Assignment 1: Examining multivariate data

Mohsen Pirmoradiyan, Ahmed Alhasan, Asad Enver, Ali Etminan, Mubarak Hussain

11/20/2019

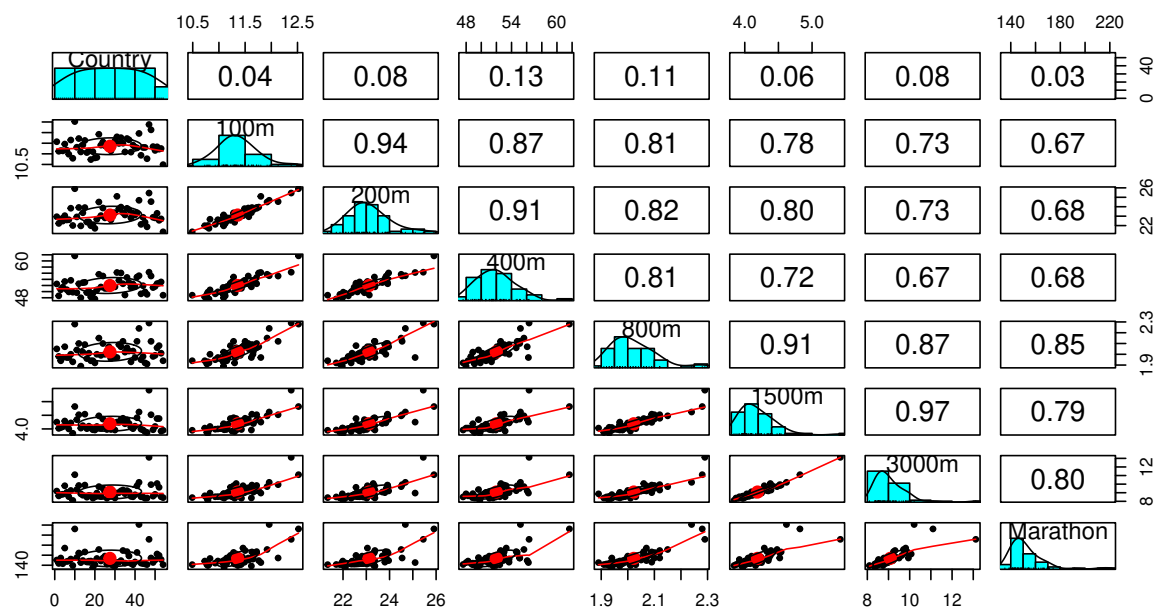
Question 1: Describing individual variables

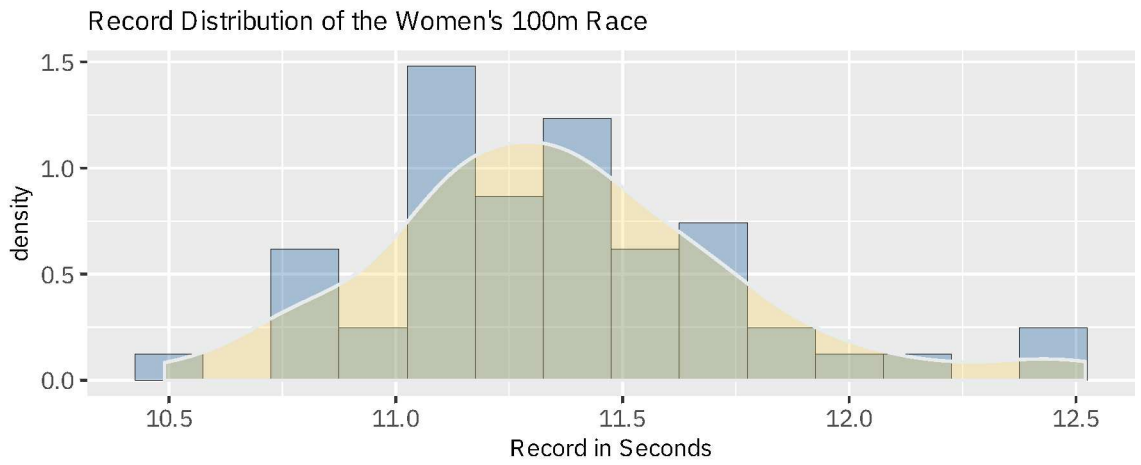
Consider the data set in the T1-9.dat
le, National track records for women. For 55 different countries we have the national records for 7 variables (100; 200; 400; 800; 1500; 3000m and marathon). Use R to do the following analyses.

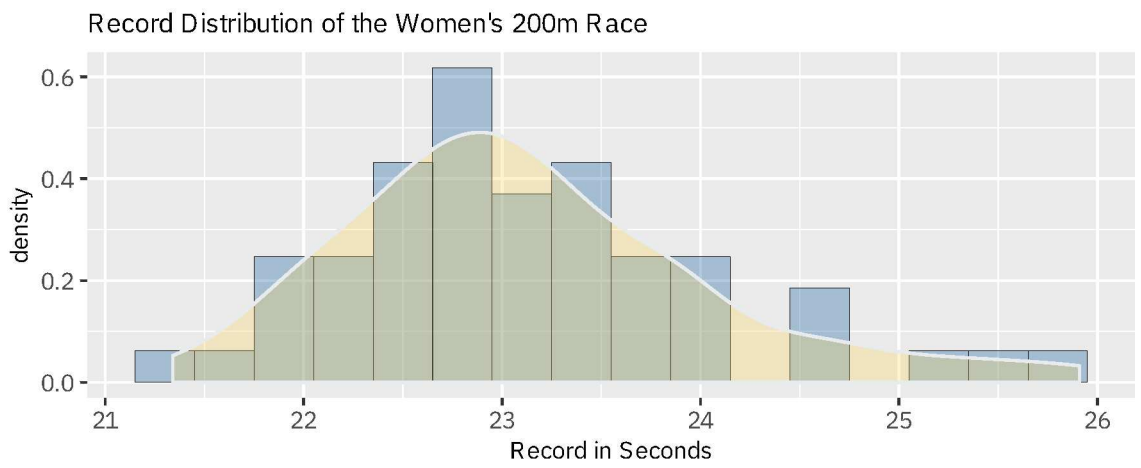
a) Describe the 7 variables with mean values, standard deviations e.t.c.

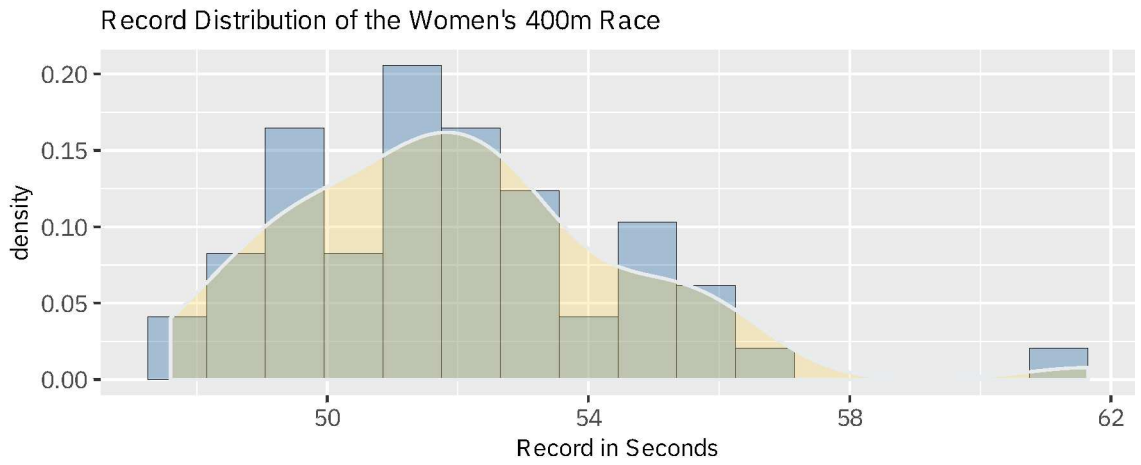
	Min.	Median	Mean	SD	Max.
100m	10.49	11.32	11.36	0.39	12.52
200m	21.34	22.98	23.12	0.93	25.91
400m	47.60	51.64	51.99	2.60	61.65
800m	1.89	2.00	2.02	0.09	2.29
1500m	3.84	4.10	4.19	0.27	5.42
3000m	8.10	8.84	9.08	0.82	13.12
Marathon	135.25	148.43	153.62	16.44	221.14

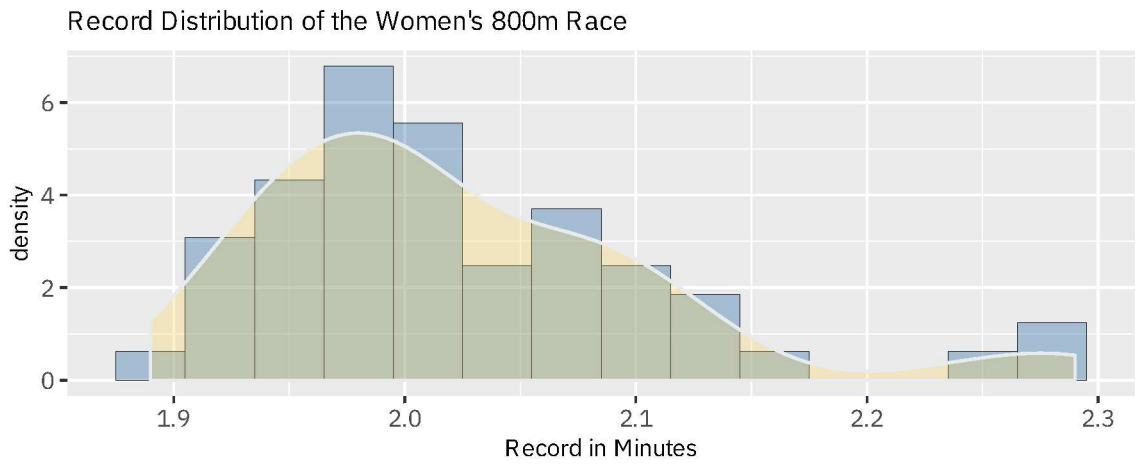
b) Illustrate the variables with different graphs (explore what plotting possibilities R has). Make sure that the graphs look attractive (it is absolutely necessary to look at the labels, font sizes, point types). Are there any apparent extreme values? Do the variables seem normally distributed? Plot 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.

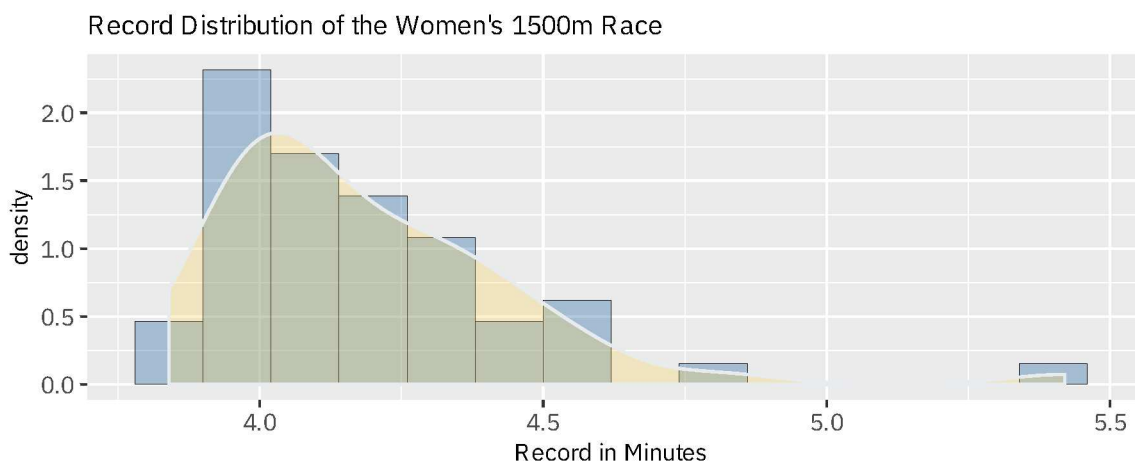


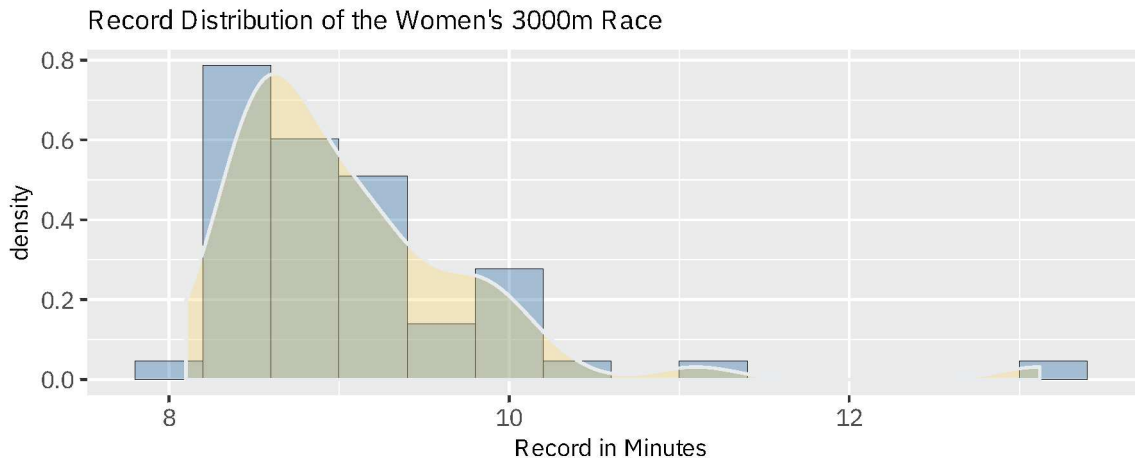

Record Distribution of the Women's 100m Race
density
1.5
1.0
0.5
0.0
10.5
11.0
11.5
12.0
12.5
Record in Seconds

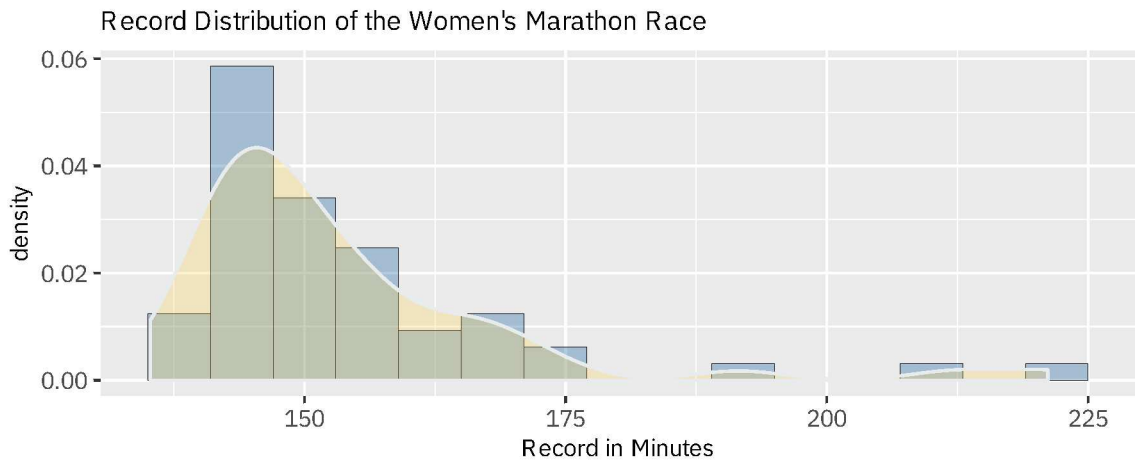

Record Distribution of the Women's 200m Race
density
0.6
0.4
0.2
0.0
21
22
23
24
25
26
Record in Seconds


Record Distribution of the Women's 400m Race
density
0.20
0.15
0.10
0.05
0.00
50
54
58
62
Record in Seconds


Record Distribution of the Women's 800m Race
density
6
4
2
0
1.9
2.0
2.1
2.2
2.3
Record in Minutes


Record Distribution of the Women's 1500m Race
density
2.0
1.5
1.0
0.5
0.0
4.0
4.5
5.0
5.5
Record in Minutes


Record Distribution of the Women's 3000m Race
density
0.8
0.6
0.4
0.2
0.0
8
10
12
Record in Minutes



Analysis:

You are correct in saying that these countries are outliers in "each variable". But note that in general, a data point being an outlier in a single histogram does not mean that it is an outlier in the entire data set: if you have, instead, an Olympics data set where there are 40 sports. In this situation, we can expect many countries outlie in at least one, but, of course, it is not quite appropriate to say that they are all "outliers".

- Looking at the scatterplots we can easily identify 3 to 5 outliers (countries) for each variable. As the race gets longer it is more easier to identify these outliers.
- The first two variables are good approximation to normal distribution, however it starts to be skewed from the third one and the longer the race the more right skewd it will get (more like gamma distribution)

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.

`## Correlation Matrix:`

	100m	200m	400m	800m	1500m	3000m	Marathon
100m	1.00	0.94	0.87	0.81	0.78	0.73	0.67
200m	0.94	1.00	0.91	0.82	0.80	0.73	0.68
400m	0.87	0.91	1.00	0.81	0.72	0.67	0.68
800m	0.81	0.82	0.81	1.00	0.91	0.87	0.85
1500m	0.78	0.80	0.72	0.91	1.00	0.97	0.79
3000m	0.73	0.73	0.67	0.87	0.97	1.00	0.80
Marathon	0.67	0.68	0.68	0.85	0.79	0.80	1.00

`## Covariance Matrix:`

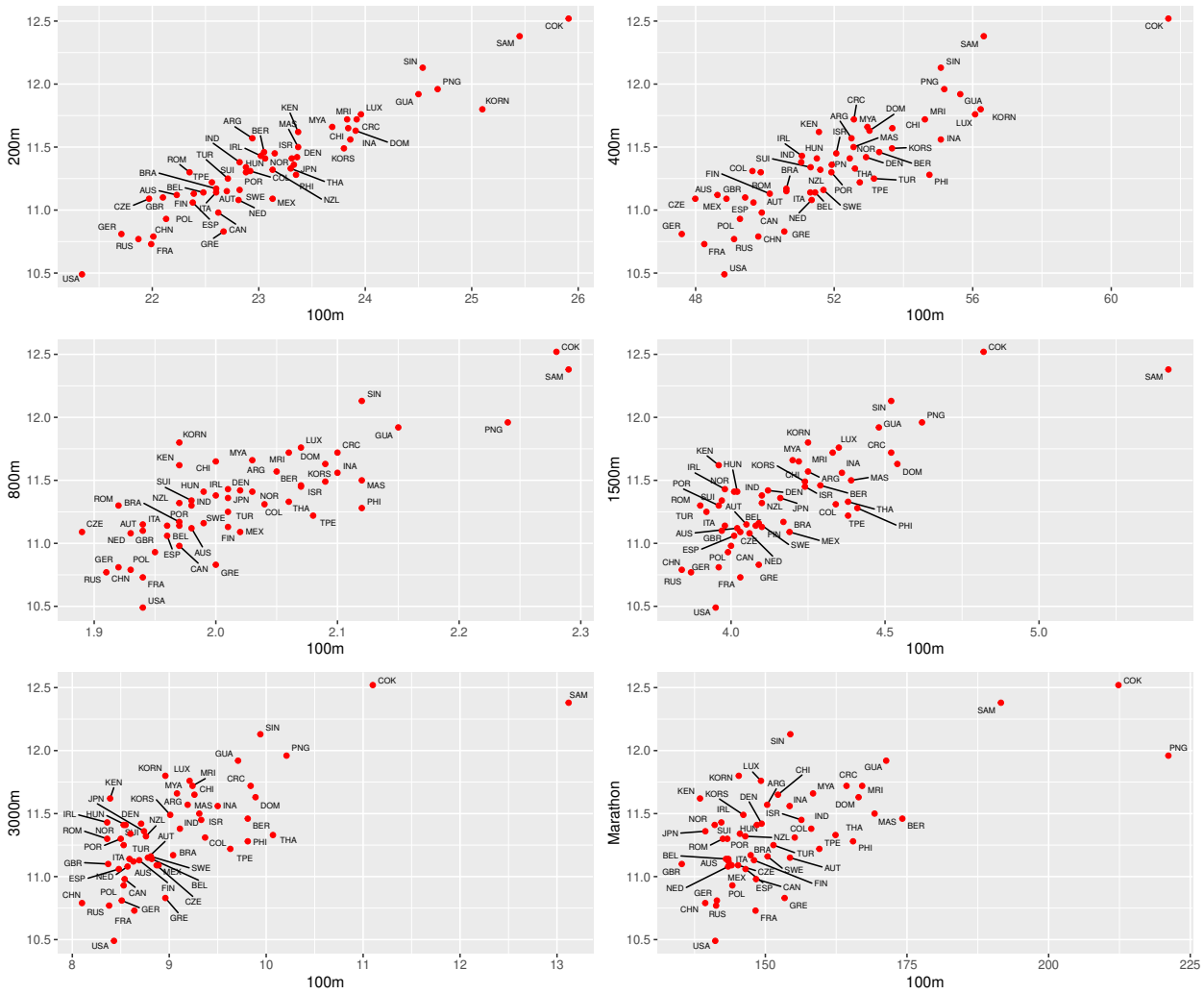
	100m	200m	400m	800m	1500m	3000m	Marathon
100m	0.16	0.34	0.89	0.03	0.08	0.23	4.33
200m	0.34	0.86	2.19	0.07	0.20	0.55	10.38
400m	0.89	2.19	6.75	0.18	0.51	1.43	28.90
800m	0.03	0.07	0.18	0.01	0.02	0.06	1.22
1500m	0.08	0.20	0.51	0.02	0.07	0.22	3.54

	100m	200m	400m	800m	1500m	3000m	Marathon
3000m	0.23	0.55	1.43	0.06	0.22	0.66	10.71
Marathon	4.33	10.38	28.90	1.22	3.54	10.71	270.27

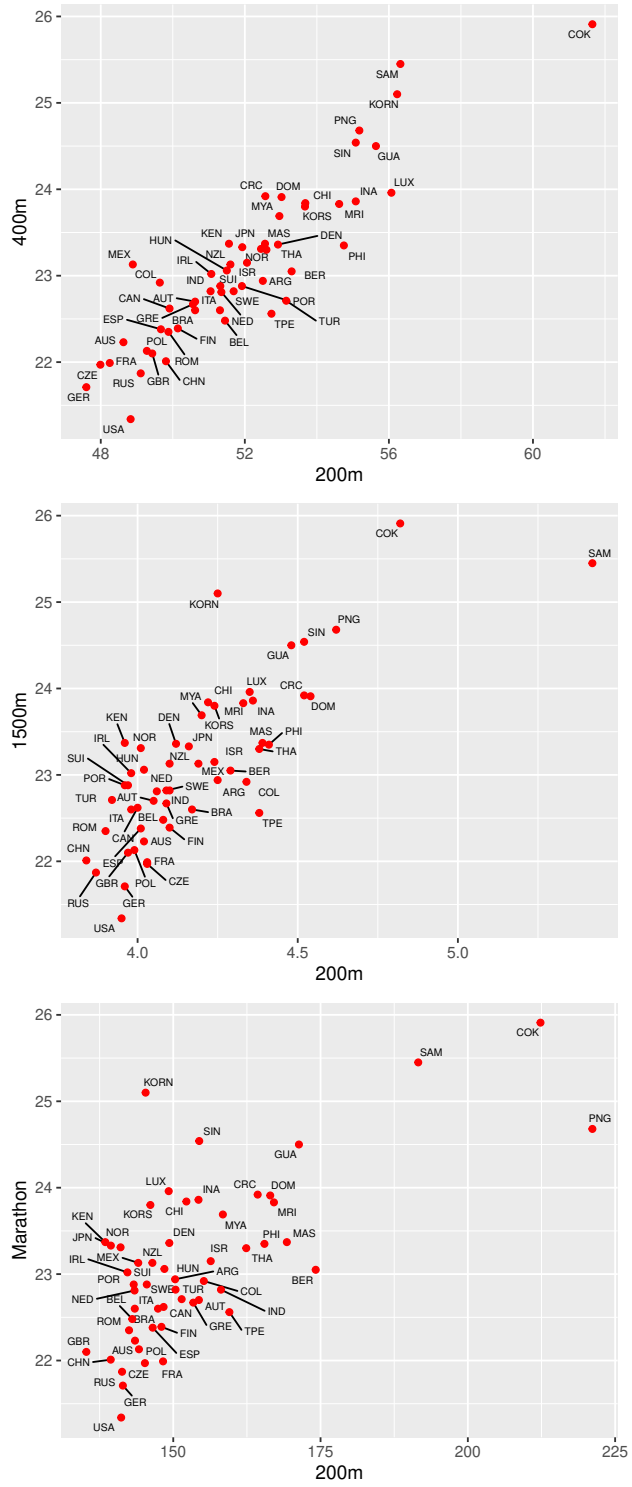
Analysis:

- There is a structure within the correlation data, it can be identified that there is high correlation between shorter races and also high correlation between longer races but the shorter and longer races are less correlated.

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




100m
200m
10.5
11.0
11.5
12.0
12.5
22
23
24
25
26
COK
SAM
KORN
PNG
SIN
GUA
DOM
LUX
CRC
INA
MRI
CHI
KORS
PHI
THA
JPN
DEN
MYA
MEX
NZL
NOR
ISR
MAS
KEN
NED
SWE
COL
HUN
BER
CAN
AUT
POR
IRL
ARG
GRE
ESP
ITA
BRA
SUI
IND
TUR
POL
FIN
BEL
ROM
FRA
CHN
GBR
AUS
TPE
RUS
CZE
GER
USA

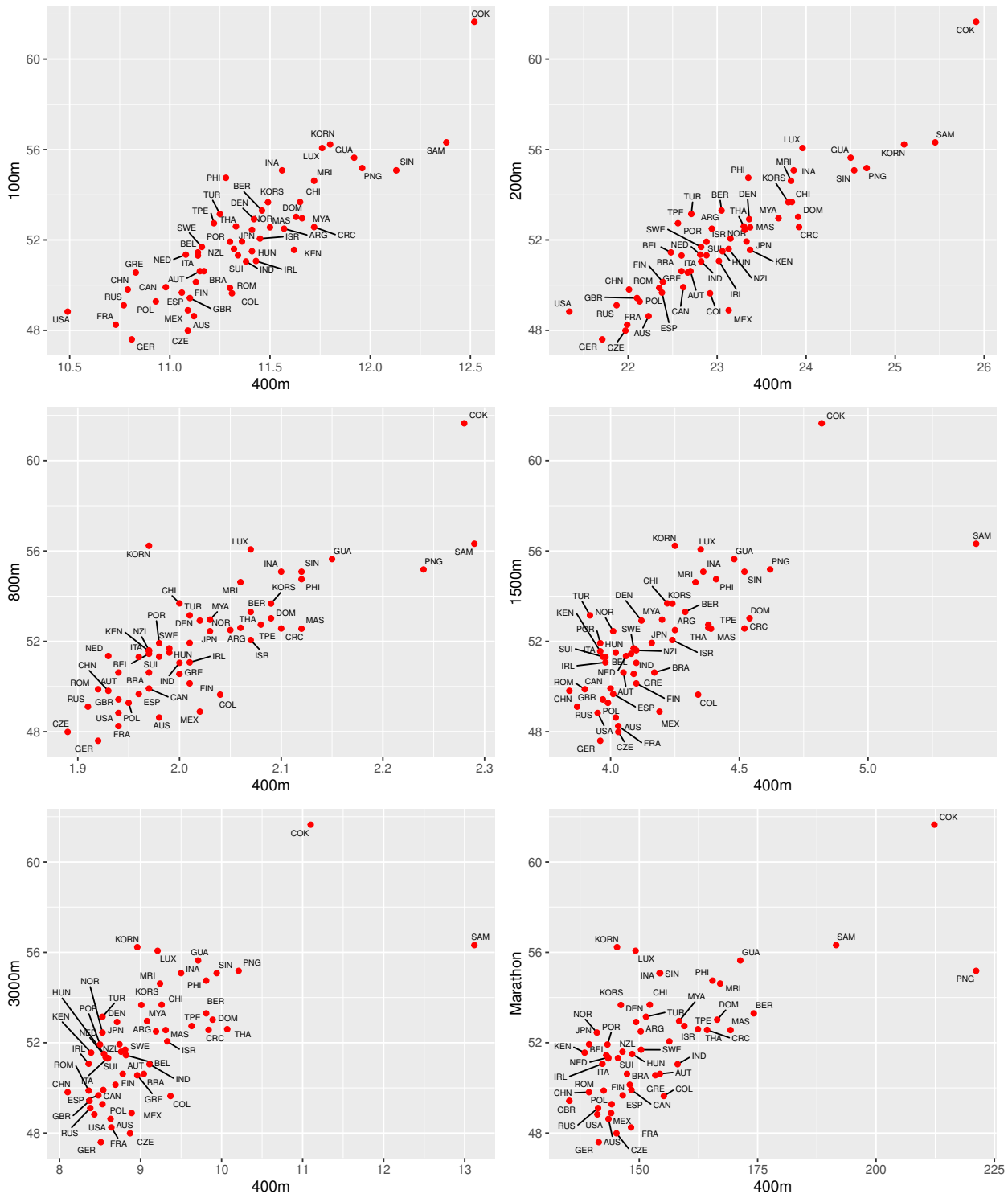

400m
200m
48
52
56
60
22
23
24
25
26
COK
SAM
KORN
PNG
SIN
GUA
CRC
DOM
CHI
INA
LUX
MYA
KORS
MRI
HUN
KEN
JPN
MAS
DEN
MEX
NZL
NOR
THA
PHI
IRL
ISR
COL
IND
SUI
BER
ARG
CAN
AUT
ITA
SWE
POR
ESP
GRE
BRA
NED
TPE
TUR
AUS
POL
FIN
BEL
ROM
FRA
GBR
CHN
CZE
RUS
GER
USA

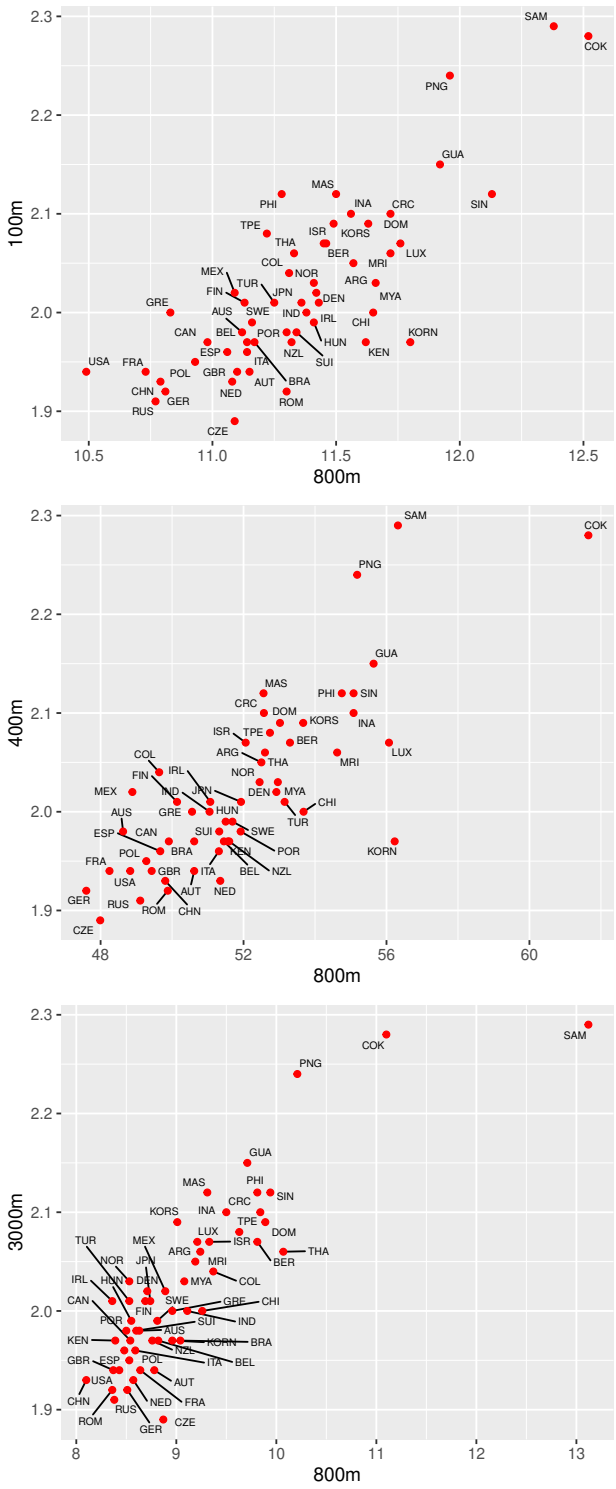
800m
200m
1.9
2.0
2.1
2.2
2.3
22
23
24
25
26
COK
SAM
KORN
SIN
PNG
GUA
LUX
DOM
CRC
CHI
MYA
MRI
KORS
INA
KEN
JPN
DEN
NOR
THA
MAS
PHI
POR
NZL
HUN
MEX
ISR
IRL
BER
NED
CAN
SUI
IND
COL
ARG
AUT
SWE
ITA
GRE
TUR
TPE
ROM
ESP
BRA
FIN
CZE
GBR
BEL
AUS
CHN
FRA
POL
RUS
GER
USA

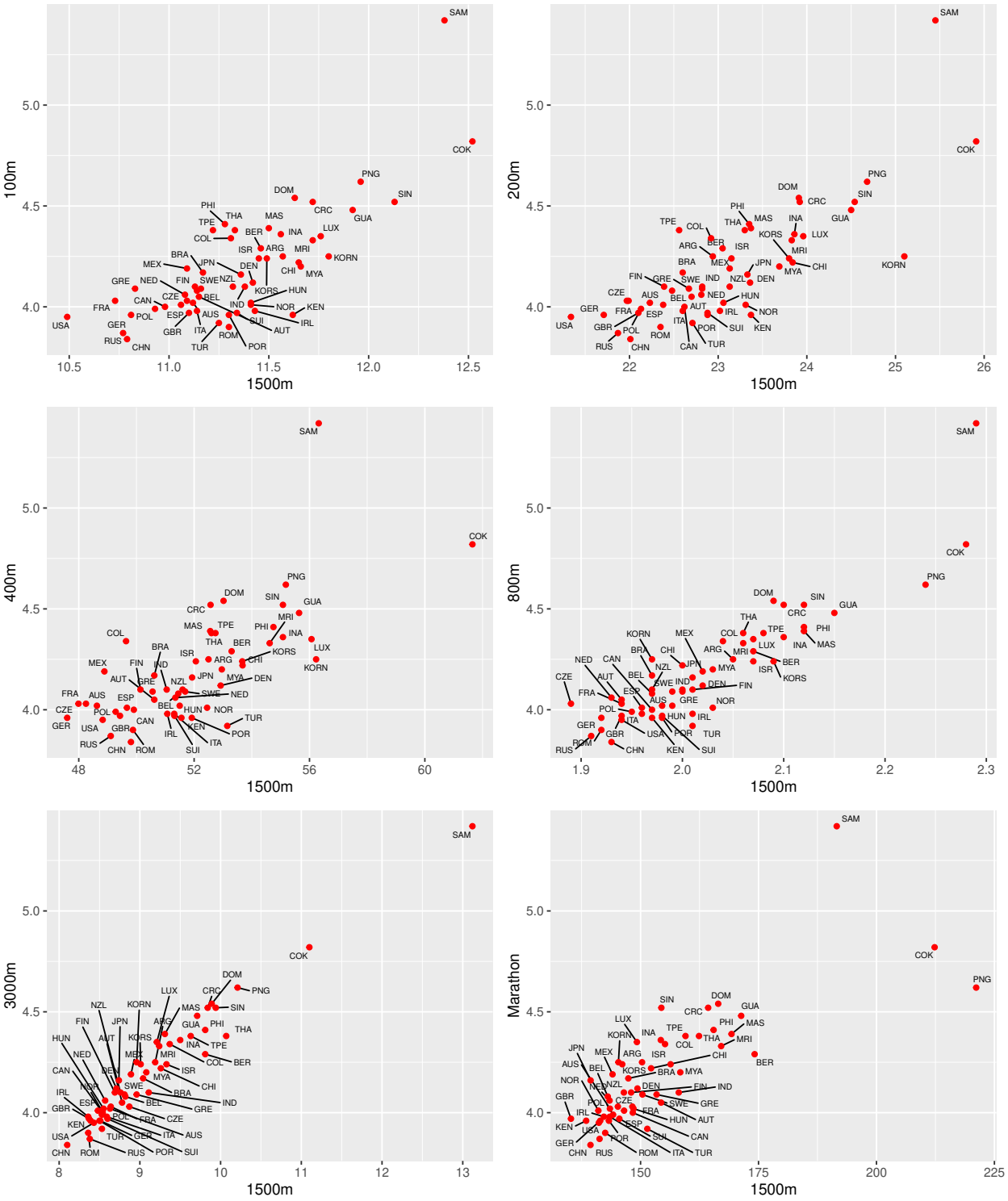
1500m
200m
4.0
4.5
5.0
22
23
24
25
26
COK
SAM
KORN
PNG
SIN
GUA
MYA
CHI
LUX
CRC
DOM
MRI
INA
KEN
DEN
KORS
IRL
NOR
JPN
MAS
PHI
ISR
THA
SUI
HUN
NZL
MEX
BER
POR
NED
SWE
ARG
COL
TUR
AUT
IND
BRA
ITA
BEL
GRE
TPE
ROM
CAN
AUS
FIN
CHN
ESP
FRA
GBR
POL
CZE
RUS
GER
USA

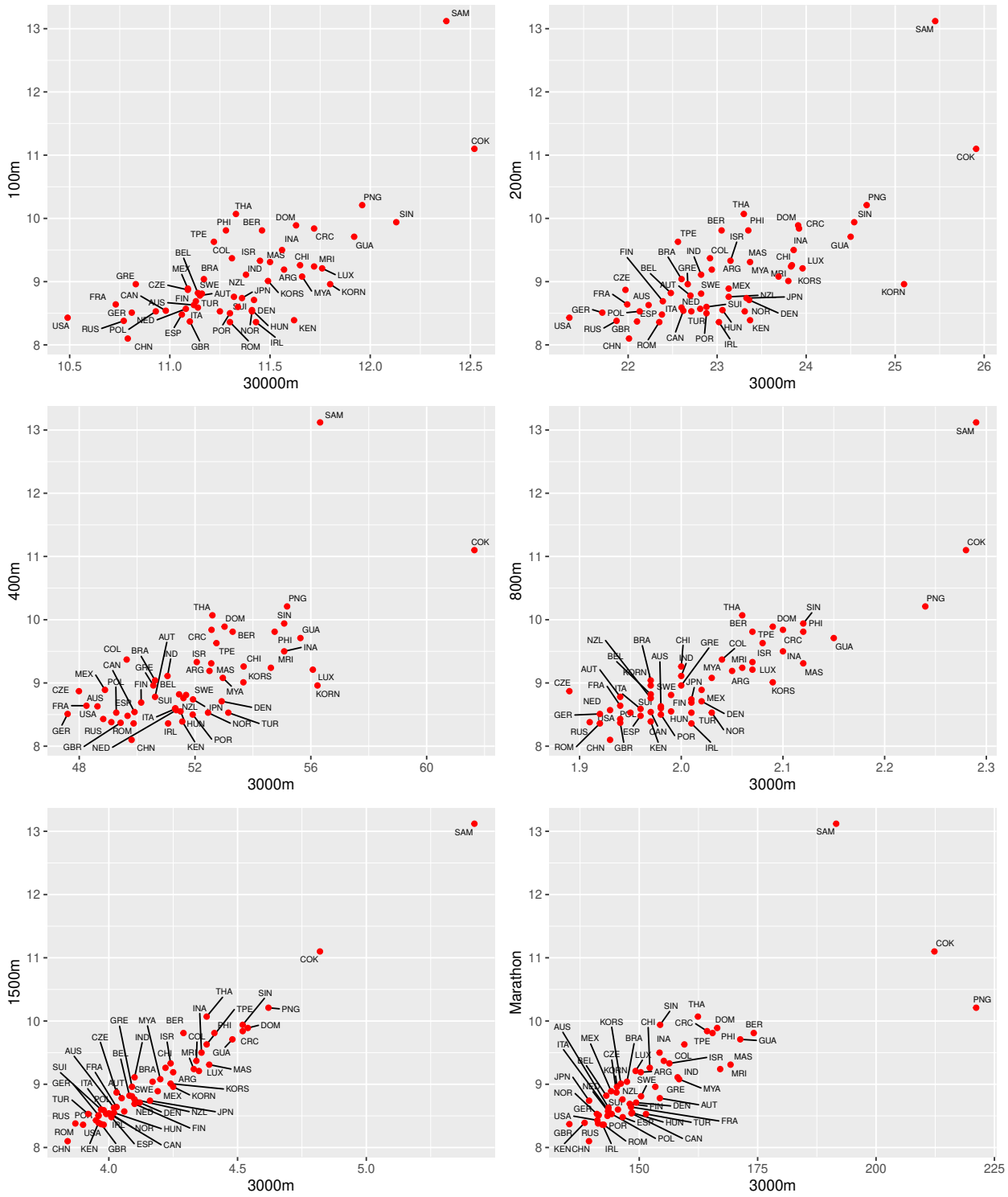
3000m
200m
8
9
10
11
12
13
22
23
24
25
26
COK
SAM
KORN
PNG
GUA
SIN
LUX
KORS
CHI
CRC
INA
DOM
MRI
KEN
DEN
MYA
IRL
NOR
JPN
MAS
PHI
THA
HUN
NZL
MEX
ISR
POR
SUI
SWE
ARG
BER
TUR
NED
GRE
IND
COL
CAN
ITA
BRA
AUT
TPE
ROM
ESP
FIN
BEL
CHN
AUS
GBR
POL
CZE
FRA
RUS
GER
USA

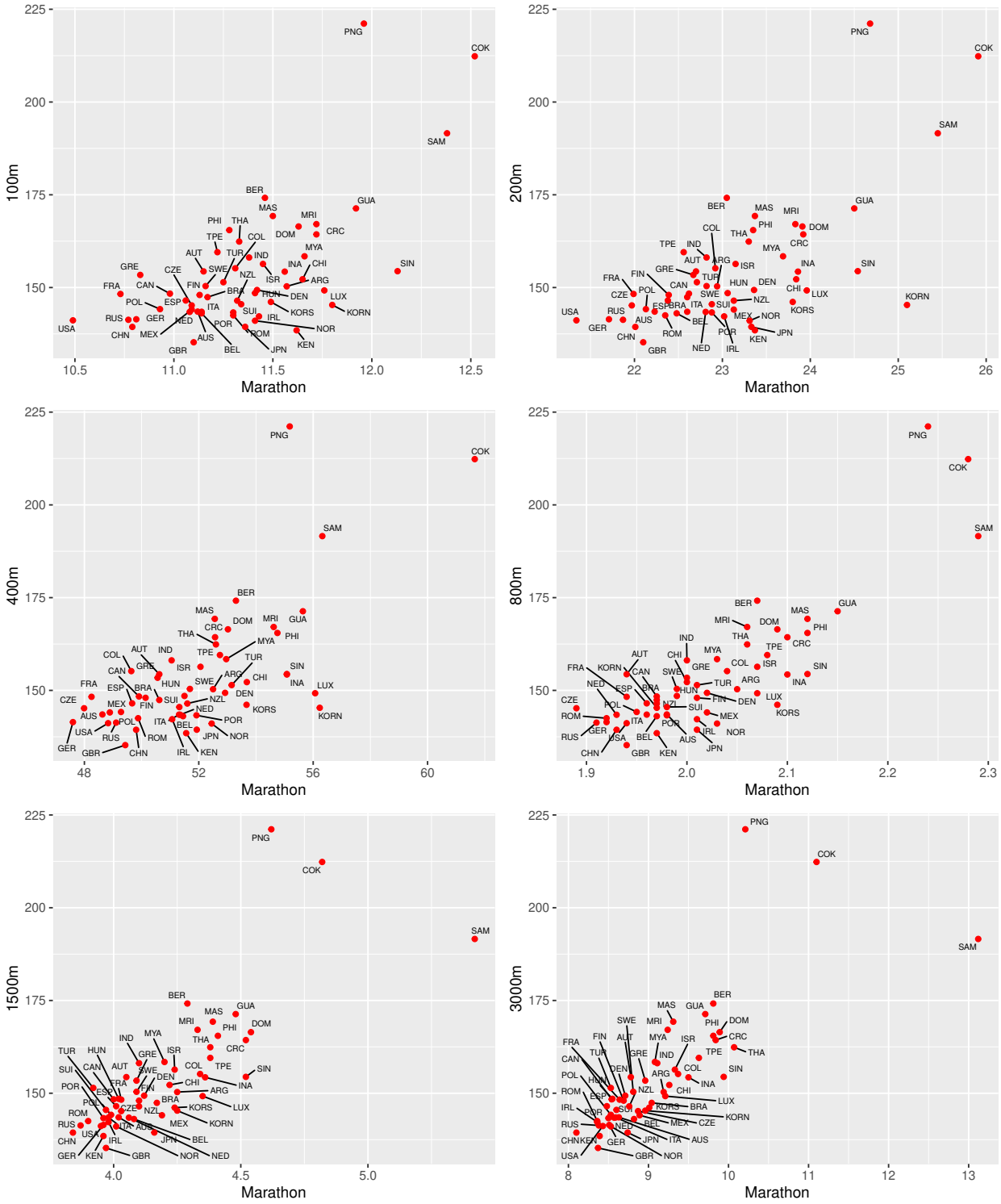
Marathon
200m
150
175
200
225
22
23
24
25
26
COK
SAM
KORN
PNG
SIN
GUA
LUX
INA
CRC
DOM
KEN
KORS
CHI
MRI
MYA
JPN
NOR
DEN
PHI
MAS
IRL
MEX
NZL
ISR
THA
POR
SUI
HUN
ARG
BER
NED
BEL
ITA
SWE
TUR
COL
CAN
AUT
IND
ROM
BRA
GRE
TPE
FIN
GBR
AUS
POL
ESP
CHN
CZE
FRA
RUS
GER
USA


100m
200m
800m
1500m
3000m
Marathon
400m
60
56
52
48
COK
SAM
KORN
GUA
LUX
INA
SIN
PNG
MRI
PHI
BER
KORS
CHI
TUR
DEN
TPE
THA
DOM
NOR
MAS
MYA
SWE
POR
JPN
ISR
ARG
CRC
BEL
NZL
HUN
KEN
NED
ITA
SUI
IND
IRL
GRE
CHN
AUT
BRA
ROM
CAN
FIN
RUS
ESP
COL
GBR
POL
MEX
USA
AUS
FRA
CZE
GER


100m
800m
200m
800m
400m
800m
1500m
800m
3000m
800m
Marathon
800m


100m
200m
400m
800m
3000m
Marathon
1500m
SAM
COK
PNG
SIN
DOM
CRC
GUA
PHI
THA
TPE
MAS
BER
INA
LUX
COL
ISR
ARG
MRI
KORN
BRA
JPN
DEN
CHI
MYA
MEX
NZL
KORS
HUN
GRE
NED
FIN
SWE
IND
CZE
BEL
NOR
KEN
CAN
AUS
IRL
FRA
ESP
SUI
AUT
POL
GER
GBR
ITA
ROM
USA
TUR
RUS
CHN
POR

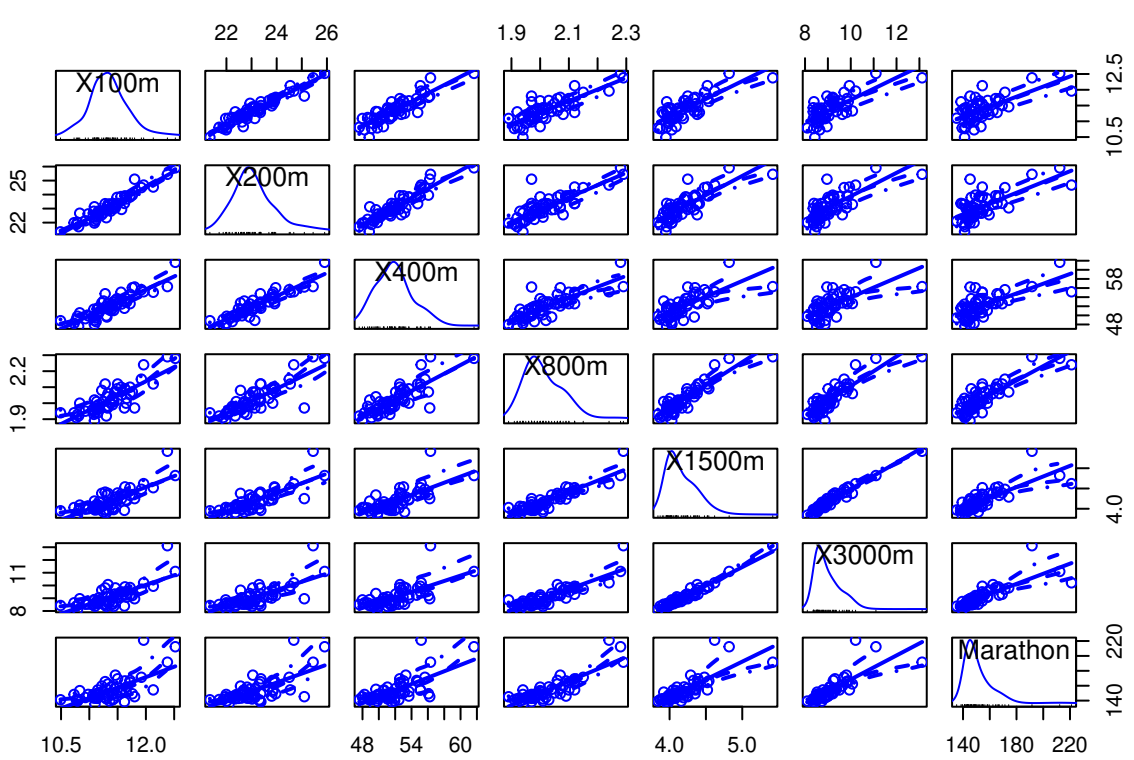

100m
30000m
200m
3000m
400m
3000m
800m
3000m
1500m
3000m
Marathon
3000m

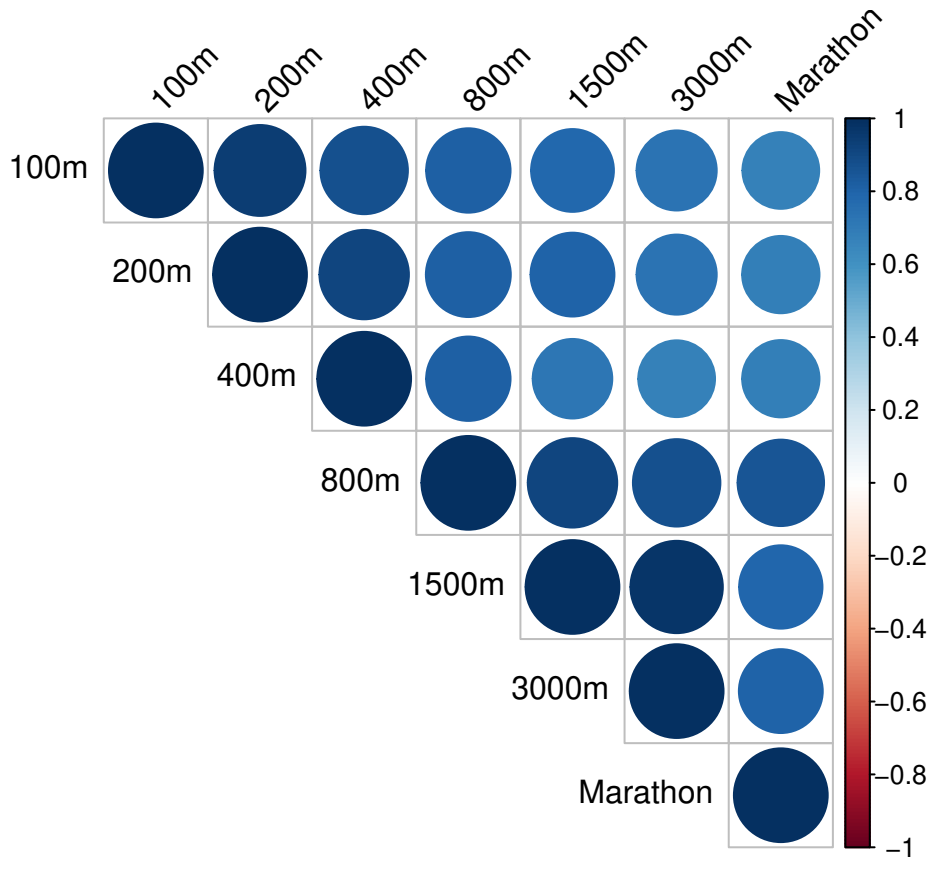


Analysis:

- Yes, especially when we compare marathon & 3000m races with the shorter races.

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.





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?

- Cook Islands(COK), Papua New Guinea(PNG) and Samoa(SAM) appear as the most extreme in almost all races, they are extreme because they acheived the worst records by a far margin from other countries.

One approach to measuring "extremism" is to look at the distance (needs to be deffined!) between an observation and the sample mean vector, i.e. we look how far one is from the average. Such a distance can be called an multivariate residual for the given observation.

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

$$d(\vec{x}, \hat{x}) = \sqrt{(\vec{x} - \hat{x})^T (\vec{x} - \hat{x})}$$

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

$$d_L(\vec{x}, \hat{x}) = (\sum_{i=1}^{p} |\vec{x} - \hat{x}|^r)^{1/r}$$

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 $\vec{x} - \bar{x}$ 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.

PNG: 67.63

COK: 59.62

SAM: 38.52

BER: 20.62

GBR: 18.59

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

$$d_V^2(\vec{x}, \hat{x}) = (\vec{x} - \hat{x})^T V^{-1} (\vec{x} - \hat{x})$$

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 (cen- tring) 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?

```
##   SAM   COK   PNG   USA   SIN
## 75.58 64.60 34.23 12.88 11.44
```

- This method does not treat the varaibales equally, thats why countries like USA and Singapore shown as extremes.

d) The most common statistical distance is the Mahalanobis distance

$$d_M^2(\vec{x}, \hat{x}) = (\vec{x} - \hat{x})^T C^{-1} (\vec{x} - \hat{x})$$

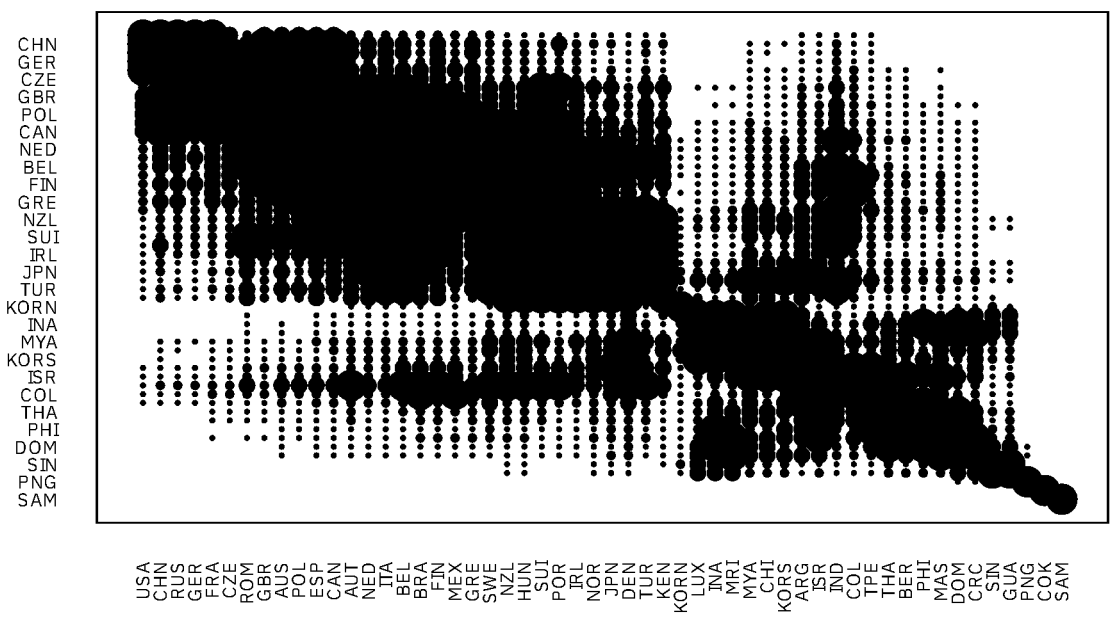
where C is the sample covariance matrix calculated from the data. With this measure we also use the relationships (covariances) between the variables (and not only the marginal variances as d_v (.,.) does). Compute the Mahalanobis distance, which countries are most extreme now?

```
##   SAM   PNG   COK  KORN   MEX
## 35.01 30.51 19.83 18.38 13.70
```

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 classiffied 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.

```
##             SWE Rank NA NA
## Euclidean    48    3  1
## Squared      50    3  1
## Mahalanobis  54    3  1
```

The second distance is sometimes called the "scaled Euclidean distance", not "squared distance".



References:

- https://www.gastonsanchez.com/visually-enforced/how-to/2014/01/15/Center-data-in-R/
- http://geog.uoregon.edu/bartlein/courses/geog495/lec18.html

 ,

Appendix

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

```r
X <- read.table("D:/Workshop/R/Multivariate Statistics/Data/T1-9.DAT")
colnames(X) <- c("Country", "100m", "200m", "400m", "800m", "1500m", "3000m", "Marathon")

Minimum               <- matrix(0,7,1)
colnames(Minimum)     <- c("Min.")

Median                <- matrix(0,7,1)
colnames(Median)      <- c("Median")

Mean                  <- matrix(0,7,1)
colnames(Mean)        <- c("Mean")

S_Deviation           <- matrix(0,7,1)
colnames(S_Deviation) <- c("SD")

Maximum               <- matrix(0,7,1)
colnames(Maximum)     <- c("Max.")


M <- function(X,var) {
  for(i in 1:var) {
    Minimum[i,]     <- min(X[,i+1])
    Median[i,]      <- median(X[,i+1])
    Mean[i,]        <- mean(X[,i+1])
    S_Deviation[i,] <- sd(X[,i+1])
    Maximum[i,]     <- max(X[,i+1])
  }
  result            <- round(cbind(Minimum, Median, Mean, S_Deviation, Maximum),2)
  rownames(result)  <- c("100m", "200m", "400m", "800m", "1500m", "3000m", "Marathon")
  return(result)
}
knitr::kable(M(X = X, var = 7))


pairs.panels(X)


ggplot(X, aes(x=X$`100m`)) +
  geom_histogram(aes(y=..density..), binwidth=.15,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Seconds") +
  ggtitle("Record Distribution of the Women's 100m Race") +
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))

ggplot(X, aes(x=X$`200m`)) +
  geom_histogram(aes(y=..density..), binwidth=.3,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Seconds") +
  ggtitle("Record Distribution of the Women's 200m Race") +
```

```
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))

ggplot(X, aes(x=X$`400m`)) +
  geom_histogram(aes(y=..density..), binwidth=.9,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Seconds") +
  ggtitle("Record Distribution of the Women's 400m Race") +
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))

ggplot(X, aes(x=X$`800m`)) +
  geom_histogram(aes(y=..density..), binwidth=.03,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Minutes") +
  ggtitle("Record Distribution of the Women's 800m Race") +
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))

ggplot(X, aes(x=X$`1500m`)) +
  geom_histogram(aes(y=..density..), binwidth=.12,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Minutes") +
  ggtitle("Record Distribution of the Women's 1500m Race") +
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))

ggplot(X, aes(x=X$`3000m`)) +
  geom_histogram(aes(y=..density..), binwidth=.4,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Minutes") +
  ggtitle("Record Distribution of the Women's 3000m Race") +
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))

ggplot(X, aes(x=Marathon)) +
  geom_histogram(aes(y=..density..), binwidth=6,
                 colour="black", fill="#00539CFF", alpha=0.3, size = 0.1) +
  geom_density(fill="#FFD662FF", color="#e9ecef", alpha=0.3) +
  xlab("Record in Minutes") +
  ggtitle("Record Distribution of the Women's Marathon Race") +
  theme(axis.text=element_text(size=9),
        axis.title=element_text(size=9),
        title = element_text(size=8))
```

```r
Corr = round(cor(X[,-1]),2)
cat("Correlation Matrix:\n")

knitr::kable(Corr)

cat("Covariance Matrix:\n")
Cov = round(cov(X[,-1]),2)
knitr::kable(Cov)
```

```r
library(ggrepel)
library("gridExtra")

p1 <- ggplot(data = as.data.frame(X), aes(y = X[,2], x = X[,3]))+
  xlab("100m") + ylab("200m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p2 <- ggplot(data = as.data.frame(X), aes(y = X[,2], x = X[,4]))+
  xlab("100m") + ylab("400m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p3 <- ggplot(data = as.data.frame(X), aes(y = X[,2], x = X[,5]))+
  xlab("100m") + ylab("800m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p4 <- ggplot(data = as.data.frame(X), aes(y = X[,2], x = X[,6]))+
  xlab("100m") + ylab("1500m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p5 <- ggplot(data = as.data.frame(X), aes(y = X[,2], x = X[,7]))+
  xlab("100m") + ylab("3000m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p6 <- ggplot(data = as.data.frame(X), aes(y = X[,2], x = X[,8]))+
  xlab("100m") + ylab("Marathon")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')
```

```r
grid.arrange(p1, p2, p3, p4, p5, p6)
```

```r
p7 <- ggplot(data = as.data.frame(X), aes(y = X[,3], x = X[,2]))+
  xlab("200m") + ylab("100m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')
```

```r
p8 <- ggplot(data = as.data.frame(X), aes(y = X[,3], x = X[,4]))+
  xlab("200m") + ylab("400m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p9 <- ggplot(data = as.data.frame(X), aes(y = X[,3], x = X[,5]))+
  xlab("200m") + ylab("800m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p10 <- ggplot(data = as.data.frame(X), aes(y = X[,3], x = X[,6]))+
  xlab("200m") + ylab("1500m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p11 <- ggplot(data = as.data.frame(X), aes(y = X[,3], x = X[,7]))+
  xlab("200m") + ylab("3000m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p12 <- ggplot(data = as.data.frame(X), aes(y = X[,3], x = X[,8]))+
  xlab("200m") + ylab("Marathon")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p13 <- ggplot(data = as.data.frame(X), aes(y = X[,4], x = X[,2]))+
  xlab("400m") + ylab("100m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p14 <- ggplot(data = as.data.frame(X), aes(y = X[,4], x = X[,3]))+
  xlab("400m") + ylab("200m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p15 <- ggplot(data = as.data.frame(X), aes(y = X[,4], x = X[,5]))+
  xlab("400m") + ylab("800m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p16 <- ggplot(data = as.data.frame(X), aes(y = X[,4], x = X[,6]))+
  xlab("400m") + ylab("1500m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p17 <- ggplot(data = as.data.frame(X), aes(y = X[,4], x = X[,7]))+
  xlab("400m") + ylab("3000m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p18 <- ggplot(data = as.data.frame(X), aes(y = X[,4], x = X[,8]))+
  xlab("400m") + ylab("Marathon")+
  geom_text_repel(label = X[,1], size = 2)+
```

```
  geom_point(color = 'red')

p19 <- ggplot(data = as.data.frame(X), aes(y = X[,5], x = X[,2]))+
  xlab("800m") + ylab("100m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p20 <- ggplot(data = as.data.frame(X), aes(y = X[,5], x = X[,3]))+
  xlab("800m") + ylab("200m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p21 <- ggplot(data = as.data.frame(X), aes(y = X[,5], x = X[,4]))+
  xlab("800m") + ylab("400m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p22 <- ggplot(data = as.data.frame(X), aes(y = X[,5], x = X[,6]))+
  xlab("800m") + ylab("1500m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p23 <- ggplot(data = as.data.frame(X), aes(y = X[,5], x = X[,7]))+
  xlab("800m") + ylab("3000m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p24 <- ggplot(data = as.data.frame(X), aes(y = X[,5], x = X[,8]))+
  xlab("800m") + ylab("Marathon")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p25 <- ggplot(data = as.data.frame(X), aes(y = X[,6], x = X[,2]))+
  xlab("1500m") + ylab("100m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p26 <- ggplot(data = as.data.frame(X), aes(y = X[,6], x = X[,3]))+
  xlab("1500m") + ylab("200m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p27 <- ggplot(data = as.data.frame(X), aes(y = X[,6], x = X[,4]))+
  xlab("1500m") + ylab("400m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p28 <- ggplot(data = as.data.frame(X), aes(y = X[,6], x = X[,5]))+
  xlab("1500m") + ylab("800m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p29 <- ggplot(data = as.data.frame(X), aes(y = X[,6], x = X[,7]))+
```

```
  xlab("1500m") + ylab("3000m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p30 <- ggplot(data = as.data.frame(X), aes(y = X[,6], x = X[,8]))+
  xlab("1500m") + ylab("Marathon")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p31 <- ggplot(data = as.data.frame(X), aes(y = X[,7], x = X[,2]))+
  xlab("30000m") + ylab("100m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p32 <- ggplot(data = as.data.frame(X), aes(y = X[,7], x = X[,3]))+
  xlab("3000m") + ylab("200m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p33 <- ggplot(data = as.data.frame(X), aes(y = X[,7], x = X[,4]))+
  xlab("3000m") + ylab("400m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p34 <- ggplot(data = as.data.frame(X), aes(y = X[,7], x = X[,5]))+
  xlab("3000m") + ylab("800m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p35 <- ggplot(data = as.data.frame(X), aes(y = X[,7], x = X[,6]))+
  xlab("3000m") + ylab("1500m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p36 <- ggplot(data = as.data.frame(X), aes(y = X[,7], x = X[,8]))+
  xlab("3000m") + ylab("Marathon")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p37 <- ggplot(data = as.data.frame(X), aes(y = X[,8], x = X[,2]))+
  xlab("Marathon") + ylab("100m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p38 <- ggplot(data = as.data.frame(X), aes(y = X[,8], x = X[,3]))+
  xlab("Marathon") + ylab("200m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p39 <- ggplot(data = as.data.frame(X), aes(y = X[,8], x = X[,4]))+
  xlab("Marathon") + ylab("400m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')
```

```r
p40 <- ggplot(data = as.data.frame(X), aes(y = X[,8], x = X[,5]))+
  xlab("Marathon") + ylab("800m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')

p41 <- ggplot(data = as.data.frame(X), aes(y = X[,8], x = X[,6]))+
  xlab("Marathon") + ylab("1500m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')
```

```r
p42 <- ggplot(data = as.data.frame(X), aes(y = X[,8], x = X[,7]))+
  xlab("Marathon") + ylab("3000m")+
  geom_text_repel(label = X[,1], size = 2)+
  geom_point(color = 'red')


grid.arrange(p7, p8, p9, p10, p11, p12)
grid.arrange(p13, p14, p15, p16, p17, p18)
grid.arrange(p19, p20, p21, p22, p23, p24)
grid.arrange(p25, p26, p27, p28, p29, p30)
grid.arrange(p31, p32, p33, p34, p35, p36)
grid.arrange(p37, p38, p39, p40, p41, p42)

library(car)

par(mar= c(1,1,1,1))

scatterplotMatrix(X[,2:8])

library(corrplot)



corrplot(Corr, type = "upper",
        tl.col = "black", tl.srt = 45)


centered           <- apply(X[,-1], 2, function(x){abs(x-mean(x))})
rownames(centered) <- X[[1]]
mat                <- sqrt(tcrossprod(centered))
dist1               <- sort(diag(mat),decreasing = TRUE)
top_5              <- round(dist1[1:5],2)
# for some reason sqrt function didnt work in rmarkdown
# so we had to copy the values from r console

var                <- matrix(0,7,7)
diag(var)          <- diag(cov(X[,-1]))
sqrd_dist          <- ((centered) %*% solve(var)) %*% t(centered)
dist2               <- (sort(diag(sqrd_dist),decreasing = TRUE))
top_5sq            <- round(dist2[1:5],2)
top_5sq
```

```r
C                 <- cov(X[,-1])
mah               <- ((centered) %*% solve(C)) %*% t(centered)
dist3             <- (sort(diag(mah),decreasing = TRUE))
top_5mh           <- round(dist3[1:5],2)
top_5mh

Sweden1 <- which(row.names(as.data.frame(dist1)) =='SWE')
Sweden2 <- which(row.names(as.data.frame(dist2)) =='SWE')
Sweden3 <- which(row.names(as.data.frame(dist3)) =='SWE')

swe             <- data.frame(c(Sweden1, Sweden2, Sweden3), 3,1)
row.names(swe) <- c("Euclidean", "Squared", "Mahalanobis")
colnames(swe)  <- c("SWE Rank")
swe

library("RMaCzek")

rownames(X) <- X[[1]]
czek        <- czek_matrix(X[,-1])
plot(czek)
```