DS5105 - Multivariate Methods I - PCA Worksheet
MSc in DS&AI

Sachith M Gunawardane
DTS 2113

11 June, 2023

Load Libraries

```
library(FactoMineR)
library(factoextra)
```

1. Load the data set into R.

```
dataset <- read.table(paste0("D:/PGIS_Data_Science_AI/DS5110_Statistical_simulation",
                             "/projects/PCA_Exercise/track_rec.csv"), header = T, sep =",")
head(dataset)
```

```
##        C1.T X100m X200m X400m X800m X1500m X5000m X10000m Marathon
## 1 Argentina 10.23 20.37 46.18  1.77   3.68  13.33   27.65   129.57
## 2 Australia  9.93 20.06 44.38  1.74   3.53  12.93   27.53   127.51
## 3   Austria 10.15 20.45 45.80  1.77   3.58  13.26   27.72   132.22
## 4   Belgium 10.14 20.19 45.02  1.73   3.57  12.83   26.87   127.20
## 5   Bermuda 10.27 20.30 45.26  1.79   3.70  14.64   30.49   146.37
## 6    Brazil 10.00 19.89 44.29  1.70   3.57  13.48   28.13   126.05
```

2. Obtain the sample correlation matrix for the data.

```
dim(dataset)
```

```
## [1] 54  9
```

```
dataset1 <- dataset[,-1]
cor(dataset1)
```

```
##              X100m     X200m     X400m     X800m    X1500m    X5000m   X10000m
## X100m    1.0000000 0.9147554 0.8041147 0.7119388 0.7657919 0.7398803 0.7147921
## X200m    0.9147554 1.0000000 0.8449159 0.7969162 0.7950871 0.7613028 0.7479519
## X400m    0.8041147 0.8449159 1.0000000 0.7677488 0.7715522 0.7796929 0.7657481
```

```
## X800m    0.7119388 0.7969162 0.7677488 1.0000000 0.8957609 0.8606959 0.8431074
## X1500m   0.7657919 0.7950871 0.7715522 0.8957609 1.0000000 0.9165224 0.9013380
## X5000m   0.7398803 0.7613028 0.7796929 0.8606959 0.9165224 1.0000000 0.9882324
## X10000m  0.7147921 0.7479519 0.7657481 0.8431074 0.9013380 0.9882324 1.0000000
## Marathon 0.6764873 0.7211157 0.7126823 0.8069657 0.8777788 0.9441466 0.9541630
##           Marathon
## X100m    0.6764873
## X200m    0.7211157
## X400m    0.7126823
## X800m    0.8069657
## X1500m   0.8777788
## X5000m   0.9441466
## X10000m  0.9541630
## Marathon 1.0000000
```

Correlation matrix shows that variables are highly correlated

3. Determine the eigen values and eigen vectors

```
# Using PCA to calculate Eigen values
# for raw data we are using covariant matrix
pca.out.raw <- PCA(dataset1,ncp = 8, scale.unit = F, graph = FALSE)
pca.out.raw$eig
```

```
##          eigenvalue percentage of variance cumulative percentage of variance
## comp 1 8.294229e+01           9.828776e+01                           98.28776
## comp 2 1.120209e+00           1.327463e+00                           99.61522
## comp 3 2.245284e-01           2.660692e-01                           99.88129
## comp 4 7.907404e-02           9.370383e-02                           99.97500
## comp 5 1.156114e-02           1.370011e-02                           99.98870
## comp 6 6.140662e-03           7.276768e-03                           99.99597
## comp 7 3.007928e-03           3.564436e-03                           99.99954
## comp 8 3.898169e-04           4.619384e-04                          100.00000
```

Eigen values

```
round(pca.out.raw$eig[,1],4)
```

```
##  comp 1  comp 2  comp 3  comp 4  comp 5  comp 6  comp 7  comp 8
## 82.9423  1.1202  0.2245  0.0791  0.0116  0.0061  0.0030  0.0004
```

Eigen Vector

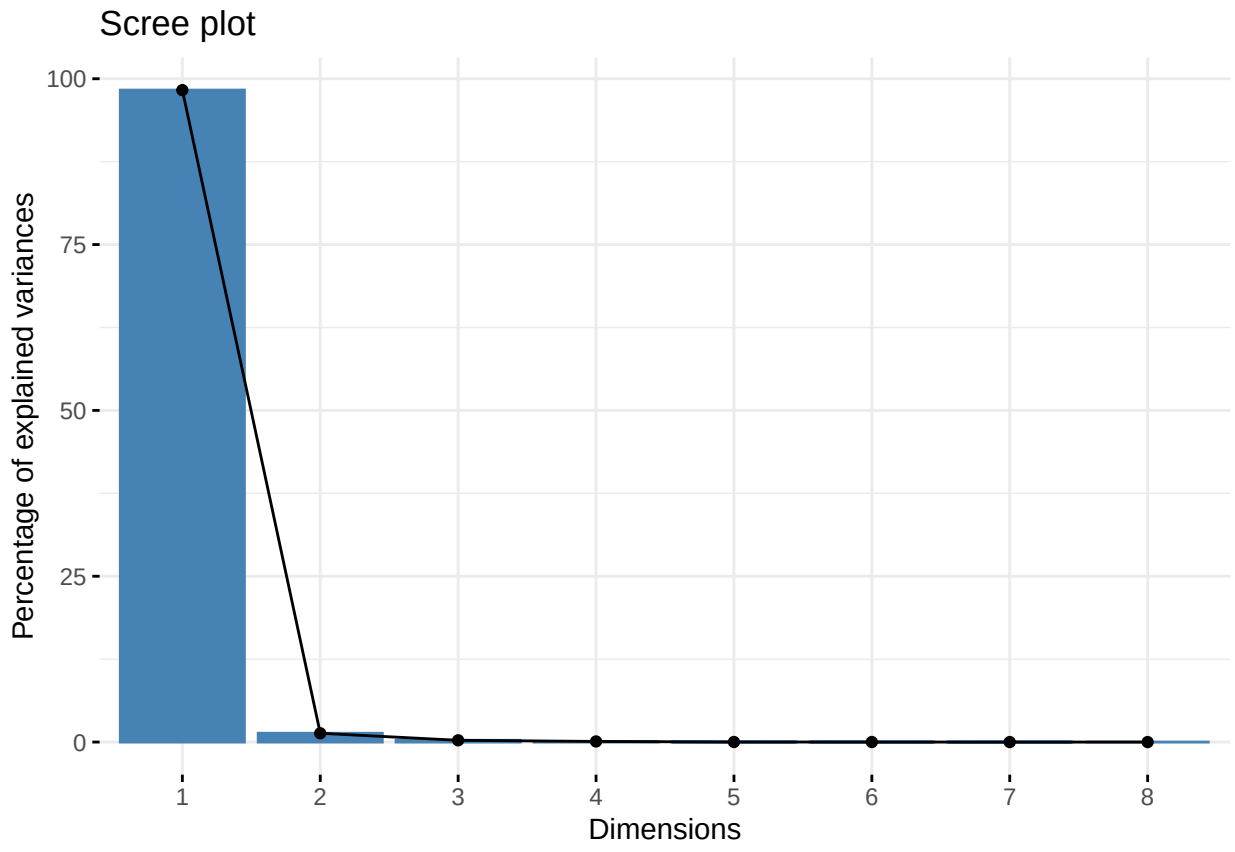
```
pca.out.raw$var$coord
```

```
##               Dim.1       Dim.2        Dim.3        Dim.4         Dim.5
## X100m    0.15046446  0.10578464 -0.003818664  0.091134829  0.0335533018
## X200m    0.39711287  0.26758673 -0.038296464  0.252345416 -0.0185439069
## X400m    1.03442059  0.96984782 -0.120099789 -0.081036675 -0.0012408232
## X800m    0.04229176  0.01487309  0.005813490  0.007417815  0.0046812093
```

```
## X1500m   0.13281554  0.03256236  0.017562605  0.018237096  0.0221198965
## X5000m   0.71577105  0.12334515  0.178528047  0.009848064  0.0888346708
## X10000m  1.59549657  0.22113946  0.413559454 -0.005089436 -0.0410734233
## Marathon 8.86648547 -0.17723792 -0.073330077 -0.003582121  0.0002718501
##                  Dim.6         Dim.7         Dim.8
## X100m     0.0691721135  1.584069e-03  1.678424e-03
## X200m    -0.0228938790 -4.036922e-03 -8.453351e-04
## X400m     0.0001473270  2.131311e-05 -4.122545e-05
## X800m    -0.0099156069  1.066479e-02  1.917502e-02
## X1500m   -0.0086289690  5.180975e-02 -4.253936e-03
## X5000m   -0.0238784876 -1.347497e-02 -6.973034e-04
## X10000m   0.0094185404  3.088398e-03  1.404636e-04
## Marathon  0.0002437045 -1.434597e-04  1.746337e-05
```

4. Obtain the scree plot for the principle components calculated on the raw data.

```
fviz_eig(pca.out.raw)
```



5. How many principle components would you like to retain based on the scree plot obtained.

These dimensions (PCs) are orthogonal to each other. When we decide PCs based on scree plot what we are looking is where the slope of percentage explained is negligible. In other words elbow joint.

In this case after principle component 2, slope is negligible. Therefore I would retain comp1 and comp2

6. Determine the proportion of variance explained by the first principle component.

First principle component (comp 1) explains 98.28% of variance.

7. Now, redo the parts (2) through (6) for the standardized data.

7.2 Obtain the sample correlation matrix for the data.

```
cor(dataset1)
```

```
##              X100m     X200m     X400m     X800m    X1500m    X5000m   X10000m
## X100m    1.0000000 0.9147554 0.8041147 0.7119388 0.7657919 0.7398803 0.7147921
## X200m    0.9147554 1.0000000 0.8449159 0.7969162 0.7950871 0.7613028 0.7479519
## X400m    0.8041147 0.8449159 1.0000000 0.7677488 0.7715522 0.7796929 0.7657481
## X800m    0.7119388 0.7969162 0.7677488 1.0000000 0.8957609 0.8606959 0.8431074
## X1500m   0.7657919 0.7950871 0.7715522 0.8957609 1.0000000 0.9165224 0.9013380
## X5000m   0.7398803 0.7613028 0.7796929 0.8606959 0.9165224 1.0000000 0.9882324
## X10000m  0.7147921 0.7479519 0.7657481 0.8431074 0.9013380 0.9882324 1.0000000
## Marathon 0.6764873 0.7211157 0.7126823 0.8069657 0.8777788 0.9441466 0.9541630
##           Marathon
## X100m    0.6764873
## X200m    0.7211157
## X400m    0.7126823
## X800m    0.8069657
## X1500m   0.8777788
## X5000m   0.9441466
## X10000m  0.9541630
## Marathon 1.0000000
```

Correlation matrix shows that variables are highly correlated

7.3 Determine the eigen values and eigen vectors

```
# Using PCA to calculate Eigen values
# for standardized data we are using correlation matrix
pca.out.sd <- PCA(dataset1,ncp = 8, graph = FALSE)
pca.out.sd$eig
```

```
##         eigenvalue percentage of variance cumulative percentage of variance
## comp 1 6.703289951             83.7911244                           83.79112
## comp 2 0.638410110              7.9801264                           91.77125
## comp 3 0.227524494              2.8440562                           94.61531
## comp 4 0.205849181              2.5731148                           97.18842
## comp 5 0.097577441              1.2197180                           98.40814
## comp 6 0.070687912              0.8835989                           99.29174
## comp 7 0.046942050              0.5867756                           99.87851
## comp 8 0.009718862              0.1214858                          100.00000
```

Eigen values

```
round(pca.out.sd$eig[,1],4)
```

```
## comp 1 comp 2 comp 3 comp 4 comp 5 comp 6 comp 7 comp 8
## 6.7033 0.6384 0.2275 0.2058 0.0976 0.0707 0.0469 0.0097
```

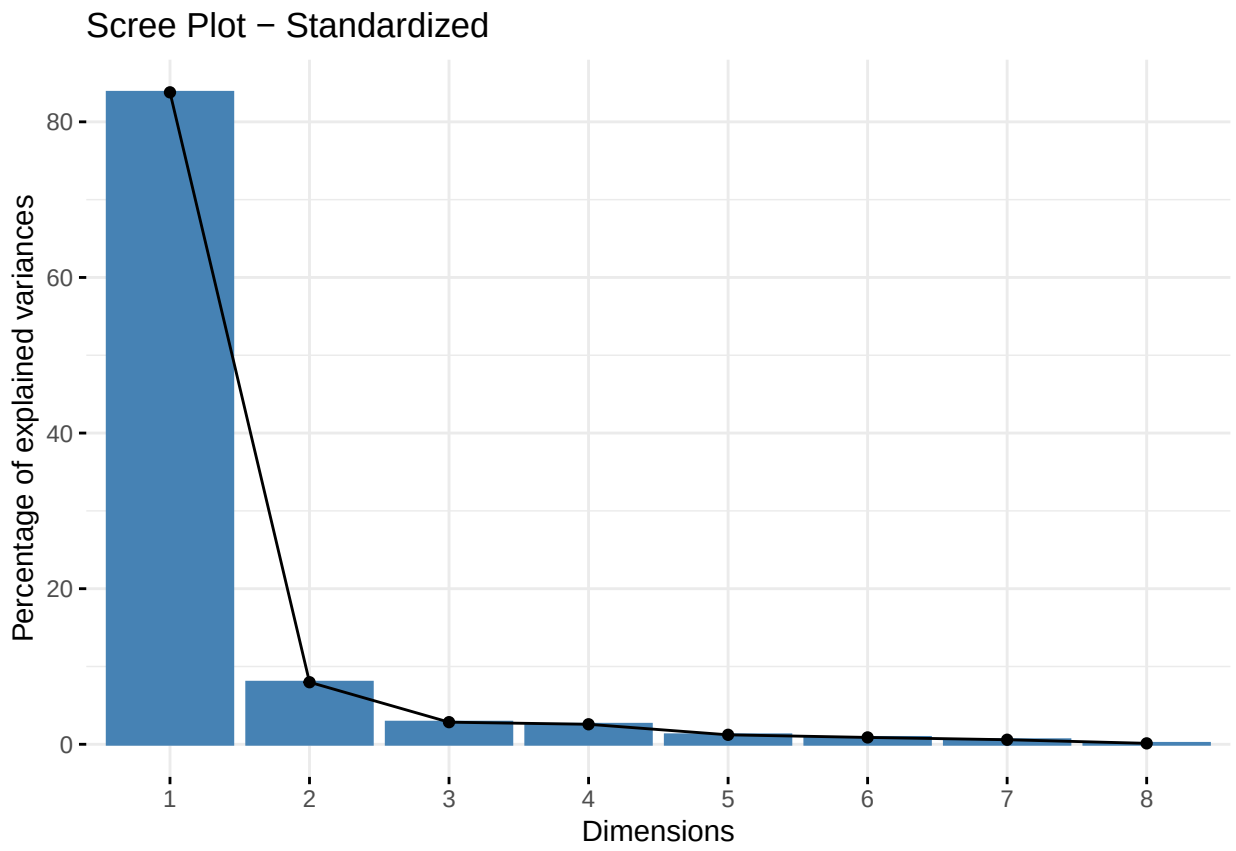
Eigen Vector

```
pca.out.sd$var$coord
```

```
##              Dim.1       Dim.2        Dim.3        Dim.4       Dim.5
## X100m    0.8605755  0.42299290  0.164019263 -0.172746424 -0.09360945
## X200m    0.8959509  0.37584469 -0.001805954 -0.098464749  0.16912991
## X400m    0.8780162  0.27592007 -0.031987545  0.386239872 -0.04154145
## X800m    0.9139768 -0.07147524 -0.373349523 -0.060923243  0.07099713
## X1500m   0.9475610 -0.12276915 -0.116515655 -0.105722740 -0.20355032
## X5000m   0.9574913 -0.23551480  0.087224857  0.024783427 -0.02243356
## X10000m  0.9474679 -0.26655325  0.116377639  0.039503854  0.01915871
## Marathon 0.9172508 -0.30886432  0.159618346 -0.008221675  0.10554831
##                Dim.6       Dim.7         Dim.8
## X100m     0.09625553  0.07532160  0.0064771301
## X200m    -0.09268091 -0.09530856 -0.0059895281
## X400m    -0.02049444  0.02460301  0.0003420605
## X800m     0.09074435  0.05608990  0.0038718020
## X1500m   -0.14085356 -0.03185703  0.0039182827
## X5000m    0.09548628 -0.07113424 -0.0695694120
## X10000m   0.07260596 -0.07607261  0.0687311626
## Marathon -0.09974715  0.12873076 -0.0068335563
```

7.4 Obtain the scree plot for the principle components calculated on the raw data.

```
fviz_eig(pca.out.sd,main = "Scree Plot - Standardized")
```



7.5 How many principle components would you like to retain based on the scree plot obtained.

With standardized scree plot, since scale has improved we can observed that elbow joint include comp 3 as well.

After principle component 3, slope is negligible. Therefore I would retain comp 1, comp 2 and comp 3

7.6 Determine the proportion of variance explained by the first principle component.

First principle component (comp 1) explains 83.79% of variance.

8. Do you see any difference between the answers you obtained for the raw data and standardized data?

Yes. there is major difference in both component when you consider variable contribution to components.

Raw Data Analysis Under raw data principle component 1 explains 98.28% of data variance and out of that 68.54% is dominated by Marathon and it's very significant different from others.

Standardized Data Analysis Component 1 explains 83.79% of data variance and out of that 12.53% is contributed by Marathon giving fair/equal opportunity for other variables. Event including Comp 2 and Comp 3 explain only 94.61% of variance.

Therefore, standardization of data provide better results by avoiding domination of few variables with large variances.

```
pca.out.raw$var$coord[,1][8]* 100/sum(pca.out.raw$var$coord[,1])
```

```
## Marathon
## 68.54722
```

```
pca.out.sd$var$coord[,1][8]* 100/sum(pca.out.sd$var$coord[,1])
```

```
## Marathon
## 12.53368
```

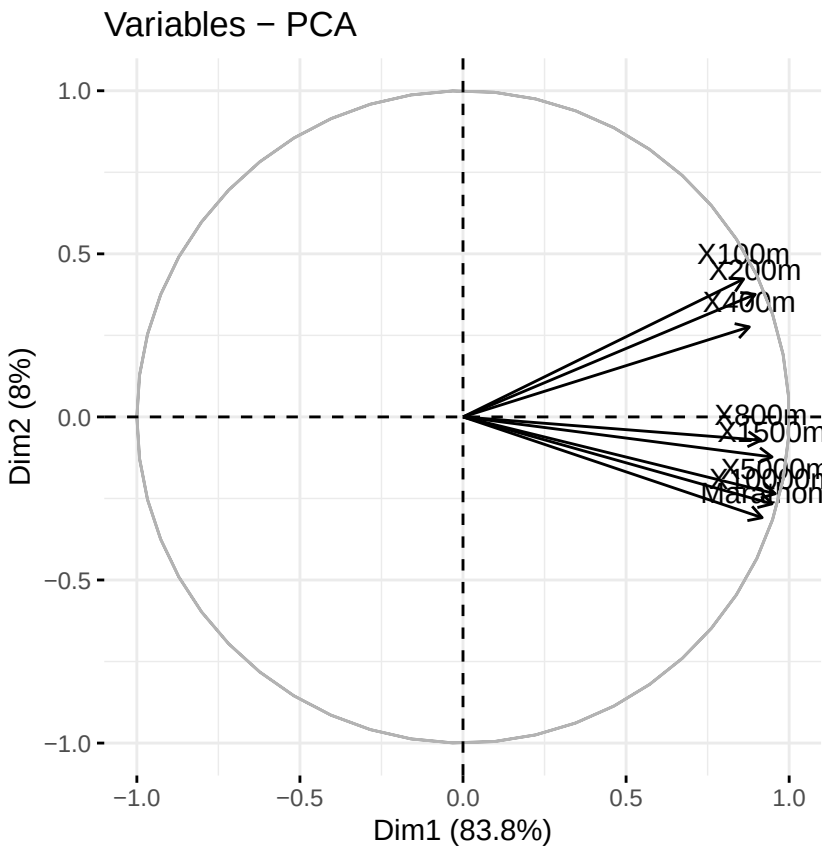
9. Is it possible to interpret the first two principle components you obtained for the standardized data? If Yes, then give a brief interpretation.

YES

Comp 1 represent 83.79% variability and comp 2 contribute to 7.98% variability.

All variables positively contribute to comp 1. Only 100m, 200m and 400m contribute positively to comp 2 and others contribute negatively to it.

```
fviz_pca_var(pca.out.sd)
```



There is no arrows driven opposite direction or perpendicular to each other. This means there is no negative correlations or independent respectively.

100m, 200m and 400m (short runs) clearly positively correlated.

1500m, 5000m, 10000m and Marathon (long distance events) are positively correlated.

800m contribute mainly to comp 1.

All variables positively contribute to comp 1. Only 100m, 200m and 400m contribute positively to comp 2 and others contribute negatively to it.

Mostly all the arrows distances are similar hence, all variables are having fairly equal representation.

10. Rank the nations based on the scores on the first principle component. Based on the rankings, identify the top 5 and last 5 countries.

```
# Calculate the scores on the first principal component
pc1_scores <- scale(dataset1) %*% pca.out.sd$var$coord[,1]

# Create a data frame with country names and their scores on the first principal component
df <- data.frame(Country = dataset$'C1.T', Rank = pc1_scores)
head(df)
```

```
##     Country      Rank
## 1 Argentina -1.077915
## 2 Australia -6.090796
## 3   Austria -1.891658
## 4   Belgium -5.125783
## 5   Bermuda  3.847706
## 6    Brazil -5.717323
```

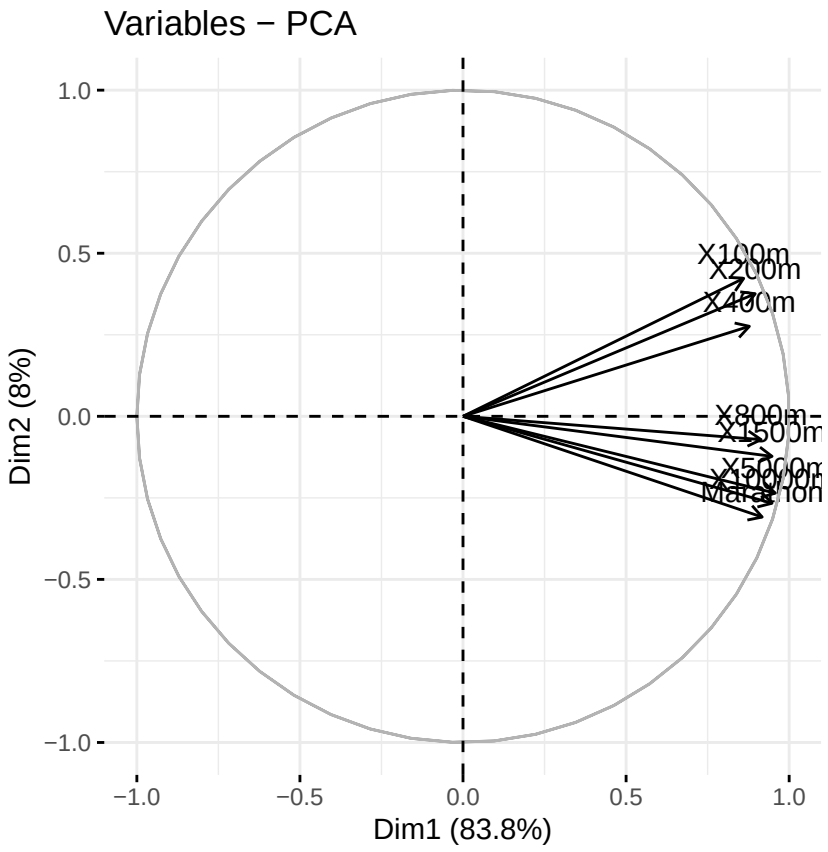
```
# Sort the data frame
sorted_df <- df[order(-df$Rank), ]

# Get the top 5 countries
top_5 <- head(sorted_df, 5)

# Get the last 5 countries
last_5 <- tail(sorted_df,5)
```

Top 5 countries are: CookIslands, Samoa, Singapore, PapuaNewGuinea, Myanmar(Burma)

Last 5 Countries are: Australia, France, Kenya, GreatBritain, U.S.A.

11. Obtain the variable correlation plot using fviz pca var() function in R.

```
fviz_pca_var(pca.out.sd)
```



12. From the above plot, identify the positively correlated variable with the first principle component.

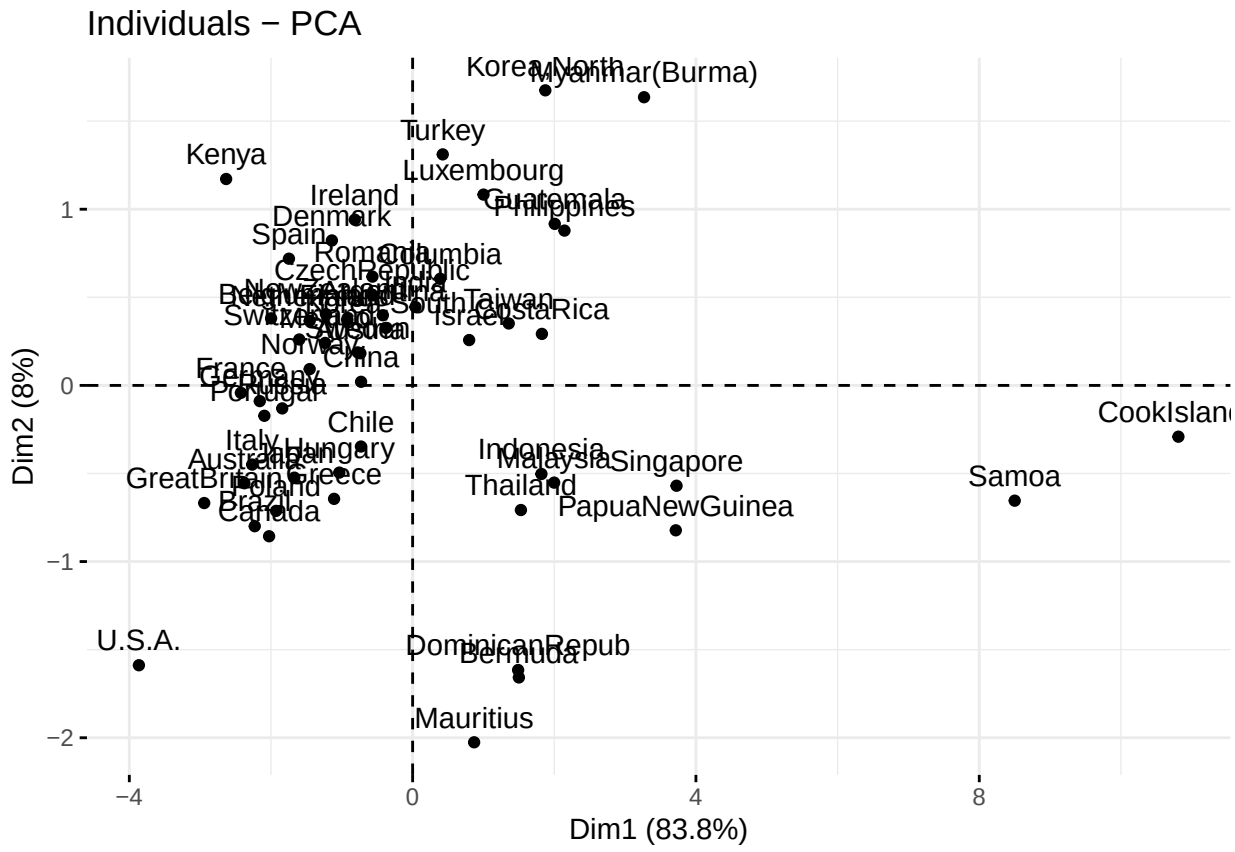
Right side of the Dim 1 are all having positive correlation ship. In this case all variables are with first principle component.

100m, 200m and 400m (short runs) clearly positively closely correlated.

800m, 1500m, 5000m, 10000m and Marathon (long distance events) are positively closely correlated.

13. Obtain the graph of individuals using fviz pca ind() function in R.

```
dataset2 <- dataset1
rownames(dataset2) <- dataset$'C1.T'

pca.out.sd2 <- PCA(dataset2,ncp = 8, graph = FALSE)

fviz_pca_ind(pca.out.sd2)
```



14. Identify possible grouping among the countries.

U.S.A. is outstanding from Europe countries

France, Germany, Russia and Portugal got close group

Indonesia, Malaysia having a group

Netherlands, New Zealand and Finland is closely associated