GR5223 - HW4

Mathieu Sauterey - UNI: mjs2364

25 March 2018

Problem 1

The data in Number_Parity.csv were collected to test a psychological model of numerical cognition: How does the processing of numbers depend on the way the numbers are presented (words versus Arabic digits)?

Thirty-two subjects were required to make a series of quick numerical judgments about two numbers presented either as two words (two vs. four, for example) or two single Arabic digits (2 vs. 4). The subjects were asked to respond "same" if the two numbers had the same numerical parity (both even or both odd) and "different" if the two numbers had a different parity (one even one odd). For each of the four combinations of parity and format, the median reaction times for correct responses were recorded for each subject.

x_1 = WordDiff = reaction time for word format, different parity x_2 = WordSame = reaction time for word format, same parity x_3 = Num_Diff = reaction time for Arabic numeral, different parity x_4 = Num_Same = reaction time for Arabic numeral, same parity

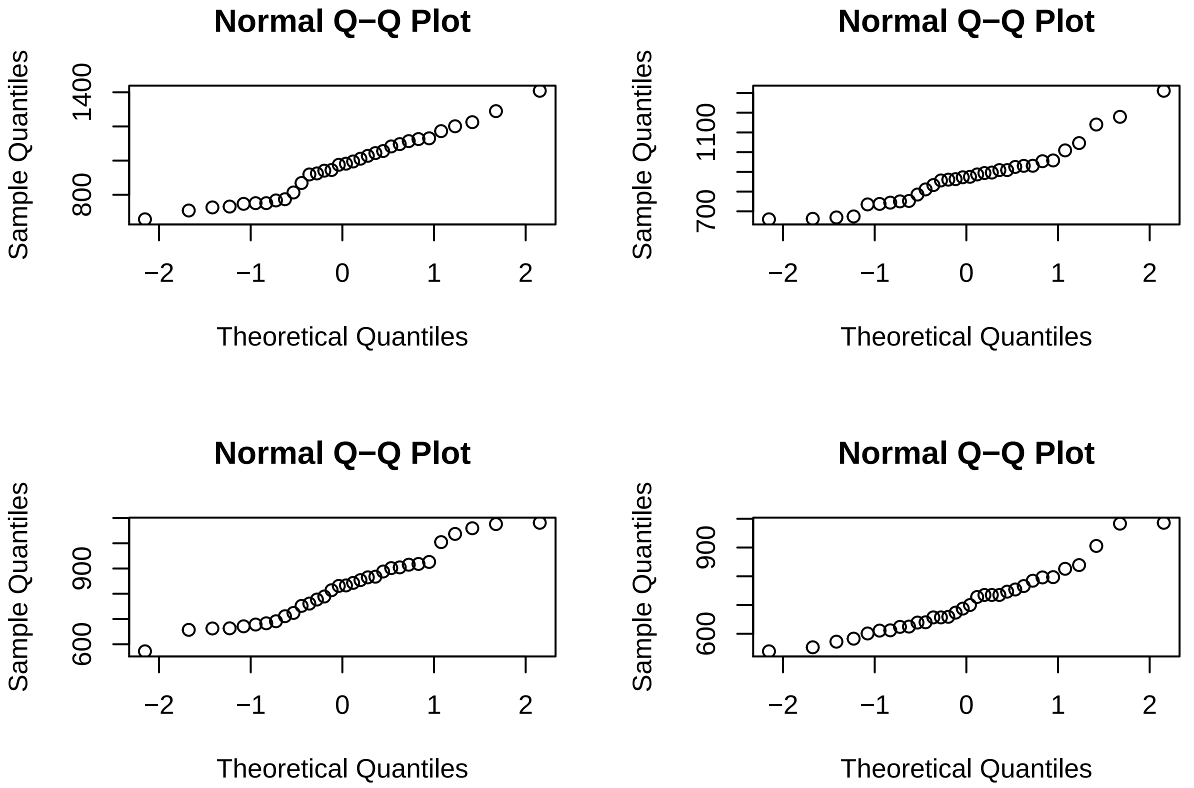
Conduct a repeated measures analysis on these data.

```r
# Loads dataset from the data CSV file
data <- read.csv("Number_Parity.csv", as.is = TRUE)
```

(a) Assess the reasonableness of assuming the data are a random sample from a multivariate normal population.

First, we construct normal probability plots (Q-Q plots) for each variable individually.

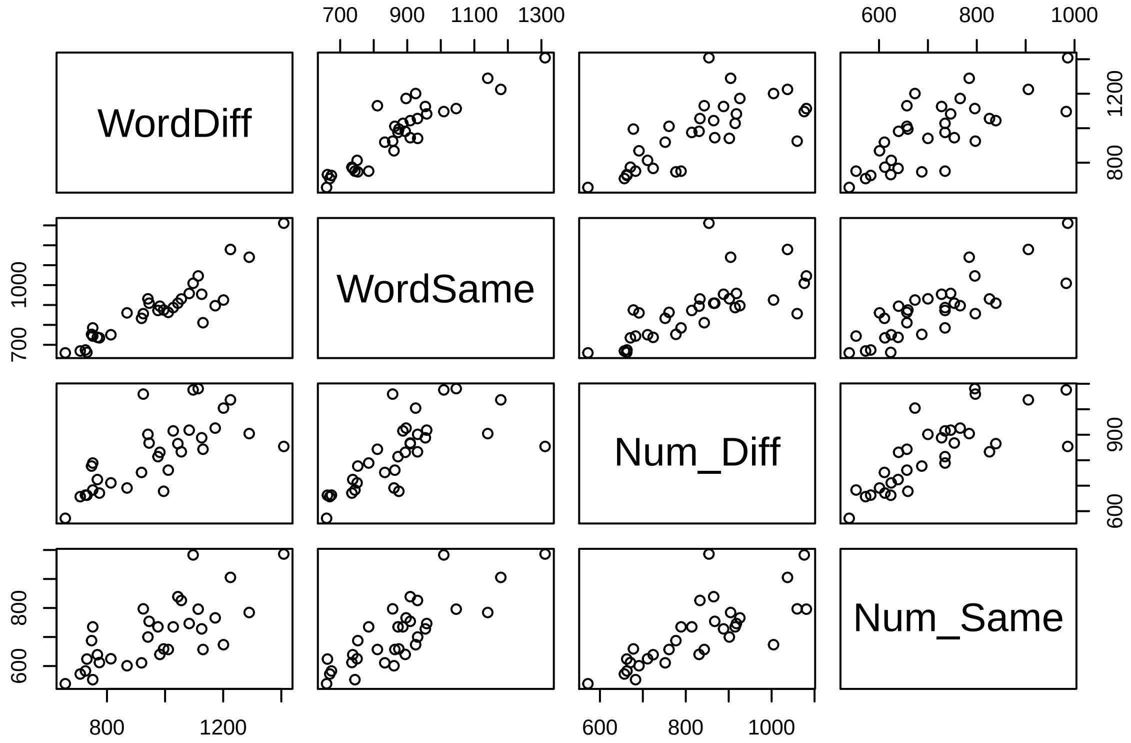
```r
# Plots the QQ-plot for each univariate variable
par(mfrow=c(2,2))
qqnorm(data$WordDiff)
qqnorm(data$WordSame)
qqnorm(data$Num_Diff)
qqnorm(data$Num_Same)
```



We observe that the QQ-plots approximately resemble straight lines. Thus the univariate distributions are approximately normally distributed.

Then, we construct the scatterplot matrix for the pairs of observations of different variables.

```
# Plots the scatterplot matrix
library(graphics)
pairs(data)
```



Looking at the scatter diagrams for each pair of variables we see an elliptical point cloud, the conditional mean functions are approximately linear and conditional variance is roughly constant.

We conclude, based on the Q-Qplots and scatter diagrams, that it is reasonbale to assume that these variables are samples from a multivariate normal distribution.

(b) Test the null hypothesis of no treatment effect. That is, find and interpret the p-value for a test of H0 : mu1 = mu2 = mu3 = mu4.

```
# Loads and calculates basic parameters
n     <- nrow(data)
p     <- ncol(data)
xbar  <- colMeans(data)
S     <- cov(data)
alpha <- 0.05

# Construct the contrast matrix for the test
c1 <- c(1,-1,1,-1)
c2 <- c(1,1,-1,-1)
c3 <- c(1,-1,-1,1)
C <- rbind(c1,c2,c3)

# Calculates T2-statistic
T2 <- as.vector(n * t(C%*%xbar) %*% solve(C %*% S %*% t(C)) %*% (C%*%xbar))
```

```r
# Finds the p-value by comparing T2 statistic to the F-distribution(p-1;n-p+1)
p.value <- 1 - pf((n-p+1)/((n-1)*(p-1)) * T2, df1=p-1, df2=n-p+1)
p.value
```

```
## [1] 2.328437e-11
```

The p-value is essentially zero so we reject the null hypothesis $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4$. We conclude that the treatment has some effect: the processing of numbers depend on the way the numbers are presented.

(c) Compute and interpret simultaneous 95% confidence intervals for

i. the contrast for parity effect (different vs. same), averaged over word format and Arabic digits

ii. the contrast for format effect (word vs. numeral), averaged over same and different parity

iii. an interaction contrast measuring the difference in parity effect for word format versus parity effect given Arabic digits (or equivalently, the difference in format effect for different parity versus format effect given same parity).

```r
# Precalculates the term containing the F-quantile
F.mult <- sqrt((n-1)*(p-1)/(n-p+1) * qf(1-alpha,p-1,n-p+1))

# Calculates the simultaneous 95% confidence intervals for each contrast
as.vector(c1%*%xbar) + F.mult*sqrt(t(c1) %*% S %*% (c1)/n) %*% c(-1,1)
```

```
##          [,1]     [,2]
## [1,] 130.4567 282.1995
```

```r
as.vector(c2%*%xbar) + F.mult*sqrt(t(c2) %*% S %*% (c2)/n) %*% c(-1,1)
```

```
##          [,1]     [,2]
## [1,] 198.1074 415.7364
```

```r
as.vector(c3%*%xbar) + F.mult*sqrt(t(c3) %*% S %*% (c3)/n) %*% c(-1,1)
```

```
##          [,1]     [,2]
## [1,] -76.6668 31.82305
```

Both the parity and format effects are statistically significant (intervals exclude zero), with the main effect of format being greater in magnitude than that of parity. The interval for the interaction contrast contains zero, so there's no evidence of a parity by format effect (or equivalently, no evidence of a format by parity effect).

Problem 2

The data file Turtles.csv, in the Data folder on Courseworks, contains measurements of carapace (shell) dimensions for 24 female and 24 male painted turtles: x1 = length, x2 = width and x3 = height, all in millimeters. Assume the female and male turtles are independent random samples from trivariate normal distributions with a common covariance matrix; denote the mean vector for female turtles by μ_1 and that of male turtles by μ_2.

```r
# Loads dataset from the data CSV file
turtles <- read.csv("Turtles.csv", as.is = TRUE)
```

```
# Extracts data for the female and male separately
female <- turtles[1:24,-4]
male <- turtles[-(1:24),-4]
```

(a) Test for equality of the two population mean vectors, H0 : mu_1 = mu_2; report and interpret a p-value.

```
# Loads and calculates basic parameters
pt <- ncol(turtles[,-4])

nf <- nrow(female)
nm <- nrow(male)

Sf <- cov(female)
Sm <- cov(male)

xbarf <- colMeans(female)
xbarm <- colMeans(male)

# Calculates the pooled sample covariance matrix
St <- 1/(nm + nf - 2) * ( (nm-1)*Sm + (nf-1)*Sf )

# Calculates the test statistic for H0 : mu_1 = mu_2
T2.t <- t(xbarf-xbarm) %*% solve((1/nm + 1/nf)* St) %*% (xbarf-xbarm)

# Finds the p-value by comparing T2 statistic to the F-distribution(p-1;nf+nm-p-1)
1 - pf((nm+nf-p-1)/((nm+nf-2)*pt)*T2.t, df1=pt, df2=nm+nf-pt-1)
```

```
##              [,1]
## [1,] 5.374197e-09
```

The p-value is essentially zero so we reject the null hypothesis $H_0 : \mu_1 = \mu_2$. The data provide no indication that the mean vectors for females and males are the same.

(b) Use the Bonferroni method to find simultaneous 95% confidence intervals for the component mean differences. Interpret your intervals.

```
# Precalculates the term containing the t-quantile
alpha <- 0.05
t.mult <- qt(1 - (alpha/(2*pt)), df = nm + nf - 2)

# Calculates the Bonferroni simultaneous 95% confidence intervals for each mean difference
(xbarf[1]-xbarm[1]) + sqrt((St[1,1]) * (1/nm + 1/nf)) * t.mult %*% t(c(-1,1))
```

```
##          [,1]     [,2]
## [1,] 10.34418 34.98915
```

```
(xbarf[2]-xbarm[2]) + sqrt((St[2,2]) * (1/nm + 1/nf)) * t.mult %*% t(c(-1,1))
```

```
##          [,1]     [,2]
## [1,] 6.738628 21.84471
```

```
(xbarf[3]-xbarm[3]) + sqrt((St[3,3]) * (1/nm + 1/nf)) * t.mult %*% t(c(-1,1))
```

```
##          [,1]     [,2]
## [1,] 6.911898 15.75477
```

As expected from the p-value, the three component mean differences are statistically significant (intervals exclude zero), with the carapace length difference being the greatest in magnitude. We can thus interpret that female turtles have larger carapace length, width and weight compared to male turtles.

Problem 3

The data file Track_Records.csv, in the Data folder on Courseworks, contains the national track records for women in n = 54 countries, in p = 7 different running events.

```
# Loads dataset from the data CSV file
track <- read.csv("Track_Records.csv", as.is = TRUE)
```

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

```
# Loads and calculates basic parameters
p.tr <- ncol(track[,-1])
n.tr <- nrow(track)
xbar.tr <- colMeans(track[,-1])
S.tr <- cov(track[,-1])

# Standardizes the data matrix
track.C <- (track[-1])- rep(1,n.tr) %*% t(xbar.tr)
track.S <- as.matrix(track.C) %*% diag(1/sqrt(diag(S.tr)))

# Obtains the sample correlation matrix R
R <- cor(track.S)
R
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## [1,] 1.0000000 0.9410886 0.8707802 0.8091758 0.7815510 0.7278784 0.6689597
## [2,] 0.9410886 1.0000000 0.9088096 0.8198258 0.8013282 0.7318546 0.6799537
## [3,] 0.8707802 0.9088096 1.0000000 0.8057904 0.7197996 0.6737991 0.6769384
## [4,] 0.8091758 0.8198258 0.8057904 1.0000000 0.9050509 0.8665732 0.8539900
## [5,] 0.7815510 0.8013282 0.7197996 0.9050509 1.0000000 0.9733801 0.7905565
## [6,] 0.7278784 0.7318546 0.6737991 0.8665732 0.9733801 1.0000000 0.7987302
## [7,] 0.6689597 0.6799537 0.6769384 0.8539900 0.7905565 0.7987302 1.0000000
```

```
# Obtains eigenvalues of R
evalues.R <- eigen(R)$values
evalues.R
```

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

```
# Obtains eigenvectors of R
evectors.R <- eigen(R)$vectors
evectors.R
```

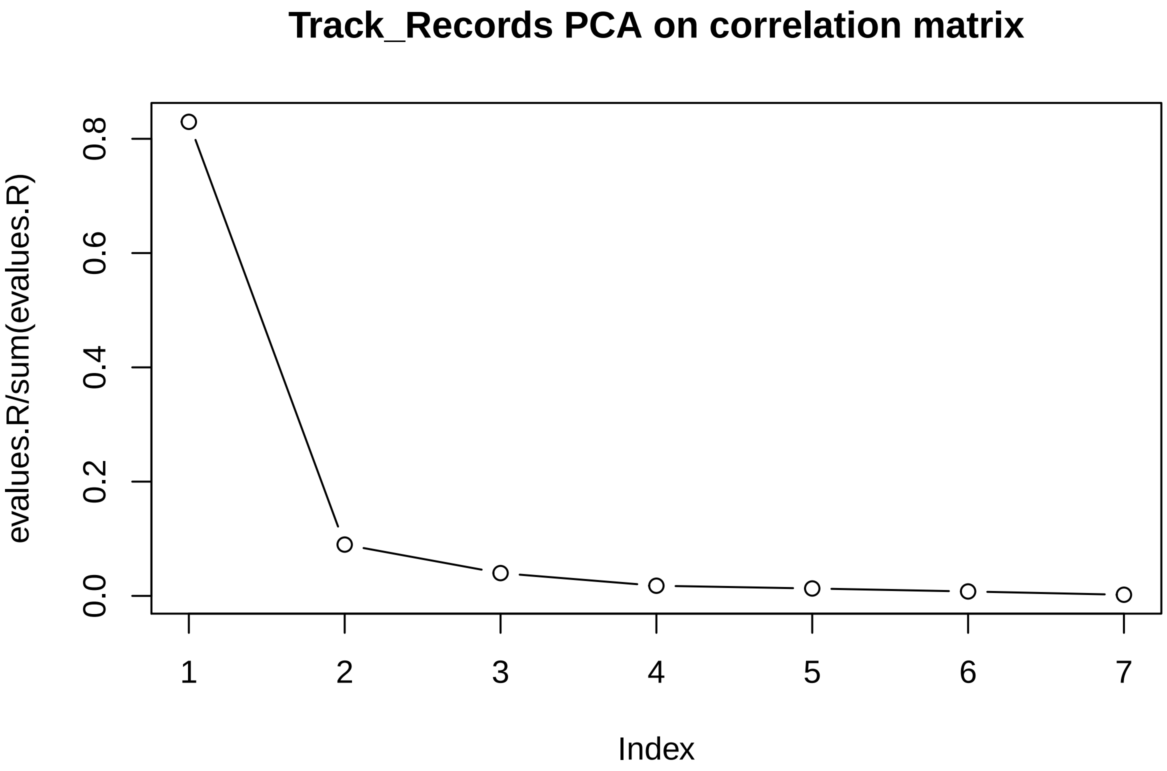
```
##            [,1]       [,2]       [,3]        [,4]        [,5]        [,6]
## [1,] -0.3777657 -0.4071756 -0.1405803  0.58706293 -0.16706891 -0.53969730
## [2,] -0.3832103 -0.4136291 -0.1007833  0.19407501  0.09350016  0.74493139
## [3,] -0.3680361 -0.4593531  0.2370255 -0.64543118  0.32727328 -0.24009405
## [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) Calculate the proportion of total (standardized) sample variance explained by each (normalized) principal component, and prepare a graphical summary in the form of a scree plot. Also find the proportion of (standardized) variance explained by the first k (normalized) principal components for k = 1, 2, . . . , 7. How many NPCs should we retain if our goal is to account for 90% of total (standardized) variance?

```
# Calculates the proportion of total (standardized) sample variance explained by each NPC
evalues.R / sum(evalues.R)
```

```
## [1] 0.829660638 0.089813346 0.039904939 0.017793531 0.012995963 0.007788403
## [7] 0.002043181
```

```
# Screeplot of the proportion of variance explained by each NPC
plot(evalues.R / sum(evalues.R), type="b", main="Track_Records PCA on correlation matrix")
```



```
# Proportion of (standardized) variance explained by the first k NPCs
cumsum((evalues.R) / sum(evalues.R))
```

```
## [1] 0.8296606 0.9194740 0.9593789 0.9771725 0.9901684 0.9979568 1.0000000
```

The print above is the proportion of total (standardized) sample variance explained by each (normalized) principal component.

Based on the cumulative variance explained that we printed above, we should retain 2 NPCs to account for 90% of total (standardized) variance.

(c) Interpret the first two NPCs.

```
# Prints eigenvectors of R
evectors.R[,1:2]
```

```
##            [,1]       [,2]
## [1,] -0.3777657 -0.4071756
## [2,] -0.3832103 -0.4136291
## [3,] -0.3680361 -0.4593531
## [4,] -0.3947810  0.1612459
## [5,] -0.3892610  0.3090877
## [6,] -0.3760945  0.4231899
## [7,] -0.3552031  0.3892153
```

From the eigenvectors printed above, we look at the first two columns which represent the first two NPCs: The first NPC is roughly a straight average across running events, namely the overall performance. The second NPC is the difference between the 800m, 1500m, 3000m, Marathon and the other running events.

(d) Rank the nations based on their score on the first (normalized) principal component. Does this ranking correspond with your previous notion of athletic excellence for the various countries?

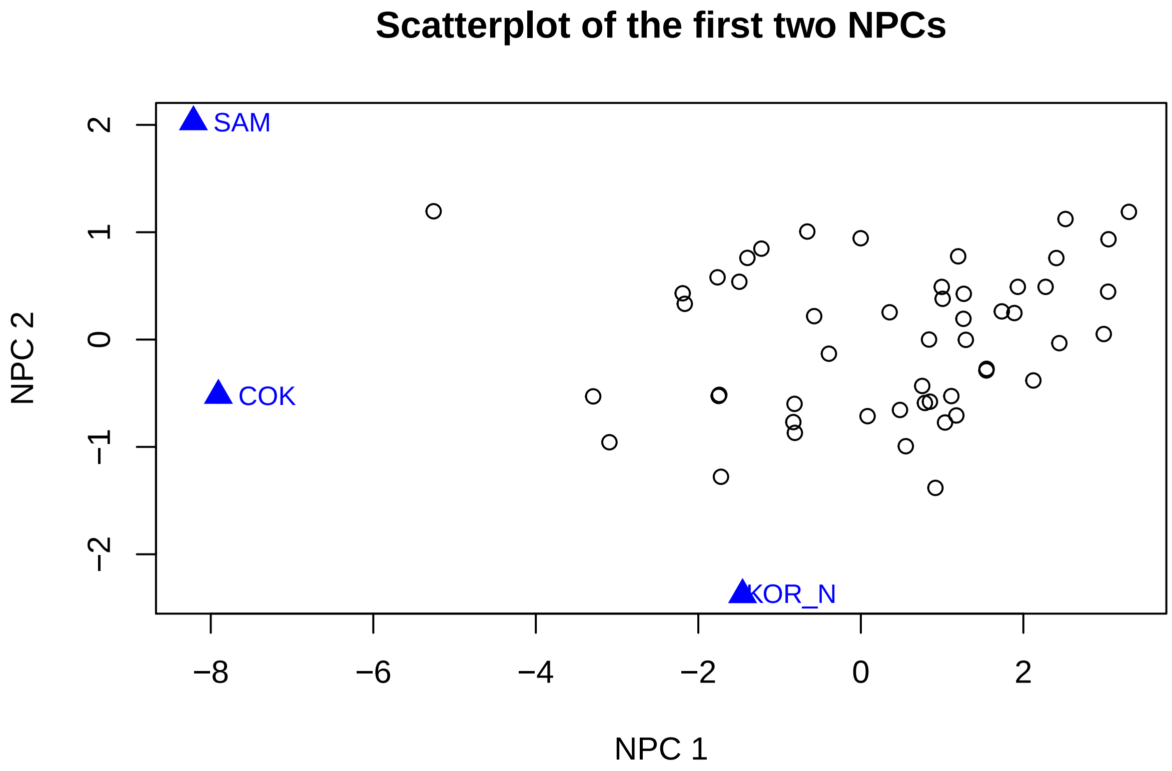
```r
# Obtains the score of the first NPC
NPC1.score <- track.S %*% evectors.R[,1]

# Ranks the nations based on their score
track$Country[order(NPC1.score, decreasing = TRUE)]
```

```
##  [1] "USA"   "GER"   "RUS"   "CHN"   "FRA"   "GBR"   "CZE"   "POL"
##  [9] "ROM"   "AUS"   "ESP"   "CAN"   "ITA"   "NED"   "BEL"   "FIN"
## [17] "AUT"   "GRE"   "POR"   "SUI"   "IRL"   "BRA"   "MEX"   "KEN"
## [25] "TUR"   "SWE"   "HUN"   "NZL"   "NOR"   "JPN"   "IND"   "DEN"
## [33] "COL"   "ARG"   "ISR"   "TPE"   "CHI"   "MYA"   "KOR_S" "THA"
## [41] "BER"   "KOR_N" "MAS"   "LUX"   "INA"   "MRI"   "PHI"   "CRC"
## [49] "DOM"   "SIN"   "GUA"   "PNG"   "COK"   "SAM"
```

Yes, this ranking does correspond with my notion of athletic excellence for the various countries, with USA being first, followed by European countries, as well as Russia and China.

(e) Make a scatterplot of the first two (normalized) principal components. Identify the points corresponding to Samoa, the Cook Islands, and North Korea, and explain what about those countries makes them stand out in the plot as they do.

```r
# Obtains the score of the second NPC
NPC2.score <- track.S %*% evectors.R[,2]

# Finds the index of each country in the data
Samoa.index <- which(track$Country == "SAM")
NKorea.index <- which(track$Country == "KOR_N")
Cook.index <- which(track$Country == "COK")
These.index <- c(Samoa.index,NKorea.index,Cook.index)

# Obtains the NPC1 and NPC2 scores of each country
Samoa.score <- c(NPC1.score[Samoa.index],NPC2.score[Samoa.index])
NKorea.score <- c(NPC1.score[NKorea.index],NPC2.score[NKorea.index])
Cook.score <- c(NPC1.score[Cook.index],NPC2.score[Cook.index])
These.score <- rbind(Samoa.score,NKorea.score,Cook.score)

# Scatterplot of first 2 NPCs with points corresponding to these countries
plot(NPC1.score, NPC2.score, xlab = "NPC 1", ylab="NPC 2",
     main = "Scatterplot of the first two NPCs")
points(These.score[,1], These.score[,2], col="blue", pch=17, cex=1.5)
text(These.score[,1]+0.6, These.score[,2], track$Country[These.index],
     col="blue", pch=17, cex=0.8)
```



These countries are outliers in the plot and they stand out due to their very low score in NPC1 or NPC2. This is caused by their low athtletic performance as accounted for by NPC1 and NPC2.

Problem 4

Continue with the women's national track records data from the previous exercise.

(a) Convert each record to an average speed for the race, measured in meters per second. Notice that the records for 800 m, 1500 m, 3000 m, and the marathon are given in minutes. The marathon is 26.2 miles, or 42,195 meters, long.

```r
# Convert each record to an average speed for the race, measured in m/s
avg.tr <- track[-1]
avg.tr <- cbind(avg.tr[,1:3],avg.tr[,4:7]*60)
avg.tr <- rep(c(100,200,400,800,1500,3000,42195), each=n.tr)/avg.tr
head(avg.tr)
```

```
##         x1       x2       x3       x4       x5       x6       x7
## 1 8.643042 8.718396 7.619048 6.504065 5.882353 5.440696 4.678353
## 2 8.992806 8.996851 8.225375 6.734007 6.218905 5.793743 4.900355
## 3 8.968610 8.810573 7.902015 6.872852 6.172840 5.694761 4.556203
## 4 8.976661 8.896797 7.774538 6.768190 6.127451 5.668934 4.916113
## 5 8.726003 8.676790 7.504690 6.441224 5.827506 5.096840 4.037490
## 6 8.952551 8.849558 7.902015 6.768190 5.995204 5.530973 4.770708
```

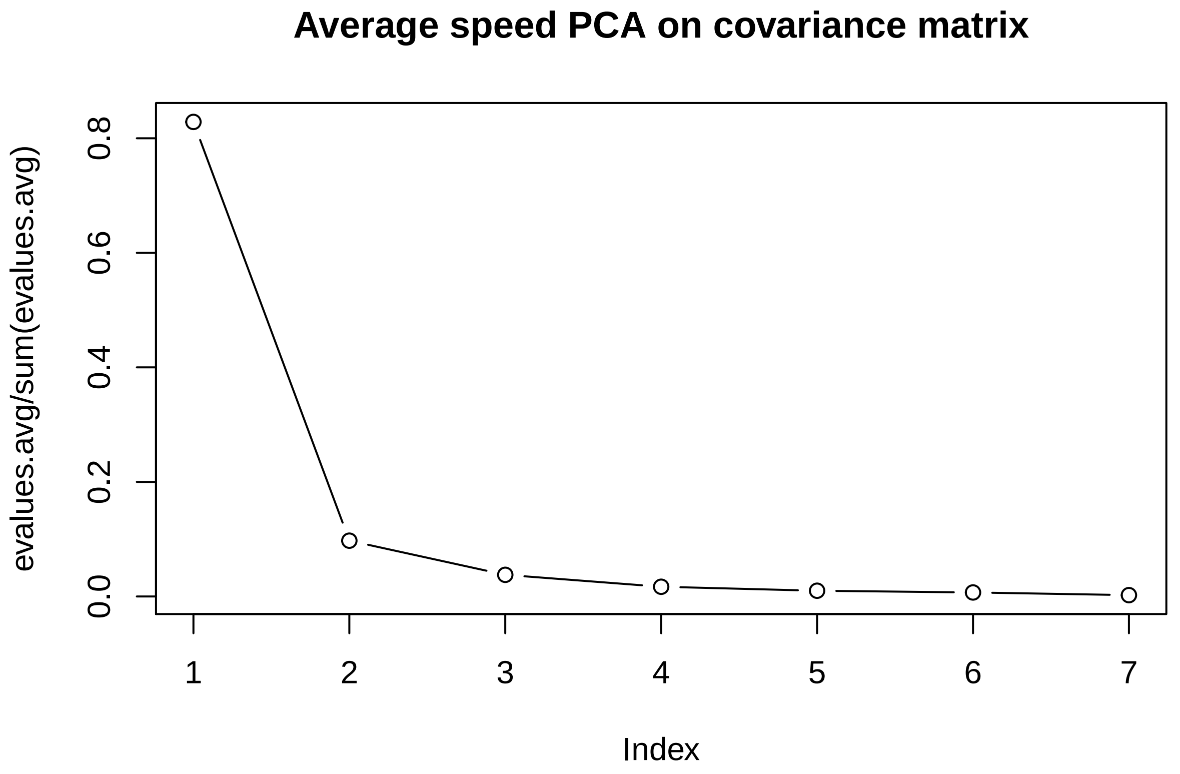
(b) Perform a principal components analysis using the covariance matrix S of the speed data. Again find the proportion of variance explained by each of the first k principal components for k = 1, 2, . . . 7. How many principal components should we retain if our goal is to account for 90% of total sample variance?

```
# Centers data and estimates the sample covariance matrix
xbar.avg.tr <- colMeans(avg.tr)
S.avg.tr <- cov(avg.tr)

# Standardizes the data matrix
avg.tr.C <- (avg.tr)- rep(1,n.tr) %*% t(xbar.avg.tr)

# Calculates and prints the eigenvalues and eigenvectors
evalues.avg <- eigen(S.avg.tr)$values
evectors.avg <- eigen(S.avg.tr)$vectors

# Calculates the proportion of total (standardized) sample variance explained by each NPC
evalues.avg / sum(evalues.avg)
```

```
## [1] 0.828538913 0.097403774 0.037774734 0.016949208 0.010016310 0.006979577
## [7] 0.002337484
```

The print above is the proportion of variance explained by each principal component.

```
# Screeplot of the proportion of variance explained by each NPC
plot(evalues.avg / sum(evalues.avg), type="b",
     main="Average speed PCA on covariance matrix")
```



```r
# Proportion of (standardized) variance explained by the first k NPCs
cumsum((evalues.avg) / sum(evalues.avg))
```

```
## [1] 0.8285389 0.9259427 0.9637174 0.9806666 0.9906829 0.9976625 1.0000000
```

Based on the cumulative variance explained that we printed above, we should retain 2 NPCs to account for 90% of total (standardized) variance.

(c) Interpret the first two principal components. Are these interpretations similar to those of the first two NPCs in the previous exercise?

```r
# Prints eigenvectors of covariance matrix S.tr
evectors.avg[,1:2]
```

```
##            [,1]        [,2]
## [1,] -0.3102442 -0.37596510
## [2,] -0.3573948 -0.43376925
## [3,] -0.3787367 -0.51873227
## [4,] -0.2993405  0.05313551
## [5,] -0.3912131  0.21084397
## [6,] -0.4595909  0.39557338
## [7,] -0.4227291  0.44458346
```

The first PC is roughly a straight average across running events, namely the overall performance. The second PC is the difference between the 800m, 1500m, 3000m, Marathon and the other running events with greater weight given to the larger distances. These interpretations are similar to those of the first two NPCs analyzed earlier.

(d) If the nations are ranked on the basis of their first principal component score, does the subsequent ranking differ notably from that in the previous exercise?

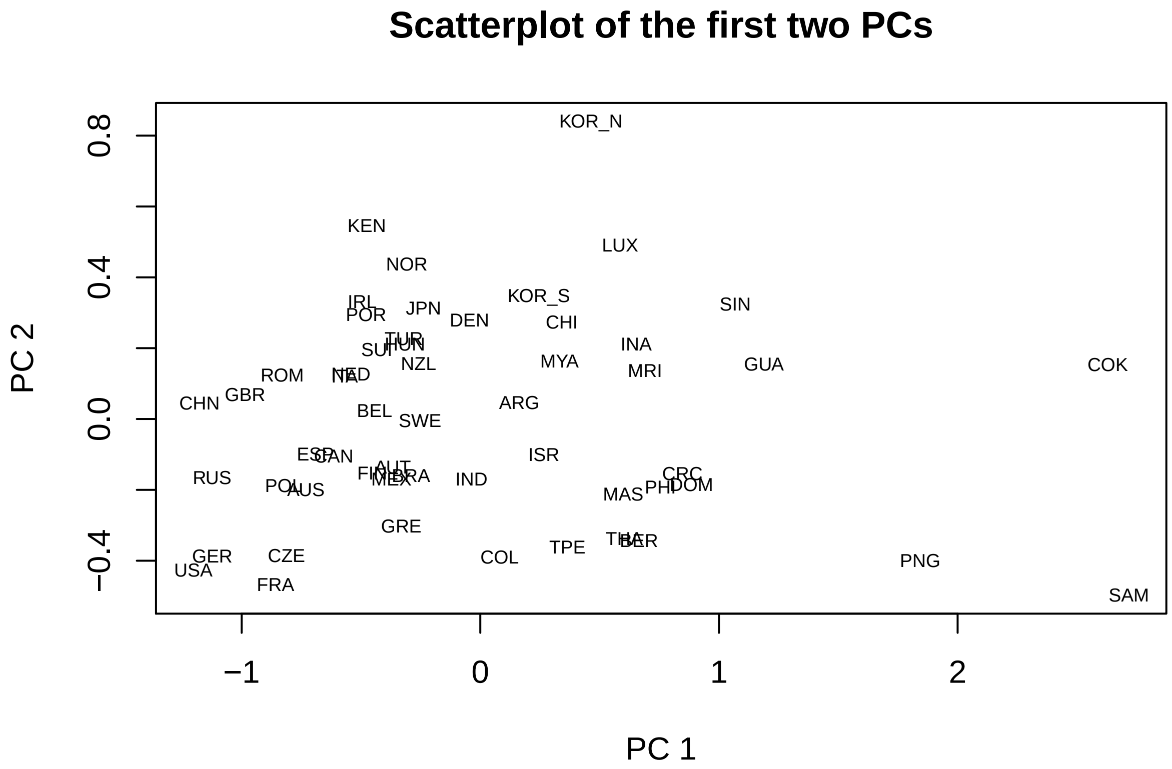
```
# Obtains the score of the first PC
PC1.score <- as.matrix(avg.tr.C) %*% evectors.avg[,1]

# Ranks the nations based on their score
track$Country[order(PC1.score)]
```

```
##  [1] "USA"   "CHN"   "RUS"   "GER"   "GBR"   "FRA"   "ROM"   "POL"  
##  [9] "CZE"   "AUS"   "ESP"   "CAN"   "ITA"   "NED"   "IRL"   "POR"  
## [17] "KEN"   "FIN"   "BEL"   "SUI"   "MEX"   "AUT"   "GRE"   "TUR"  
## [25] "HUN"   "NOR"   "BRA"   "NZL"   "SWE"   "JPN"   "DEN"   "IND"  
## [33] "COL"   "ARG"   "KOR_S" "ISR"   "MYA"   "CHI"   "TPE"   "KOR_N"
## [41] "LUX"   "MAS"   "THA"   "INA"   "BER"   "MRI"   "PHI"   "CRC"  
## [49] "DOM"   "SIN"   "GUA"   "PNG"   "COK"   "SAM"
```

No, this ranking does not differ significantly from that in the previous exercise.

(e) Make a scatterplot of the first two principal component scores, and label the points by the countries' abbreviations. Comment on the difference between this plot and the corresponding plot for the previous exercise.

```
# Obtains the score of the second PC
PC2.score <- as.matrix(avg.tr.C) %*% evectors.avg[,2]

# Scatterplot of first 2 NPs with countries' labels
plot(PC1.score, PC2.score, xlab = "PC 1", ylab="PC 2",
     main = "Scatterplot of the first two PCs", cex=0)
text(PC1.score, PC2.score, track$Country, cex = 0.6)
```



This plot is fairly similar to that of the previous exercise, with Cook Islands, Samoa and North Korea outlying. Also, we see the same pattern that European countries, China, US and Russia are being clustered. The only difference is that signs have been inverted because in the last exercise a shorter time meant good performance but in this case a larger average speed means good performance.