Multivariate Statistical Methods - Lab 03

Maximilian Pfundstein (maxpf364), Hector Plata (hecpl268), Aashana Nijhawan(aasni448), Lakshidaa Saigiridharan (laksa656)

2019-12-04

Contents

1 Question 1: Principal components, including interpretation of them

Solve Exercise 8.18 of *Johnson, Wichern.* The data on the national track records for women, which you have studied earlier, can be found in the file `T1-9.dat`

a) Obtain the sample correlation matrix **R** for these data, and determine its eigenvalues and eigenvectors.

```r
data = read.table("T1-9.dat")
features = c("Country", "100", "200", "400", "800", "1500", "3000m", "Marathon")
colnames(data) = features

# Getting the sample correlation matrix and
# eigenvalues and egenvectors.
X = data[, 2:8]
X_corr = cor(X)
X_eigen = eigen(X_corr)


print("Sample Correlation Matrix")
```

```
## [1] "Sample Correlation Matrix"
```

```r
print(X_corr)
```

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

```r
print("Eigenvalues")
```

```
## [1] "Eigenvalues"
```

```r
print(X_eigen$values)
```

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

```r
print("Eigenvectors")
```

```
## [1] "Eigenvectors"
```

```r
print(X_eigen$vectors)
```

```
##            [,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
```

b) 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 the total (standardized) sample variance explained by the two components.

```r
Z = scale(X)
Z_corr = cor(Z)
Z_eigen = eigen(Z_corr)

print("First principal component of the standardized variables.")
```

```
## [1] "First principal component of the standardized variables."
```

```r
print(Z_eigen$vectors[, 1])
```

```
## [1] -0.3777657 -0.3832103 -0.3680361 -0.3947810 -0.3892610 -0.3760945
## [7] -0.3552031
```

```r
print("Second principal component of the standardized variables.")
```

```
## [1] "Second principal component of the standardized variables."
```

```r
print(Z_eigen$vectors[, 2])
```

```
## [1] -0.4071756 -0.4136291 -0.4593531  0.1612459  0.3090877  0.4231899
## [7]  0.3892153
```

```r
print("Correlation of the standardized variables")
```

```
## [1] "Correlation of the standardized variables"
```

```r
print(Z_corr)
```

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

```r
print("Cumulative percentage of the total variance explained by the first two components")
```

```
## [1] "Cumulative percentage of the total variance explained by the first two components"
```

```r
print(sum(Z_eigen$values[1:2]) / 7)
```

```
## [1] 0.919474
```

```r
print("(91.9474%)")
```

```
## [1] "(91.9474%)"
```

c) Interpret the two principal components obtained in Part b. (Note that the first component is essentially a normalized unit vector and might measure the athletic excellence of a given nation. The second component might measure the relative strength of a nation at the various runnning distances.)

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) 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?

```r
Y_1 = as.matrix(Z) %*% Z_eigen$vectors[, 1]
rank = list(Country=data$Country, Score=Y_1)
rank = data.frame(rank)
ordered_idxs = order(rank$Score, decreasing=TRUE)
ordered_rank = rank[ordered_idxs, ]

print("Top 10 countries")
```

```
## [1] "Top 10 countries"
```

```r
print(ordered_rank[1:10,])
```

```
##    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
```

```r
print("Bottom 10 countries")
```

```
## [1] "Bottom 10 countries"
```

```r
print(ordered_rank[44:54,])
```

```
##    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 bototm are underdeveloped nations that always lack performance on competitive sports.