas how good a country is good long distance runs.

Factor 2: The interpretation of this factor is how good a nation is at sprint runs.

correlation matrix

```
##
## Call:
## factanal(x = my_data[, -1], factors = 2, covmat = R_matrix, n.obs = nrow(my_data))
##
## Uniquenesses:
##    V2    V3    V4    V5    V6    V7    V8
## 0.094 0.024 0.152 0.144 0.016 0.028 0.338
##
## Loadings:
##    Factor1 Factor2
## V2 0.461   0.833
## V3 0.455   0.877
## V4 0.401   0.829
## V5 0.732   0.566
## V6 0.882   0.454
## V7 0.918   0.361
## V8 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
```

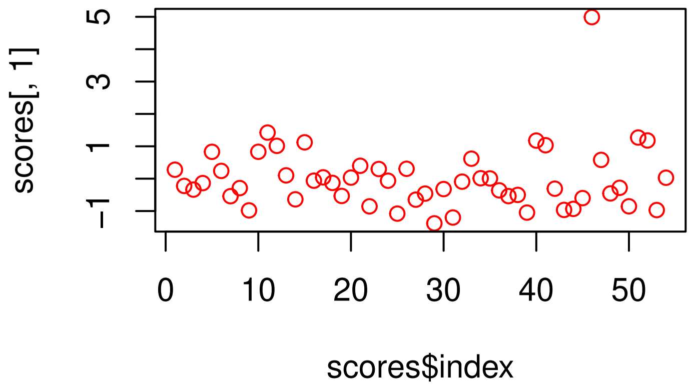


From the correlation matrix ML analysis above, same result as the analysis with the covariance matrix.

Outliners Scores

Scores factor 1

From the belowed graph, we can easily spot the outliner from the factor 1 (scores >2), which is the country of SAM.

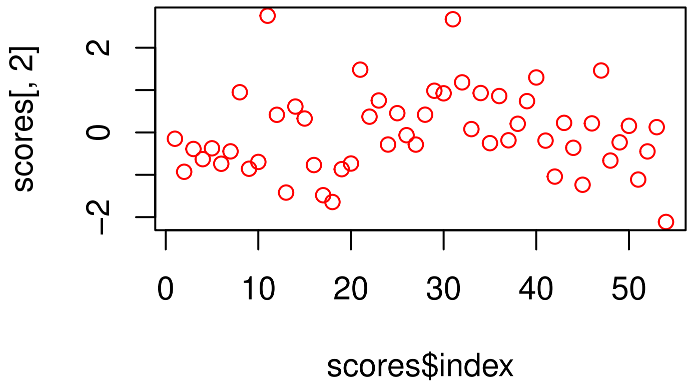


```
## [1] 46
```

```
## [1] SAM
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

Scores factor 2

From the below graph, we want to spot the outliner from the factor 2 (scores >1.5), so we have the outliners of COK and KORN



```
## [1] 11 31
```

```
## [1] COK  KORN
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

Principle Component Analysis

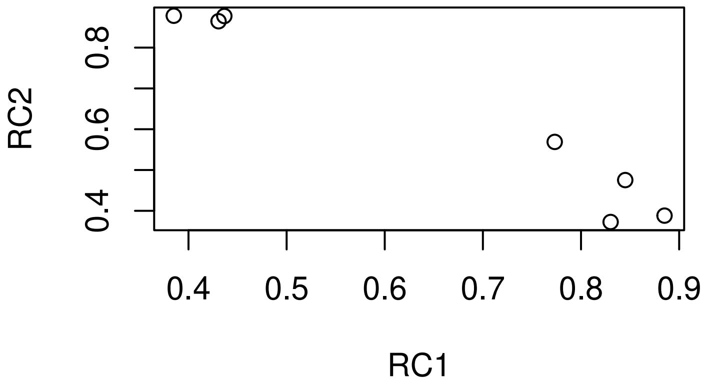
correlation matrix

```
##
## Attaching package: 'psych'
```

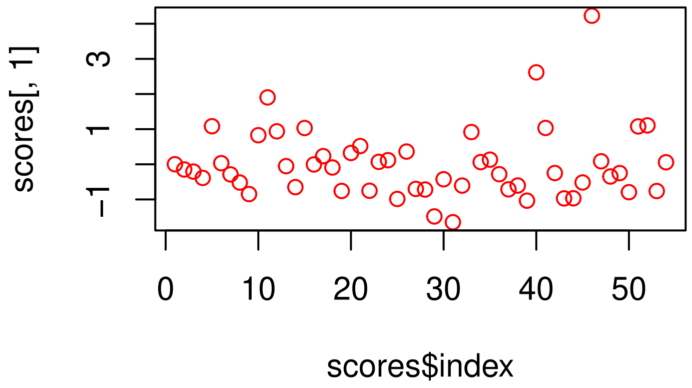
```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
## Principal Components Analysis
## Call: principal(r = my_data[, -1], nfactors = 2, n.obs = 54, scores = TRUE,
##     oblique.scores = FALSE, method = "regression")
## Standardized loadings (pattern matrix) based upon correlation matrix
##     RC1  RC2   h2    u2 com
## V2 0.43 0.86 0.93 0.067 1.5
## V3 0.44 0.88 0.96 0.040 1.5
## V4 0.39 0.88 0.92 0.081 1.4
## V5 0.77 0.57 0.92 0.079 1.8
## V6 0.85 0.48 0.94 0.060 1.6
## V7 0.89 0.39 0.93 0.066 1.4
## V8 0.83 0.37 0.83 0.172 1.4
##
##                        RC1  RC2
## SS loadings           3.31 3.13
## Proportion Var        0.47 0.45
## Cumulative Var        0.47 0.92
## Proportion Explained  0.51 0.49
## Cumulative Proportion 0.51 1.00
##
## Mean item complexity =  1.5
## Test of the hypothesis that 2 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.03
##  with the empirical chi square  2.63  with prob <  0.96
##
## Fit based upon off diagonal values = 1
```

```
##        V2        V3        V4        V5        V6        V7        V8
## 0.4306309 0.4365153 0.3850257 0.7732014 0.8450596 0.8850790 0.8301493
```



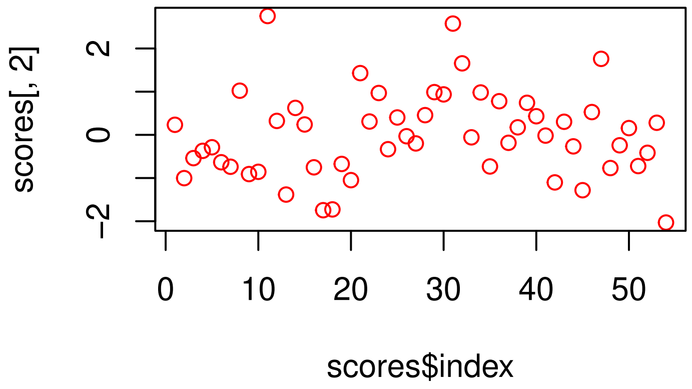
Using PCA from the belowed graph, scores plot and outliers from that plot for FACTOR 1 (scores>2), Outliers seems to be PNG and SAM.



```
## [1] 40 46
```

```
## [1] PNG SAM
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

From the belowed graph, the outliner that plotted for Factor2 (scores>2), which seems to be COK and KORN.



```
## [1] 11 31
```

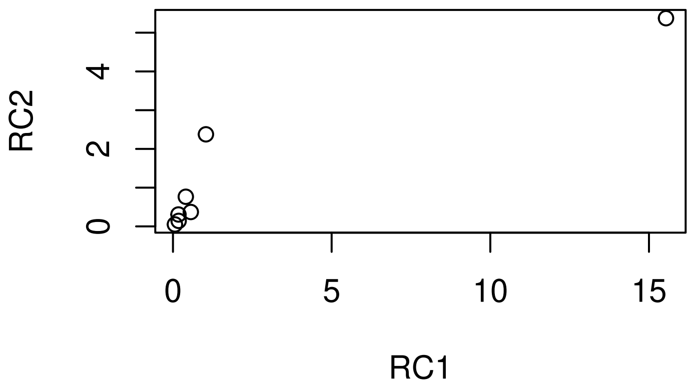
```
## [1] COK  KORN
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

covaiance matrix

Form factor 1 Marathon load very highly on this factor. Form factor 2 “800m” load higly.

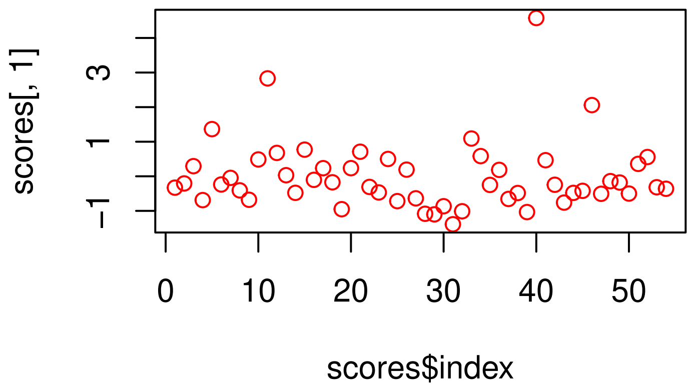
```
##          V2          V3          V4          V5          V6          V7
##  0.17270136  0.40383119  1.03817773  0.06099288  0.17879844  0.56119868
##          V8
## 15.53651648
```

```
## Principal Components Analysis
## Call: principal(r = my_data[, -1], nfactors = 2, n.obs = 54, covar = TRUE,
##     scores = TRUE, method = "regression")
## Unstandardized loadings (pattern matrix) based upon covariance matrix
##      RC1  RC2      h2      u2   H2      U2
## V2  0.17 0.31 1.2e-01 0.03100 0.80 2.0e-01
## V3  0.40 0.77 7.5e-01 0.11435 0.87 1.3e-01
## V4  1.04 2.38 6.7e+00 0.02014 1.00 3.0e-03
## V5  0.06 0.05 6.3e-03 0.00126 0.83 1.7e-01
## V6  0.18 0.14 5.2e-02 0.02200 0.70 3.0e-01
## V7  0.56 0.37 4.5e-01 0.21213 0.68 3.2e-01
## V8 15.54 5.37 2.7e+02 0.00026 1.00 9.5e-07
##
##                           RC1   RC2
## SS loadings           243.00 35.37
## Proportion Var          0.87  0.13
## Cumulative Var          0.87  1.00
## Proportion Explained    0.87  0.13
## Cumulative Proportion   0.87  1.00
##
##  Standardized loadings (pattern matrix)
##    item  RC1  RC2   h2      u2
## V2    1 0.44 0.78 0.80 2.0e-01
## V3    2 0.43 0.82 0.87 1.3e-01
## V4    3 0.40 0.92 1.00 3.0e-03
## V5    4 0.70 0.58 0.83 1.7e-01
## V6    5 0.66 0.52 0.70 3.0e-01
## V7    6 0.69 0.46 0.68 3.2e-01
## V8    7 0.95 0.33 1.00 9.5e-07
##
##                  RC1  RC2
## SS loadings     2.83 3.05
## Proportion Var  0.40 0.44
## Cumulative Var  0.40 0.84
## Cum. factor Var 0.48 1.00
##
## Mean item complexity =  1.6
## Test of the hypothesis that 2 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.02
##  with the empirical chi square  1.37  with prob <  0.99
##
## Fit based upon off diagonal values = 1
```



Scores factor 1

The outliners seems to be COK and PNG when the scores > 2.5

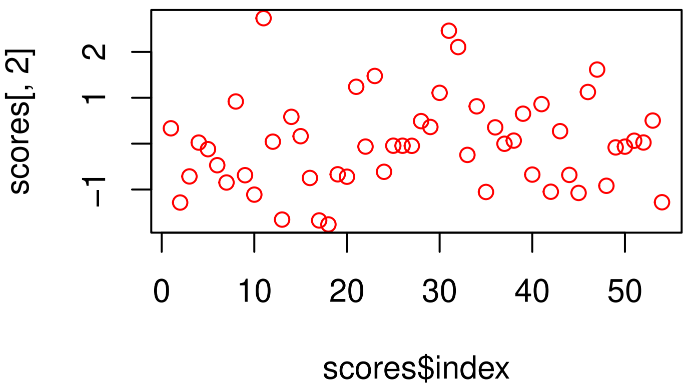


```
## [1] 11 40
```

```
## [1] COK PNG
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

Scores factor 2

According to the plot, it is not easily to pinpoint which countries are outliers here. But COK, PNG and SAM has high scores values, when scores>2.



```
## [1] 11 40 46
```

```
## [1] COK PNG SAM
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

Conclusion

The result from the two methods(PCA and ML) are quite similar.

The meaning of varimax of the rotation parametier is that aims when rotating the factors, it will clarify the structure of the loading matrix.

#Apprendix

```
############-----------QUESTION 1----------------##############

my_data <- read.delim(file="C:\\Users\\Suat\\Desktop\\Master_courses\\732A97_Multivariate\\Materials\\T
colnames(my_data)<-c("Country","100m","200m","400m","800m","1500m","3000m","Marathon")

###### A.

#Correlation matrix
R_matrix<-cor(my_data[,-1])
#Eigenvalues and eigenvectors
eigen(R_matrix)

###### B.
r1_yz = eigenvect[,1] * sqrt(eigenval[1])
r2_yz = eigenvect[,2] * sqrt(eigenval[2])

cor_matr<-rbind(r1_yz, r2_yz)
colnames(cor_matr) <- c("z1", "z2","z3","z4","z5","z6","z7")

cat("The correlation of the standardized vaiables vs components:
    ")
cor_matr

a <- sum( abs(eigenval[1]))
b <- sum( abs(eigenval[2]))

paste("cumulative precentage :", (a+b)/ sum(abs(eigenval)))


my_PCA<-prcomp(my_data[,-1],scale = TRUE)
summary(my_PCA) #PC1 and PC2 together explains 91.9 % percent of the variation

###### C.
#install.packages("factoextra")
library(factoextra)

fviz_contrib(my_PCA,choice = "var",axes = 1) #Contribution to PC1
#All variables contributes equally to dimension 1

fviz_contrib(my_PCA,choice = "var",axes = 2) #Contribution to PC2
#This component should be interpreted as IN THE QUESTION!

#But from PC1 values from each variable, we can see that the short distance races have negative values

###### D.
scores_frame<-data.frame("Country"=my_data$Country,"Scores"=my_PCA$x[,1])
scores_frame<-scores_frame[order(scores_frame$Scores,decreasing = TRUE),]
head(scores_frame) #Seems reasonable


############-----------QUESTION 2---------------##############

###### USING FACTANAL() HERE.

##### Covariance matrix

S_matrix<-cov.wt(my_data[,-1])
R_matrix<-cor(my_data[,-1])

my_fact<-factanal(my_data[,-1],factors=2,covmat=S_matrix,n.obs=nrow(my_data))
cov_loads<-my_fact$loadings[,1:2]

plot(cov_loads);text(cov_loads,labels=names(my_data[,-1]))
#For factor 1: High values for long distance runs and low values for sprint runs.
#Factor 1 interpretas how good a country is good long distance runs.

#Factor 2: The interpretation of this factor is how good a nation is at sprint runs.


##### correlation matrix

my_fact_cor<-factanal(my_data[,-1],factors=2,covmat=R_matrix,n.obs=nrow(my_data))
cor_loads<-my_fact_cor$loadings[,1:2]
```

```
plot(cor_loads);text(cor_loads,labels=names(my_data[,-1]))

#Same result as the analysis with the covariance matrix.

#Outliers. Dont know how to specify Cov or Cor matrix here when calculating scores. These function runs

my_fact3<-factanal(x=my_data[,-1],factors=2,scores="regression")

scores<-as.data.frame(my_fact3$scores[,c(1:2)])
scores$index<-1:54
plot(y=scores[,1],x=scores$index,col="red")
scores$index[which(scores[,1]>2)]
my_data[46,1] #Outlier from scores for factor 1. SAM


plot(y=scores[,2],x=scores$index,col="red")
scores$index[which(scores[,2]>1.5)]
my_data[c(11,31),1] #Outlier from scores for factor 2. COK and KORN


#### USING PC now. psych package. Rotation is by default "varimax"
install.packages("psych")
library(psych)

#Using correlation matrix
cor_fact<-principal(my_data[,-1],nfactors=2,method = "regression",n.obs = 54,scores=TRUE,oblique.scores
cor_fact$loadings[,1]

plot(cor_fact$loadings) #Same result as with factanal()

#Scores plot and outliers from that plot for FACTOR 1

scores<-as.data.frame(cor_fact$scores[,c(1:2)])
scores$index<-1:54
plot(y=scores[,1],x=scores$index,col="red")
scores$index[which(scores[,1]>2)]
my_data[c(40,46),1] #Outliers seems to be PNG and SAM

#Scores plot and outliers from that plot for FACTOR 2

scores<-as.data.frame(cor_fact$scores[,c(1:2)])
scores$index<-1:54
plot(y=scores[,2],x=scores$index,col="red")
scores$index[which(scores[,2]>2)]
my_data[c(11,31),1] #Outliers seems to be COK and KORN

#USING COVARIANCE MATRIX

cov_fact<-principal(my_data[,-1],nfactors=2,method = "regression",n.obs = 54,scores=TRUE,covar = TRUE)
cov_fact$loadings[,1]
plot(cov_fact$loadings) #For factor 1 Marathon load very highly on this factor. For factor 2 "800m" loa

#Scores plot for factor 1
```

```r
scores<-as.data.frame(cov_fact$scores[,c(1:2)])
scores$index<-1:54
plot(y=scores[,1],x=scores$index,col="red")
scores$index[which(scores[,1]>2.5)]
my_data[c(11,40),1] #Outliers seems to be COK and PNG

#Scores plot for factor 2

scores<-as.data.frame(cov_fact$scores[,c(1:2)])
scores$index<-1:54
plot(y=scores[,2],x=scores$index,col="red")
scores$index[which(scores[,1]>2)]
my_data[c(11,40,46),1] #Hard to pinpoint which countries are outliers here. But COK, PNG and SAM has hi
```