Multivariate Statistical Methods

Assignment 3

Allan Gholmi, Emma Wallentinsson, Rasmus Holm

2017-12-15

Question 1

a)

The sample covariance matrix for the national tracks data is:

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

The eigenvalues are the following:

```
5.80762446 0.62869342 0.27933457 0.12455472 0.09097174 0.05451882 0.01430226
```

And the corresponding eigenvectors are (one vector per column):

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

b)

The first two principal components for the standardized variables is:

```
> princomp_1
[1] -0.3777657 -0.3832103 -0.3680361 -0.3947810 -0.3892610 -0.3760945 -0.3552031
> princomp_2
[1] -0.4071756 -0.4136291 -0.4593531 0.1612459 0.3090877 0.4231899 0.3892153
```

The cumulative percentage of the total sample variance explained by the two first principal components are 0.919474.

c)

The first component PC1, seems to have similar correlations to all of the variables, being aroundmeaning that the first principal component is moderately and negatively correlated with all variables. The second component PC2, are positively and moderately correlated with the first 3 variables which feels plausible since these 3 variables has a lower distance then the rest. The rest of the variables, are negatively and moderately correlate. PC1 might be called athletic excellence component and PC2 might be called distance component.

d)

When we rank the scores of the different countries and check the countries with lowest scores, we recognize them from previous labs as countries who have bad results, e.g they have been outliers. The result makes sense.

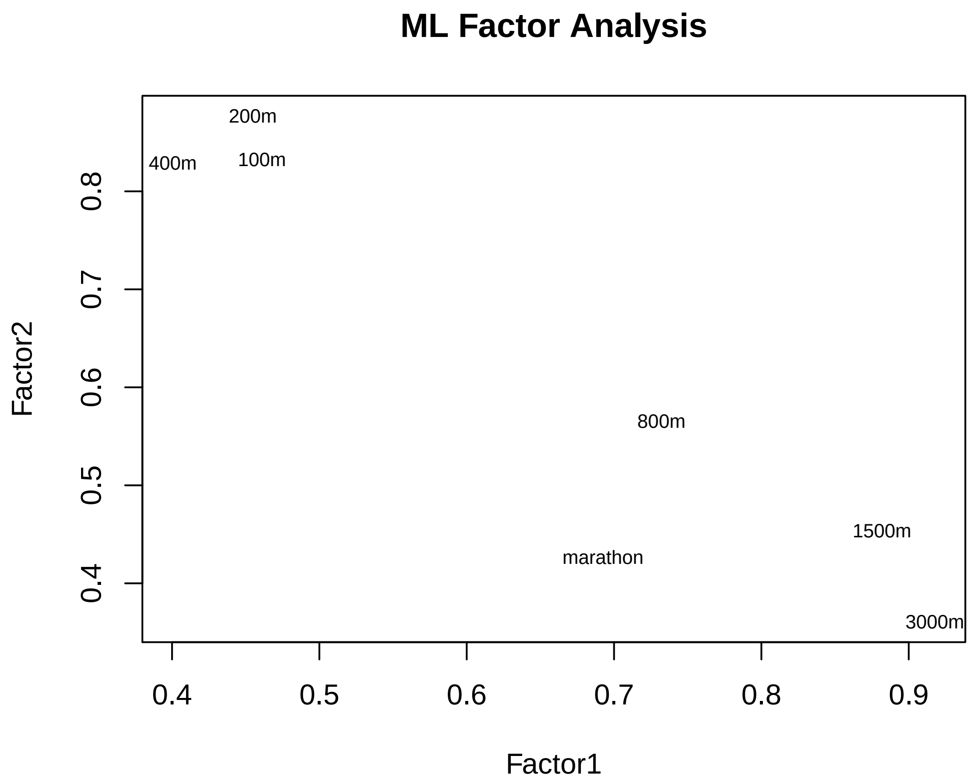
```
[,1]  [,2]
[1,] "SAM" "-8.21341512287609"
[2,] "COK" "-7.90622722445813"
[3,] "PNG" "-5.25744974658153"
[4,] "GUA" "-3.29412379863809"
[5,] "SIN" "-3.09391951725173"
[6,] "DOM" "-2.19240980880379"
[7,] "CRC" "-2.16681150553093"
[8,] "PHI" "-1.76353368162812"
```

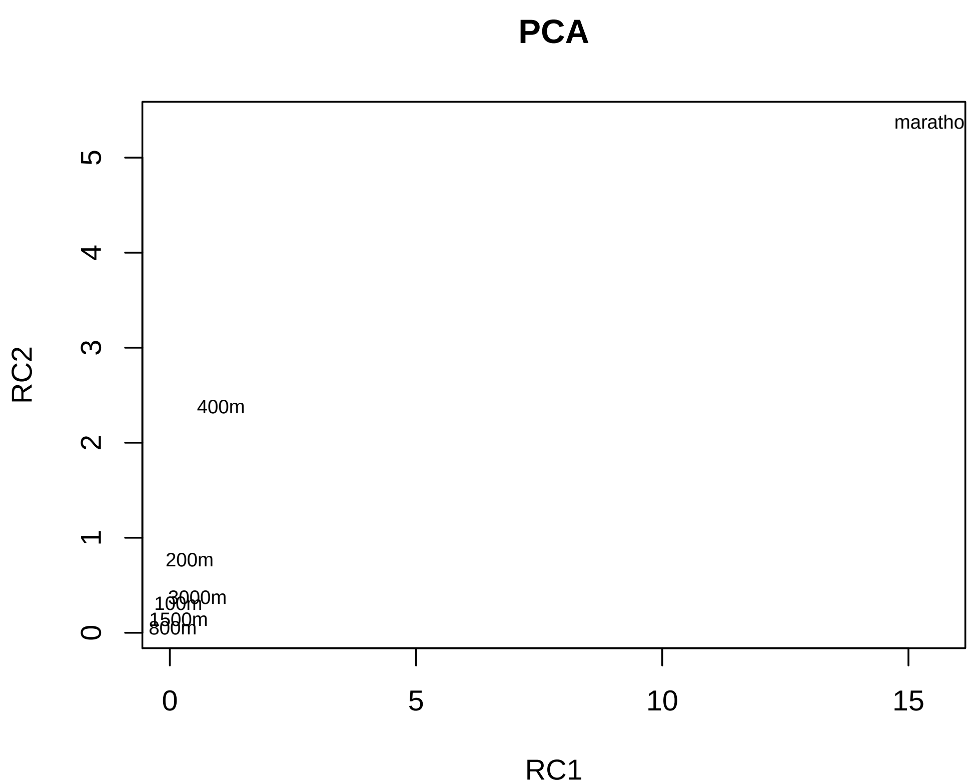
Question 2

```r
library(psych)

data <- read.table("../data/T1-9.dat")
names(data) <- c("country", "100m", "200m", "400m", "800m", "1500m", "3000m", "marathon")
numeric_data <- data[, -1]
countries <- as.character(data$country)

S <- cov(numeric_data)
R <- cor(numeric_data)
factors <- 2

print(S)
#>                100m       200m       400m         800m      1500m
#> 100m     0.15531572  0.3445608  0.8912960 0.027703564 0.08389119
#> 200m     0.34456080  0.8630883  2.1928363 0.066165898 0.20276331
#> 400m     0.89129602  2.1928363  6.7454576 0.181807932 0.50917683
#> 800m     0.02770356  0.0661659  0.1818079 0.007546925 0.02141457
#> 1500m    0.08389119  0.2027633  0.5091768 0.021414570 0.07418270
#> 3000m    0.23388281  0.5543502  1.4268158 0.061379315 0.21615514
#> marathon 4.33417757 10.3849876 28.9037314 1.219654647 3.53983732
#>                3000m   marathon
#> 100m      0.23388281   4.334178
#> 200m      0.55435017  10.384988
#> 400m      1.42681579  28.903731
#> 800m      0.06137932   1.219655
#> 1500m     0.21615514   3.539837
#> 3000m     0.66475793  10.706091
#> marathon 10.70609113 270.270150
```

Since the data is measured in different units it is more appropriate to use the correlation matrix. We can see that the covariances of marathon is huge compared to the other variables which will pose a problem.

Analysis on Covariance Matrix

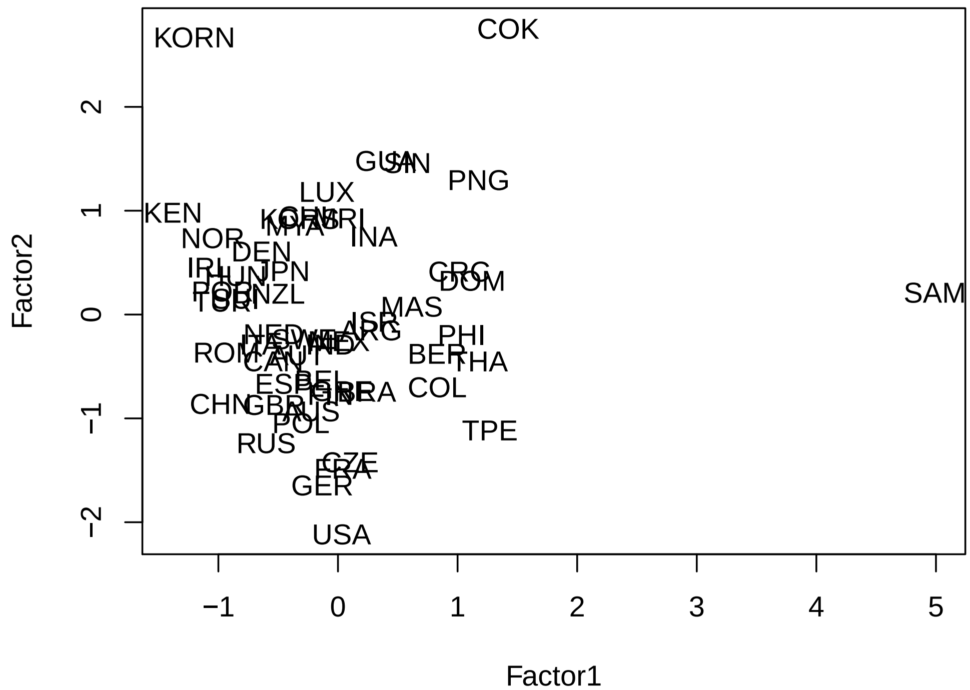




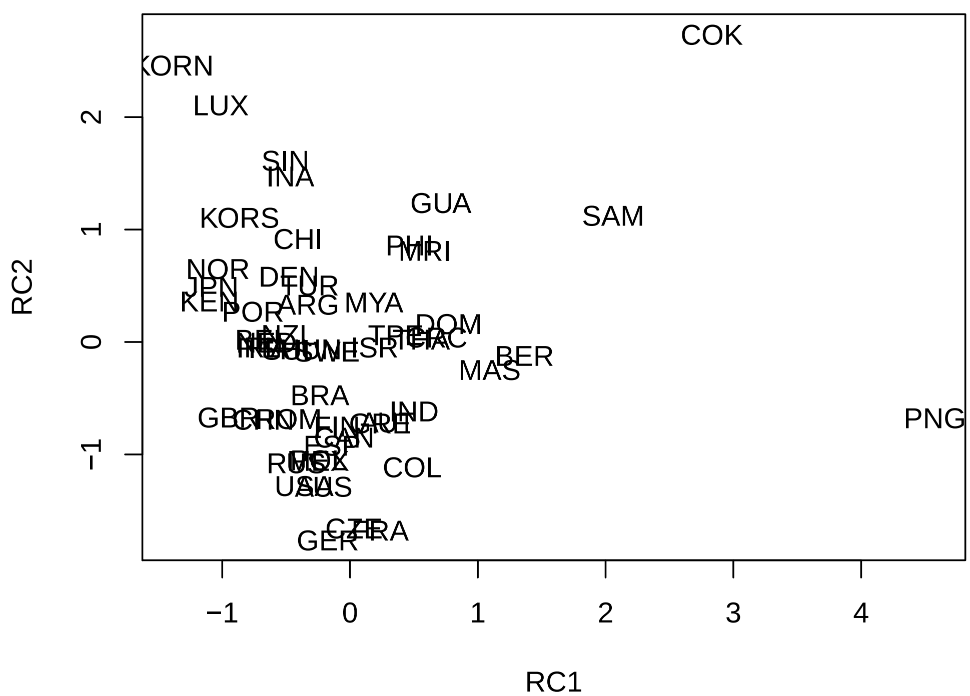
```
#> [1] "PCA"
#>
#> Loadings:
#>          RC1    RC2
#> 100m      0.173  0.307
#> 200m      0.404  0.765
#> 400m      1.038  2.376
#> 800m
#> 1500m     0.179  0.142
#> 3000m     0.561  0.371
#> marathon 15.537  5.375
#>
#>                    RC1    RC2
#> SS loadings    243.005 35.375
#> Proportion Var  34.715  5.054
#> Cumulative Var  34.715 39.768
#> [1] "FA"
#>
#> 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
```

We can see that the first principal component explains about 87% of the variance and the largest loading is associated with the marathon which is clear from the plot. The other component explains about 13% of the variance and is thus not very informative. These two components do not help us very much in understanding the nature of the data because we did not normalize the data.

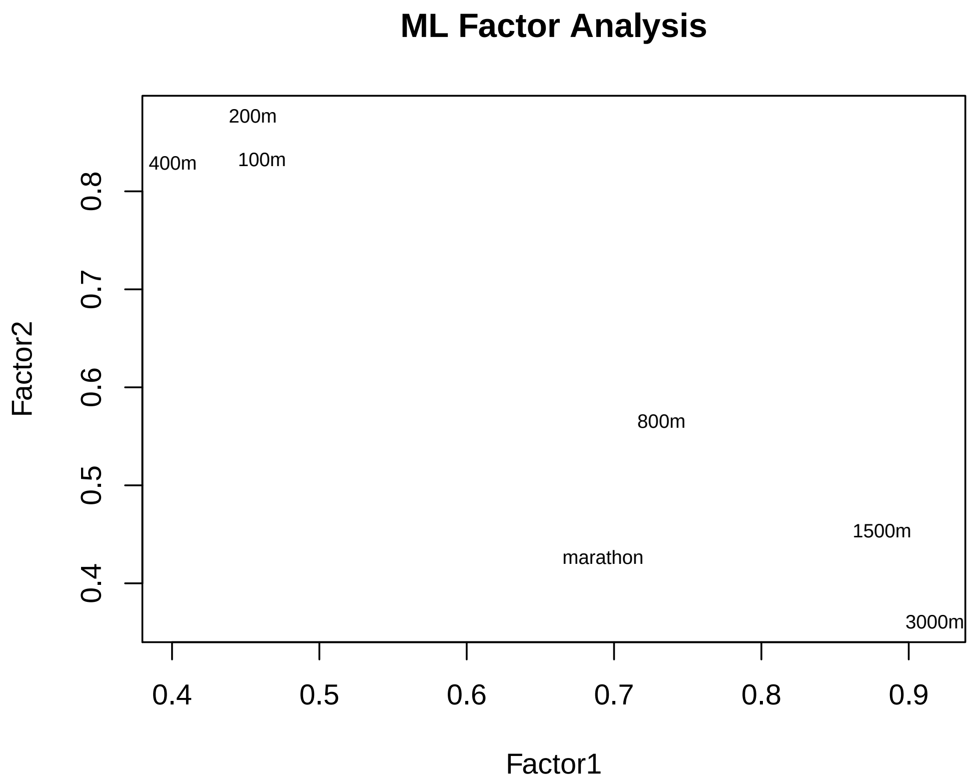
ML Factor Analysis

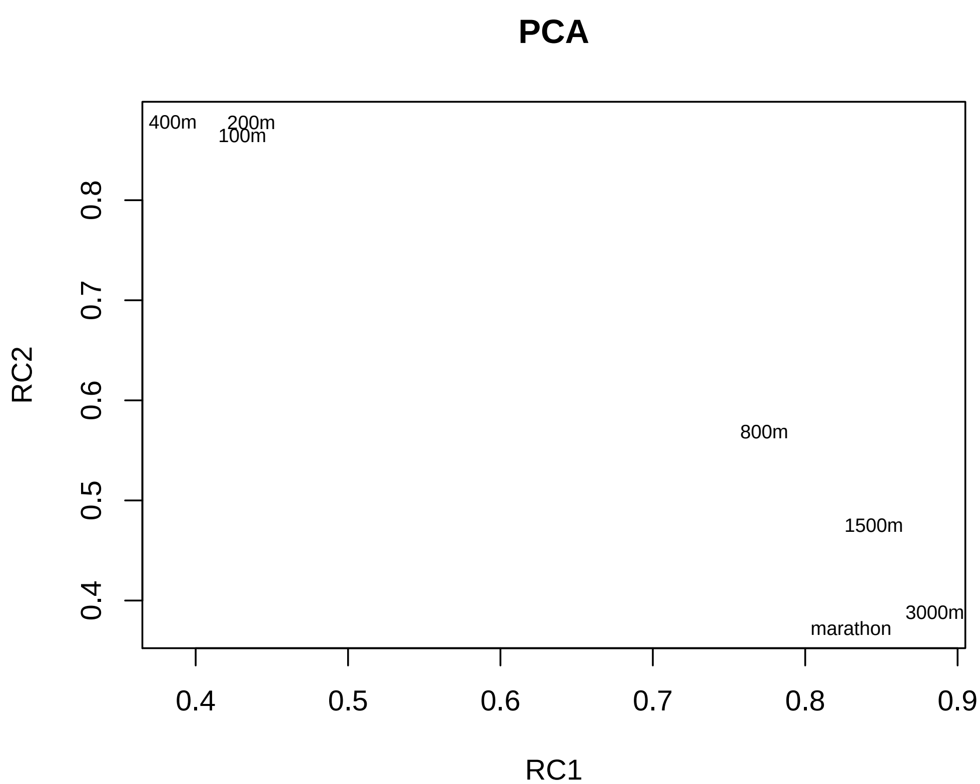


PCA



Analysis on Correlation Matrix

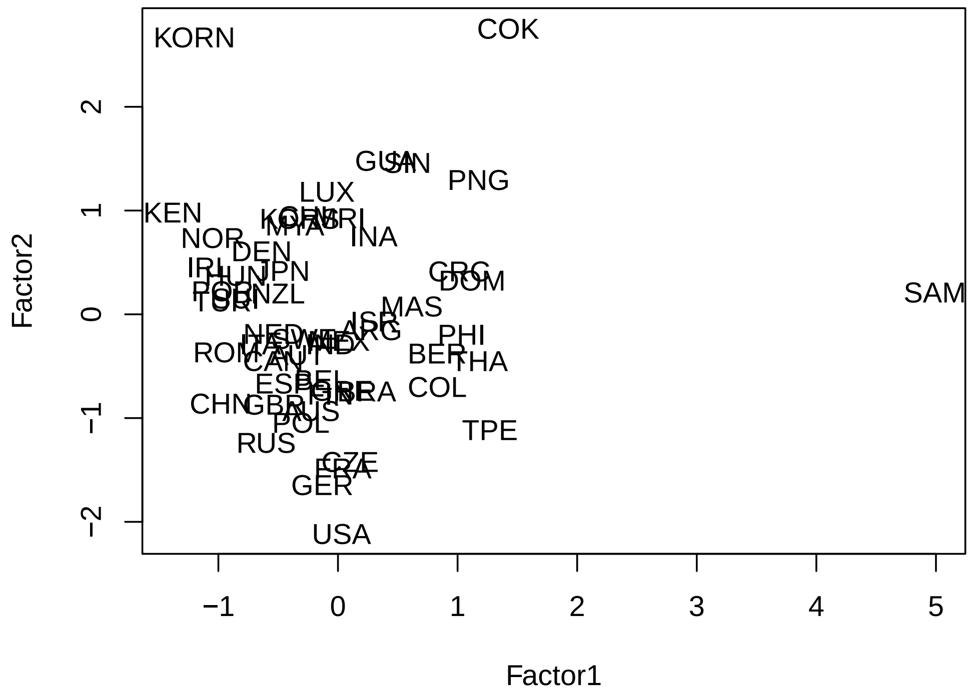




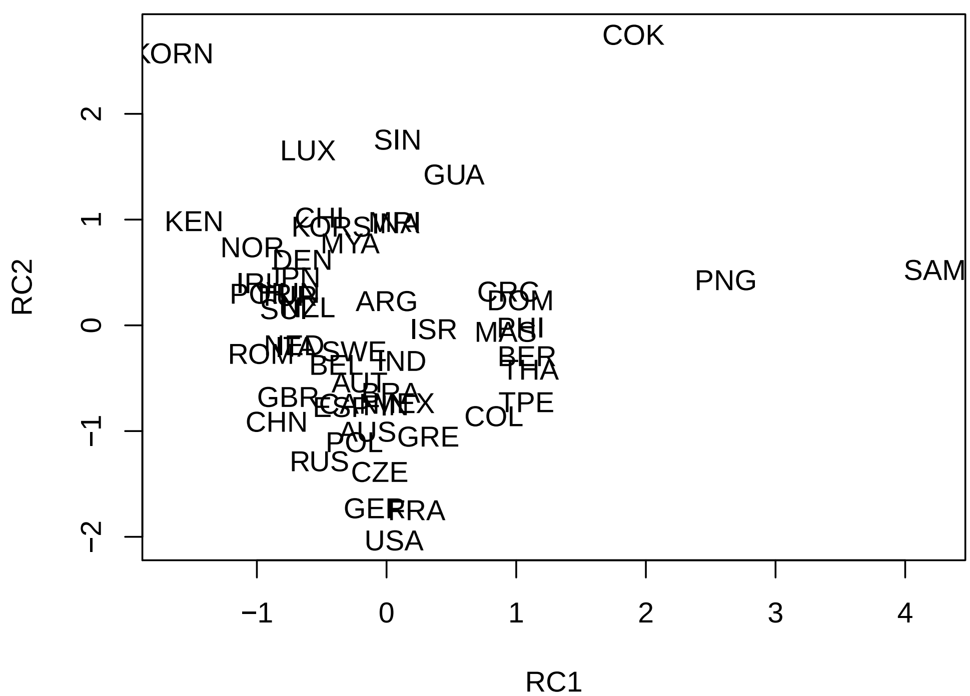
```
#> [1] "PCA"
#>
#> Loadings:
#>          RC1   RC2
#> 100m     0.431 0.865
#> 200m     0.437 0.877
#> 400m     0.385 0.878
#> 800m     0.773 0.569
#> 1500m    0.845 0.475
#> 3000m    0.885 0.388
#> marathon 0.830 0.373
#>
#>                  RC1   RC2
#> SS loadings    3.309 3.128
#> Proportion Var 0.473 0.447
#> Cumulative Var 0.473 0.919
#> [1] "FA"
#>
#> 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
```

Now the first two principal components explains about the same amount of variance and in total almost 92% so its a decent fit. Similar values are true for the factors and so the two solutions give similar results. The first factor/principal component seem to represent shorter races since those load highly on it and the other represent longer races, but the opposite loadings are still rather high. So these factors could be interpreted as representing speed versus endurance.

ML Factor Analysis



PCA



We can see from the plots that the factor and principal component scores indicate that North Korea, Cook Islands, Samoa, and Papua New Guinea are outliers.

Setting rotation to varimax means that the algorithm rotates the loadings such as to maximize their variances. As a result of this rotation, each variable loads more heavily on a single factor making the factors easier to interpret.

Note that we get the same results for the factor analysis on the covariance matrix as with the correlation matrix and that is because the factanal function internally normalizes the covariance matrix. If that was not the case we would get different result because factor analysis is trying to approximate the covariance matrix as $\Sigma = LL^T + \Psi$ and correlation matrix is a covariance matrix.

Appendix

Code

```
## Question 1
data<- read.table("T1-9.dat")
rownames(data)<- data[,1]
data<- data[,-1]
colnames(data)<- c("100m","200m","400m","800m","1500m","3000m","marathon")

#solve 8.18;
#####################
#a) obtain sample corr and determine eigenvalues and eigvectors
scaled<- scale(data)
#sample correlation
corrdata<- cor(scaled)

#egienvalues
eigenvalues<- eigen(corrdata)$values
eigenvalues
#eigenvectors in the columns
eigenvectors<- eigen(corrdata)$vectors
eigenvectors

##############
#b) determine first 2 princomp for standardized vars, table with corr and components,
# cumul percentage of total (standardized) sample var explained by 2 comps

#first princomp:
princomp_1 <- eigenvectors[,1]
princomp_1
#scnd princomp
princomp_2 <- eigenvectors[,2]
princomp_2

#cumulative percentage variance explained by 2 first comps
#is the sum of 2 first eigenvals divided by the sum of eigenvalues.
(eigenvalues[1] + eigenvalues[2])/sum(eigenvalues)

############
#c) interpret 2 comps from b. first might measure atletich excellence, scnd relative strength of nation
colnames(scaled)
princomp_1
princomp_2

#d) rank nations based on score from frst princomp. does this correspond to inituive notion of atletic
# excellence for various countries?

first_comp_scores<- matrix(0,nrow=nrow(scaled), ncol=2)
for (i in 1:nrow(scaled)){
  first_comp_scores[i,1]<-rownames(scaled)[i]
  first_comp_scores[i,2]<-as.numeric(sum(princomp_1*scaled[i,]))
}
```

```r
ordered_scores<- first_comp_scores[order(first_comp_scores[,2], decreasing = T),]
#something wrong with ordering, lowest are in the middle..
ordered_scores[33:40,]
#here we have the countries we have seen previous in labs that have had bad timeresults, and
#they also have lowest scores here, seems legit.
head(ordered_scores)
#USA, germany, russia seems to be countries with good results, sounds legit

## Question 2
library(psych)

data <- read.table("../data/T1-9.dat")
names(data) <- c("country", "100m", "200m", "400m", "800m", "1500m", "3000m", "marathon")
numeric_data <- data[, -1]
countries <- as.character(data$country)

S <- cov(numeric_data)
R <- cor(numeric_data)
factors <- 2

print(S)

S_principal <- principal(S, factors, rotate="varimax", covar=TRUE)
S_factanalysis <- factanal(numeric_data, factors=factors, covmat=S, rotation="varimax")

S_factoranalysis_loadings <- S_factanalysis$loadings[, 1:2]
S_principal_loadings <- S_principal$loadings[, 1:2]

old <- par(mfrow=c(2, 1))
plot(S_factoranalysis_loadings, type="n", main="ML Factor Analysis")
text(S_factoranalysis_loadings, labels=names(numeric_data), cex=.7)

plot(S_principal_loadings, type="n", main="PCA")
text(S_principal_loadings, labels=names(numeric_data), cex=.7)
par(old)
print("PCA")
S_principal$loadings

print("FA")
S_factanalysis$loadings
factor_scores <- factanal(numeric_data, factors=factors,
                          rotation="varimax", scores="regression")$scores
principal_scores <- principal(numeric_data, factors, scores=TRUE, covar=TRUE)$scores

old <- par(mfrow=c(2, 1))
plot(factor_scores, type="n", main="ML Factor Analysis")
text(factor_scores, labels=countries)

plot(principal_scores, type="n", main="PCA")
text(principal_scores, labels=countries)
par(old)

R_principal <- principal(R, factors, rotate="varimax", covar=FALSE)
```

```r
R_factanalysis <- factanal(numeric_data, factors=factors, covmat=R, rotation="varimax")

R_factoranalysis_loadings <- R_factanalysis$loadings[, 1:2]
R_principal_loadings <- R_principal$loadings[, 1:2]

old <- par(mfrow=c(2, 1))
plot(R_factoranalysis_loadings, type="n", main="ML Factor Analysis")
text(R_factoranalysis_loadings, labels=names(numeric_data), cex=.7)

plot(R_principal_loadings, type="n", main="PCA")
text(R_principal_loadings, labels=names(numeric_data), cex=.7)
par(old)
print("PCA")
R_principal$loadings

print("FA")
R_factanalysis$loadings
factor_scores <- factanal(numeric_data, factors=factors,
                          rotation="varimax", scores="regression")$scores
principal_scores <- principal(numeric_data, factors, scores=TRUE, covar=FALSE)$scores

old <- par(mfrow=c(2, 1))
plot(factor_scores, type="n", main="ML Factor Analysis")
text(factor_scores, labels=countries)

plot(principal_scores, type="n", main="PCA")
text(principal_scores, labels=countries)
par(old)
```