Assignment 1 - Examining multivariate data

GROUP 03

Agustin Valencia - aguva779
Bayu Brahmantio - baybr878
Joris van Doorn - jorva845
Marcos F Mourao - marfr825

18 December 2019

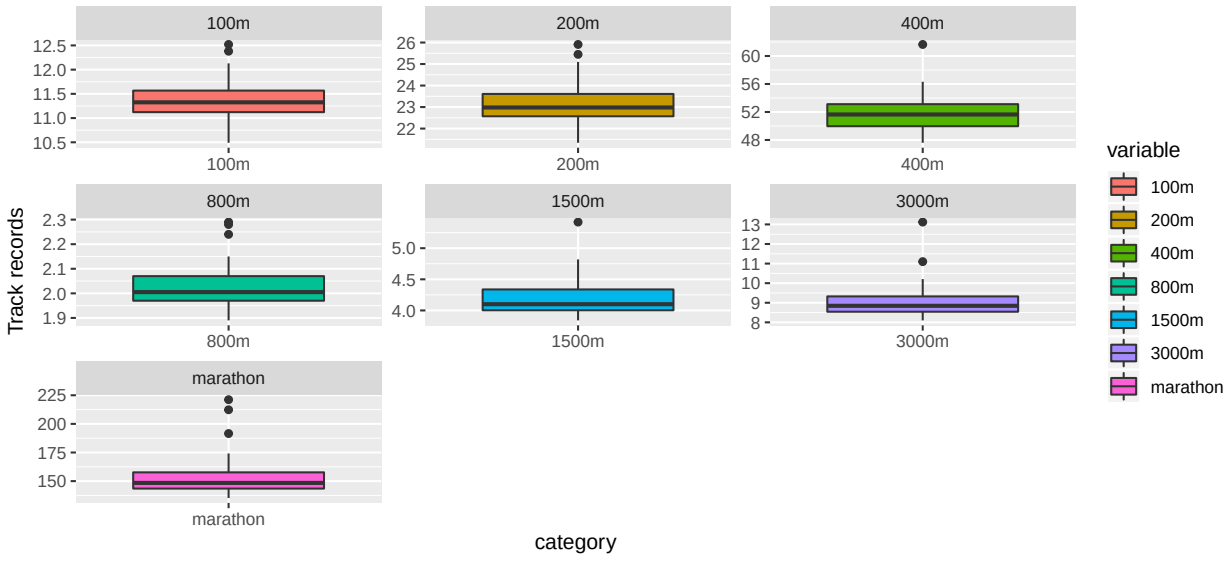
Question 1: Describing individual variables

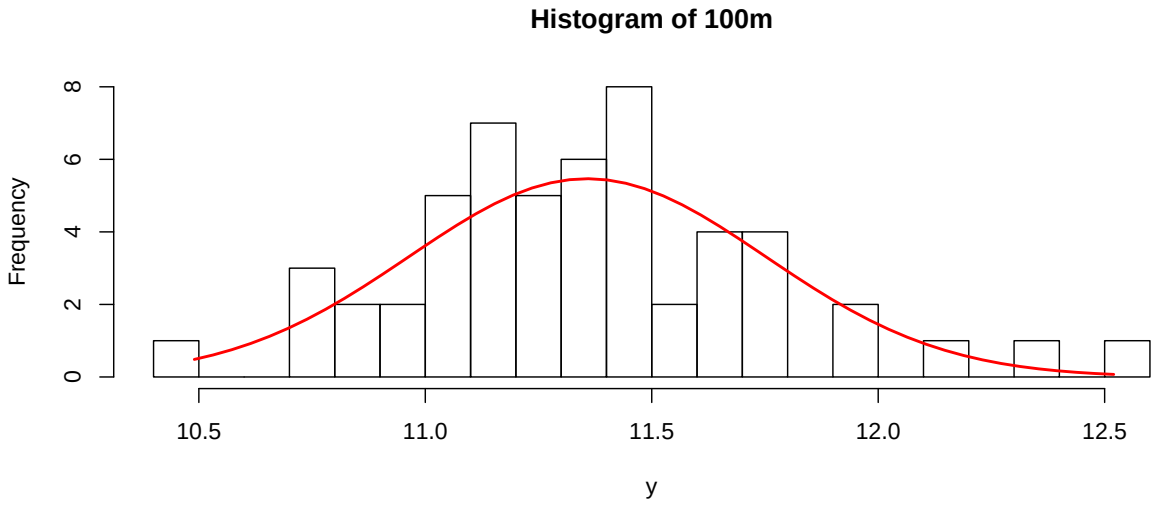
a) Describe the 7 variables with mean values, standard deviations e.t.c.

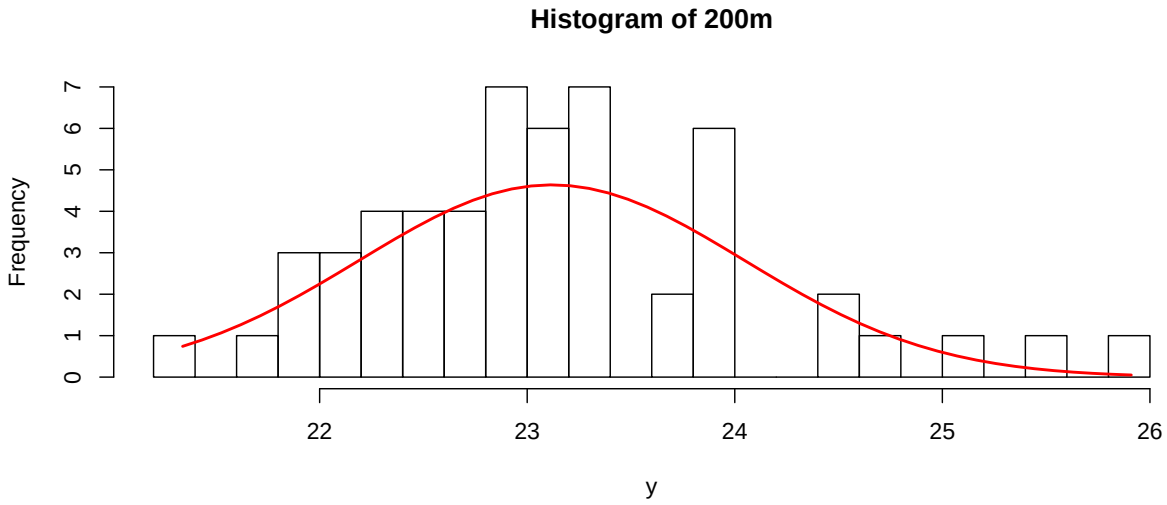
```
## ** Summarizing data **

##
##
## *data path:  data/T1-9.dat

##
##
## *Column means:

##       100m       200m       400m       800m      1500m      3000m
##  11.357778  23.118519  51.989074   2.022407   4.189444   9.080741
##   marathon
## 153.619259

##
## *Variances:

## [1] 1.553157e-01 8.630883e-01 6.745458e+00 7.546925e-03 7.418270e-02
## [6] 6.647579e-01 2.702702e+02

##
## *Total Sample Variance:

## [1] 278.7805

##
## *Generalized Sample Variance:

## [1] 8.195897e-07
```

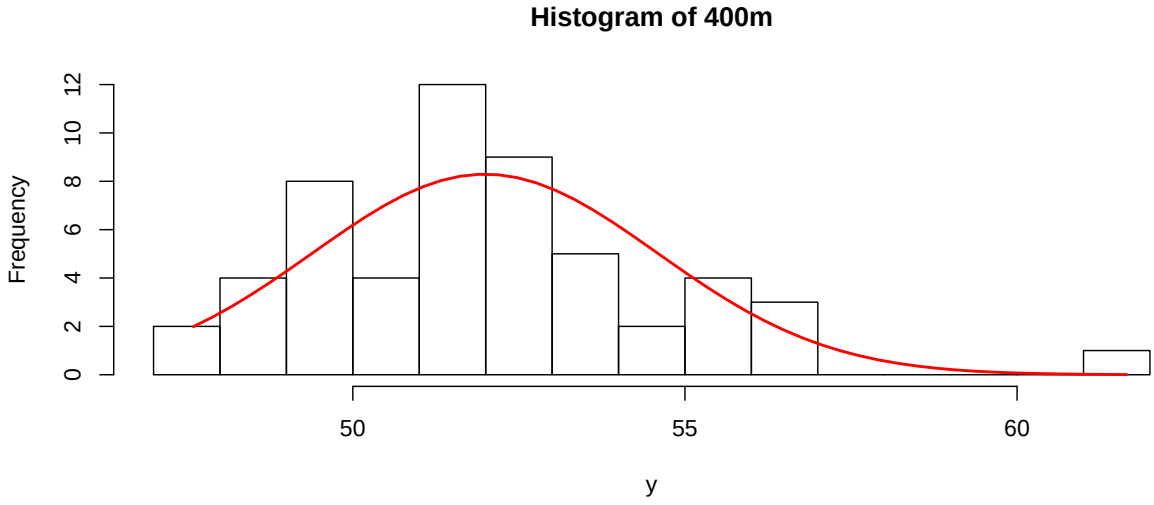
b) Illustrate the variables with different graphs (explore what plotting possibilities R has). Make sure that the graphs look attractive (it is absolutely necessary to look at the labels, font sizes, point types). Are there any apparent extreme values? Do the variables seem normally distributed? Plot the best fitting (match the mean and standard deviation, i.e. method of moments) Gaussian density curve on the data's histogram. For the last part you may be interested in the hist() and density() functions.

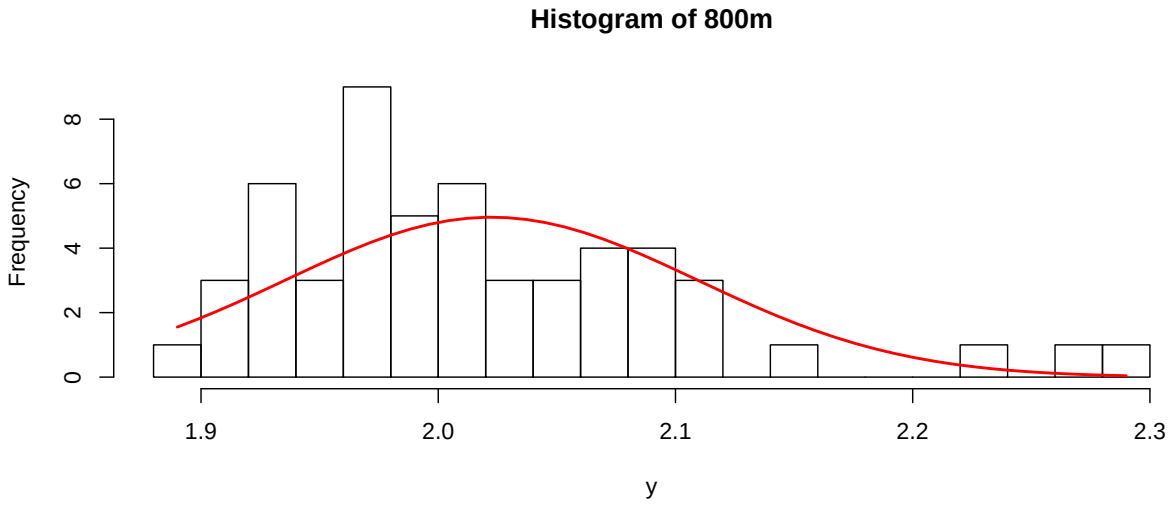
```
## Using country as id variables
```

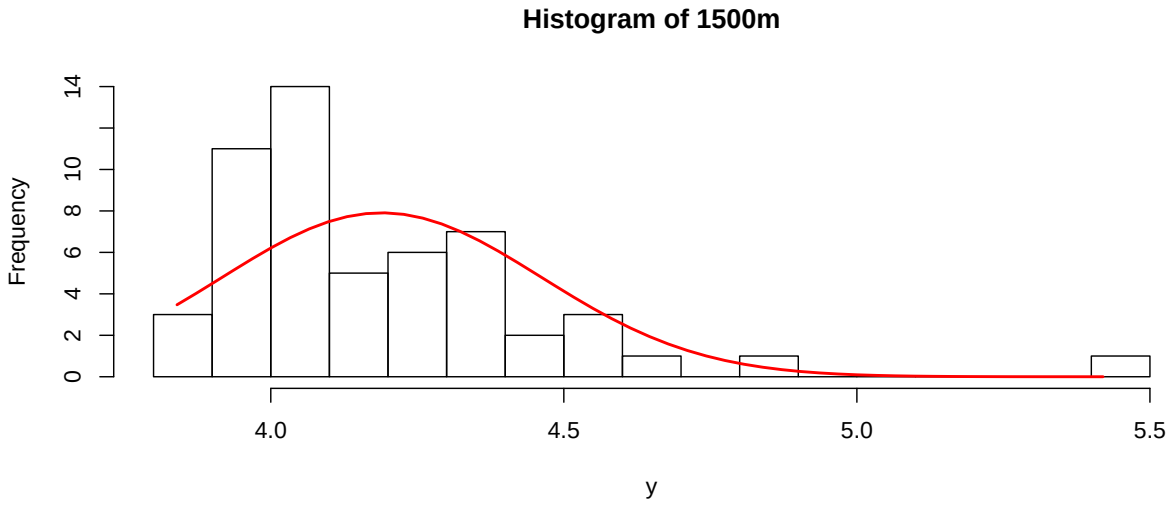

100m
200m
400m
800m
1500m
3000m
marathon
Track records
category
variable
100m
200m
400m
800m
1500m
3000m
marathon

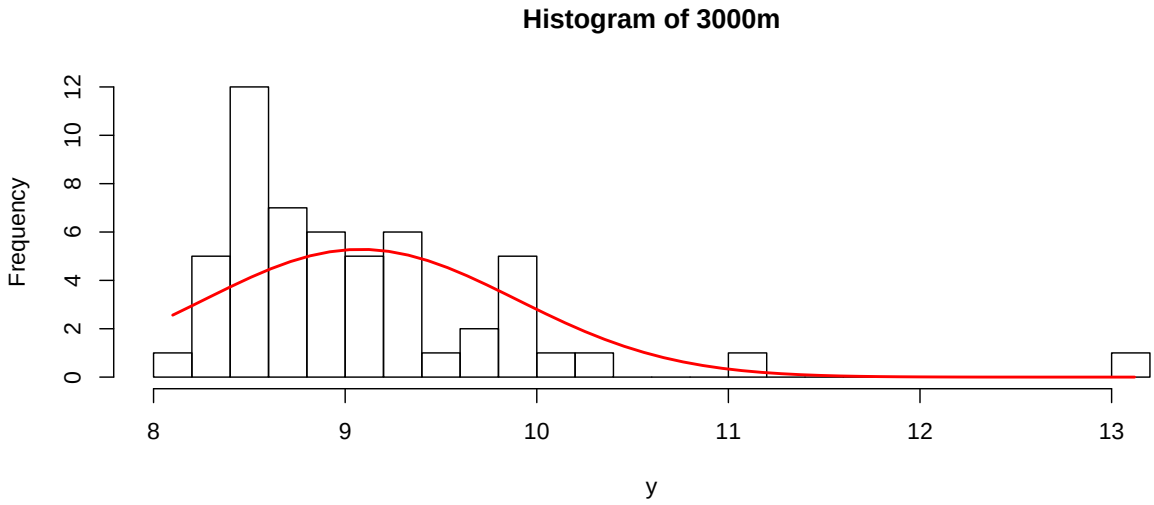

Histogram of 100m
Frequency
y

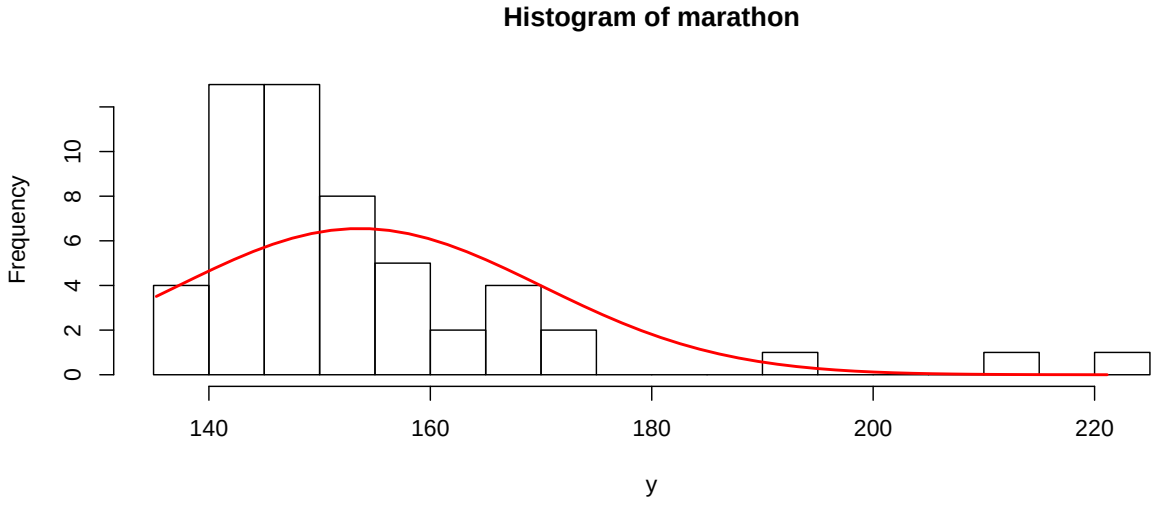

Histogram of 200m
Frequency
0
1
2
3
4
5
6
7
22
23
24
25
26
y


Histogram of 400m
Frequency
0
2
4
6
8
10
12
50
55
60
y


Histogram of 800m
Frequency
0
2
4
6
8
1.9
2.0
2.1
2.2
2.3
y


Histogram of 1500m
Frequency
0
2
4
6
8
10
14
4.0
4.5
5.0
5.5
y





The first categories (100m, 200m and 400m) seem normally distributed by looking at the histograms. The longer races are have more skewed to the right histograms.

Question 2: Relationships between the variables

a) Compute the covariance and correlation matrices for the 7 variables. Is there any apparent structure in them? Save these matrices for future use.

Covariance Matrix:

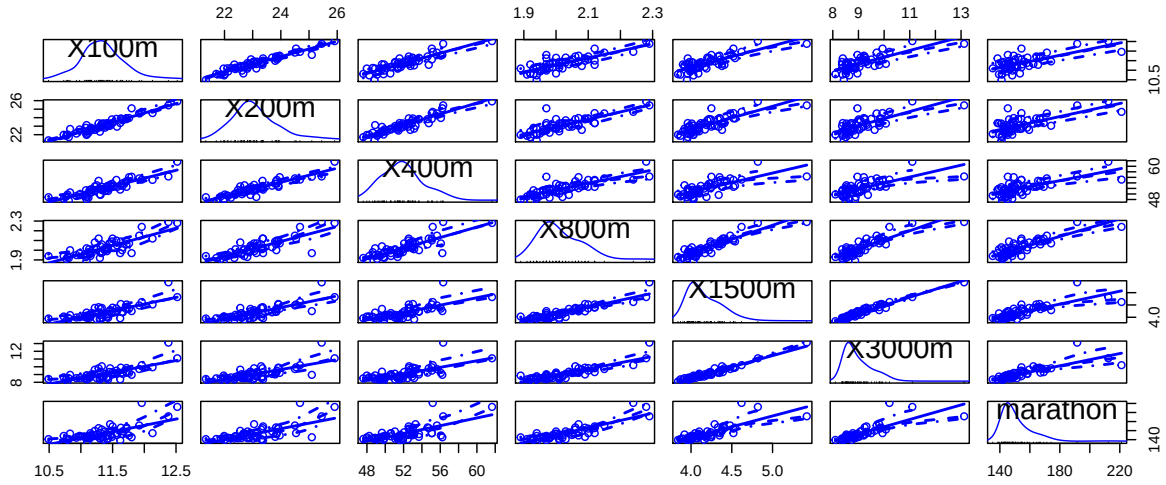
	100m	200m	400m	800m	1500m	3000m	marathon
100m	0.1553157	0.3445608	0.8912960	0.0277036	0.0838912	0.2338828	4.334178
200m	0.3445608	0.8630883	2.1928363	0.0661659	0.2027633	0.5543502	10.384988
400m	0.8912960	2.1928363	6.7454576	0.1818079	0.5091768	1.4268158	28.903731
800m	0.0277036	0.0661659	0.1818079	0.0075469	0.0214146	0.0613793	1.219655
1500m	0.0838912	0.2027633	0.5091768	0.0214146	0.0741827	0.2161551	3.539837
3000m	0.2338828	0.5543502	1.4268158	0.0613793	0.2161551	0.6647579	10.706091
marathon	4.3341776	10.3849876	28.9037314	1.2196546	3.5398373	10.7060911	270.270150

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

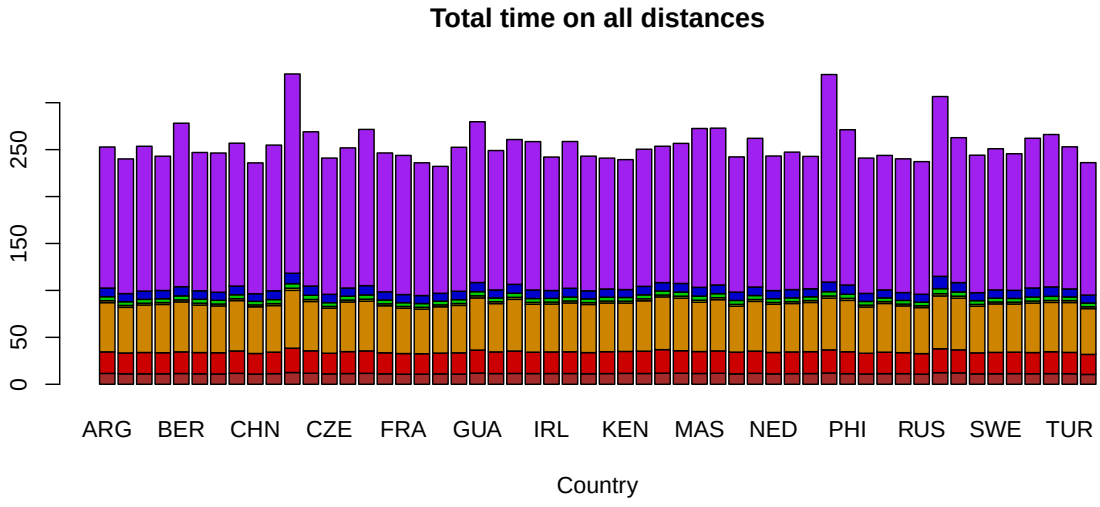
When comparing the shorter distances, to the middle-long and long distances in the correlation matrix, it can be concluded that countries that have a high performance in "shorter" races (100m, 200m and 400m) do not necessarily have high performance in "long distance" races (800m, 1500m, 3000m and marathon). This is coherent to the fact that short and long races require different training. Normally, the athletes are different altogether in these different categories.

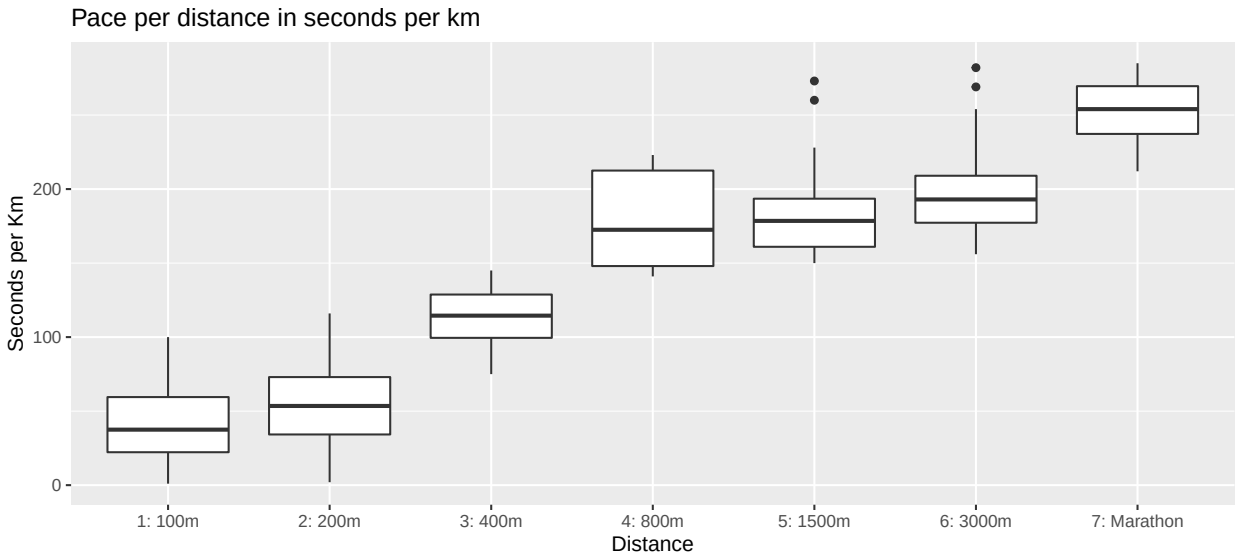
b) Generate and study the scatterplots between each pair of variables. Any extreme values?



The closer two distances are two each other, the more clustered the distribution of track records. Which makes intuitive sense, because if a country has no great runners on the 100m, then it is like that they also don't have great athletes on the 200m (and visa versa) because those are often the same people. The reversed logic applies here, because a country can have great sprinters, but no long-distance runne. These events are less correlated on each other because they require specific training.

c) Explore what other plotting possibilities R offers for multivariate data. Present other (at least two) graphs that you find interesting with respect to this data set.

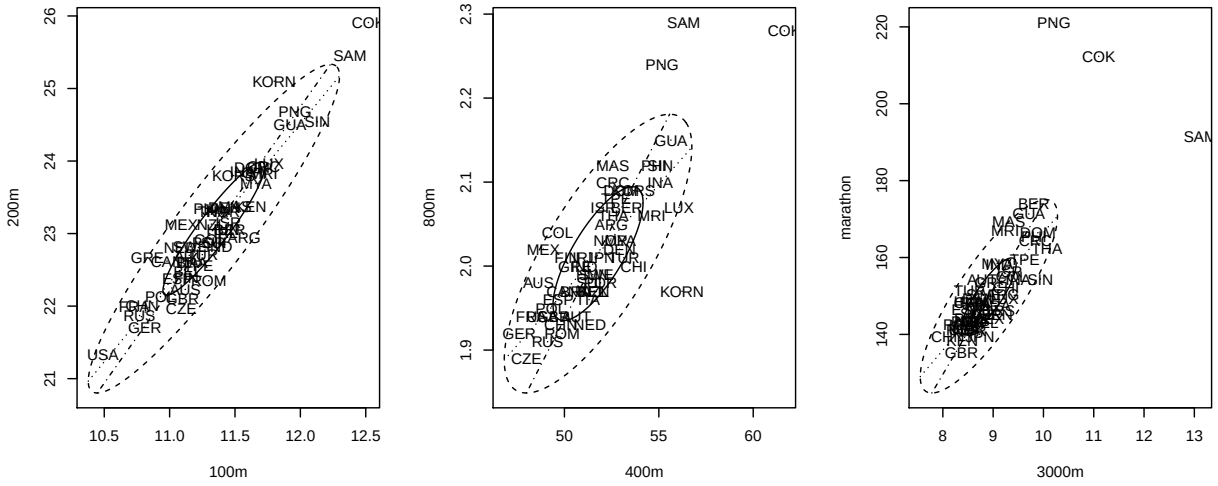



Pace per distance in seconds per km
Seconds per Km
200
100
0
1: 100m
2: 200m
3: 400m
4: 800m
5: 1500m
6: 3000m
7: Marathon
Distance

Question 3: Examining for extreme values

a) Look at the plots (esp. scatterplots) generated in the previous question. Which 3–4 countries appear most extreme? Why do you consider them extreme?

s we can see from the scatterplots in 2b, there are some points that stand out in each bivariate plot. If we take a closer look at few plots of variable pairs:



we can see that some countries are consistently located out of the "fence" such as `PNG`, `COK`, `SAM`, and `KORN`.

b) The most common residual is the Euclidean distance between the observation and sample mean vector, i.e.

$$d(\overrightarrow{x}, \overline{x}) = \sqrt{(\overrightarrow{x} - \overline{x})^T (\overrightarrow{x} - \overline{x})}$$

This distance can be immediately generalized to the L^r, r > 0 distance as

$$d_{L^r}(\overrightarrow{x}, \overline{x}) = \left(\sum_{i=1}^{p} |\overrightarrow{x}_i - \overline{x}_i|^r \right)^{1/r}$$

where p is the dimension of the observation (here p = 7).

Compute the squared Euclidean distance (i.e. r = 2) of the observation from the sample mean for all 55 countries using R's matrix operations. First center the raw data by the means to get x - mean for each country. Then do a calculation with matrices that will result in a matrix that has on its diagonal the requested squared distance for each country. Copy this diagonal to a vector and report on the five most extreme countries. In this questions you MAY NOT use any loops.

```
## Warning in sqrt((centered) %*% t(centered)): NaNs produced

## [1] "Top 5 distance extreme countries:"

## [1] PNG COK SAM BER GBR
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA

## [1] "Sweden's distance rank:  48"
```

c) The different variables have different scales so it is possible that the distances can be dominated by some few variables. To avoid this we can use the squared distance,

$$d^2_{\mathbf{V}}(\overrightarrow{x}, \overline{x}) = (\overrightarrow{x} - \overline{x})^T \mathbf{V^{-1}} (\overrightarrow{x} - \overline{\mathbf{x}}),$$

where V is a diagonal matrix with variances of the appropriate variables on the diagonal. The effect is that for each variable the squared distance is divided by its variance and we have a scaled independent distance.

It is simple to compute this measure by standardizing the raw data with both means (centering) and standard deviations (scaling), and then compute the Euclidean distance for the normalized data. Carry out these computations and conclude which countries are the most extreme ones. How do your conclusions compare with the unnormalized ones?

```
## [1] "Top 5 distance extreme countries:"
```

```
## [1] SAM COK PNG USA SIN
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

```
## [1] "Sweden's distance rank:  50"
```

The euclidean distance for normalized data results in a different country list for most extreme ones, altough some countries are in both lists, but not necessarily in the same positions: Samoa (SAM), Cook Islands (COK) and Papua-New Guinea (PNG). This second set of extremes is more consistent because the for the first three race categories (100m, 200m and 400m), time is measured in seconds while the others (800m, 1500m, 3000m and marathon) are measured in minutes.

d) The most common statistical distance is the Mahalanobis distance,

$$d^2_{\mathrm{M}}(\overrightarrow{x}, \overline{x}) = (\overrightarrow{x} - \overline{x})^T \mathbf{C^{-1}} (\overrightarrow{x} - \overline{\mathbf{x}}),$$

where C is the sample covariance matrix calculated from the data. With this measure we also use the relationships (covariances) between the variables (and not only the marginal variances as dV(· , ·) does). Compute the Mahalanobis distance, which countries are most extreme now?

```
## [1] "Top 5 distance extreme countries:"
```

```
## [1] SAM  PNG  KORN COK  MEX
## 54 Levels: ARG AUS AUT BEL BER BRA CAN CHI CHN COK COL CRC CZE DEN ... USA
```

```
## [1] "Sweden's distance rank:  54"
```

The most extreme countries given by the Mahalanobis distance are: Samoa (SAM), Papua New Guinea (PNG), North Korea (KORN), Cook Islands (COK) and Mexico (MEX).

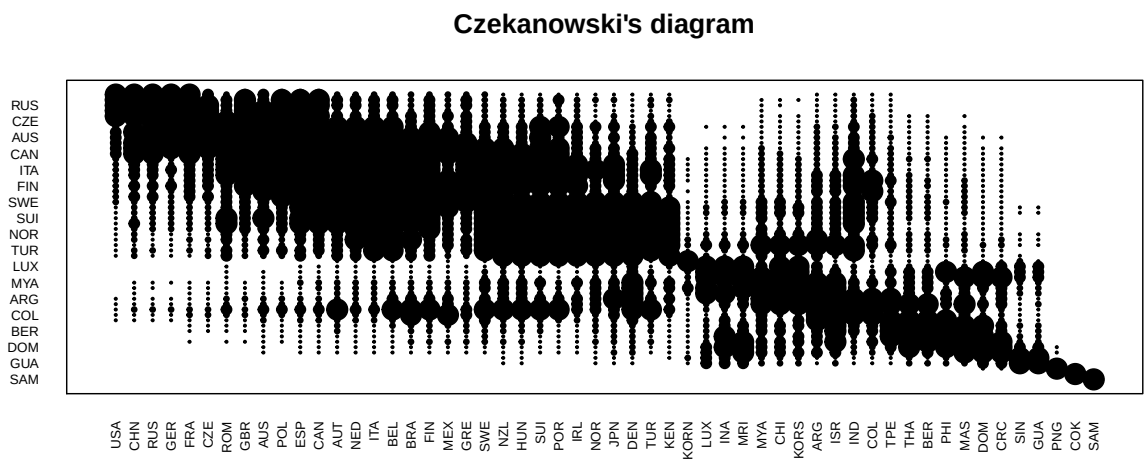
e) Compare the results in b)–d). Some of the countries are in the upper end with all the measures and perhaps they can be classified as extreme. Discuss this. But also notice the different measures give rather different results (how does Sweden behave?). Summarize this graphically. Produce Czekanowski's diagram using e.g. the `RMaCzek` package. In case of problems please describe them.

In this case, extreme countries normally have poor performance in races, but that is not always the case. For some distances definitions, high performance countries like USA and Great Britain appear as extremes

as well. So "extremism" is not a measure of how "slow" a country is, but rather how far from the overall mean the country is. Graphically, this means that the region of equal euclidean distance (a hypersphere) gets distorted to regions of equal statistical distance (hyperellipsoids) with different statistical weights applied: the marginal variances in c) and variance-covariances in d).

By ranking the countries in decreasing order of distance, Sweden's lowers in position in the rank for the different distances definitions in questions b), c), and d) respectively (48th, 50th and 54th place). The low mahalonobis distance from Sweden to the mean indicates that Sweden is not particularly bad at any type of races compared to other countries.

The Czekanowski's diagram is showed below:



Appendix

```r
RNGversion("3.5.1")
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(fig.width = 9, fig.height = 4.1)
knitr::opts_chunk$set(tidy.opts = list(width.cutoff = 5), tidy = TRUE)
library(car)
library(RMaCzek)
library(knitr)
library(reshape)
library(ggplot2)
library(tidyr)
library(MVA)
data_path <- "data/T1-9.dat"
data <- read.table(data_path)
colnames(data) <- c("country", "100m", "200m", "400m", "800m",
    "1500m", "3000m", "marathon")
data <- data[, 2:8]
# Means
column_means <- colMeans(data)

# Variance
# -
# Covariance
# Matrix
vars <- var(data)

# Correlations
cors <- cor(data)

# Total
# sample
# variance
total_sample_var <- sum(diag(vars))

# Generalized
# sample
# variance
gen_sample_var <- det(vars)

cat("** Summarizing data **")
cat("\n\n*data path: ", data_path)

cat("\n\n*Column means: \n")
print(column_means)

cat("\n*Variances: \n")
variances <- c()
for (i in 1:7) {
    variances[i] <- var(data[, i])
}
print(variances)
```

```r
cat("\n*Total Sample Variance: \n")
print(total_sample_var)

cat("\n*Generalized Sample Variance: \n")
print(gen_sample_var)
# Boxplot
# per
# race
# category
# melted
# data
data <- read.table(data_path)
colnames(data) <- c("country", "100m", "200m", "400m", "800m",
    "1500m", "3000m", "marathon")
meltdata <- melt(data)
p <- ggplot(data = meltdata, aes(x = variable, y = value)) +
    geom_boxplot(aes(fill = variable)) + labs(x = "category",
    y = "Track records")
boxp <- p + facet_wrap(~variable, scales = "free")
boxp
# identifying
# outliets
# in
# boxplots
# for
# each
# category:

detect.outlier <- function(x) {
    # note:
    # x
    # should
    # be a
    # column
    # of
    # the
    # data.
    # e.g.
    # data$`100m`
    p <- boxplot(x)
    # numeric
    # outliers
    num_out <- p$out

    ind <- c()
    for (i in 1:length(num_out)) {
        ind[i] <- which(x == num_out[i])
    }
    out_country <- data[ind, 1]
    return(out_country)
}

detect.outlier(data$`100m`)
```

```r
detect.outlier(data$`200m`)
detect.outlier(data$`400m`)
detect.outlier(data$`800m`)
detect.outlier(data$`1500m`)
detect.outlier(data$`3000m`)
detect.outlier(data$marathon)
fit_normals <- function(y, var_name) {
    h <- hist(y, breaks = 20, main = paste("Histogram of", var_name))
    x_normal <- seq(min(y), max(y), length = 50)
    y_normal <- dnorm(x_normal, mean = mean(y), sd = sd(y))
    y_normal <- y_normal * diff(h$mids[1:2]) * length(y)
    lines(x_normal, y_normal, col = "red", lwd = 2)
}

for (i in c(2:8)) {
    fit_normals(data[, i], colnames(data)[i])
}

kable(vars)
kable(cors)
scatterplotMatrix(data[, 2:8])
barplot(t(data[, 2:8]), main = "Total time on all distances",
    xlab = "Country", col = c("brown", "red3", "orange3", "yellow3",
        "green3", "blue3", "purple"), names.arg = t(data[, 1]),
    legend = rownames(t(data[, 1])))
pace <- as.matrix(data[, 2:8])
pace <- as.numeric(pace)
distance <- rep(NA, 378)

for (i in 1:54) {
    pace[i] <- pace[i] * 10
    distance[i] <- "1: 100m"
}
for (i in 55:108) {
    pace[i] <- pace[i] * 5
    distance[i] <- "2: 200m"
}
for (i in 109:162) {
    pace[i] <- pace[i] * 2.5
    distance[i] <- "3: 400m"
}
for (i in 163:216) {
    minutes <- floor(pace[i])
    min_to_sec <- 60 * minutes
    sec <- (pace[i] - minutes) * 100
    pace[i] <- (sec + min_to_sec) * 1.25
    distance[i] <- "4: 800m"
}
for (i in 217:270) {
    minutes <- floor(pace[i])
    min_to_sec <- 60 * minutes
    sec <- (pace[i] - minutes) * 100
    pace[i] <- (sec + min_to_sec) * (2/3)
```

```
    distance[i] <- "5: 1500m"
}
for (i in 271:324) {
    minutes <- floor(pace[i])
    min_to_sec <- 60 * minutes
    sec <- (pace[i] - minutes) * 100
    pace[i] <- (sec + min_to_sec)/3
    distance[i] <- "6: 3000m"
}
for (i in 325:378) {
    minutes <- floor(pace[i])
    min_to_sec <- 60 * minutes
    sec <- (pace[i] - minutes) * 100
    pace[i] <- (sec + min_to_sec)/42.195
    distance[i] <- "7: Marathon"
}
speed <- cbind(pace, distance)

speed <- as.data.frame(speed)
speed$pace <- as.numeric(speed$pace)

# ggplot(speed,
# aes(x
# =
# distance,
# y =
# pace))
# +
# geom_point()
# +
# ggtitle('Pace
# per
# distance
# in
# seconds
# per
# km')
# +
# xlab('Distance')
# +
# ylab('Seconds
# per
# Km')


ggplot(speed, aes(x = distance, y = pace)) + geom_boxplot() +
    ggtitle("Pace per distance in seconds per km") + xlab("Distance") +
    ylab("Seconds per Km")

data <- read.table(data_path)
colnames(data) <- c("country", "100m", "200m", "400m", "800m",
    "1500m", "3000m", "marathon")
bivariate_boxplot = function(data, vars) {
```

```
    x = data[, vars]
    bvbox(x, cex = 0.01, xlab = colnames(data)[vars[1]], ylab = colnames(data)[vars[2]])
    with(x, text(x[, 1], x[, 2], cex = 1, labels = as.character(data[,
        1])))
}
attach(data)
par(mfrow = c(1, 3))
bivariate_boxplot(data, c(2, 3))
bivariate_boxplot(data, c(4, 5))
bivariate_boxplot(data, c(7, 8))
# euclidean
# distance
records <- as.matrix(data[, 2:8])

# center
# the
# data
# deviation
# matrix
centered <- scale(x = records, center = TRUE, scale = FALSE)

euclid_dist <- sqrt((centered) %*% t(centered))

# diagonal
euclid_diag <- sort(diag(euclid_dist), decreasing = TRUE, index.return = TRUE)
# extreme
# values
# (top
# 5)
euclid_extremes <- head(euclid_diag, n = 5)
# extract
# extreme
# countries
ind <- head(euclid_extremes$ix, 5)
euclid_extreme_countries <- data[ind, 1]
# Sweden's
# index
SWE_ind <- which(data[, 1] == "SWE")
# Sweden's
# Position
SWE_rank_euclid <- which(euclid_extremes$ix == SWE_ind)

print("Top 5 distance extreme countries:")
euclid_extreme_countries
paste("Sweden's distance rank: ", SWE_rank_euclid)
# diagonal
# of
# variance
# covariance
# matrix
covars <- cov(records)
V <- matrix(0, nrow = ncol(records), ncol = ncol(records))
diag(V) <- diag(covars)
```

```r
# compute
# distance
dist3c <- (centered %*% solve(V) %*% t(centered))^(1/2)

# diagonal
diag3c <- sort(diag(dist3c), decreasing = TRUE, index.return = TRUE)
# extreme
# values
# (top
# 5)
extremes3c <- head(diag3c, n = 5)
# extract
# extreme
# countries
ind3c <- head(extremes3c$ix, 5)
extreme_countries3c <- data[ind3c, 1]
# Sweden's
# Position
SWE_rank_3c <- which(extremes3c$ix == SWE_ind)

print("Top 5 distance extreme countries:")
extreme_countries3c
paste("Sweden's distance rank: ", SWE_rank_3c)
dist3d <- (centered %*% solve(covars) %*% t(centered))^(1/2)
# diagonal
diag3d <- sort(diag(dist3d), decreasing = TRUE, index.return = TRUE)
# extreme
# values
# (top
# 5)
extremes3d <- head(diag3d, n = 5)
# extract
# extreme
# countries
ind3d <- head(extremes3d$ix, 5)
extreme_countries3d <- data[ind3d, 1]
# Sweden's
# Position
SWE_rank_3d <- which(extremes3d$ix == SWE_ind)

print("Top 5 distance extreme countries:")
extreme_countries3d
paste("Sweden's distance rank: ", SWE_rank_3d)
library(car)
df = data.matrix(data)
rownames(df) = data[, 1]
df = df[, 2:8]
x = czek_matrix(df)
plot(x)
```