732A97 - Group 19

Lab 3 - "Principal component analysis and factor analysis"

Alexander Karlsson (aleka769)
Ruben Munoz (rubmu773)
Mariano Mariani (marma330)
Raymond Sseguya (rayss753)

2019-12-08

Contents

Packages used can be seen below.

```
library(dplyr)
library(psych)
library(ggplot2)

theme_set(theme_bw())
```

Read data:

```
trackData = read.table(file        = "T1-9.dat",
                       col.names   = c("Country", "100m", "200m", "400m",
                                       "800m", "1500m", "3000m", "42000m"),
                       check.names = FALSE)
```

Note: All code is displayed in the document, therefore we have no appendix.

Question 1 - Principal component analysis

Eigendecomposition on correlation matrix

Eigendecomposition in R is made with function `eigen()`, then the results are printed using `xtable`. Note that some modification to the labels are made in most tables.

Eigenvalues:

```
X = trackData[-1] %>% as.matrix()
R = cor(X)
E = eigen(R)

# generate table:
# xtable::xtable(matrix(E$values, ncol = 1))
```

k	λ_k
1	5.81
2	0.63
3	0.28
4	0.12
5	0.09
6	0.05
7	0.01

Eigenvectors:

```
# generate table:
# xtable::xtable(E$vectors)
```

	$\vec{e}_1$	$\vec{e}_2$	$\vec{e}_3$	$\vec{e}_4$	$\vec{e}_5$	$\vec{e}_6$	$\vec{e}_7$
100m	-0.38	-0.41	-0.14	0.59	-0.17	0.54	0.09
200m	-0.38	-0.41	-0.10	0.19	0.09	-0.74	-0.27
400m	-0.37	-0.46	0.24	-0.65	0.33	0.24	0.13
800m	-0.39	0.16	0.15	-0.30	-0.82	-0.02	-0.20
1500m	-0.39	0.31	-0.42	-0.07	0.03	-0.19	0.73
3000m	-0.38	0.42	-0.41	-0.08	0.35	0.24	-0.57
42000m	-0.36	0.39	0.74	0.32	0.25	-0.05	0.08

Principal components

$cov(Z) = cor(X)$, where Z represents scaled data (centered and scaled). Eigendecomposition on either $cov(Z)$ or $cor(X)$ gives the same eigenvalues and eigenvectors. Using these, one can calculate the correlation between principal components (Y) and scaled data (Z) as:

$$\rho_{Y_i,Z_k} = e_{ik}\sqrt{\lambda_i},$$

where Y_i are the principal components; $Y_i = \mathbf{e}_i'\mathbf{Z}$.

```
Z = scale(X)
# The following check holds:
# all.equal(cov(Z),R)
```

```r
# calculate correlation of std. variables with components:
e12 = matrix(E$vectors[,1:2], ncol = 2)
l12 = diag(E$values[1:2] %>% sqrt())

# generate table:
# xtable::xtable(e12 %*% l12)
```

k	ρ_{Y_k,Z_1}	ρ_{Y_k,Z_2}
1	-0.91	-0.32
2	-0.92	-0.33
3	-0.89	-0.36
4	-0.95	0.13
5	-0.94	0.25
6	-0.91	0.34
7	-0.86	0.31

Since $\sum_k Var(Z_k) = p = \sum_k \lambda_k$ (for eigendecomposition on correlation matrix; R!), each eigenvalue (λ_k) can be divided by the sum of variance (p) to get the proportion of variance that the corresponding eigenvector spans. The cumulative sum of the first k eigenvalues divided by this p corresponds to the proportion of standardized variance in the first k components.

```r
p = 7

# The following check holds:
# sum(E$values) == 7

# generate table:
# xtable::xtable(matrix(c(E$values/p, cumsum(E$values / p)), ncol = 2))
```

k	λ_k/p	$\sum_k \lambda_k/p$
1	0.83	0.83
2	0.09	0.92
3	0.04	0.96
4	0.02	0.98
5	0.01	0.99
6	0.01	1.00
7	0.00	1.00

The first two principal components stand for approximately 92% of the total standardized variance.

Interpretation of components

The first component can be interpreted as the general ability one country has when considering running. This interpretation is made because the scores are considered to be similar (subjective).

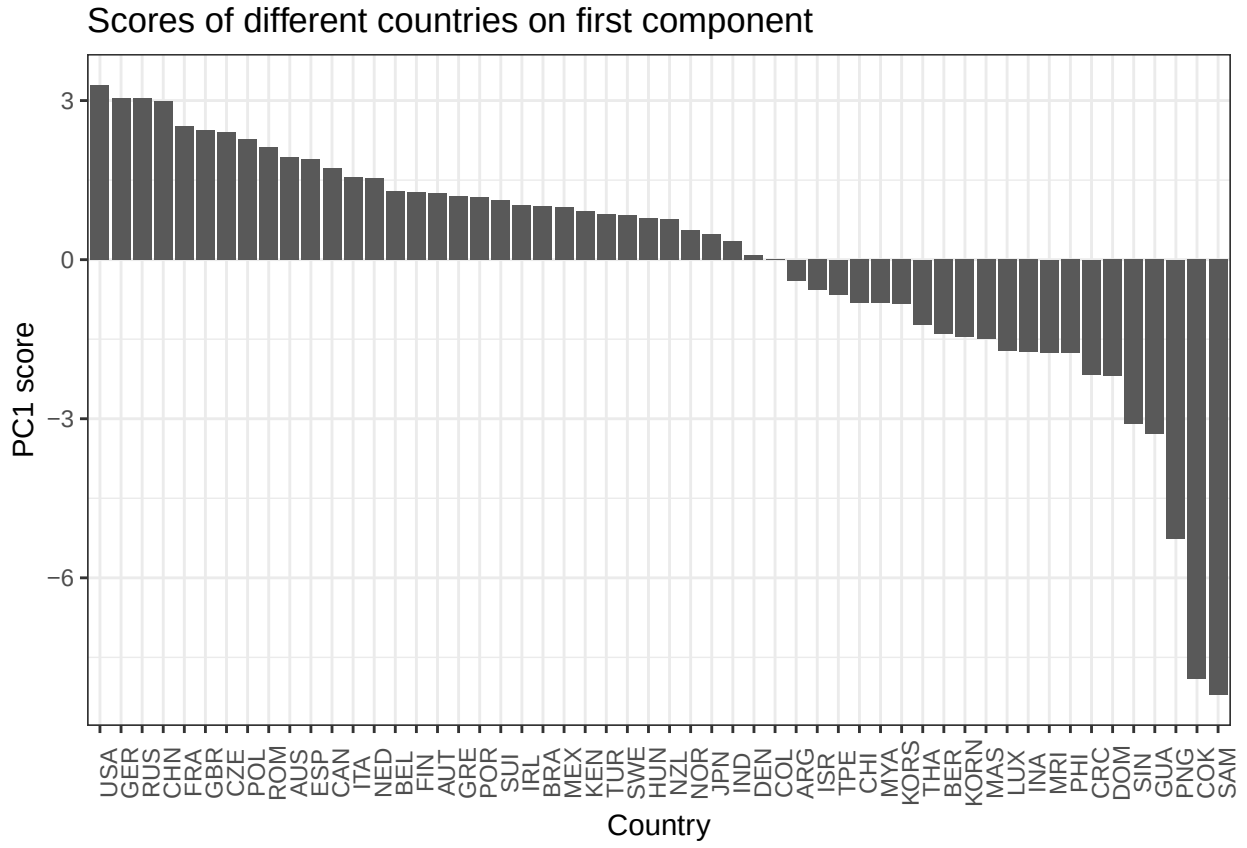
The second component seem to discriminate between shorter and longer distances, and can thus be seen as a contrast in these seemingly different disciplines. Assuming that the interpretation is correct, the following holds:

- A negative score is an indication of bad performance in the distance.
- A positive score is an indication of good performance in the distance.
- A score close to 0 is an indication of average performance in the distance.

Score-based ranking on PC1

First, the scores are sorted, then plotted, as this gives an easy interpretation.

```
# Calculate scores and prep for plot:
yHat = data.frame(Country = trackData$Country,
                  Score = Z %*% e12[,1]) %>%
  arrange(., desc(Score)) %>%
  mutate(., Country = factor(Country, levels = Country))

# Plot:
ggplot(yHat, aes(x = Country, y = Score)) + geom_bar(stat = "identity") +
  theme(axis.text.x = element_text(angle = 90)) +
  labs(y = "PC1 score", title = "Scores of different countries on first component")
```



From the plot, assuming that the interpretation made in the previous question makes sense, we can make (a small selection of) interpretations:

- USA is way above average, and also the best.
- Sweden is "lagom", as was concluded in Lab02.
- Cook Islands & Samoa are extremely bad, which was also concluded in Lab02.
- Colombia has almost exacly a score of 0, which is very close to average.

Of course, in hindsight everything makes sense... But that USA would be in the top was expected due to a history of medals in various running disciplines. Germany and Russia are a bit unknown, but large countries with good athletes in many different sports, so maybe not so suprising after all. Samoa and Cook islands was exactly as expected, as we have seen their bad performances in different multivariate analyses in previous labs.

Question 2 - Factor analysis

Below, the basic variables are defined for each fit. Also, the critical value is computed, this will also remain the same for each model test.

```
# Define baics:
p = 7     # Nr of variables
m = 2     # Nr of factors
n = 54    # Nr of data points
a = 0.05  # Significance level

# Compute Barlett's correction:
Barlett_corr = n - 1 - (2*p+4*m+5)/6

# Compute critical value:
crit_df   = ((p-m)^2 - p - m) / 2
chi2_crit = qchisq(p = 1 - a, df = crit_df)

# Add rownames to matrix (for easier plotting) :
rownames(X) = trackData$Country
```

For both methods (ML and PC), the hypotheses are constructed as below.

$$H_0 : \Sigma = L^T L + \Psi$$
$$H_a : \Sigma \neq L^T L + \Psi$$

Where Σ will stand for correlation matrix when correlation matrix is used, and stand for covariance matrix when covariance matrix is used... With both estimation methods of $\hat{\Psi}$ and $\hat{L}$, the sampling distribution is approxiamtely χ^2-distributed with $\big((p-m)^2 - p - m\big)/2$ degrees of freedom. In this case, $m = 2$, $p = 7$. The test statistic, using Barlett's correction, is calculated as:

$$\chi^2_{test} = \left[n - 1 - \frac{(2p) + 4m + 5}{6}\right]\left[ln\left(\frac{det(\hat{\mathbf{L}}\hat{\mathbf{L}}^T + \hat{\mathbf{\Psi}})}{det(\mathbf{S}_n)}\right)\right]$$

Where $\mathbf{S}_n = \frac{n-1}{n}\mathbf{S}$ and $\hat{\mathbf{L}}$ is obtained in the fit for both `factanal()` and `principal()`. Using $\alpha = 0.05$, a critical value is obtained: $\chi^2_{crit} \approx 15.5$.

In all cases, we reject the null hypothesis if $\chi^2_{test} > \chi^2_{crit}$, which means that there is not sufficient evidence under the null hypothesis to believe in the model. The above formula can be seen as:

$$[correction] \cdot \left[ln\left(\frac{Maximized\ likelihood\ under\ H_0}{Maximized\ likelihood}\right)\right]$$

If the likelihood fraction is small, there is not enough evidence to believe in H_0 and thus not sufficient evidence to believe in the model.

Inference on (sample) covariance matrix

```
# Using covariance first:
S = cov(X)

# Fit models:
factMod = factanal(x = X, factors = 2, covmat = S)
pcaMod  = principal(r = S, nfactors = 2, covar = T)

# Check models:
factMod$loadings
```

```
##
## 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
## 42000m 0.693   0.427
##
##                Factor1 Factor2
## SS loadings      3.216   2.987
## Proportion Var   0.459   0.427
## Cumulative Var   0.459   0.886
```

Since the loadings on the first component are fairly uneven, the interpretation becomes *performance on longer distances*. The second component is sort of the complement; *performance on shorter distances*. Both factors explain approximately the same amount of variance.

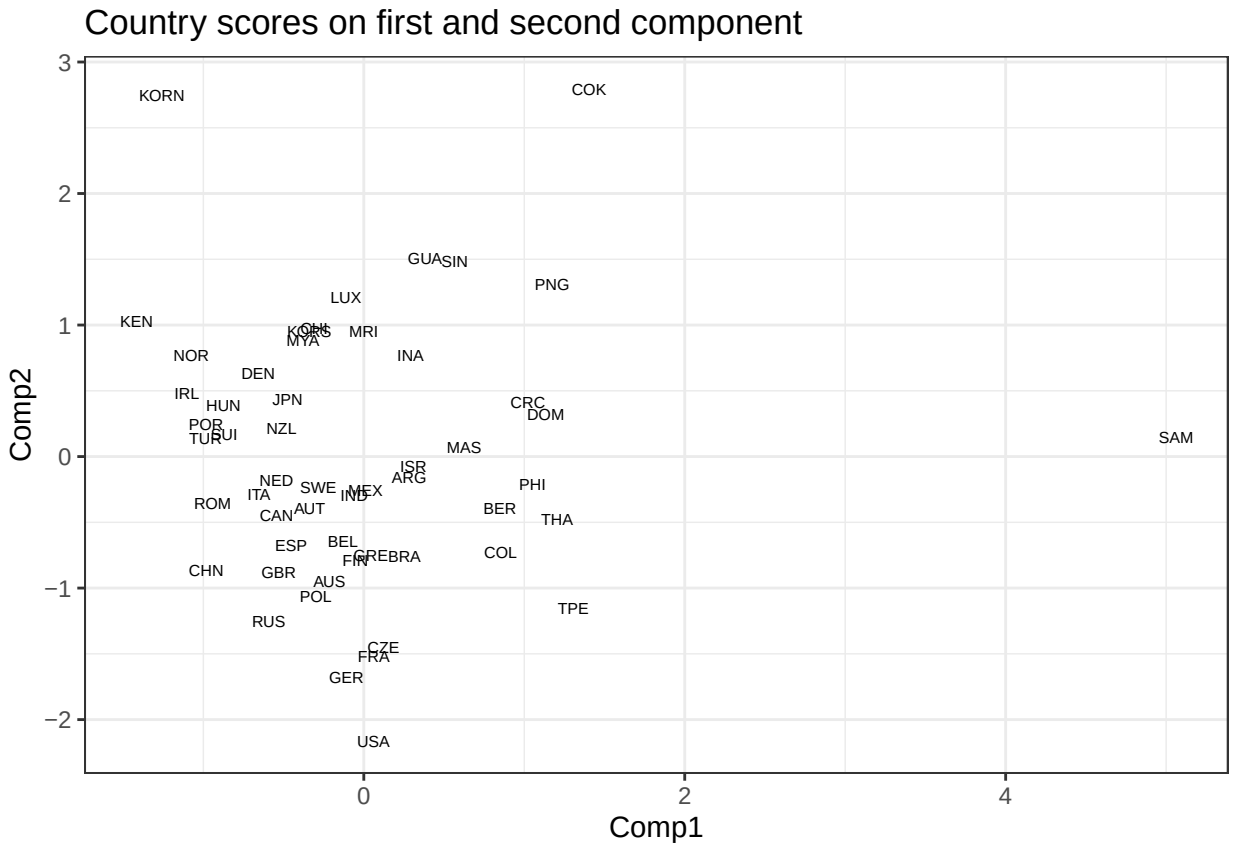
```
# Compute matrices:
S_n      = ((n-1)/n) * S          # Cov estimate (unbiased)
L_hat    = factMod$loadings       # Fitted loadings (ML estimates)
LL_hat   = L_hat %*% t(L_hat)     # ...
Psi_hat  = diag(1 - diag(LL_hat)) # Error covariances (diagonal matrix => uncorrelated)

# Compute matrix fraction (likelihood ratio):
lh_ratio = det(LL_hat + Psi_hat) / det(S_n)

# Compute test statistic:
chi2_test = Barlett_corr * log(lh_ratio)

d1 = data.frame(ModelType     = "MLmod_CovMatrix",
                TestStatistic = chi2_test,
                CriticalValue = chi2_crit,
                TestOutcome   = ifelse(test = chi2_test > chi2_crit,
                                       yes  = "Reject Null",
                                       no   = "Fail to reject Null"))
d1
```

```
##         ModelType TestStatistic CriticalValue TestOutcome
## 1 MLmod_CovMatrix      154.0517      15.50731 Reject Null
```

With 5 % significance level, the null hypothesis is rejected. There is no reason to believe in a two-factor (ML)

model, according to the test.

```
factor.scores(x = X, f = factMod)$scores %>%
  as.data.frame(.) %>% setNames(., c("Comp1", "Comp2")) %>%
  tibble::rownames_to_column(., "Country") %>%
  ggplot(., aes(Comp1, Comp2)) +
  geom_text(aes(label = Country), size = 2) +
  labs(title = "Country scores on first and second component")
```



- SAM is extreme.
- KORN and COK are outliers.
- USA and KEN are perhaps on the border.

The procedure is repeated for PC model below.

```
# Check models:
pcaMod$loadings
```

```
##
## 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
## 42000m 15.537  5.375
##
##                    RC1    RC2
## SS loadings    243.005 35.375
## Proportion Var  34.715  5.054
## Cumulative Var  34.715 39.768
```

The first component for PC model on covariance matrix has a pretty straightforward interpretation; *performance on marathon distance*. The second is also interpreted as *performance on marathon distance*. Here, the first component is dominant, as it explains approximately 7 times as much of total variance as the second component. But together, these components only explain approximately 39 % of variance, so the model is not good. Most probably, the different scales contributes to this phenomena, where the variance for marathon distance is multiple times larger than for other distances.

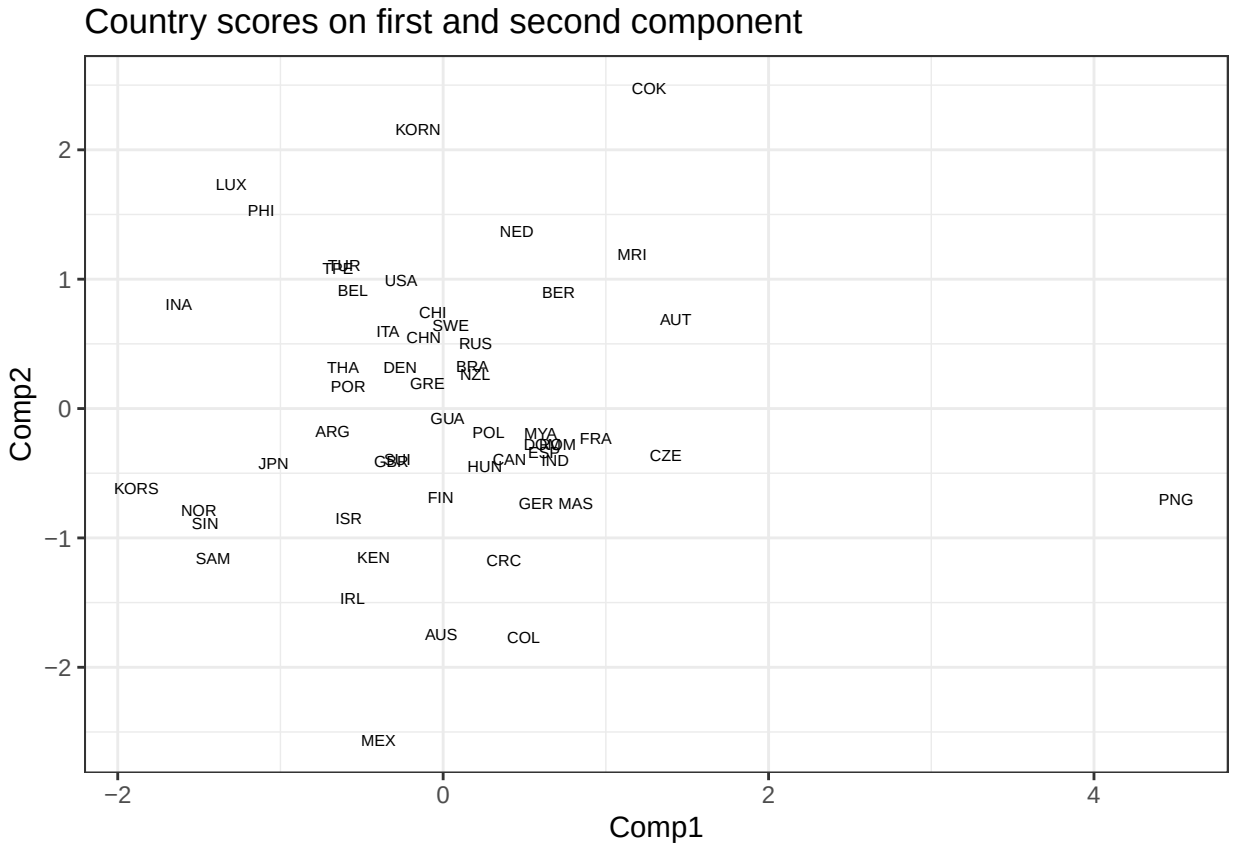
```
# Compute matrices:
S_n      = ((n-1)/n) * S          # Cov estimate (unbiased)
L_hat    = pcaMod$loadings        # Fitted loadings (ML estimates)
LL_hat   = L_hat %*% t(L_hat)     # ...
Psi_hat  = diag(1 - diag(LL_hat)) # Error covariances (diagonal matrix => uncorrelated)

# Compute matrix fraction (likelihood ratio):
lh_ratio = det(LL_hat + Psi_hat) / det(S_n)

# Compute test statistic:
chi2_test = Barlett_corr * log(lh_ratio)

d2 = data.frame(ModelType     = "PCmod_CovMatrix",
                TestStatistic = chi2_test,
                CriticalValue = chi2_crit,
                TestOutcome   = ifelse(test = chi2_test > chi2_crit,
                                       yes  = "Reject Null",
                                       no   = "Fail to reject Null"))
d2
```

```
##         ModelType TestStatistic CriticalValue TestOutcome
## 1 PCmod_CovMatrix      975.8928      15.50731 Reject Null
```

With 5 % significance level, the null hypothesis is rejected. There is no reason to believe in a two-factor (PC) model, according to the test.

```
factor.scores(x = X, f = pcaMod)$scores %>%
  as.data.frame(.) %>% setNames(., c("Comp1", "Comp2")) %>%
  tibble::rownames_to_column(., "Country") %>%
```

```
ggplot(., aes(Comp1, Comp2)) +
geom_text(aes(label = Country), size = 2) +
labs(title = "Country scores on first and second component")
```



- PNG is extreme.
- MEX and COK are outliers.

Inference on (sample) correlation matrix

```
# Using correlation instead of covariance:
R = cor(X)

# Fit models:
factMod = factanal(x = X, factors = 2, covmat = R)
pcaMod  = principal(r = R, nfactors = 2, covar = F)

# Check models:
factMod$loadings
```

```
##
## 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
## 42000m 0.693   0.427
##
##                Factor1 Factor2
## SS loadings      3.216   2.987
## Proportion Var   0.459   0.427
## Cumulative Var   0.459   0.886
```

Again, the first factor is interpreted as *performance on longer distances*. The second factor is the complement; *performance on shorter distances*. These two factors explain approximately the same amount of variance, and combined they explain approximately 89 % of variance.

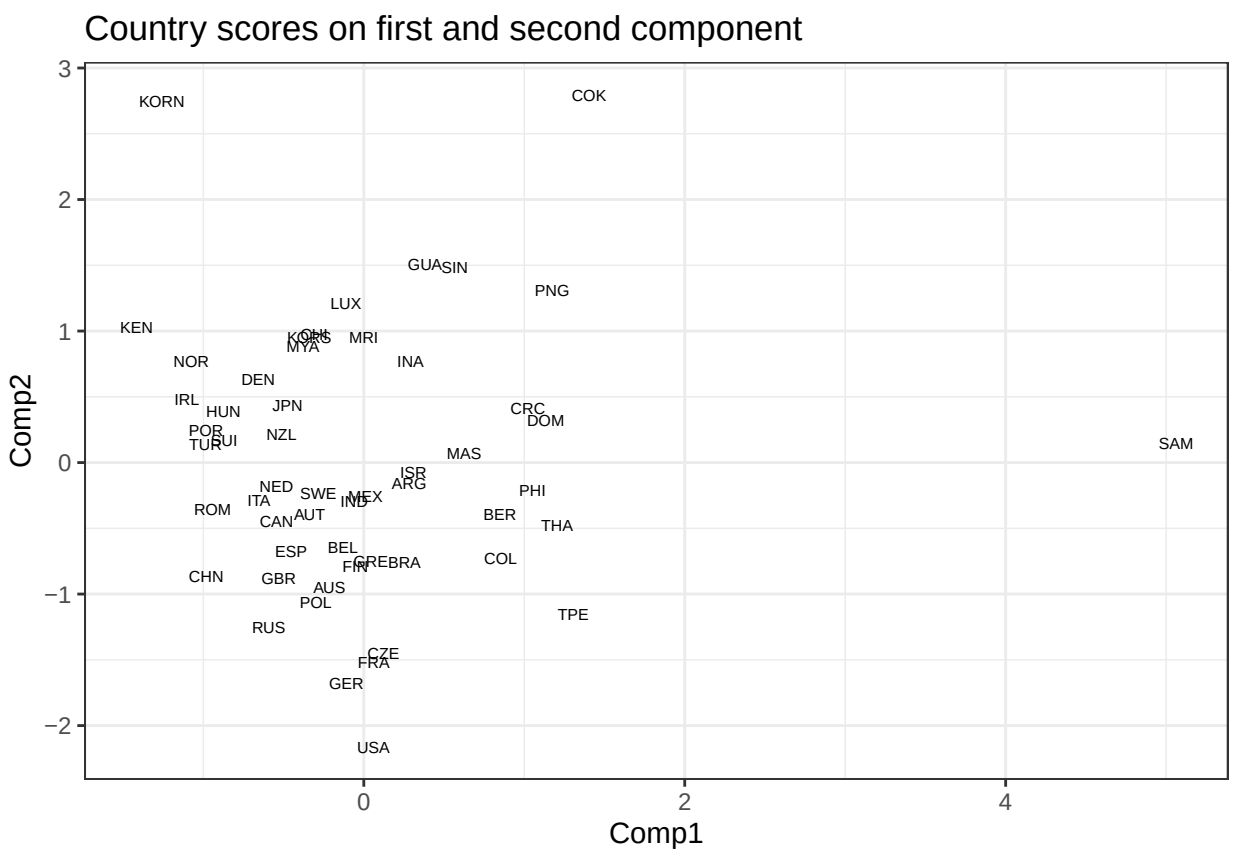
```
# Compute matrices:
R_n      = ((n-1)/n) * R          # Cor estimate
L_hat    = factMod$loadings       # Fitted loadings (ML estimates)
LL_hat   = L_hat %*% t(L_hat)     # ...
Psi_hat  = diag(1 - diag(LL_hat)) # Error covariances (diagonal matrix => uncorrelated)

# Compute matrix fraction (likelihood ratio):
lh_ratio = det(LL_hat + Psi_hat) / det(R_n)

# Compute test statistic:
chi2_test = Barlett_corr * log(lh_ratio)

d3 = data.frame(ModelType     = "MLmod_CorrMatrix",
                TestStatistic = chi2_test,
                CriticalValue = chi2_crit,
                TestOutcome   = ifelse(test = chi2_test > chi2_crit,
                                       yes  = "Reject Null",
                                       no   = "Fail to reject Null"))
d3
```

```
##          ModelType TestStatistic CriticalValue TestOutcome
## 1 MLmod_CorrMatrix      37.77713      15.50731 Reject Null
```

With 5 % significance level, the null hypothesis is rejected. There is no reason to believe in a two-factor (ML)

model, according to the test. Although, this test statistic is much closer than the previous ones.

```
factor.scores(x = X, f = factMod)$scores %>%
  as.data.frame(.) %>% setNames(., c("Comp1", "Comp2")) %>%
  tibble::rownames_to_column(., "Country") %>%
  ggplot(., aes(Comp1, Comp2)) +
  geom_text(aes(label = Country), size = 2) +
  labs(title = "Country scores on first and second component")
```



- SAM is extreme.
- KORN and COK are outliers.
- KEN and USA are perhaps on the border.

```
# Check models:
pcaMod$loadings
```

```
##
## 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
## 42000m 0.830 0.373
##
##                  RC1   RC2
## SS loadings    3.309 3.128
## Proportion Var 0.473 0.447
## Cumulative Var 0.473 0.919
```

Unlike the previous PC model (on the covariance matrix), the loadings are much more even, and so are the proportions of explained variance. The first component is interpreted as *performance on longer distances*. The second component is interpreted as the *performance on shorter distances*. The first two components explain approximately the same amount of variance, and combined they stand for approximately 99.2 %.

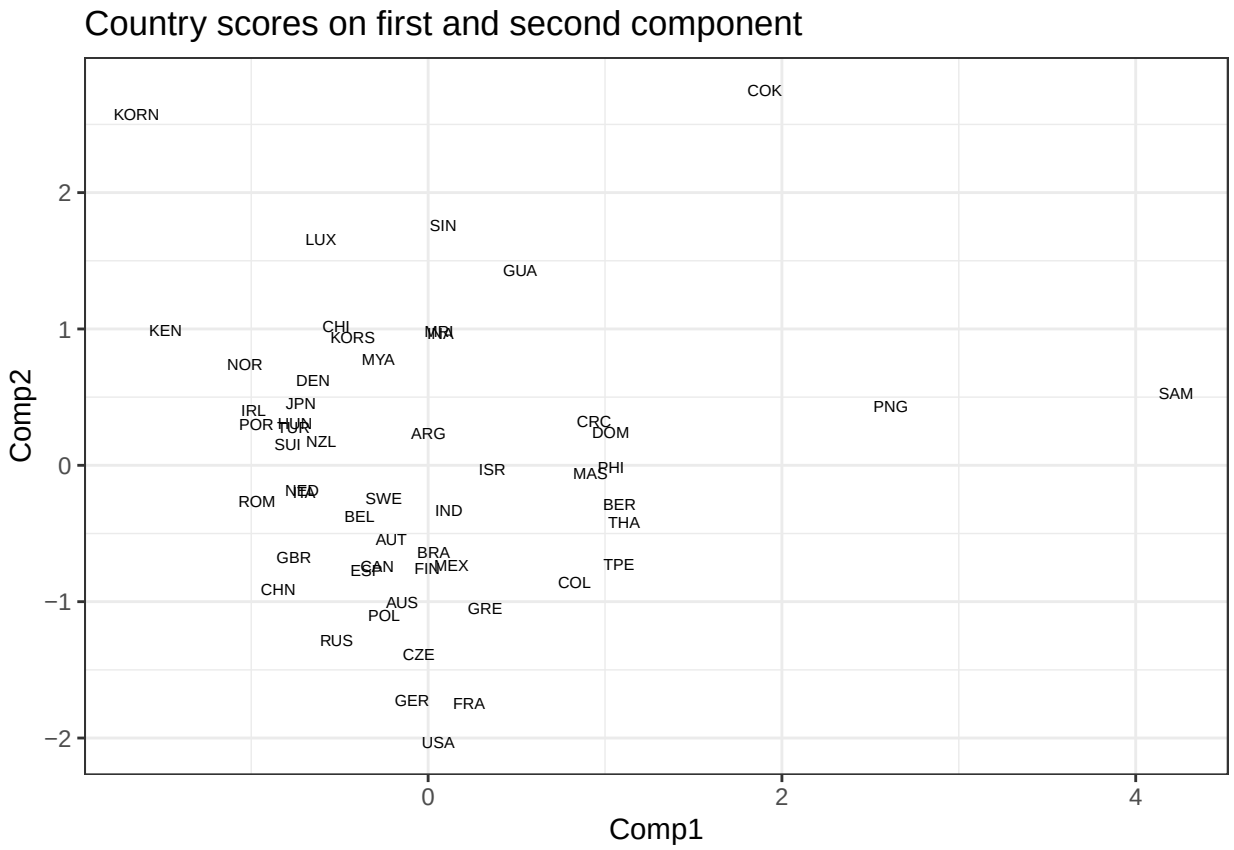
```
# Compute matrices:
R_n      = ((n-1)/n) * R          # Cor estimate
L_hat    = pcaMod$loadings        # Fitted loadings (PC estimates)
LL_hat   = L_hat %*% t(L_hat)     # ...
Psi_hat  = diag(1 - diag(LL_hat)) # Error covariances (diagonal matrix => uncorrelated)

# Compute matrix fraction (likelihood ratio):
lh_ratio = det(LL_hat + Psi_hat) / det(R_n)

# Compute test statistic:
chi2_test = Barlett_corr * log(lh_ratio)

d4 = data.frame(ModelType     = "PCmod_CorrMatrix",
                TestStatistic = chi2_test,
                CriticalValue = chi2_crit,
                TestOutcome   = ifelse(test = chi2_test > chi2_crit,
                                       yes  = "Reject Null",
                                       no   = "Fail to reject Null"))

rbind(d1,d2,d3,d4)
```

```
##          ModelType TestStatistic CriticalValue         TestOutcome
## 1  MLmod_CovMatrix     154.05170      15.50731         Reject Null
## 2  PCmod_CovMatrix     975.89282      15.50731         Reject Null
## 3 MLmod_CorrMatrix      37.77713      15.50731         Reject Null
## 4 PCmod_CorrMatrix      12.17582      15.50731 Fail to reject Null
```

For this model (bottom row), we fail to reject the null hypothesis for the first time on 5 % significance. In other words, we believe in a model for the first time.

```
factor.scores(x = X, f = pcaMod)$scores %>%
  as.data.frame(.) %>% setNames(., c("Comp1", "Comp2")) %>%
```

```
  tibble::rownames_to_column(., "Country") %>%
  ggplot(., aes(Comp1, Comp2)) +
  geom_text(aes(label = Country), size = 2) +
  labs(title = "Country scores on first and second component")
```



- SAM, PNG, COK are extreme.
- KORN is an outlier.
- KEN seem to be on the border.