Multivariate Statistical Methods

Assignment 1. Examining Multivariate Data

Ahmet Hakan Akdeve(ahmak554), Jooyoung Lee(joole336), Weng Hang Wong(wonwo535), Zhixuan Duan(zhidu838)

2019 11 23

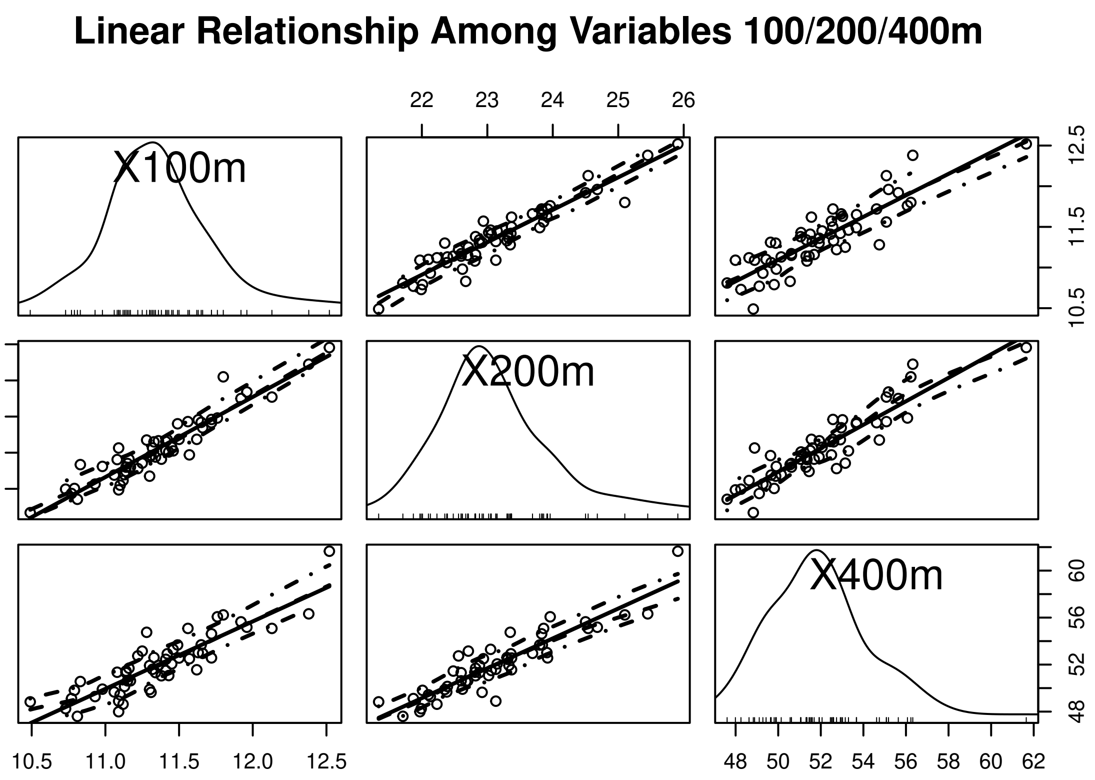
Question 1.

a) Means and standard deviations of 100m, 200m, 400m, 800m, 1500m, 3000m, and marathon records are calculated as below.

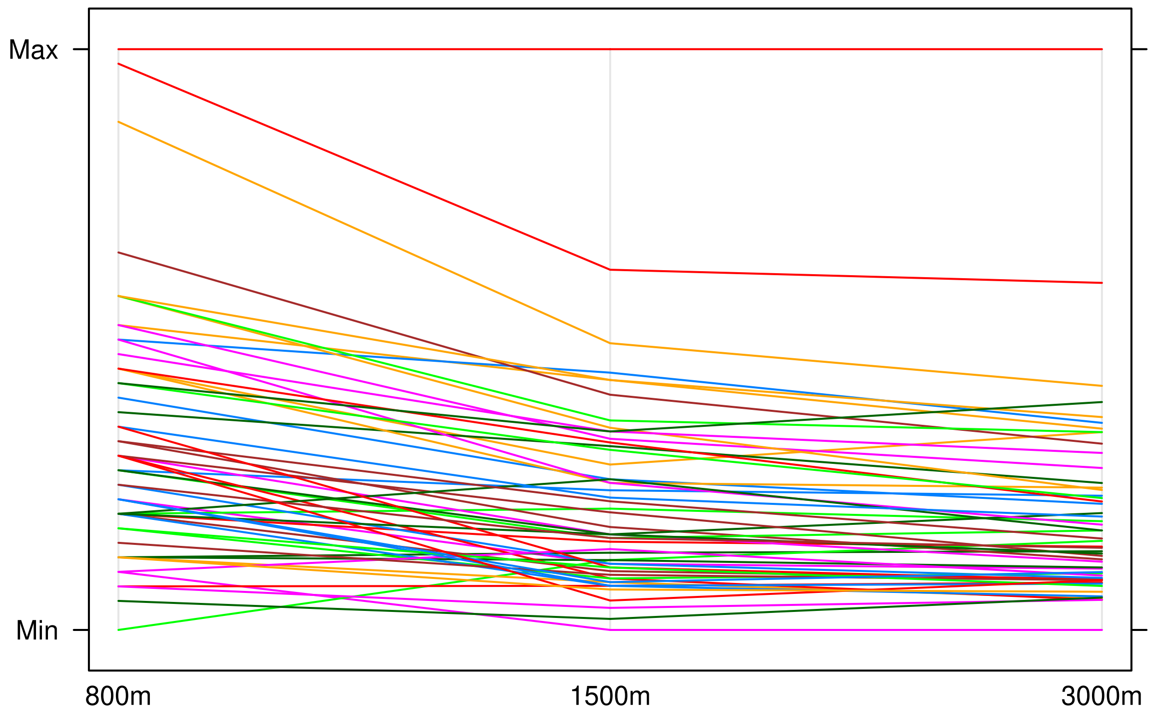
```
##   variable       mean     std_dev
## 1     100m  11.357778  0.39410116
## 2     200m  23.118519  0.92902547
## 3     400m  51.989074  2.59720188
## 4     800m   2.022407  0.08687304
## 5    1500m   4.189444  0.27236502
## 6    3000m   9.080741  0.81532689
## 7 Marathon 153.619259 16.43989508
```

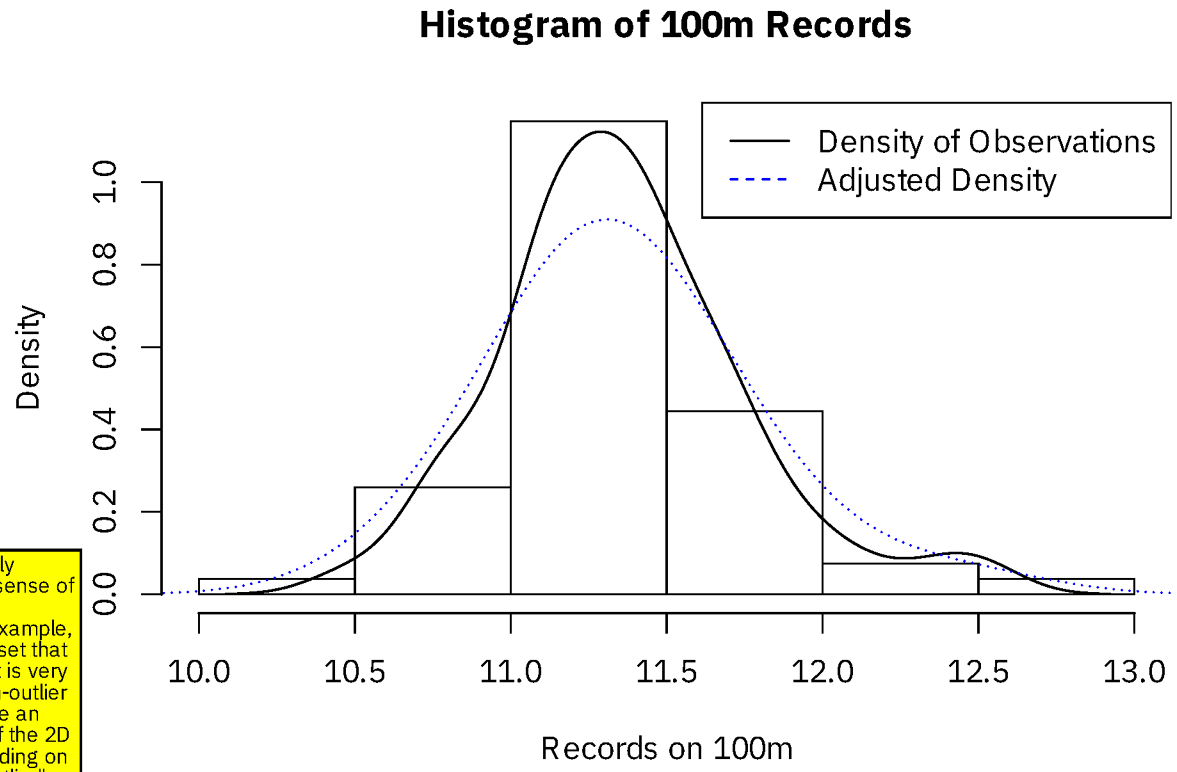
b) Followings are plotted to check if extreme values exist, and the observations are normally distributed.

```
## Loading required package: carData
```


Linear Relationship Among Variables 100/200/400m
X100m
X200m
X400m
22
23
24
25
26
12.5
11.5
10.5
26
24
22
60
56
52
48
10.5
11.0
11.5
12.0
12.5
48
50
52
54
56
58
60
62

Participating Countries' Rerods in 800/1500/3000m





The scatter plot only shows outlier in a sense of some marginal distributions. For example, if you have a data set that has 50 variables, it is very possible that a non-outlier data point looks like an outlier in a some of the 2D scatter plot, depending on how you define "outlier".

First two plots shows linear relationships among variables and observations at different variables respectively. The first plot shows outliers with empty dots. Also, it is possible to see two extreme values, which are red and orange colored lines, on the second plot.

Third plot is histogram of observations on variable 100m. The fitted line showing approximate density of the data is similar to blue dotted line, which represents gaussian distribution. It is plausible to say this data is normally distributed.

Question 2.

a) Covariance and correlation matrices are followings:

```
## [1] "Covariance Matrix: "
```

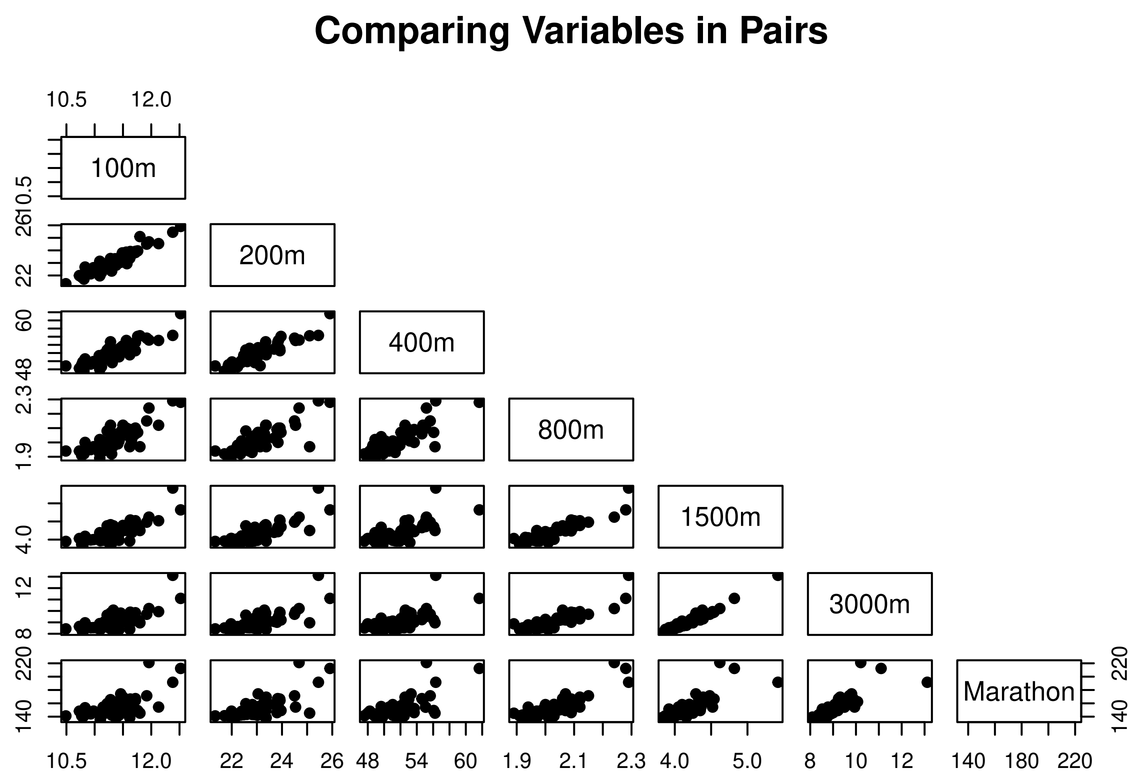
```
##           100m   200m   400m  800m 1500m  3000m Marathon
## 100m     0.155  0.345  0.891 0.028 0.084  0.234    4.334
## 200m     0.345  0.863  2.193 0.066 0.203  0.554   10.385
## 400m     0.891  2.193  6.745 0.182 0.509  1.427   28.904
## 800m     0.028  0.066  0.182 0.008 0.021  0.061    1.220
## 1500m    0.084  0.203  0.509 0.021 0.074  0.216    3.540
## 3000m    0.234  0.554  1.427 0.061 0.216  0.665   10.706
## Marathon 4.334 10.385 28.904 1.220 3.540 10.706  270.270
```

```
## [1] "Coorrelation Matrix: "
```

```
##          100m  200m  400m  800m 1500m 3000m Marathon
## 100m     1.000 0.941 0.871 0.809 0.782 0.728    0.669
## 200m     0.941 1.000 0.909 0.820 0.801 0.732    0.680
## 400m     0.871 0.909 1.000 0.806 0.720 0.674    0.677
## 800m     0.809 0.820 0.806 1.000 0.905 0.867    0.854
## 1500m    0.782 0.801 0.720 0.905 1.000 0.973    0.791
## 3000m    0.728 0.732 0.674 0.867 0.973 1.000    0.799
## Marathon 0.669 0.680 0.677 0.854 0.791 0.799    1.000
```

The elements in the matrices are rounded to three decimal places. It is possible to observe that both covariance and correlation matrices are symmetric matrices.

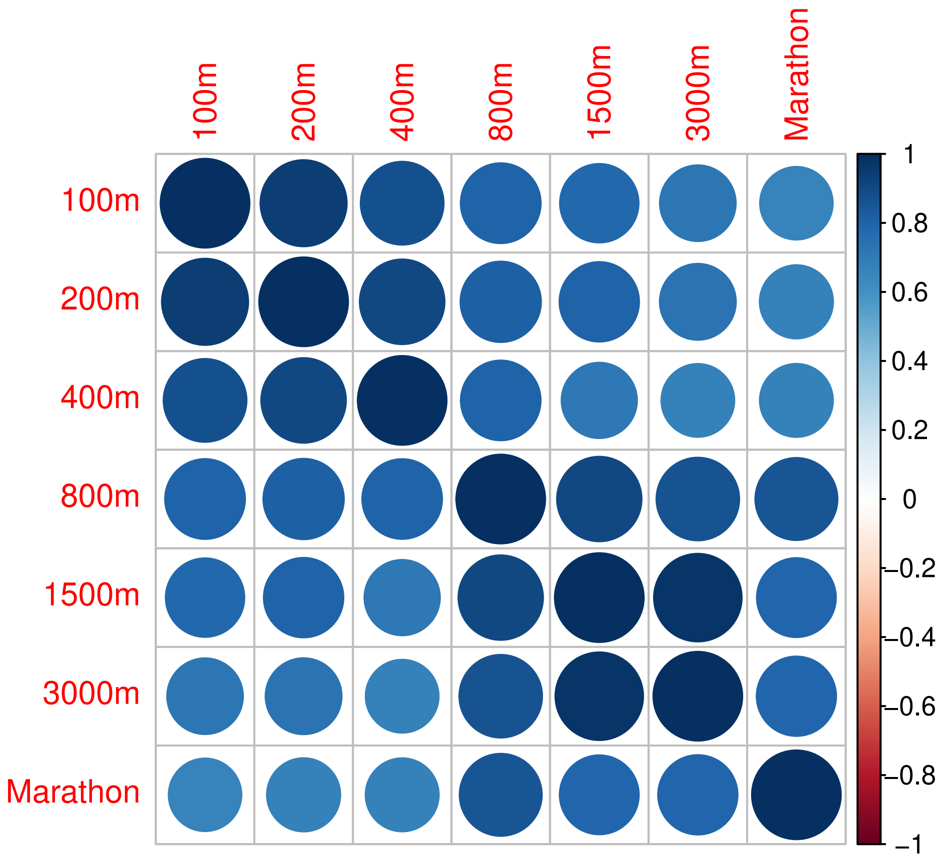
b) Extreme values exist in every scatterplot.



c) Chernoff face diagram and pairwise correlation plot can be used to describe multivariate data.


```
## effect of variables:
##  modified item       Var
##  "height of face   " "100m"
##  "width of face    " "200m"
##  "structure of face" "400m"
##  "height of mouth  " "800m"
##  "width of mouth   " "1500m"
##  "smiling          " "3000m"
##  "height of eyes   " "Marathon"
##  "width of eyes    " "100m"
##  "height of hair   " "200m"
##  "width of hair   "  "400m"
##  "style of hair   "  "800m"
##  "height of nose  "  "1500m"
##  "width of nose   "  "3000m"
##  "width of ear    "  "Marathon"
##  "height of ear   "  "100m"
```

```
## corrplot 0.84 loaded
```



In Chernoff face diagram, COK and SAM are clearly deviated from the other ones. The Size of faces and hair design are different from most of the other countries. Pairwise correlation plot shows all variables have positive correlation to the other variables. However in most cases, correlation coefficient decreases as the difference in race length increases.

Question 3.

a) Mean-corrected data is used to produce the following plots.

```
## NULL
```

```
## NULL
```

```
## NULL
```

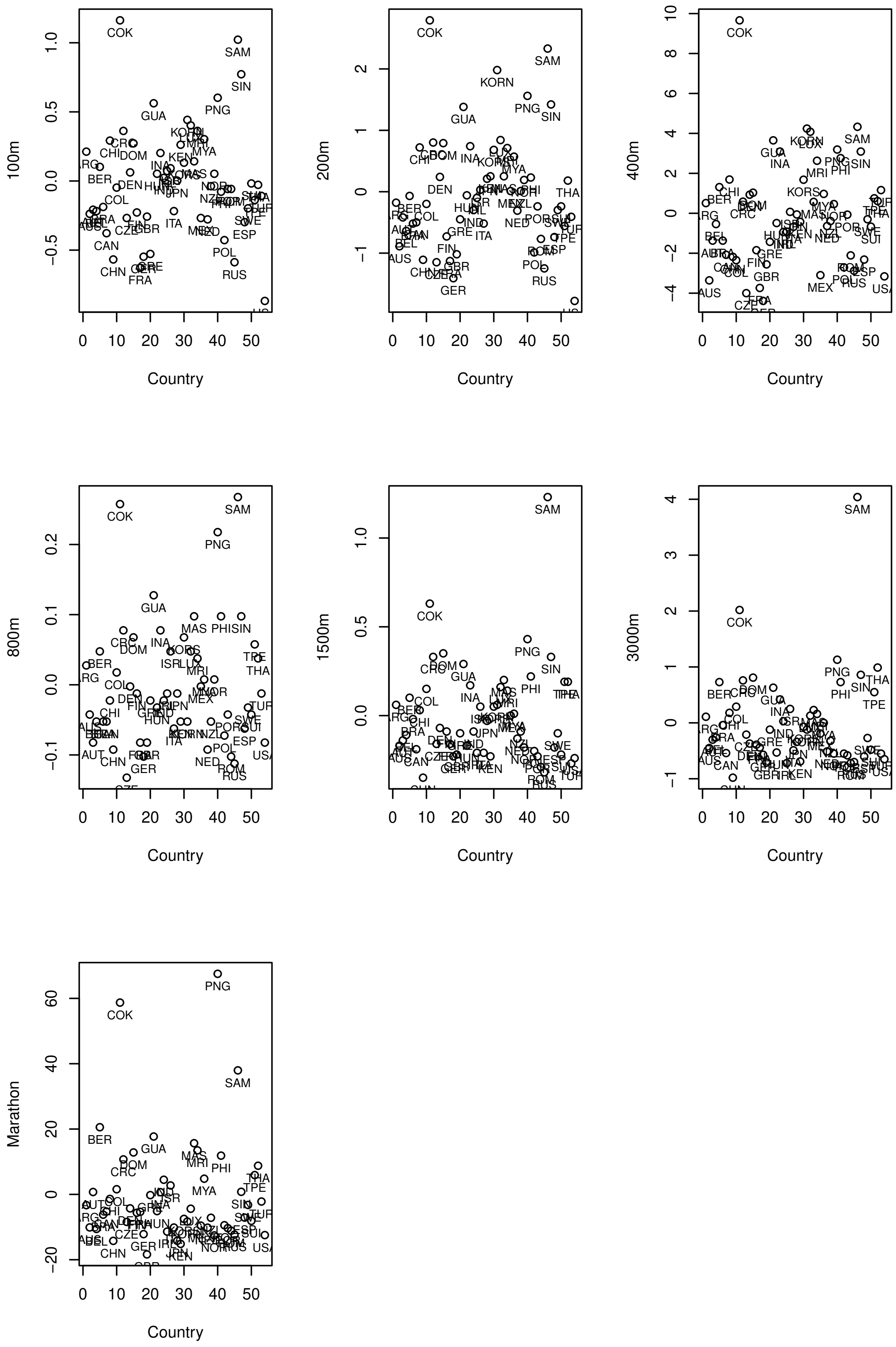
```
## NULL
```

```
## NULL
```

```
## NULL
```

```
## NULL
```

You can use par(mar=...) to reduce the space between the plots.



Extreme observations are defined to be the ones that are deviated a lot from 0, so that the actual value is greatly different from the mean value of the variable. In general, COK, SAM and PNG are likely to be considered as extreme countreis.

b) By using squared Euclidean distance, following countries are turend out to be the most extreme countreis; they have the longest distance.

```
##      PNG      COK      SAM      BER      GBR
## 67.62796 59.61517 38.52476 20.61606 18.59146
```

c) According to the scaled independent distance, countries below are the most extreme countreis.

```
##      SAM      COK      PNG      USA      SIN
## 75.58280 64.60116 34.22891 12.87689 11.44486
```

The last two countries are different from unnormalized analysis. Before the distance is normalized, the result was easily dominated by few variables. However, by using inverse diagonal matrix of variance, it is possible to get the extreme values based on the same scale.

d) The solution can be given by using the in-built R-function "mahalanobis". However, matrix multiplication is utilized to obtain the following result:

```
##    Country d2values
## 46     SAM 35.01406
## 40     PNG 30.50725
## 31    KORN 26.16714
## 11     COK 19.83400
## 35     MEX 14.23093
```

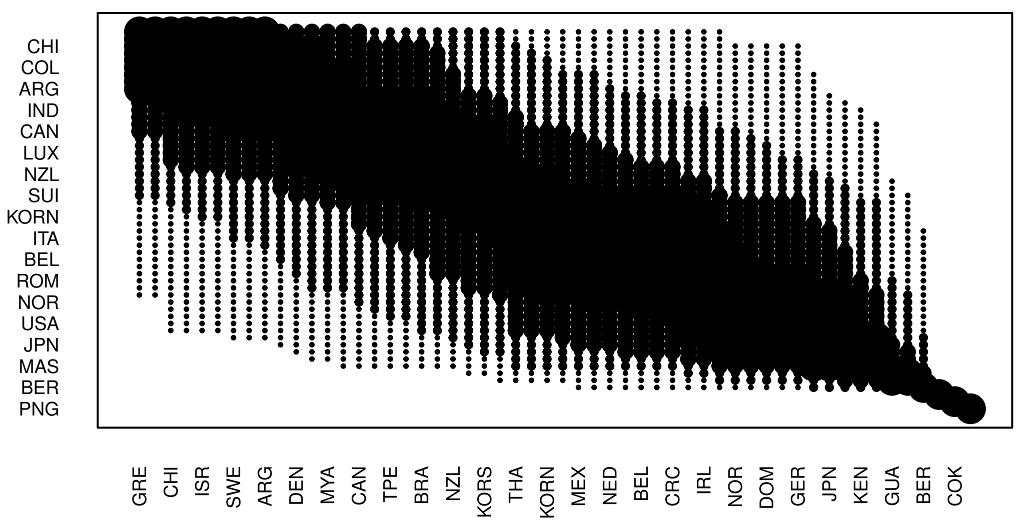
e)

Each method used in b-d always includes SAM, PNG and COK as three of five observations that are most deviated from the mean. However, the other two are different for each method. Distances are measured as a deviation from the mean, as a scale-independent deviation from the mean, and as a deviation from the mean that are scale-independent but relationship among variables are considered. In all methods, SAM, PNG, and COK are considered as extreme observations; it will be safe to consider them as extreme outlier countries.

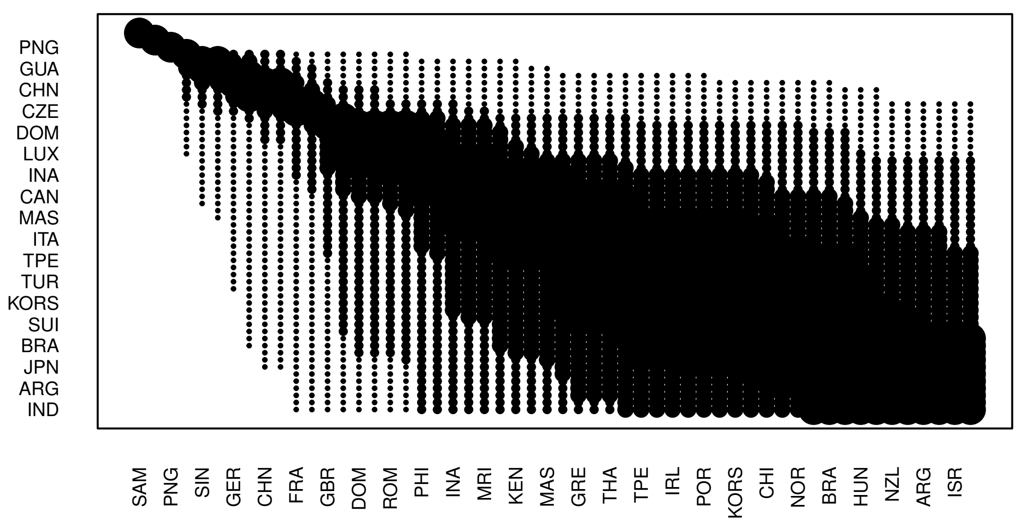
The following plot shows the distance when each method is utilized for detecting extremes. Except COK, PNG and SAM, all other countries are defined to be extreme only once. In case of Sweden, Euclidean distance is not appeared on the group because it is almost the same as mahalabonis distance that the red point is hidden behind the green point.

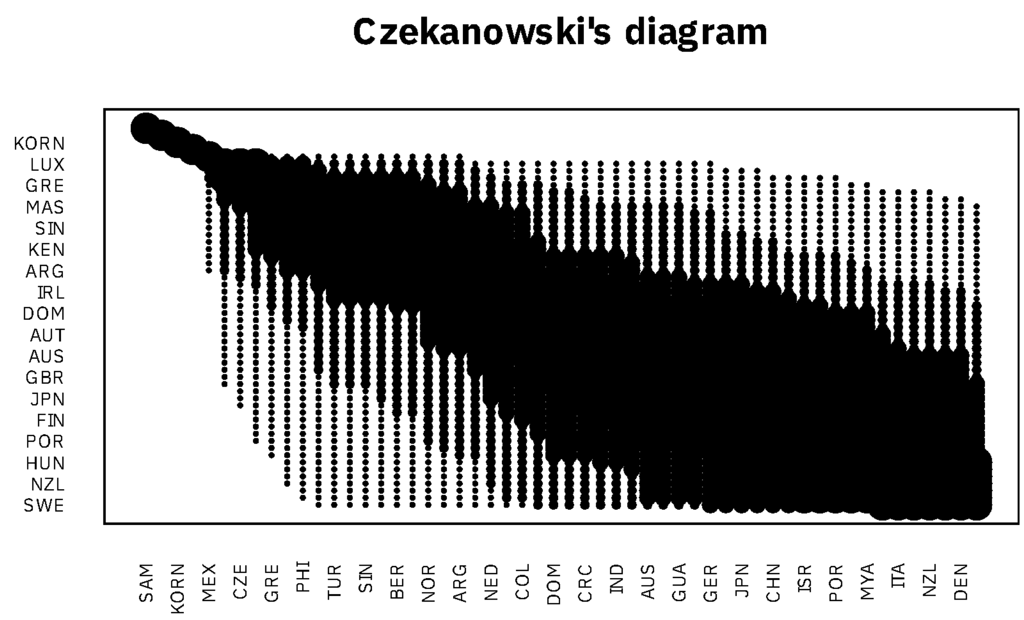
```
## Registered S3 method overwritten by 'seriation':
##   method         from
##   reorder.hclust gclus
```

Czekanowski's diagram



Czekanowski's diagram





Above plots are Czekanowski's diagram, which shows the similarity between each variables. Bigger dot placed between two countries means there is a big similarity between them. Each diagram represents the result based on Euclidean distance, scale-independent distance and mahalabonis distance respectively.

For each method, extreme values tend to not have dot as a relationship between all another countries. But it cannot be said that they do not have any similarity with other countries, because first of all, the size of the dots are classified by the distances. There could be weak similarity between certain countries, but such similarity is so weak that it is classified to have no dot between them on the diagram. Also, not all countries are appeared in axis.

Notice the definition of metric guarantees that d(x,y)=0 iff x=y. So it is mathematically trivial that the distance of any of those points are non-zero.

Appendix

```
#importing data
data <- read.delim(file="C:/Users/Young/Documents/Multivariate/732A97_HT2019_Materials/T1-9.dat", sep="`
colnames(data)<-c("Country","100m","200m","400m","800m","1500m","3000m","Marathon")
##Q1. a
charac <- data.frame(variable = as.numeric(), mean=as.numeric(), std_dev=as.numeric(), stringsAsFactors
for (i in 2:length(data)) {
  mean_v <- mean(data[[i]])
  std_dev_v <- sd(data[[i]])
  charac <- rbind(charac, data.frame(variable = colnames(data)[i], mean=mean_v, std_dev=std_dev_v))
}
##Q1. b
## scatter_plot
#install.packages("car")
library(car)
scatterplotMatrix(data[,2:4], col = "black", main="Linear Relationship Among Variables 100/200/400m")
```

```r
## profile_plot
library(lattice)
parallelplot(~data[5:7], horizontal.axis = FALSE, main="Participating Countries' Rerods in 800/1500/3000
## hist graph and density fitting line
y <- data$"100m"
hist(y, probability = TRUE, main="Histogram of 100m Records", xlab="Records on 100m")
lines(density(y))
lines(density(y, adjust = 2), lty = "dotted", col="blue")
legend(x="topright", legend=c("Density of Observations", "Adjusted Density"), col=c("black", "blue"), l
##Q2. a
data_mat <- as.matrix(data[,-1])
for (i in 1:7) {
  colnames(data_mat)[i] <- paste("V", i, sep='')
}
mean_vect <- as.vector(charac[,2])
mean_mat <- matrix(0, ncol=7, nrow=nrow(data_mat))
for (i in 1:nrow(data_mat)) {
  mean_mat[i,] <- mean_vect
}
mean_correct_mat <- data_mat - mean_mat
cov_mat<-cov(data[,-1])
cor_mat<-cor(data[,-1])
print('Covariance Matrix: ')
round(cov_mat, digits = 3)
print("Coorrelation Matrix: ")
round(cor_mat, digits = 3)
##Q2. b
pairs(data[,-1],pch=19,upper.panel = NULL, main = "Comparing Variables in Pairs")
##Q2. c
#install.packages("aplpack")
library(aplpack)
faces(data[,-1],labels = data[,1])
#Pairwise correlation between the variables. Plot.
#install.packages("corrplot")
library(corrplot)
corrplot(cor_mat)
##Q3. a
mean_correct_mat2<-as.data.frame(mean_correct_mat)
mean_correct_mat2$country<-data$Country
rownames(mean_correct_mat2)<-data[,1]
colnames(mean_correct_mat2)<-c(colnames(data[,-1]),"country")
par(mfrow=c(3,3))
for(i in 1:ncol(mean_correct_mat2[,-8])){
 a<-plot(x=1:nrow(mean_correct_mat2),y=mean_correct_mat2[,i],
         xlab = "Country",ylab=colnames(mean_correct_mat2)[i])
  text(1:nrow(mean_correct_mat2), mean_correct_mat2[,i], labels=mean_correct_mat2$country, cex= 0.7, po
  print(a)
}
##Q3. b
dis <- (diag(mean_correct_mat %*% t(mean_correct_mat))^(1/2))
Eulidean_dis <- matrix(dis)
row.names(Eulidean_dis) <- data[,1]
extreme_dis <- Eulidean_dis[order(Eulidean_dis, decreasing = TRUE)[1:5],]
```

```r
print(extreme_dis)
eulidean_extreme<-sort(dis, decreasing = TRUE)[1:5]
##Q3. c
v_1 <- as.vector(diag(cov_mat))
v_1 <- diag(v_1)
v_1 <- solve(v_1)
scale_dist <- matrix(0, nrow=nrow(mean_correct_mat))
rownames(scale_dist) <- country
colnames(scale_dist) <- "Scaled Distance"
for (i in 1:nrow(mean_correct_mat)) {
  scale_dist[i] <- t(as.matrix(mean_correct_mat[i,])) %*% v_1 %*% (as.matrix(mean_correct_mat[i,]))
}
scale_extreme <- scale_dist[order(scale_dist, decreasing = TRUE)[1:5],]
##Q3. d
d2values<-vector()
for(i in 1:nrow(data)){
  d2values[i]<-t(mean_correct_mat[i,])%*%solve(cov_mat)%*%mean_correct_mat[i,]
}
d2frame<-data.frame("Country"=data$Country,d2values)
d2frame<-d2frame[order(d2values,decreasing = TRUE),]
head(d2frame)[1:5,]
mat_2 <-  matrix(d2values)
row.names(mat_2) <- data[,1]
d2_extreme <- mat_2[order(mat_2, decreasing = TRUE)[1:5],]
##Q3. e
swe_frame<-data.frame(rep("SWE",3),c(Eulidean_dis["SWE",][[1]],scale_dist["SWE",],d2frame["49",2]),
                      c("sq_dist","extreme_dis","mahal_dis"))
colnames(swe_frame)<-colnames(distance_frame)
sq_dist_frame<-data.frame("Country"=names(sq_dist),"distance"=sq_dist);rownames(sq_dist_frame)<-1:nrow(:
extreme_dis_frame<-data.frame("Country"=names(extreme_dis),"distance"=extreme_dis);rownames(extreme_dis_
mah_frame<-as.data.frame(d2frame[1:5,]);rownames(mah_frame)<-1:5;colnames(mah_frame)<-c("Country","dist
dist_names<-c(rep("sq_dist",5),rep("extreme_dis",5),rep("mahal_dis",5))
distance_frame<-data.frame(rbind(sq_dist_frame,extreme_dis_frame,mah_frame),dist_names)
distance_frame<-rbind(distance_frame,swe_frame)
ggplot(distance_frame,aes(x=Country,y=distance,color=dist_names))+geom_point(size=3)+theme_bw()+
  theme(legend.title = element_blank())+ylab("Distance")
library(RMaCzek)
#euli_dis_czek
euli_dist <- czek_matrix(Eulidean_dis)
plot.czek_matrix(euli_dist)
#scale_dis_czek
sca_dist <- czek_matrix(scale_dist)
plot.czek_matrix(sca_dist)
#d2_dis_czek
d2_dis <- czek_matrix(mat_2)
plot.czek_matrix(d2_dis)
```