Lab3-Principle component and factor analysis

Andreas C Charitos[andch552]

2/21/2021

Contents

Problem 1

Data Overview

Table of the first 3 lines of the data

Table 1: First lines of national track data

Country	100m	200m	400m	800m	1500m	3000m	Marathon
ARG	11.57	22.94	52.50	2.05	4.25	9.19	150.32
AUS	11.12	22.23	48.63	1.98	4.02	8.63	143.51
AUT	11.15	22.70	50.62	1.94	4.05	8.78	154.35
BEL	11.14	22.48	51.45	1.97	4.08	8.82	143.05
BER	11.46	23.05	53.30	2.07	4.29	9.81	174.18
BRA	11.17	22.60	50.62	1.97	4.17	9.04	147.41

a)

Question: Obtain the sample correlation matrix R for these data and determine its eigenvalues and eigenvectors.

Correlation matrix

Table 2: Correlation matrix

	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

Eigenvalues

```
## Eigenvalues:
##  5.807624 0.6286934 0.2793346 0.1245547 0.09097174 0.05451882 0.01430226
```

Eigenvectors

Table 3: Eigenvectors

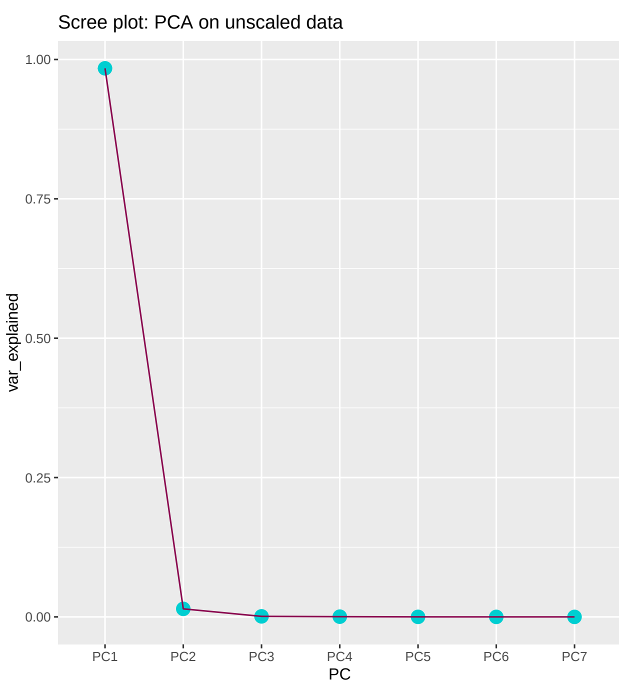
	COMP1	COMP2	COMP3	COMP4	COMP5	COMP6	COMP7
100m	-0.3777657	-0.4071756	-0.1405803	0.5870629	-0.1670689	0.5396973	0.0889393
200m	-0.3832103	-0.4136291	-0.1007833	0.1940750	0.0935002	-0.7449314	-0.2656566
400m	-0.3680361	-0.4593531	0.2370255	-0.6454312	0.3272733	0.2400940	0.1266044
800m	-0.3947810	0.1612459	0.1475424	-0.2952080	-0.8190547	-0.0165065	-0.1952131
1500m	-0.3892610	0.3090877	-0.4219855	-0.0666904	0.0261310	-0.1889877	0.7307682
3000m	-0.3760945	0.4231899	-0.4060627	-0.0801570	0.3516980	0.2404997	-0.5715064
Marathon	-0.3552031	0.3892153	0.7410610	0.3210764	0.2470082	-0.0482699	0.0820840

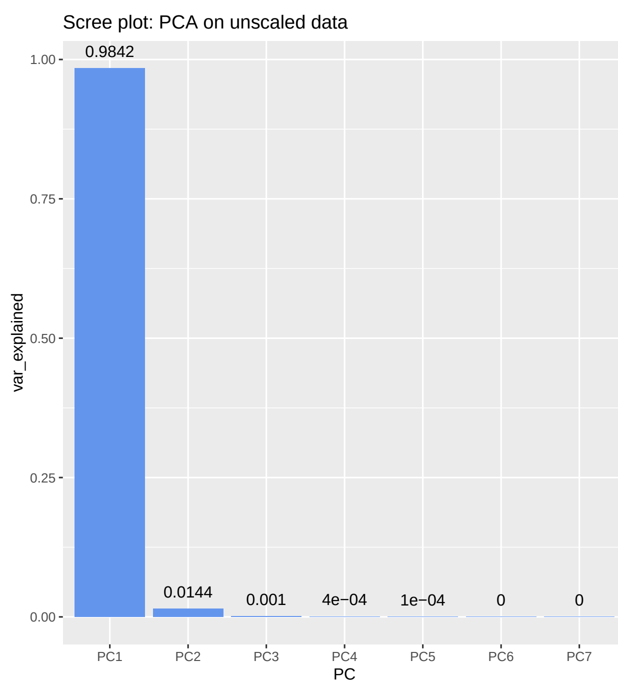
Variance importance

PC	var_explained
PC1	0.9841530
PC2	0.0144080
PC3	0.0009623
PC4	0.0004109
PC5	0.0000543
PC6	0.0000093

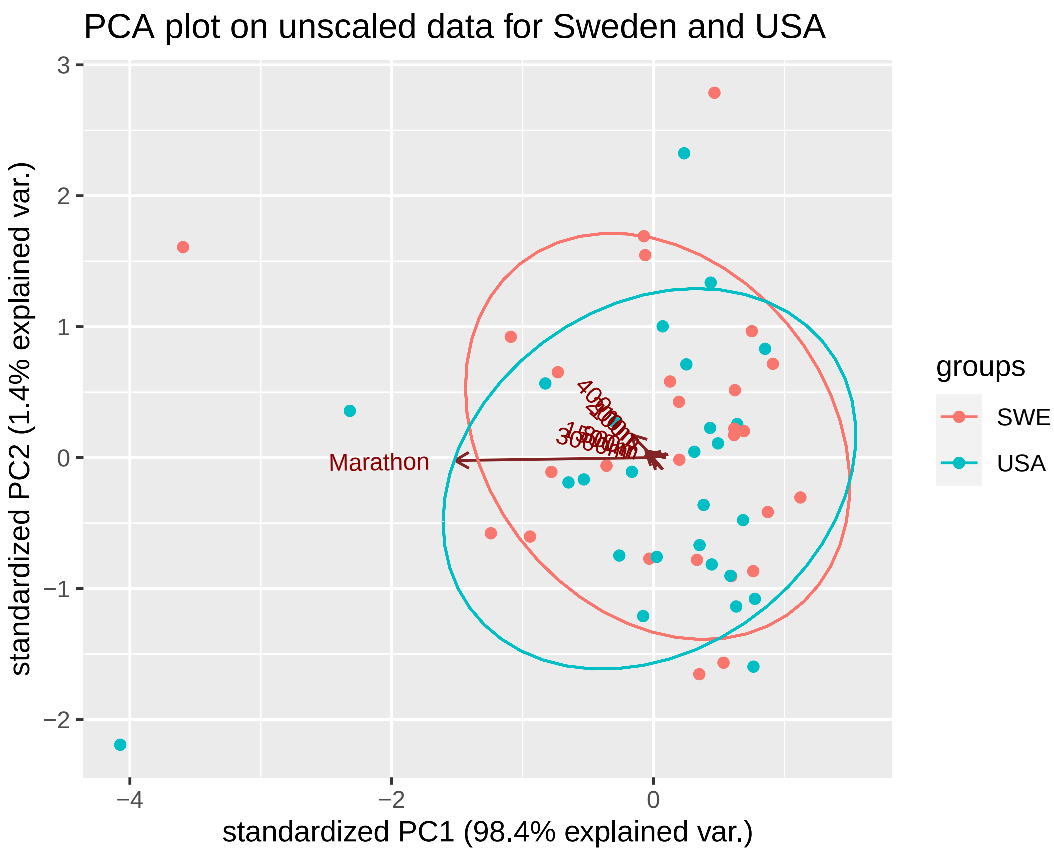
```
## Cumulative percentage of the total variance explained by the first two components:  99.85611 %
```

Scree plots





PCA plot



b)

Question: Determine the first two principal components for the standardized variables. Prepare a table showing the correlations of the standardized variables with the components and the cumulative percentage of total (standardized) sample variance explained by the two components.

First two principal components on scaled data

```
## ============================================================

## First principal component for scaled variables:
##  -0.3777657 -0.3832103 -0.3680361 -0.394781 -0.389261 -0.3760945 -0.3552031

## ============================================================

## Second  principal component for scaled variables:
##  -0.4071756 -0.4136291 -0.4593531 0.1612459 0.3090877 0.4231899 0.3892153
```

Correlation of standardized variables with components

Table 5: Correlation of standardized variables with components

Comp1	Comp2
-0.9103780	-0.3228503
-0.9234990	-0.3279673
-0.8869307	-0.3642220
-0.9513832	0.1278522
-0.9380805	0.2450762
-0.9063506	0.3355481
-0.8560043	0.3086096

Correlation matrix on scaled data

Table 6: Correlation matrix scaled variables

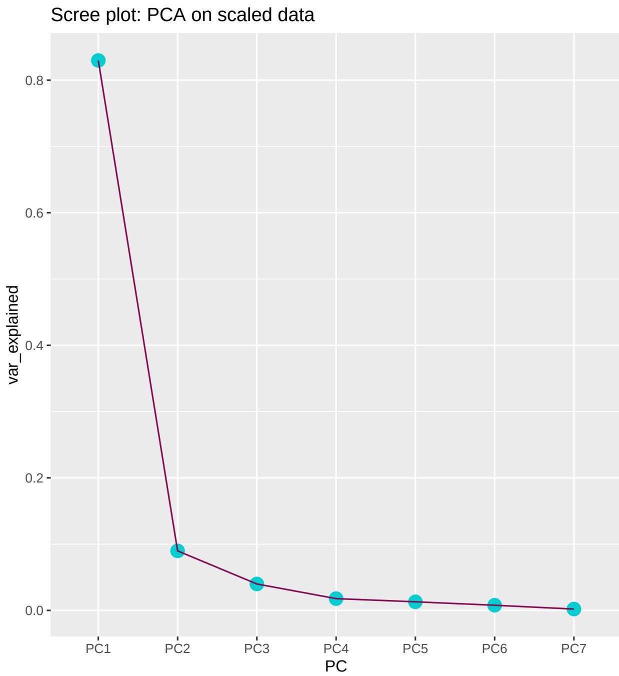
	100m	200m	400m	800m	1500m	3000m	Marathon
100m	1.0000000	0.9410886	0.8707802	0.8091758	0.7815510	0.7278784	0.6689597
200m	0.9410886	1.0000000	0.9088096	0.8198258	0.8013282	0.7318546	0.6799537
400m	0.8707802	0.9088096	1.0000000	0.8057904	0.7197996	0.6737991	0.6769384
800m	0.8091758	0.8198258	0.8057904	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

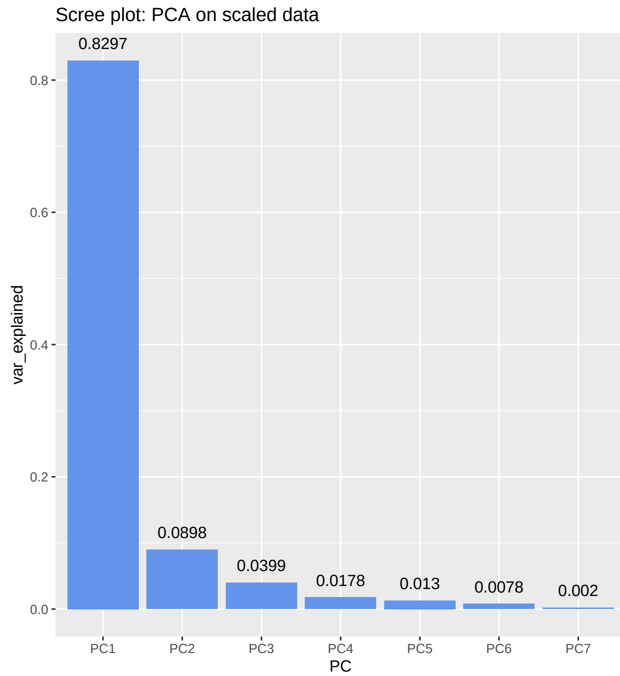
Variance importance

PC	var_explained
PC1	0.8296606
PC2	0.0898133
PC3	0.0399049
PC4	0.0177935
PC5	0.0129960
PC6	0.0077884

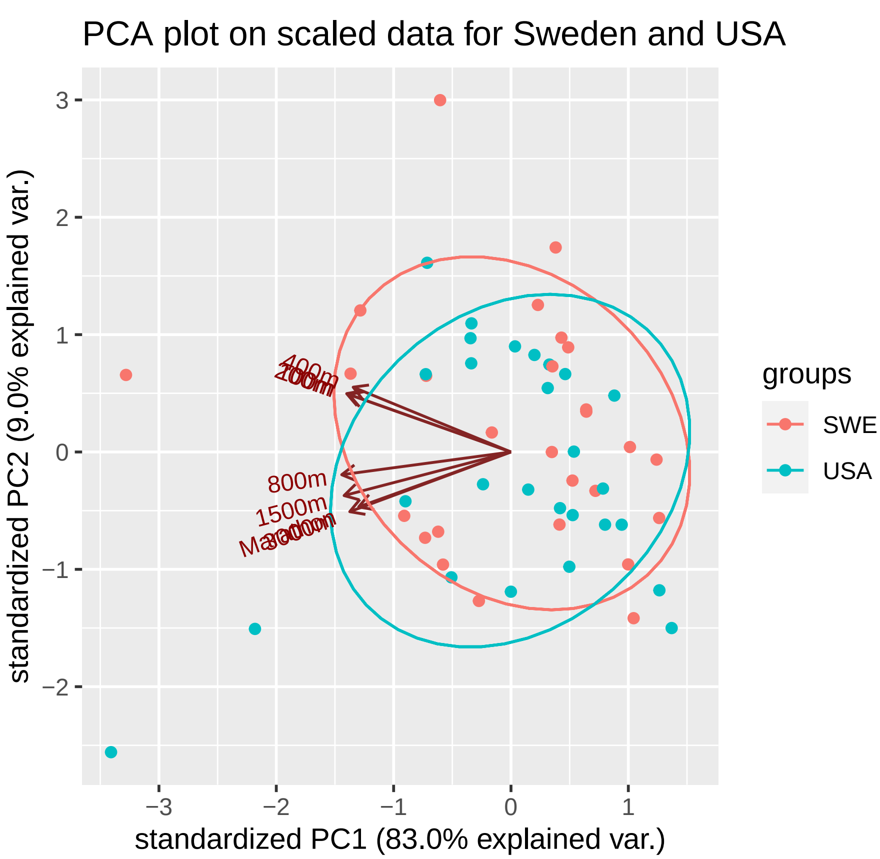
```
## Cumulative percentage of the total variance explained by the first two components:  91.9474 %
```

Scree plots





PCA plot



c)

Question: Interpret the two principal components obtained in Part b. (Note the first component is essentially a normalized unit vector and might measure the athletic excellence of a ginven nation. The second component might measute reative strength of a nation at the various running distances.)

Answer: Most of the values of the first components are pretty close. In some sense, this component measures the average time on each of the tracks. So its an equally weighted performance measure. The second component seems to be a measure of strenght regarding the distance of the runs. If the new component Y is positive, it means that nation better at shorter distances while if it's negative, it means that it performs better at longer distances.

d)

Question: Rank the nations based on their score on the first principal component. Does this ranking correspond with your intuitive notion of athletic excellence for the various countries?

Score ranking

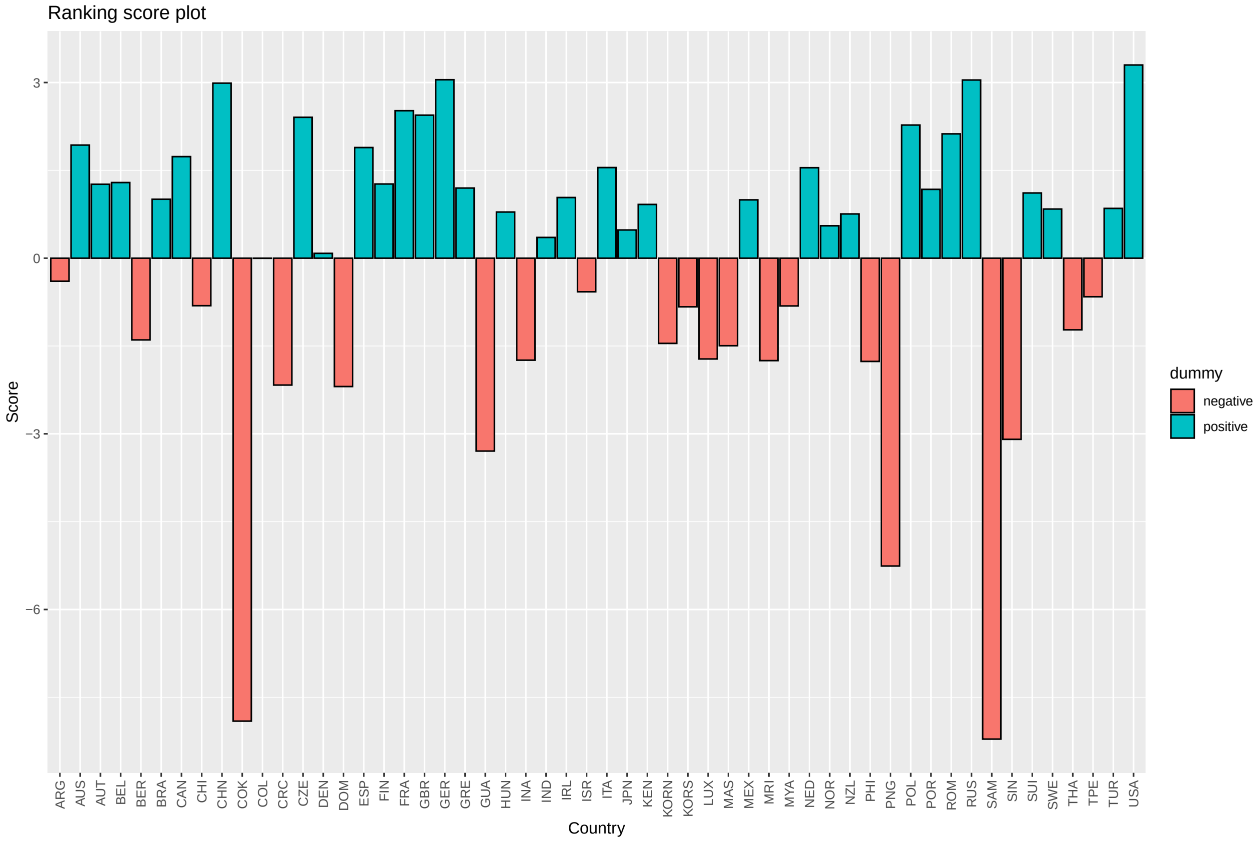
Table 8: Top 10 countries

	Country	Score
54	USA	3.299149
18	GER	3.047517
45	RUS	3.042948
9	CHN	2.989467
17	FRA	2.518346
19	GBR	2.442706
13	CZE	2.406030
42	POL	2.273766
44	ROM	2.123006
2	AUS	1.931643

Table 9: Last 10 countries

	Country	Score
32	LUX	-1.721468
23	INA	-1.741942
34	MRI	-1.749728
41	PHI	-1.763534
12	CRC	-2.166812
15	DOM	-2.192410
47	SIN	-3.093920
21	GUA	-3.294124
40	PNG	-5.257450
11	COK	-7.906227
46	SAM	-8.213415

This ranking makes sense since the countries on top are mostly developed nations who always perform well on sports while the ones at the bottom are underdeveloped nations that always lack performance on competitive sports.



Problem 2

Question: Perform a factor analysis of the national track records for women data given on the previous table. Use the sample covariance matrix S and interpret the factors. Compute factor scores and check for outliers in the data. Repeat the analysis with the sample correlation matrix R. Does it make a difference if R, rather than S, is factored?. Explain.

a) Analysis for covariance matrix S

Maximum likelihood for covariance matrix

```
##
## Call:
## factanal(x = dt[, 2:8], factors = 2, covmat = cov(dt[, 2:8]))
##
## 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
##
## The degrees of freedom for the model is 8 and the fit was 0.6481
```

We can see that both components are explain approximately the same amount of variance. Looking at the values we see that factor1 puts more emphasis on longer distances (800m and more), while factor2 focuses mainly on shorter distances (400m and less). The model seems consistent as both components explain around 89 percent of the variance.

PCA for covariance matrix

```
## Principal Components Analysis
## Call: principal(r = cov(dt[, 2:8]), nfactors = 2, covar = T)
## Unstandardized loadings (pattern matrix) based upon covariance matrix
##            RC1  RC2      h2      u2   H2      U2
## 100m      0.17 0.31 1.2e-01 0.03100 0.80 2.0e-01
## 200m      0.40 0.77 7.5e-01 0.11435 0.87 1.3e-01
## 400m      1.04 2.38 6.7e+00 0.02014 1.00 3.0e-03
## 800m      0.06 0.05 6.3e-03 0.00126 0.83 1.7e-01
## 1500m     0.18 0.14 5.2e-02 0.02200 0.70 3.0e-01
## 3000m     0.56 0.37 4.5e-01 0.21213 0.68 3.2e-01
## Marathon 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
## 100m        1 0.44 0.78 0.80 2.0e-01
## 200m        2 0.43 0.82 0.87 1.3e-01
## 400m        3 0.40 0.92 1.00 3.0e-03
## 800m        4 0.70 0.58 0.83 1.7e-01
## 1500m       5 0.66 0.52 0.70 3.0e-01
## 3000m       6 0.69 0.46 0.68 3.2e-01
## Marathon    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
##
## Fit based upon off diagonal values = 1
```

```
##
## 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
```

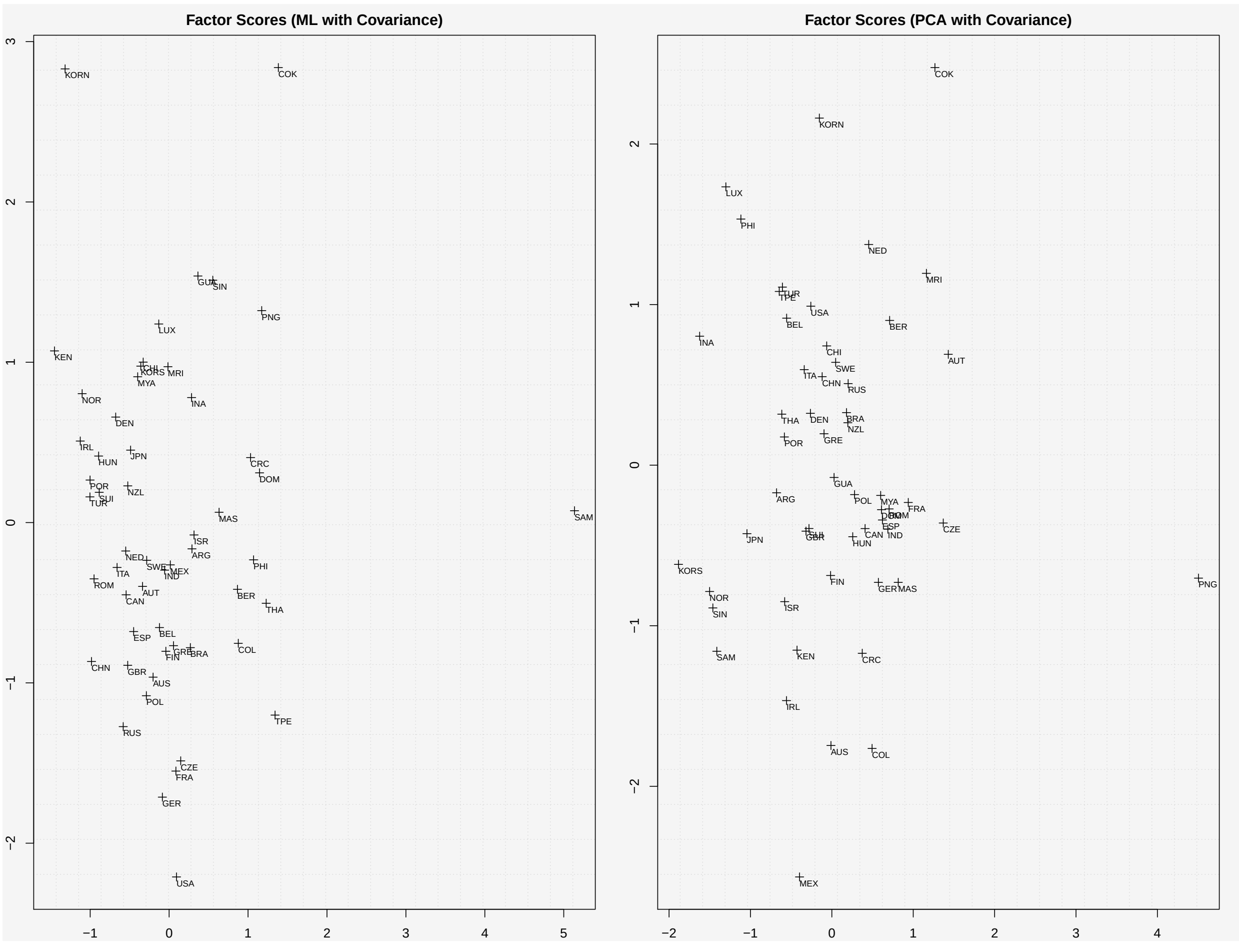
This time it seems that both factors focus quite a lot on the 400m track and the marathon. Component 2 puts a but more emphasis on the shorter tracks. So it seems like these components provide less interpretability. Looking at the difference we see that both components together explain less than 50 percent of the variance, which means that we lose quite a lot of the original variance. This also explains why it is quite hard to explain the components in terms of the given data.

Loadings

Table 10: Loadings for covariance matrix S from factor analysis ML and PCA

	Factor1	Factor2	RC1	RC2
100m	0.4610766	0.8325215	0.1727014	0.3073853
200m	0.4548942	0.8768695	0.4038312	0.7652819
400m	0.4005507	0.8291157	1.0381777	2.3764474
800m	0.7321983	0.5655217	0.0609929	0.0506720
1500m	0.8819073	0.4538146	0.1787984	0.1421843
3000m	0.9177249	0.3605910	0.5611987	0.3710620
Marathon	0.6925367	0.4269358	15.5365165	5.3746209

Outliers plots covariance matrix



According to Factors scores plot(ML with covariance)

- SAM is extreme
- KORN and COK are outiers
- USE and KEN are on the boarder

According to Factors scores plot(PCA with covariance)

- PNG is extreme
- MEX and COK are outliers

b) Analysis for the correlation matrix R

Maximum likelihood for correlation matrix

```
##
## Call:
## factanal(x = dt[, 2:8], factors = 2, covmat = cor(dt[, 2:8]))
##
## 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
##
## The degrees of freedom for the model is 8 and the fit was 0.6481
```

Comparing with using the covariance matrix, we get the same explanation for the variables. Component 1 is still responsible for the longer tracks while component 2 is responsible for the shorter tracks. Both explain almost 90 percent of the variance which is quite good. As it seems there is no difference at all. Although, it seems there is a difference between using the covariance or correlation using the ML method.

PCA for correlation matrix

```
## Principal Components Analysis
## Call: principal(r = cor(dt[, 2:8]), nfactors = 2, covar = FALSE, scores = TRUE)
## Standardized loadings (pattern matrix) based upon correlation matrix
##           RC1  RC2   h2    u2 com
## 100m     0.43 0.86 0.93 0.067 1.5
## 200m     0.44 0.88 0.96 0.040 1.5
## 400m     0.39 0.88 0.92 0.081 1.4
## 800m     0.77 0.57 0.92 0.079 1.8
## 1500m    0.85 0.48 0.94 0.060 1.6
## 3000m    0.89 0.39 0.93 0.066 1.4
## Marathon 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
##
## Fit based upon off diagonal values = 1
```

```
##
## 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
```

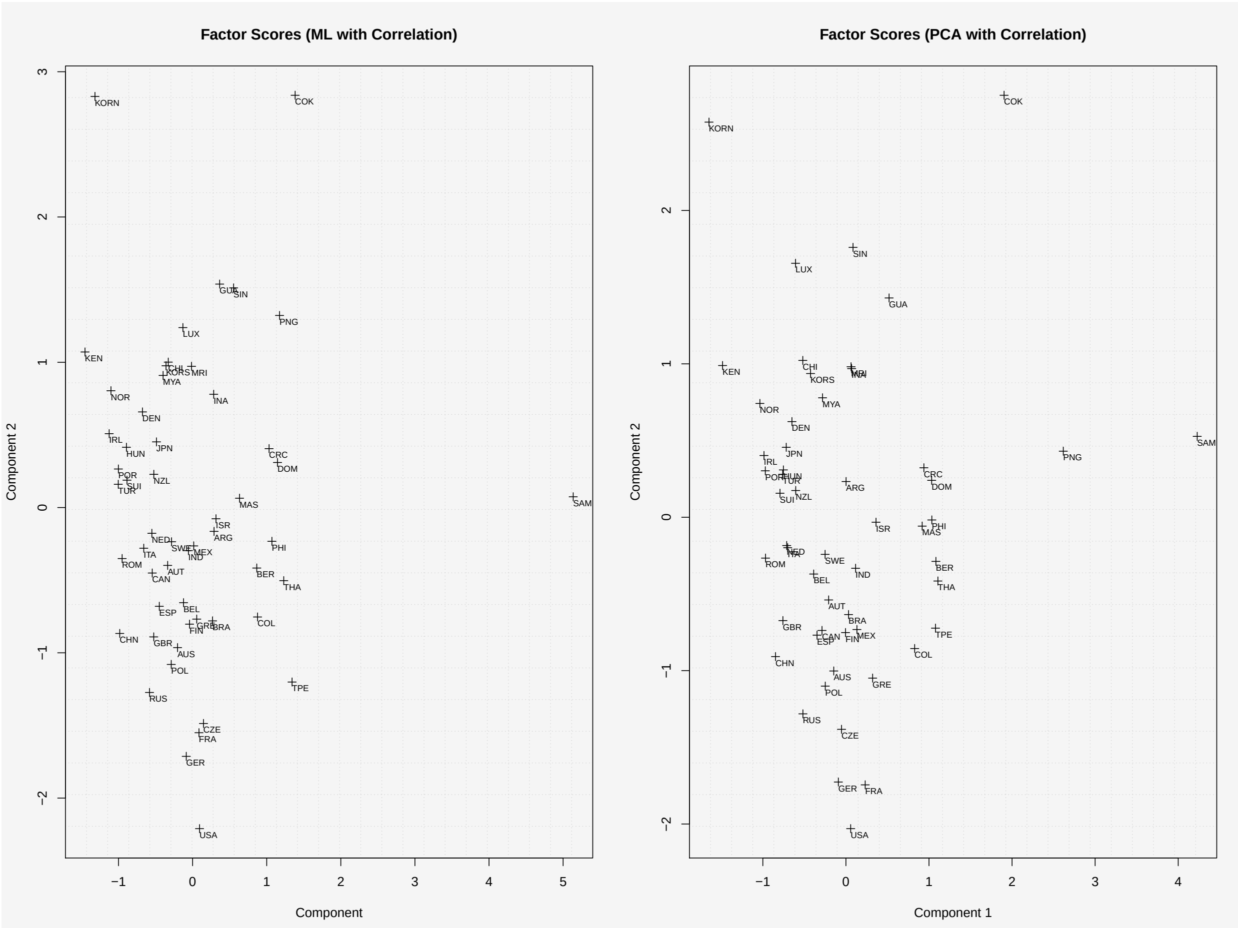
As we see the values changed and it looks the we almost found the same components compared to the ML method in both cases (using S and R). Component 1 is again explaining the variance for the longer tracks and components to the variance for the shorter tracks. We see a slight improvement for the cumulative variance, as we actually explain more than 90 percent of the variance.

Loadings

Table 11: Loadings for correlation matrix R from factor analysis ML and PCA

	Factor1	Factor2	RC1	RC2
100m	0.4610766	0.8325215	0.4306309	0.8646256
200m	0.4548942	0.8768695	0.4365153	0.8774209
400m	0.4005507	0.8291157	0.3850257	0.8780996
800m	0.7321983	0.5655217	0.7732014	0.5688899
1500m	0.8819073	0.4538146	0.8450596	0.4753226
3000m	0.9177249	0.3605910	0.8850790	0.3881997
Marathon	0.6925367	0.4269358	0.8301493	0.3726062

Outliers plots correlation matrix



According to Factors scores plot(ML with correlation)

- SAM is extreme
- KORN and COK are outiers

According to Factors scores plot(PCA with correlation)

- SAM, PNG, COK are extreme
- KORN is an outlier
- KEN seem to bee on boarder

Appendix

```r
## ----message=FALSE, echo=F-----------------------------------------------------
# import libraries ------------------------------------------------------
library(knitr)
library(corrplot)
library(ggbiplot)
library(tidyverse)
library(gridExtra)
library(psych)
knitr::opts_chunk$set(echo = F)


## ----------------------------------------------------------------------------
# Problem 1 -------------------------------------------------------------

# a) --------------------------------------------------

dt = read.table("T1-9.dat")
colnames(dt) = c("Country", "100m", "200m", "400m", "800m", "1500m", "3000m", "Marathon")
kable(head(dt), caption = "First lines of national track data")



## ----------------------------------------------------------------------------
# sample correlation matrix
dt_corr = cor(dt[, 2:8])
dt_eigen = eigen(dt_corr)


## ----------------------------------------------------------------------------
# correlation matrix
kable(dt_corr, caption="Correlation matrix")


## ----------------------------------------------------------------------------
# eigen values
cat("Eigenvalues: \n", dt_eigen$values)


## ----------------------------------------------------------------------------
# eigen vectors
dt_eigen_vectors = dt_eigen$vectors
row.names(dt_eigen_vectors) = colnames(dt[,2:8])
colnames(dt_eigen_vectors) = c("COMP1", "COMP2", "COMP3","COMP4","COMP5","COMP6","COMP7")

kable(dt_eigen_vectors, caption = "Eigenvectors")



## ----------------------------------------------------------------------------
# variance importance
pca_obj <- prcomp(dt[,2:8], scale. = F)
```

```r
var_explained_dt <- data.frame(PC= paste0("PC",1:7),
                               var_explained=(pca_obj$sdev)^2/sum((pca_obj$sdev)^2))

kable(head(var_explained_dt))



## ----------------------------------------------------------------------------
cat("Cumulative percentage of the total variance explained by the first two components: ",
    sum(var_explained_dt$var_explained[1:2])*100,"%")


## ----fig.height=6.5, fig.width=12, position="h"------------------------------
# scree plots
g1 = var_explained_dt %>%
  ggplot(aes(x=PC,y=var_explained, group=1))+
  geom_point(size=4, col='darkturquoise')+
  geom_line(col='deeppink4')+
  labs(title="Scree plot: PCA on unscaled data")

g2 = var_explained_dt %>%
  ggplot(aes(x=PC,y=var_explained))+
  geom_col(fill="cornflowerblue")+
  labs(title="Scree plot: PCA on unscaled data")+
   geom_text(aes(label = round(var_explained,4), vjust = -1))

grid.arrange(g1, g2, ncol = 2)



## ---- fig.align="center"-----------------------------------------------------
## PCA computation
dt.pca = dt %>%
  select(2:8) %>%
  prcomp(scale. = F, center = TRUE)

dt.pca %>%
  ggbiplot::ggbiplot(scale = 1,
                     groups=dt$Country[which(unique(dt$Country)%in%c("SWE","USA"))],
                     ellipse = T)+labs(title = "PCA plot on unscaled data for Sweden and USA")



## ----------------------------------------------------------------------------
dt_scaled = scale(dt[,2:8])
dt_scaled_corr = cor(dt_scaled)
dt_scaled_eigen = eigen(dt_scaled_corr)
cat("============================================================\n")
cat("First principal component for scaled variables: \n", dt_scaled_eigen$vectors[,1],"\n")
cat("============================================================\n")
cat("Second  principal component for scaled variables: \n", dt_scaled_eigen$vectors[,2])
```

```r
## ---------------------------------------------------------------------------
# ----------------------------------------------
# calculate correlation of std. variables with components
e = matrix(dt_eigen$vectors[,1:2], ncol=2)
l = diag(dt_eigen$values[1:2]%>%sqrt())
cor_mat = as.matrix(e%*%l)
cor_dt = as.data.frame(cor_mat)
colnames(cor_dt) = c('Comp1', 'Comp2')

kable(cor_dt, caption ='Correlation of standardized variables with components')



## ---------------------------------------------------------------------------
kable(dt_scaled_corr, caption="Correlation matrix scaled variables")



## ---------------------------------------------------------------------------
# ----------------------------------------------
pca_obj.scaled <- prcomp(dt[,2:8], scale. = T)
var_explained_dt.scaled <- data.frame(PC= paste0("PC",1:7),
                              var_explained=(pca_obj.scaled$sdev)^2/sum((pca_obj.scaled$sdev)^2))

kable(head(var_explained_dt.scaled))



## ---------------------------------------------------------------------------
cat("Cumulative percentage of the total variance explained by the first two components: ",
    sum(var_explained_dt.scaled$var_explained[1:2])*100,"%")


## ---- fig.height=6.5, fig.width=12,position="h"------------------------------
g1 = var_explained_dt.scaled %>%
  ggplot(aes(x=PC,y=var_explained, group=1))+
  geom_point(size=4, col='darkturquoise')+
  geom_line(col='deeppink4')+
  labs(title="Scree plot: PCA on scaled data")

g2 = var_explained_dt.scaled %>%
  ggplot(aes(x=PC,y=var_explained))+
  geom_col(fill="cornflowerblue")+
  labs(title="Scree plot: PCA on scaled data")+
   geom_text(aes(label = round(var_explained,4), vjust = -1))


grid.arrange(g1, g2, ncol = 2)



## ---- fig.align="center"----------------------------------------------------
## PCA computation
```

```r
dt_scaled.pca = dt %>%
  select(2:8) %>%
  prcomp(scale. = TRUE, center = TRUE)


dt_scaled.pca %>%
  ggbiplot::ggbiplot(scale = 1,
                     groups=dt$Country[which(unique(dt$Country)%in%c("SWE","USA"))],
                     ellipse = T)+labs(title = "PCA plot on scaled data for Sweden and USA")



## ----------------------------------------------------------------------------
Y_1 = as.matrix(dt_scaled) %*% dt_scaled_eigen$vectors[, 1]
rank = list(Country=dt$Country, Score=Y_1)
rank = data.frame(rank)
ordered_idxs = order(rank$Score, decreasing=TRUE)
ordered_rank = rank[ordered_idxs, ]
# top 10 countries
#  cat("Top 10 countries: \n")
kable(ordered_rank[1:10,], caption = "Top 10 countries")



## ----------------------------------------------------------------------------
# last 10 countries
# cat("Last 10 countries: \n")
kable(ordered_rank[44:54,], caption = "Last 10 countries")



## ---- fig.width=12,fig.height=8---------------------------------------------
ordered_rank$dummy<-ifelse(ordered_rank$Score>0,"positive","negative")

ggplot(ordered_rank,aes(Country,Score,fill=dummy))+
  geom_bar(stat="identity",col="black")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(title = "Ranking score plot")



## ----------------------------------------------------------------------------

# Maximum likelihood estimation and PCA for S
fit_fact_S = factanal(dt[,2:8],
                      factors=2,
                      covmat = cov(dt[, 2:8]))

fit_pca_S = principal(cov(dt[, 2:8]), nfactors=2, covar=T)

# Maximum likelihood estimation and PCA for R
fit_fact_R = factanal(dt[,2:8],
                      factor=2,
```

```r
                      covmat = cor(dt[,2:8]))

fit_pca_R = principal(cor(dt[,2:8]),nfactors = 2, covar = FALSE, scores = TRUE)



## ----------------------------------------------------------------------------
fit_fact_S

# cat ("Degres of freedom: ",fit_fact_S$dof,
#       "and fit for model is: ", fit_fact_S$criteria[1])

# scores
fit_fact_S_scores = factanal(dt[,2:8],
                             factors=2,
                             scores="Bartlett")$scores


## ----------------------------------------------------------------------------
fit_pca_S

fit_pca_S$loadings

fit_pca_S_scores = factor.scores(dt[,2:8],
                                 f=fit_pca_S)$scores


## ----------------------------------------------------------------------------
cov_mat_loadings = cbind(fit_fact_S$loadings,fit_pca_S$loadings)

kable(cov_mat_loadings, caption = "Loadings for covariance matrix S from factor analysis ML and PCA")



## ---- fig.width=16, fig.height=12--------------------------------------------

par(bg="whitesmoke",mfrow=c(1,2),mar=c(2,2,2,2))
# factor scores plot ML
plot(fit_fact_S_scores[,1], fit_fact_S_scores[,2],
     pch=3,
     panel.first = grid(25,25),
     main="Factor Scores (ML with Covariance)",
     xlab = "Component 1", ylab = "Component 2")
text(x = fit_fact_S_scores[,1],
     y = fit_fact_S_scores[,2],
     labels = dt[,1], adj = c(0,1.5),cex=0.7)
# text(3, 2, "#SAM is extreme\n#KORN and COK are outiers\n#USE and KEN are on the boarder")
# We can see that the outliers are "SAM", "KORN" and "COK"

# factors scores plot PCA
plot(fit_pca_S_scores[,1], fit_pca_S_scores[,2],
     pch=3,
     panel.first = grid(25,25),
```

```
     main="Factor Scores (PCA with Covariance)",
     xlab = "Component 1", ylab = "Component 2")
text(x = fit_pca_S_scores[,1],
     y = fit_pca_S_scores[,2],
     labels = dt[,1], adj = c(0,1.5),cex=0.7)
# text(3, 2, "#SAM is extreme\n#KORN and COK are outiers\n#USE and KEN are on the boarder")
# The outliers this time are "MEX", "COK" and "PNG".


## ----------------------------------------------------------------------------

fit_fact_R
#
# cat ("Degres of freedom: ",fit_fact_R$dof,
#       "and fit for model is: ", fit_fact_R$criteria[1])

# scores
fit_fact_R_scores = factanal(dt[,2:8],
                             factors=2,
                             scores="Bartlett")$scores



## ----------------------------------------------------------------------------
fit_pca_R

fit_pca_R$loadings

fit_pca_R_scores = factor.scores(dt[,2:8],
                                 f=fit_pca_R)$scores




## ----------------------------------------------------------------------------

cov_mat_loadings = cbind(fit_fact_R$loadings,fit_pca_R$loadings)

kable(cov_mat_loadings, caption = "Loadings for correlation matrix R from factor analysis ML and PCA")



## ----fig.width=16, fig.height=12-----------------------------------------------

par(bg="whitesmoke",mfrow=c(1,2))
# factor scores plot ML
plot(fit_fact_R_scores[,1], fit_fact_R_scores[,2],
     pch=3,
     panel.first = grid(25,25),
     main="Factor Scores (ML with Correlation)",
     xlab = "Component", ylab = "Component 2")
text(x = fit_fact_R_scores[,1],
     y = fit_fact_R_scores[,2],
     labels = dt[,1], adj = c(0,1.5),cex=0.7)
```

```r
# We can see that the outliers are "SAM", "KORN" and "COK"

# factors scores plot PCA
plot(fit_pca_R_scores[,1], fit_pca_R_scores[,2],
     pch=3,
     panel.first = grid(25,25),
     main="Factor Scores (PCA with Correlation)",
     xlab = "Component 1", ylab = "Component 2")
text(x = fit_pca_R_scores[,1],
     y = fit_pca_R_scores[,2],
     labels = dt[,1], adj = c(0,1.5),cex=0.7)
# The outliers this time are "MEX", "COK" and "PNG".



## ----code=readLines(knitr::purl("/home/quartermaine/Courses/Multivariate-Statistical-Methods/labs/Ass
## NA
```