2-Práctica de Análisis de Componentes Principales en R

Vamos a realizar un Análisis de Componentes principales sobre la matriz de datos correspondientes a los resultados en la prueba femenina de heptathlon de las Olimpiadas de Río de 2016 de las 29 primeras clasificadas.

1º parte de la Práctica

1. Localizar la tabla de datos en Internet y prepararla como una matriz de 29 x 8, las columnas corresponden a las 7 pruebas del heptathlon, más la puntuación total obtenida por cada atleta, y las filas a las 29 primeras clasificadas.

2. Analizar el tratamiento que hay que dar a las dos últimas atletas que figuran con 0 en la carrera de 800m.

3. Realizar los pasos efectuados en la 1ª práctica.

4. Vamos a utilizar también diversas librerías aparte de prcomp que realizan Análisis de Componentes Principales, como princomp, principal, pca.

5. Introducir el siguiente código en R que nos va a permitir realizar un gráfico de la matriz de correlaciones entre las variables. Como es lógico las funciones y librerías nuevas que aparezcan requerirán la instalación de los paquetes correspondientes. Debes asegurarte de tener la última versión de R instalada. Este código viene referido a los datos de la 1ª práctica.

   Creamos la matriz de correlaciones pero esta matriz la hemos de transformar en una lista, para ello empleamos la función melt del paquete reshape, en este punto hemos pasado de una matriz de correlaciones a una lista con pares de variables y su correspondiente coeficiente de correlación. A esta lista le asignamos los nombres que deseamos con la función names y por último generamos un vector que denominamos escala que nos permitirá establecer los colores que deseamos utilizar cuando pintemos nuestra matriz de correlaciones.

   El código de ggplot2 que utilizamos para pintarla es muy sencillo:

   La clave es geom_tile que nos permite pintar mapas que estarán compuestos de cuadrados y representarán el coeficiente de correlación entre las variables. Con scale_fill_continous especificamos el rango de colores y el número de grupos que utilizamos, como vemos es el objeto escala el que nos indica el número de grupos. Por último añadimos un título y obtenemos un gráfico de correlaciones.

```
#Cargamos el paquete ggplot2
library(ggplot2)
datos <- cor(heptathlon)
#Cargamos el paquete reshape2, la función melt del paquete reshape2, convierte la matriz en una
lista con pares de variables y su correspondiente coeficiente de correlación
library(reshape2)
datos.lista  <-  melt(datos)
names(datos.lista) <- c("Variable_1","Variable_2","Correlacion")
escala <- seq(-1,1,0.1)
(p <- ggplot(datos.lista, aes(Variable_1, Variable_2, fill=Correlacion))
   geom_tile(aes(fill=Correlacion))
   scale_fill_continuous(low = "white", high = "steelblue" ,breaks=escala)
labs(title="Matriz de correlaciones",
        plot.title = element_text(face="bold", size=14)))
```

6. Por último vas a obtener las conclusiones correspondientes a las relaciones entre las atletas, a las relaciones entre las pruebas del heptathlon y a las relaciones entre las atletas y las pruebas.

2ª parte de la Práctica

1. Realizar el análisis paso a paso según el desarrollo teórico estudiado en ACP con el siguiente guión:

    a. Leer la tabla con las 29 atletas.
    b. Las 2 últimas atletas en la prueba de 800m, una no terminó y la otra fue descalificada, por lo que seguiremos el estudio con las 27 primeras atletas. Lo que haremos es convertir los ceros en NA (valores no disponibles) y consideraremos el estudio para los casos completos, es decir las 27 primeras.
    c. Realizar las transformaciones de valores complementarios para las pruebas de velocidad.
    d. Eliminar la columna score de la tabla de partida y convertir la tabla en matriz.
    e. Centrar la matriz y tipificar las variables.
    f. Dividir por $\sqrt{n}$.
    g. Calcular X'X para obtener la matriz de correlaciones. Averigua cómo obtiene el programa R la matriz de correlaciones.
    h. Comparar el resultado del apartado anterior con la matriz de correlaciones de la tabla inicial obtenida con la orden 'cor', ambas deben de coincidir.
    i. Calcular los autovalores y autovectores de la matriz de correlaciones (descomposición en valores singulares de una matriz o autodescomposición de una matriz).
    j. Representar gráficamente las 2 primeras componentes de las pruebas con sus nombres.
    k. Afinar un poco más el apartado anterior y en el mismo gráfico dibujar unos vectores que salgan del origen y terminen en cada una de las variables (al modo biplot).
    l. Representar gráficamente las 2 primeras componentes de las atletas con sus nombres.
    m. Por último comparar los resultados obtenidos con los de la primera parte, ¿salen las gráficas igual?

1° parte de la Práctica

El tiempo realmente es 13,48

La marca más alta registrada en cada caso se resalta en amarillo

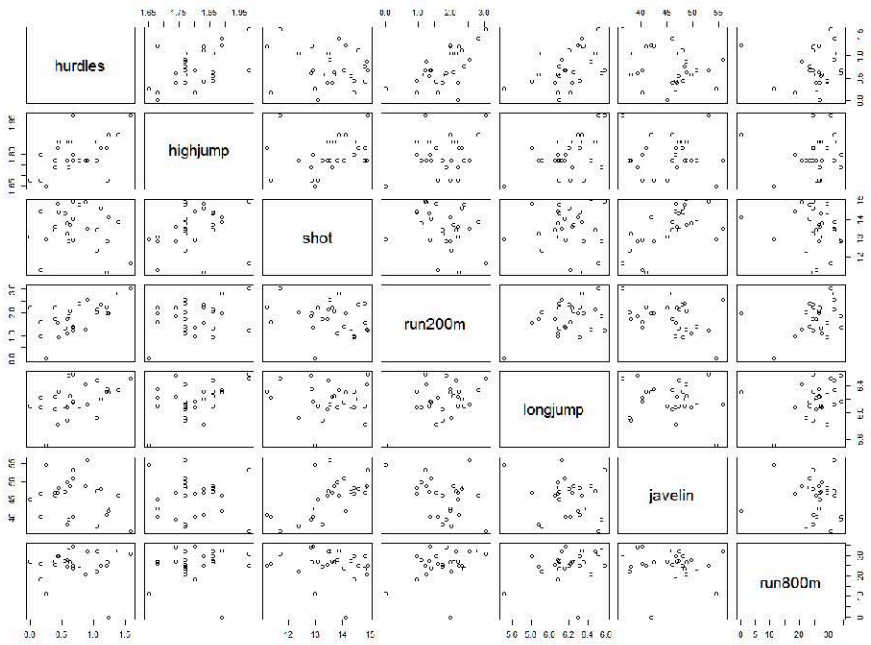
Río-2016

	Atleta	100_valla	S_altura	L_peso	200m	S_longitu	L_jabalin	800_m	Puntos
1	N_Thiam_(BEL)	13.56	1.98	14.91	25.10	6.58	53.13	136,54	**6810**
2	J_Ennis-Hill_(GBR)	12.84	1.89	13.86	23.49	6.34	46.06	129,07	**6775**
3	B_Theisen_Eaton_(CAN)	13.18	1.86	13.45	24.18	6.48	47.36	129,50	**6653**
4	L_Ikauniece_(LAT)	13.33	1.77	13.52	23.76	6.12	55.93	129,43	**6617**
5	C_Schafer_(GER)	13.12	1.83	14.57	23.99	6.20	47.99	136,52	**6540**
6	K_Johnson-Thomps_(GBR)	12.65	1.98	11.68	23.26	6.51	36.36	130,47	**6480**
7	Y_Rodriguez_(CUB)	13.61	1.86	13.69	24.26	6.25	48.89	134,65	**6452**
8	G_Zsivoczky-Farkas_(HUN)	13.79	1.86	14.39	25.38	6.31	48.07	131,76	**6442**
9	J_Oeser_(GER)	13.69	1.86	14.28	24.99	6.10	47.22	133,82	**6401**
10	A_Vetter_(NED)	13.47	1.77	14.78	23.93	6.10	48.42	137,71	**6394**
11	A_N_Djimou_Ida_(FRA)	13.37	1.77	14.88	25.07	6.43	48.76	140,36	**6383**
12	B_Nwaba_(USA)	13.81	1.83	14.81	24.77	5.81	46.85	131,61	**6309**
13	N_Broersen_(NED)	13.56	1.77	14.04	24.94	6.15	50.80	137,55	**6300**
14	C_Rath_(GER)	13.63	1.74	12.83	24.48	6.55	39.39	127,22	**6267**
15	E_Aguilar_(COL)	13.84	1.74	13.60	24.12	6.23	46.90	134,32	**6263**
16	Xenia Krizsan (HUN)	13.66	1.77	13.78	25.24	6.08	49.78	133,46	**6257**
17	K_Williams_(USA)	13.04	1.83	11.21	24.09	6.31	40.93	136,24	**6221**
18	H_Miller-Koch_(USA)	13.56	1.80	12.91	24.97	6.16	40.25	126,82	**6213**
19	N_Visser_(NED)	13.02	1.68	12.84	24.34	6.35	42.48	134,47	**6190**
20	A_Jones_(BAR)	13.00	1.89	14.09	24.35	6.30	42.00	161,12	**6173**
21	Ivona Dadic (AUT)	13.84	1.77	13.43	24.60	6.05	46.08	135,64	**6155**
22	E_Klucinova_(CZE)	14.07	1.80	14.41	25.37	6.08	46.73	142,81	**6077**
23	Vanessa Chefer_(BRA)	14.24	1.68	13.06	24.11	6.10	45.05	134,20	**6024**
24	K_Cachova_(CZE)	13.19	1.77	12.38	24.32	5.91	37.77	138,95	**5958**
25	H_Kasyanova_(UKR)	13.66	1.77	13.25	24.60	5.88	38.10	136,58	**5951**
26	A_Felix_(PUR)	14.07	1.68	11.36	24.74	6.22	40.17	135,32	**5805**

27	S_Yfantidou_(GRE)	13.99	1.65	12.97	26.32	5.51	54.57	150,08	**5613**
28	A_Fodorova_(UKR)	14.10	1.80	14.38	25.4	6.00	35.44	0	**5038**
29	U_Osazuwa_(NGR)	13.75	1.77	13.15	24.67	5.72	33.42	0	**4916**

```
> X <- read.table("2acp-16.txt" ,header=T)
> X
                      hurdles highjump  shot run200m longjump javelin run800m score
 N_Thiam_(BEL)          13.56     1.98 14.91   25.10     6.58   53.13 136.54  6810
 J_Ennis-Hill_(GBR)     12.84     1.89 13.86   23.49     6.34   46.06 129.07  6775
 B_Theisen_Eaton_(CAN)  13.18     1.86 13.45   24.18     6.48   47.36 129.50  6653
 L_Ikauniece_(LAT)      13.33     1.77 13.52   23.76     6.12   55.93 129.43  6617
 C_Schafer_(GER)        13.12     1.83 14.57   23.99     6.20   47.99 136.52  6540
 K_Johnson_(GBR)        12.65     1.98 11.68   23.26     6.51   36.36 130.47  6480
 Y_Rodriguez_(CUB)      13.61     1.86 13.69   24.26     6.25   48.89 134.65  6452
 G_Zsivoczkyarkas_(HUN) 13.79     1.86 14.39   25.38     6.31   48.07 131.76  6442
 J_Oeser_(GER)          13.69     1.86 14.28   24.99     6.10   47.22 133.82  6401
 A_Vetter_(NED)         13.47     1.77 14.78   23.93     6.10   48.42 137.71  6394
 A_N_Djimou_Ida_(FRA)   13.37     1.77 14.88   25.07     6.43   48.76 140.36  6383
 B_Nwaba_(USA)          13.81     1.83 14.81   24.77     5.81   46.85 131.61  6309
 N_Broersen_(NED)       13.56     1.77 14.04   24.94     6.15   50.80 137.55  6300
 C_Rath_(GER)           13.63     1.74 12.83   24.48     6.55   39.39 127.22  6267
 E_Aguilar_(COL)        13.84     1.74 13.60   24.12     6.23   46.90 134.32  6263
 Xenia_Krizsan_(HUN)    13.66     1.77 13.78   25.24     6.08   49.78 133.46  6257
 K_Williams_(USA)       13.04     1.83 11.21   24.09     6.31   40.93 136.24  6221
 H_Miller-Koch_(USA)    13.56     1.80 12.91   24.97     6.16   40.25 126.82  6213
 N_Visser_(NED)         13.02     1.68 12.84   24.34     6.35   42.48 134.47  6190
 A_Jones_(BAR)          13.00     1.89 14.09   24.35     6.30   42.00 161.12  6173
 Ivona_Dadic_(AUT)      13.84     1.77 13.43   24.60     6.05   46.08 135.64  6155
 E_Klucinova_(CZE)      14.07     1.80 14.41   25.37     6.08   46.73 142.81  6077
 Vanessa_Chefer_(BRA)   14.24     1.68 13.06   24.11     6.10   45.05 134.20  6024
 K_Cachova_(CZE)        13.19     1.77 12.38   24.32     5.91   37.77 138.95  5958
 H_Kasyanova_(UKR)      13.66     1.77 13.25   24.60     5.88   38.10 136.58  5951
 A_Felix_(PUR)          14.07     1.68 11.36   24.74     6.22   40.17 135.32  5805
 S_Yfantidou_(GRE)      13.99     1.65 12.97   26.32     5.51   54.57 150.08  5613
A_Fodorova_(UKR)        14.10     1.80 14.38   25.40     6.00   35.44   0.00  5038
U_Osazuwa_(NGR)         13.75     1.77 13.15   24.67     5.72   33.42   0.00  4916

> X <- replace(X, X==0, NA)
> X <- subset(X,complete.cases(X))

> X$hurdles <- max(X$hurdles) -X$hurdles
> X$run200m <- max(X$run200m) -X$run200m
> X$run800m <- max(X$run800m) -X$run800m

> score <- which(colnames(X) == "score")
> X <- X[,-score]
> X
                      hurdles highjump  shot run200m longjump javelin run800m
 N_Thiam_(BEL)           0.68     1.98 14.91    1.22     6.58   53.13   24.58
 J_Ennis-Hill_(GBR)      1.40     1.89 13.86    2.83     6.34   46.06   32.05
 B_Theisen_Eaton_(CAN)   1.06     1.86 13.45    2.14     6.48   47.36   31.62
 L_Ikauniece_(LAT)       0.91     1.77 13.52    2.56     6.12   55.93   31.69
 C_Schafer_(GER)         1.12     1.83 14.57    2.33     6.20   47.99   24.60
 K_Johnson_(GBR)         1.59     1.98 11.68    3.06     6.51   36.36   30.65
 Y_Rodriguez_(CUB)       0.63     1.86 13.69    2.06     6.25   48.89   26.47
 G_Zsivoczkyarkas_(HUN)  0.45     1.86 14.39    0.94     6.31   48.07   29.36
 J_Oeser_(GER)           0.55     1.86 14.28    1.33     6.10   47.22   27.30
 A_Vetter_(NED)          0.77     1.77 14.78    2.39     6.10   48.42   23.41
 A_N_Djimou_Ida_(FRA)    0.87     1.77 14.88    1.25     6.43   48.76   20.76
 B_Nwaba_(USA)           0.43     1.83 14.81    1.55     5.81   46.85   29.51
 N_Broersen_(NED)        0.68     1.77 14.04    1.38     6.15   50.80   23.57
 C_Rath_(GER)            0.61     1.74 12.83    1.84     6.55   39.39   33.90
 E_Aguilar_(COL)         0.40     1.74 13.60    2.20     6.23   46.90   26.80
 Xenia_Krizsan_(HUN)     0.58     1.77 13.78    1.08     6.08   49.78   27.66
 K_Williams_(USA)        1.20     1.83 11.21    2.23     6.31   40.93   24.88
 H_Miller-Koch_(USA)     0.68     1.80 12.91    1.35     6.16   40.25   34.30
 N_Visser_(NED)          1.22     1.68 12.84    1.98     6.35   42.48   26.65
 A_Jones_(BAR)           1.24     1.89 14.09    1.97     6.30   42.00    0.00
 Ivona_Dadic_(AUT)       0.40     1.77 13.43    1.72     6.05   46.08   25.48
 E_Klucinova_(CZE)       0.17     1.80 14.41    0.95     6.08   46.73   18.31
 Vanessa_Chefer_(BRA)    0.00     1.68 13.06    2.21     6.10   45.05   26.92
 K_Cachova_(CZE)         1.05     1.77 12.38    2.00     5.91   37.77   22.17
 H_Kasyanova_(UKR)       0.58     1.77 13.25    1.72     5.88   38.10   24.54
 A_Felix_(PUR)           0.17     1.68 11.36    1.58     6.22   40.17   25.80
 S_Yfantidou_(GRE)       0.25     1.65 12.97    0.00     5.51   54.57   11.04
```

```
> plot(X)
```




```
> Z <- round(cor(X),2)
> Z
         hurdles highjump  shot run200m longjump javelin run800m
hurdles     1.00     0.52 -0.15    0.62     0.46   -0.27    0.03
highjump    0.52     1.00  0.25    0.29     0.50   -0.03    0.08
shot       -0.15     0.25  1.00   -0.25    -0.03    0.62   -0.16
run200m     0.62     0.29 -0.25    1.00     0.43   -0.35    0.33
longjump    0.46     0.50 -0.03    0.43     1.00   -0.18    0.30
javelin    -0.27    -0.03  0.62   -0.35    -0.18    1.00   -0.10
run800m     0.03     0.08 -0.16    0.33     0.30   -0.10    1.00

> X_pca <- prcomp(X, scale=T)
> X_pca

Standard deviations:
[1] 1.6422272 1.2875146 1.0090571 0.7751684 0.6638359 0.5935750 0.4830158

Rotation:
                PC1         PC2         PC3         PC4         PC5         PC6         PC7
hurdles   0.4823786 -0.14319190  0.36173020 -0.34266244  0.05182669 -0.3868867  0.58836496
highjump  0.3490533 -0.48915795  0.14785050  0.32317736 -0.56007084 -0.2072972 -0.39488155
shot     -0.2126808 -0.65171727 -0.06296514 -0.08932881 -0.12918701  0.5852192  0.39864318
run200m   0.4932162  0.05060990 -0.05984647 -0.61295457  0.03493213  0.4355074 -0.42894581
longjump  0.4398372 -0.23931968 -0.17113888  0.48329701  0.68795159  0.1130882 -0.01839809
javelin  -0.3255236 -0.49942083 -0.22804117 -0.40002830  0.33059101 -0.4855649 -0.29584479
run800m   0.2399060  0.07892245 -0.87089264 -0.02910766 -0.28837009 -0.1632056  0.25907858

> summary(X_pca)
Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6     PC7
Standard deviation     1.6422 1.2875 1.0091 0.77517 0.66384 0.59357 0.48302
Proportion of Variance 0.3853 0.2368 0.1455 0.08584 0.06295 0.05033 0.03333
Cumulative Proportion  0.3853 0.6221 0.7675 0.85338 0.91634 0.96667 1.00000

> center <- X_pca$center
> center
  hurdles   highjump       shot    run200m   longjump    javelin    run800m
 0.7292593  1.8000000 13.5177778  1.7729630  6.1892593 45.7792593 25.3340741

> scale <- X_pca$scale
> scale
   hurdles   highjump       shot    run200m   longjump    javelin
0.40252019 0.08278796 1.03168396 0.65683867 0.23840699 5.12609382
   run800m
6.84449534

> scale(X, center = center, scale = scale)
                          hurdles   highjump       shot    run200m   longjump    javelin    run800m
N_Thiam_(BEL)          -0.1223771  2.1742292  1.349465801 -0.84185507  1.63896514  0.05476699  0.95232541
J_Ennis-Hill_(GBR)      1.6663530  1.0871146  0.331712264  1.60927955  0.63228323  0.30837140  0.89135096
B_Theisen_Eaton_(CAN)   0.8216749  0.7247431 -0.065696260  0.55879328  1.21951434  1.98020971  0.90127703
L_Ikauniece_(LAT)       0.4490228 -0.3623715  0.002153976  1.19821971 -0.29050851  0.43127200 -0.10409248
```

```
C_Schafer_(GER)          0.9707357  0.3623715  1.019907513  0.84805762  0.04505212  -1.83751207  0.75380393
K_Johnson_(GBR)          2.1383790  2.1742292 -1.781337937  1.95944164  1.34534958   0.60684428  0.16107550
Y_Rodriguez_(CUB)       -0.2465945  0.7247431  0.166933120  0.43699778  0.25477752   0.44687843  0.57088056
G_Zsivoczkyarkas_(HUN) -0.6937770  0.7247431  0.845435478 -1.26813935  0.50644799   0.28106016  0.27877038
J_Oeser_(GER)          -0.4453423  0.7247431  0.738813679 -0.67438624 -0.37439867   0.51515654 -0.27283574
A_Vetter_(NED)           0.1012142 -0.3623715  1.223458220  0.93940425 -0.37439867   0.58148384 -0.64860855
A_N_Djimou_Ida_(FRA)     0.3496489 -0.3623715  1.320387128 -0.79618175  1.00978895   0.20888044  0.59215072
B_Nwaba_(USA)           -0.7434640  0.3623715  1.252536892 -0.33944859 -1.59080597   0.97944769 -0.25014757
N_Broersen_(NED)        -0.1223771 -0.3623715  0.506184299 -0.59826405 -0.16467327  -1.24641871  1.21465737
C_Rath_(GER)            -0.2962814 -0.7247431 -0.666655491  0.10206013  1.51312990   0.21863446  0.20786985
E_Aguilar_(COL)         -0.8179944 -0.7247431  0.079697102  0.65013992  0.17088736   0.78046577  0.32981876
Xenia_Krizsan_(HUN)     -0.3708119 -0.3623715  0.254169137 -1.05499721 -0.45828883  -0.94599503 -0.06438818
K_Williams_(USA)         1.1694835  0.3623715 -2.236903805  0.69581323  0.50644799  -1.07864964  1.27137780
H_Miller-Koch_(USA)     -0.1223771  0.0000000 -0.589112365 -0.64393737 -0.12272819  -0.64362054  0.18659969
N_Visser_(NED)           1.2191705 -1.4494862 -0.656962601  0.31520227  0.67422831  -0.73725909 -3.59239855
A_Jones_(BAR)            1.2688574  1.0871146  0.554648753  0.29997783  0.46450292   0.05866860  0.02069245
Ivona_Dadic_(AUT)       -0.8179944 -0.3623715 -0.085082042 -0.08063314 -0.58412407   0.18547080 -0.99602115
E_Klucinova_(CZE)       -1.3893943  0.0000000  0.864821259 -1.25291491 -0.45828883  -0.14226413  0.22488598
Vanessa_Chefer_(BRA)    -1.8117333 -1.4494862 -0.443719002  0.66536435 -0.37439867  -1.56244882 -0.44866906
K_Cachova_(CZE)          0.7968314 -0.3623715 -1.102835579  0.34565114 -1.17135518  -1.49807232 -0.11260055
H_Kasyanova_(UKR)       -0.3708119 -0.3623715 -0.259554077 -0.08063314 -1.29719042  -1.09425607  0.06606879
A_Felix_(PUR)           -1.3893943 -1.4494862 -2.091510443 -0.29377528  0.12894228   1.71490048 -2.02691485
S_Yfantidou_(GRE)       -1.1906465 -1.8118577 -0.530955020 -2.69923658 -2.84915835   1.43398482 -0.10692850

attr(,"scaled:center")
   hurdles   highjump       shot    run200m   longjump    javelin    run800m
 0.7292593  1.8000000 13.5177778  1.7729630  6.1892593 45.7792593 25.3340741
attr(,"scaled:scale")
   hurdles   highjump       shot    run200m   longjump    javelin    run800m
0.40252019 0.08278796 1.03168396 0.65683867 0.23840699 5.12609382 7.05213348
```

```
> scale(X, center = center, scale = scale) %*%X_pca$rotation[,1] ##coordenadas Xu##
                          [,1]
N_Thiam_(BEL)           0.2260969
J_Ennis-Hill_(GBR)      2.3951904
B_Theisen_Eaton_(CAN)   1.5887567
L_Ikauniece_(LAT)       0.1244757
C_Schafer_(GER)         0.6505642
K_Johnson_(GBR)         4.5064455
Y_Rodriguez_(CUB)       0.2672145
G_Zsivoczkyarkas_(HUN) -0.6727212
J_Oeser_(GER)          -0.6408868
A_Vetter_(NED)         -0.2723651
A_N_Djimou_Ida_(FRA)   -0.5320837
B_Nwaba_(USA)          -1.2915869
N_Broersen_(NED)       -1.0395230
C_Rath_(GER)            1.1589018
E_Aguilar_(COL)        -0.2899863
Xenia_Krizsan_(HUN)    -1.2562643
K_Williams_(USA)        2.0248050
H_Miller-Koch_(USA)     0.3508171
N_Visser_(NED)          0.9281708
A_Jones_(BAR)           0.6039843
Ivona_Dadic_(AUT)      -0.8137975
E_Klucinova_(CZE)      -1.9730021
Vanessa_Chefer_(BRA)   -1.0217630
K_Cachova_(CZE)         0.5486902
H_Kasyanova_(UKR)      -0.3998344
A_Felix_(PUR)          -0.4474626
S_Yfantidou_(GRE)      -4.7228361
```

```
> drop(scale(X, center = center, scale = scale) %*%X_pca$rotation[,1])## sin dimensión
          N_Thiam_(BEL)      J_Ennis-Hill_(GBR)   B_Theisen_Eaton_(CAN)
              0.2260969               2.3951904               1.5887567
     L_Ikauniece_(LAT)         C_Schafer_(GER)         K_Johnson_(GBR)
              0.1244757               0.6505642               4.5064455
     Y_Rodriguez_(CUB)  G_Zsivoczkyarkas_(HUN)           J_Oeser_(GER)
              0.2672145              -0.6727212              -0.6408868
        A_Vetter_(NED)    A_N_Djimou_Ida_(FRA)           B_Nwaba_(USA)
             -0.2723651              -0.5320837              -1.2915869
      N_Broersen_(NED)           C_Rath_(GER)         E_Aguilar_(COL)
             -1.0395230               1.1589018              -0.2899863
   Xenia_Krizsan_(HUN)        K_Williams_(USA)     H_Miller-Koch_(USA)
             -1.2562643               2.0248050               0.3508171
        N_Visser_(NED)          A_Jones_(BAR)       Ivona_Dadic_(AUT)
              0.9281708               0.6039843              -0.8137975
     E_Klucinova_(CZE)    Vanessa_Chefer_(BRA)         K_Cachova_(CZE)
             -1.9730021              -1.0217630               0.5486902
     H_Kasyanova_(UKR)          A_Felix_(PUR)       S_Yfantidou_(GRE)
             -0.3998344              -0.4474626              -4.7228361
```

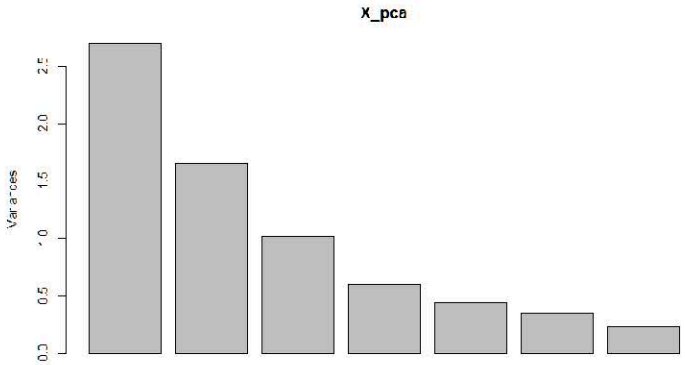
```
> predict(X_pca)[,1] ##coordenadas Xu
          N_Thiam_(BEL)      J_Ennis-Hill_(GBR)   B_Theisen_Eaton_(CAN)
              0.2260969               2.3951904               1.5887567
     L_Ikauniece_(LAT)         C_Schafer_(GER)         K_Johnson_(GBR)
              0.1244757               0.6505642               4.5064455
     Y_Rodriguez_(CUB)  G_Zsivoczkyarkas_(HUN)           J_Oeser_(GER)
              0.2672145              -0.6727212              -0.6408868
        A_Vetter_(NED)    A_N_Djimou_Ida_(FRA)           B_Nwaba_(USA)
             -0.2723651              -0.5320837              -1.2915869
      N_Broersen_(NED)           C_Rath_(GER)         E_Aguilar_(COL)
             -1.0395230               1.1589018              -0.2899863
   Xenia_Krizsan_(HUN)        K_Williams_(USA)     H_Miller-Koch_(USA)
             -1.2562643               2.0248050               0.3508171
```

```
      N_Visser_(NED)          A_Jones_(BAR)       Ivona_Dadic_(AUT)
           0.9281708             0.6039843            -0.8137975
   E_Klucinova_(CZE)  Vanessa_Chefer_(BRA)       K_Cachova_(CZE)
          -1.9730021            -1.0217630             0.5486902
   H_Kasyanova_(UKR)          A_Felix_(PUR)       S_Yfantidou_(GRE)
          -0.3998344            -0.4474626            -4.7228361
```

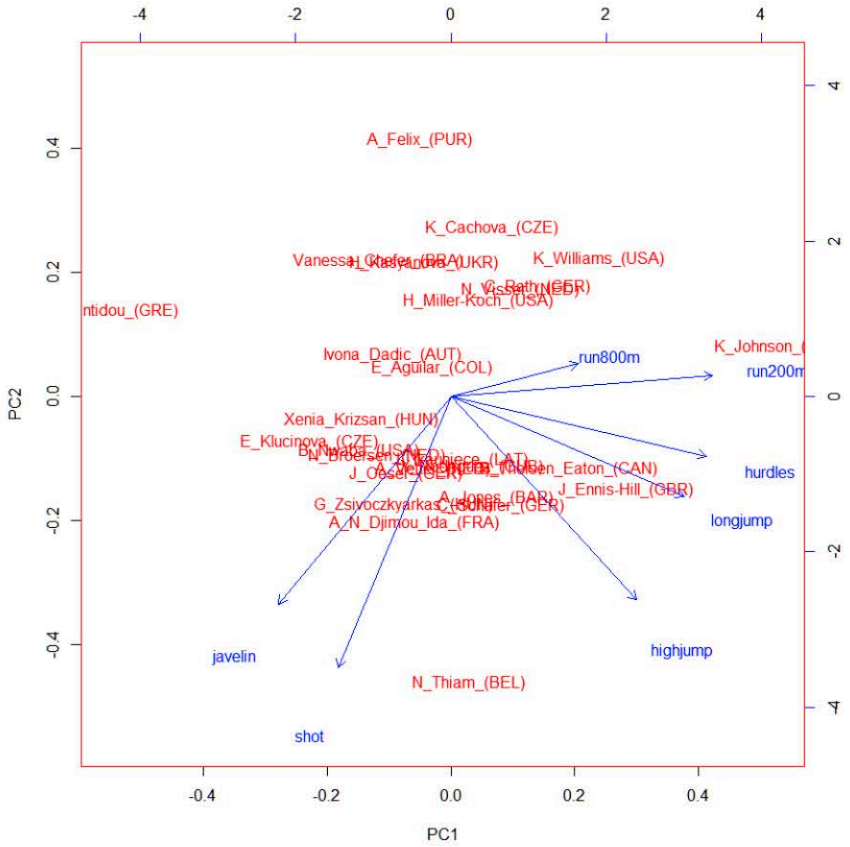
```
> X_pca$x[,1] ##coordenadas Xu
```

```
      N_Thiam_(BEL)      J_Ennis-Hill_(GBR)  B_Theisen_Eaton_(CAN)
          0.2260969             2.3951904             1.5887567
   L_Ikauniece_(LAT)        C_Schafer_(GER)       K_Johnson_(GBR)
          0.1244757             0.6505642             4.5064455
   Y_Rodriguez_(CUB) G_Zsivoczkyarkas_(HUN)         J_Oeser_(GER)
          0.2672145            -0.6727212            -0.6408868
      A_Vetter_(NED)   A_N_Djimou_Ida_(FRA)          B_Nwaba_(USA)
          -0.2723651            -0.5320837            -1.2915869
    N_Broersen_(NED)        C_Rath_(GER)          E_Aguilar_(COL)
          -1.0395230             1.1589018            -0.2899863
  Xenia_Krizsan_(HUN)      K_Williams_(USA)   H_Miller-Koch_(USA)
          -1.2562643             2.0248050             0.3508171
      N_Visser_(NED)          A_Jones_(BAR)       Ivona_Dadic_(AUT)
           0.9281708             0.6039843            -0.8137975
   E_Klucinova_(CZE)  Vanessa_Chefer_(BRA)       K_Cachova_(CZE)
          -1.9730021            -1.0217630             0.5486902
   H_Kasyanova_(UKR)          A_Felix_(PUR)       S_Yfantidou_(GRE)
          -0.3998344            -0.4474626            -4.7228361
```

```
> plot(X_pca) #para que salga el gráfico, X_pca no puede ser una matriz
```



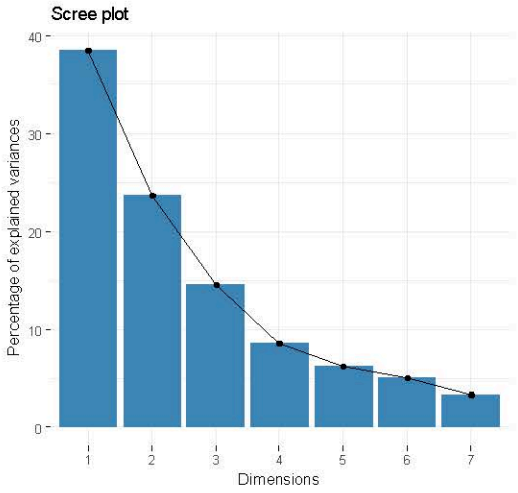

```
> biplot(X_pca, col=c("red", "blue"))
```



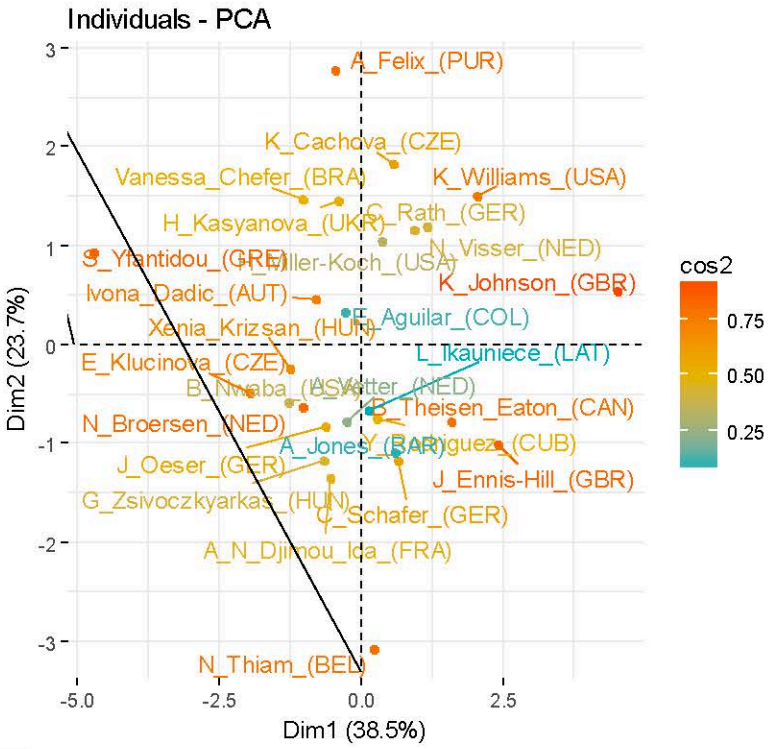
Vamos a utilizar el paquete de visualización *factoextra*

fviz {factoextra} es una función genérica de visualización de resultados de Análisis Multivariantes, ver ?fviz

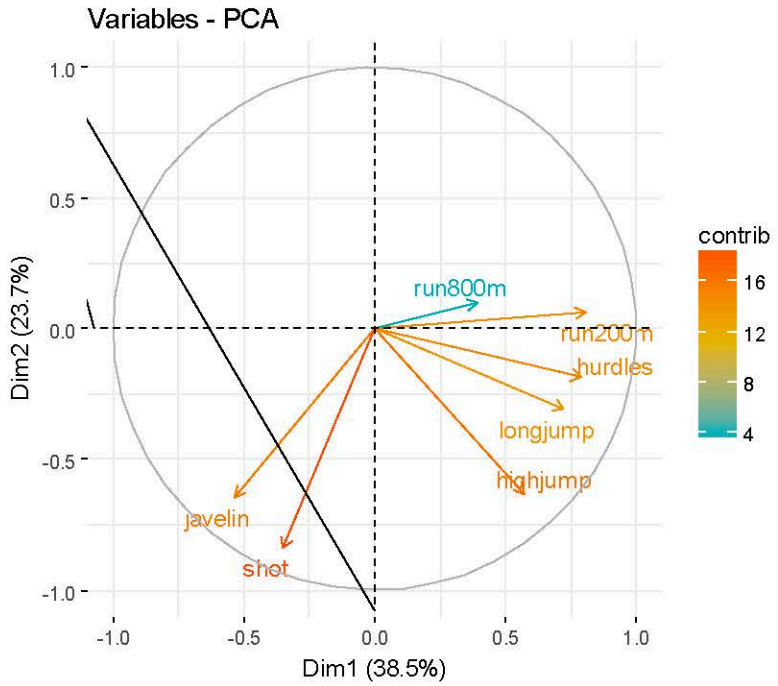
```
> install.packages("factoextra")
> get_eig(X_pca)
      eigenvalue variance.percent cumulative.variance.percent
Dim.1  2.6969102        38.527288                    38.52729
Dim.2  1.6576940        23.681342                    62.20863
Dim.3  1.0181963        14.545661                    76.75429
Dim.4  0.6008860         8.584086                    85.33838
Dim.5  0.4406781         6.295401                    91.63378
Dim.6  0.3523312         5.033303                    96.66708
Dim.7  0.2333042         3.332918                   100.00000
> fviz_eig(X_pca)
```



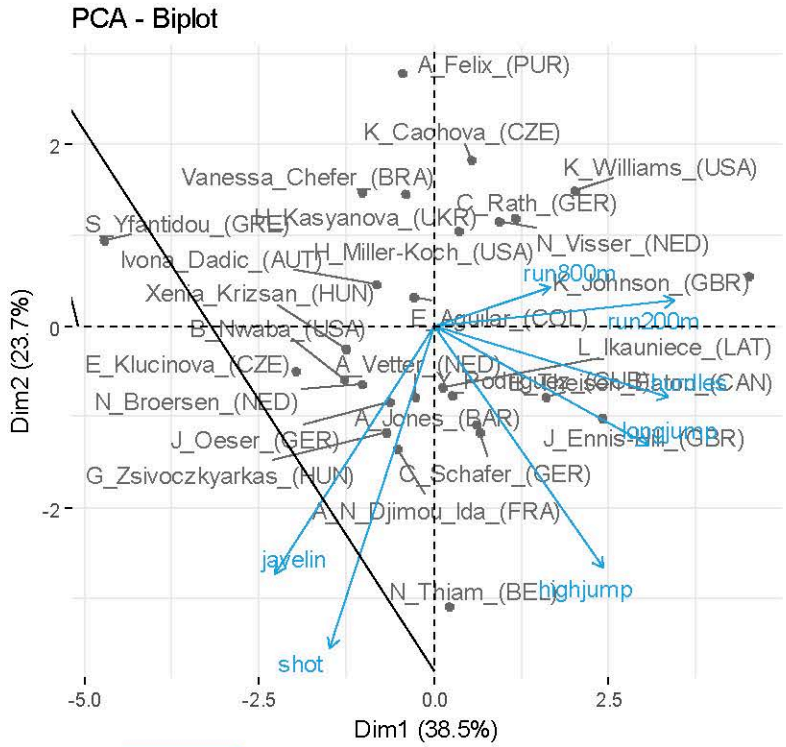

```
> fviz_pca_ind(X_pca,
+              col.ind = "cos2", # Color by the quality of representation
+              gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
+              repel = TRUE     # Avoid text overlapping
+ )
```




```
> fviz_pca_var(X_pca,
+              col.var = "contrib", # Color by contributions to the PC
+              gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
+              repel = TRUE     # Avoid text overlapping
+ )
```




```
> fviz_pca_biplot(X_pca, repel = TRUE,
+                 col.var = "#2E9FDF", # Variables color
+                 col.ind = "#696969"  # Individuals color
+ )
```

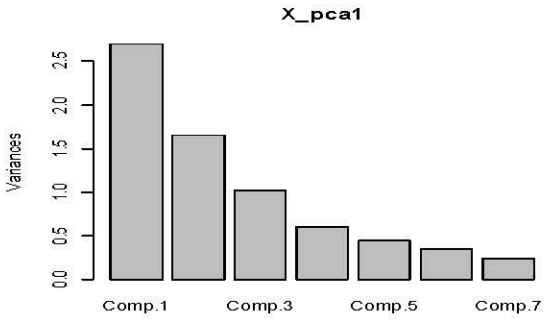


Vamos a utilizar también la librería *princomp* para realizar el ACP

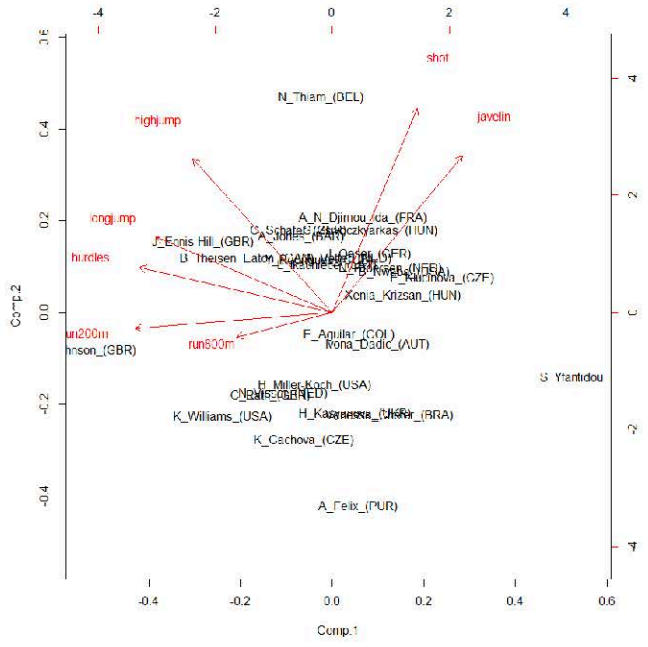
```
> X_pca1 <- princomp(X, cor=T)
> X_pca1

Call:
princomp(x = X, cor = T)
Standard deviations:
   Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7
1.6422272 1.2875146 1.0090571 0.7751684 0.6638359 0.5935750 0.4830158
 7  variables and  27 observations.

> summary(X_pca1, loadings=T)
Importance of components:
                          Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6
```

```
Standard deviation     1.6422272 1.2875146 1.0090571 0.77516840 0.66383589 0.59357497
Proportion of Variance 0.3852729 0.2368134 0.1454566 0.08584086 0.06295401 0.05033303
Cumulative Proportion  0.3852729 0.6220863 0.7675429 0.85338378 0.91633779 0.96667082
                           Comp.7
Standard deviation     0.48301577
Proportion of Variance 0.03332918
Cumulative Proportion  1.00000000
Loadings:
         Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7
hurdles  -0.482  0.143 -0.362  0.343         -0.387  0.588
highjump -0.349  0.489 -0.148 -0.323  0.560 -0.207 -0.395
shot      0.213  0.652                0.129  0.585  0.399
run200m  -0.493                0.613         0.436 -0.429
longjump -0.440  0.239  0.171 -0.483 -0.688  0.113
javelin   0.326  0.499  0.228  0.400 -0.331 -0.486 -0.296
run800m  -0.240         0.871         0.288 -0.163  0.259
```

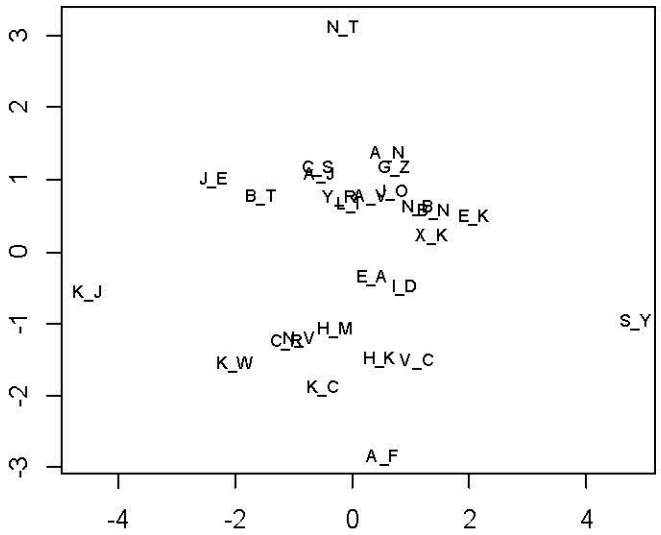
```
> plot(X_pca1)
```




```
> biplot(X_pca1)
```




```
> plot(X_pca1$scores[,1], X_pca1$scores[,2],
+ xlab="PC1",ylab="PC2",type="n",lwd=2)

> text(X_pca1$scores[,1],X_pca1$scores[,2],
+labels=abbreviate(row.names(X)),cex=0.7,lwd=2)
```



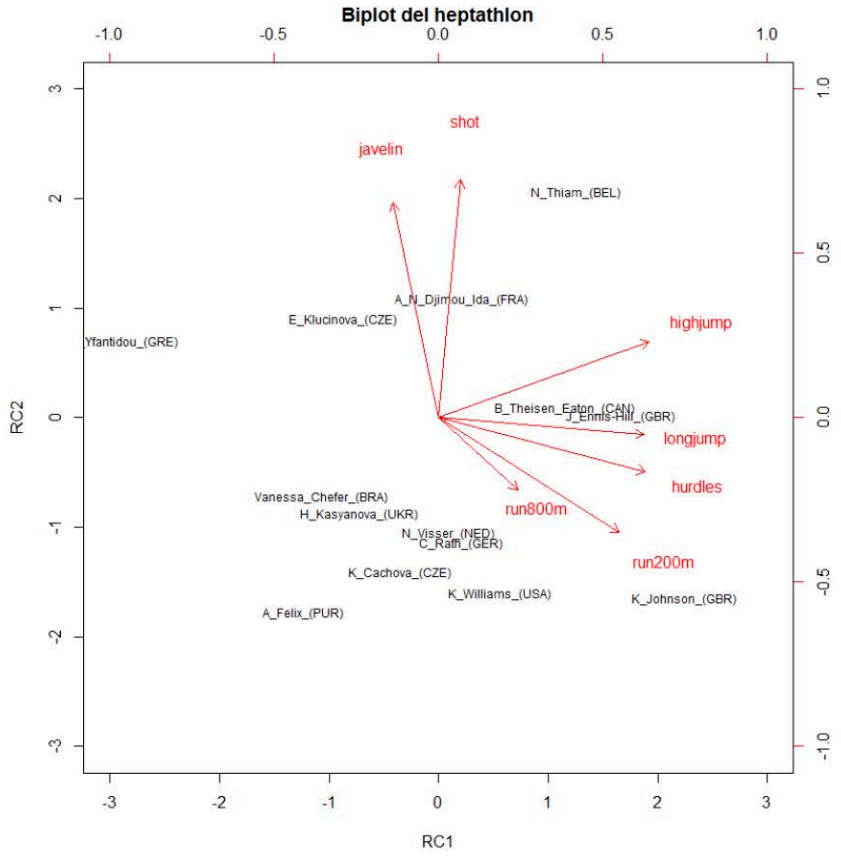
Vamos a utilizar la librería *principal y pca* del paquete *psych* para realizar el ACP

```
> library("psych", lib.loc="~/R/win-library/3.4")
> principal(Z)
> pca(Z)

Principal Components Analysis
Call: principal(r = Z)
Standardized loadings (pattern matrix) based upon correlation matrix
           PC1   h2   u2 com
hurdles   0.79 0.63 0.37   1
highjump  0.57 0.33 0.67   1
shot     -0.35 0.12 0.88   1
run200m   0.81 0.66 0.34   1
longjump  0.72 0.52 0.48   1
```

trabaja con la matriz de correlaciones, covarianzas o la matriz de datos, los vectores propios son reescalados por la raíz cuadrada de los valores propios: 0.79=0.48 x 1.6422272 donde 0.48 es la 1ª componente de hurdles en prcomp y 1.6422272 la desviac.típica. h2=factores comunes y u2=factores específicos

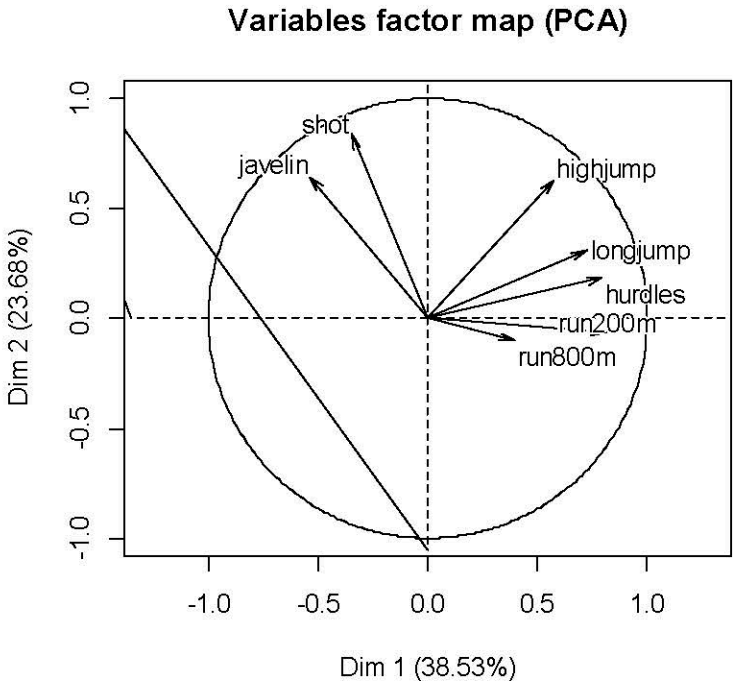
```
javelin  -0.53 0.28 0.72   1
run800m   0.39 0.15 0.85   1

                 PC1
SS loadings     2.69
Proportion Var  0.38

Mean item complexity =  1
Test of the hypothesis that 1 component is sufficient.

The root mean square of the residuals (RMSR) is  0.19

Fit based upon off diagonal values = 0.67
```

```
> biplot(pca(X,2,rotate="varimax"), labels=rownames(X))
```



Librería *PCA* del paquete *FactoMineR* para realizar el ACP

```
> PCA(X)
**Results for the Principal Component Analysis (PCA)**
The analysis was performed on 27 individuals, described by 7 variables
*The results are available in the following objects:

   name               description
1  "$eig"             "eigenvalues"
2  "$var"             "results for the variables"
3  "$var$coord"       "coord. for the variables"
4  "$var$cor"         "correlations variables - dimensions"
5  "$var$cos2"        "cos2 for the variables"
6  "$var$contrib"     "contributions of the variables"
7  "$ind"             "results for the individuals"
8  "$ind$coord"       "coord. for the individuals"
9  "$ind$cos2"        "cos2 for the individuals"
10 "$ind$contrib"     "contributions of the individuals"
11 "$call"            "summary statistics"
12 "$call$centre"     "mean of the variables"
13 "$call$ecart.type" "standard error of the variables"
14 "$call$row.w"      "weights for the individuals"
15 "$call$col.w"      "weights for the variables"
```




```
> summary(PCA(X))

Call:
PCA(X = X)


Eigenvalues
                       Dim.1   Dim.2   Dim.3   Dim.4   Dim.5   Dim.6   Dim.7
Variance               2.697   1.658   1.018   0.601   0.441   0.352   0.233
% of var.             38.527  23.681  14.546   8.584   6.295   5.033   3.333
Cumulative % of var.  38.527  62.209  76.754  85.338  91.634  96.667 100.000

Individuals (the 10 first)
                        Dist    Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3
N_Thiam_(BEL)          3.534 |  0.230  0.073  0.004 |  3.144 22.081  0.791 |  0.277
J_Ennis-Hill_(GBR)     2.877 |  2.441  8.182  0.720 |  1.028  2.360  0.128 |  0.310
B_Theisen_Eaton_(CAN)  2.011 |  1.619  3.600  0.648 |  0.791  1.399  0.155 |  0.693
L_Ikauniece_(LAT)      2.615 |  0.127  0.022  0.002 |  0.689  1.061  0.069 |  1.172
C_Schafer_(GER)        1.774 |  0.663  0.604  0.140 |  1.195  3.189  0.453 | -0.280
K_Johnson_(GBR)        4.787 |  4.592 28.962  0.920 | -0.556  0.691  0.013 | -0.634
Y_Rodriguez_(CUB)      1.146 |  0.272  0.102  0.056 |  0.772  1.330  0.454 |  0.347
G_Zsivoczkyarkas_(HUN) 2.066 | -0.686  0.645  0.110 |  1.192  3.174  0.333 |  0.822
J_Oeser_(GER)          1.449 | -0.653  0.586  0.203 |  0.851  1.618  0.345 |  0.309
A_Vetter_(NED)         1.765 | -0.278  0.106  0.025 |  0.791  1.398  0.201 | -0.035
                         ctr   cos2
N_Thiam_(BEL)          0.279  0.006 |
J_Ennis-Hill_(GBR)     0.349  0.012 |
B_Theisen_Eaton_(CAN)  1.748  0.119 |
L_Ikauniece_(LAT)      4.994  0.201 |
C_Schafer_(GER)        0.284  0.025 |
K_Johnson_(GBR)        1.462  0.018 |
Y_Rodriguez_(CUB)      0.439  0.092 |
G_Zsivoczkyarkas_(HUN) 2.459  0.158 |
J_Oeser_(GER)          0.347  0.045 |
A_Vetter_(NED)         0.004  0.000 |

Variables
                        Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
hurdles                 0.792 23.269  0.628 |  0.184  2.050  0.034 | -0.365 13.085
highjump                0.573 12.184  0.329 |  0.630 23.928  0.397 | -0.149  2.186
shot                   -0.349  4.523  0.122 |  0.839 42.474  0.704 |  0.064  0.396
run200m                 0.810 24.326  0.656 | -0.065  0.256  0.004 |  0.060  0.358
longjump                0.722 19.346  0.522 |  0.308  5.727  0.095 |  0.173  2.929
javelin                -0.535 10.597  0.286 |  0.643 24.942  0.413 |  0.230  5.200
run800m                 0.394  5.755  0.155 | -0.102  0.623  0.010 |  0.879 75.845
                        cos2
hurdles                0.133 |
highjump               0.022 |
shot                   0.004 |
run200m                0.004 |
longjump               0.030 |
javelin                0.053 |
run800m                0.772 |
```

```
> plot(PCA(X))
```

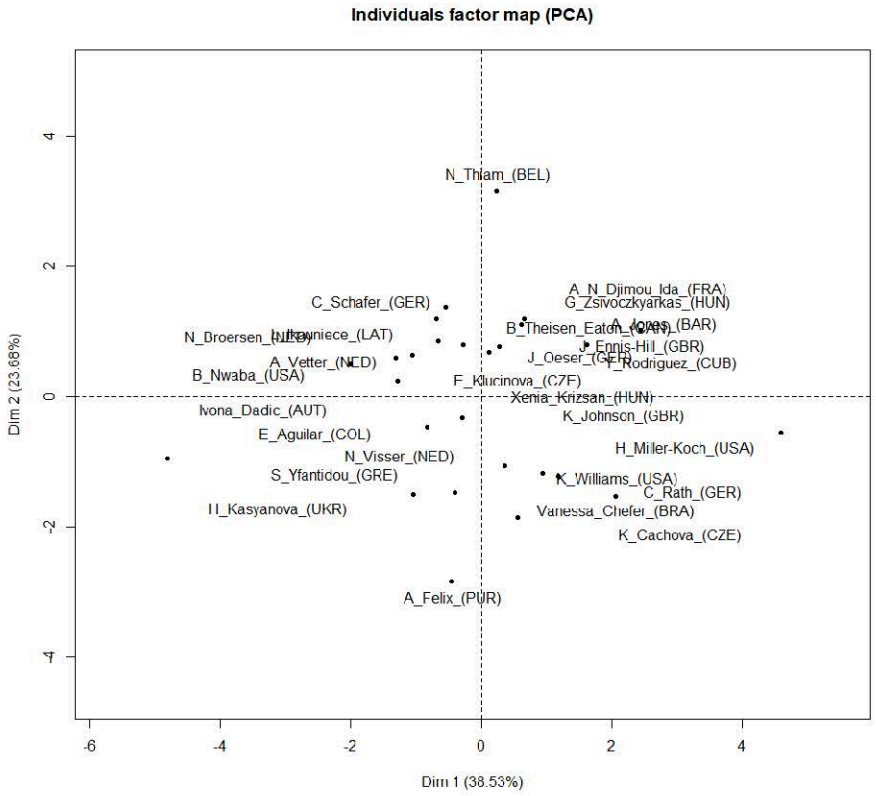


Gráfico de correlaciones

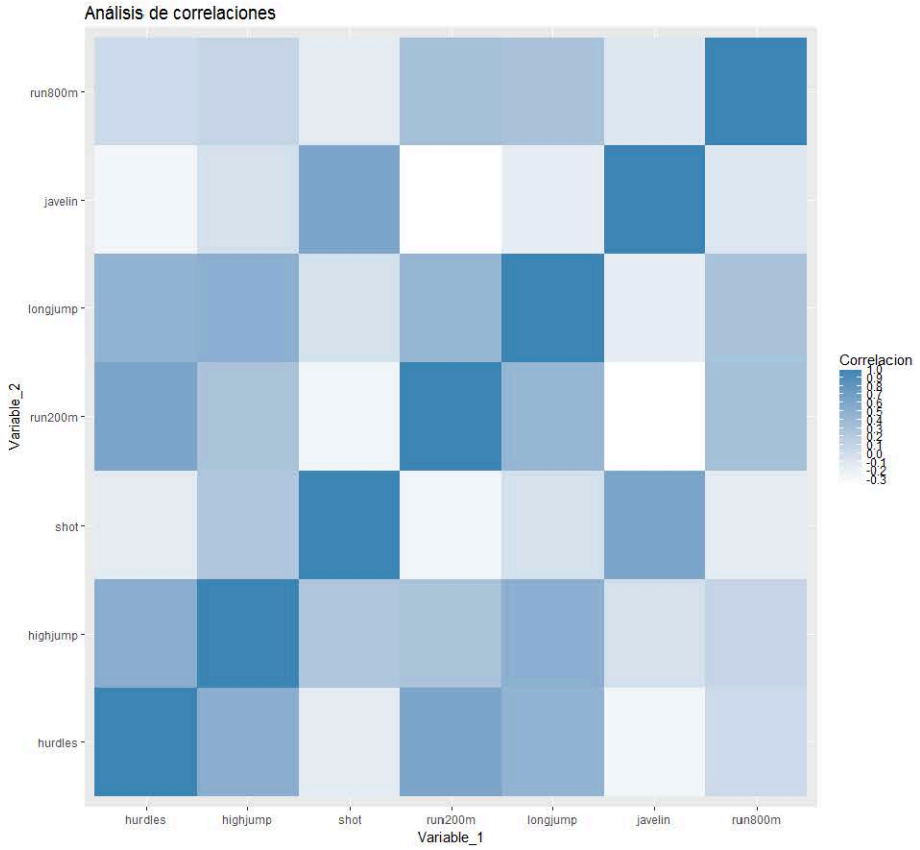
```
> library(ggplot2)
> library(reshape2)
> datoscor <- round((Z),2)
> datoscor
         hurdles highjump  shot run200m longjump javelin run800m
hurdles     1.00     0.52 -0.15    0.62     0.46   -0.27    0.03
highjump    0.52     1.00  0.25    0.29     0.50   -0.03    0.08
shot       -0.15     0.25  1.00   -0.25    -0.03    0.62   -0.16
run200m     0.62     0.29 -0.25    1.00     0.43   -0.35    0.33
longjump    0.46     0.50 -0.03    0.43     1.00   -0.18    0.30
javelin    -0.27    -0.03  0.62   -0.35    -0.18    1.00   -0.10
run800m     0.03     0.08 -0.16    0.33     0.30   -0.10    1.00
> datos.lista <- melt(datoscor)
> datos.lista
       Var1     Var2 value
1   hurdles  hurdles  1.00
2  highjump  hurdles  0.52
3      shot  hurdles -0.15
4   run200m  hurdles  0.62
5  longjump  hurdles  0.46
6   javelin  hurdles -0.27
7   run800m  hurdles  0.05
8   hurdles highjump  0.52
9  highjump highjump  1.00
10     shot highjump  0.25
11  run200m highjump  0.29
12 longjump highjump  0.50
13  javelin highjump -0.03
14  run800m highjump  0.13
15  hurdles     shot -0.15
16 highjump     shot  0.25
17     shot     shot  1.00
18  run200m     shot -0.25
19 longjump     shot -0.03
20  javelin     shot  0.62
21  run800m     shot -0.13
22  hurdles  run200m  0.62
23 highjump  run200m  0.29
```

```
24     shot  run200m -0.25
25  run200m  run200m  1.00
26 longjump  run200m  0.43
27  javelin  run200m -0.35
28  run800m  run200m  0.34
29  hurdles longjump  0.46
30 highjump longjump  0.50
31     shot longjump -0.03
32  run200m longjump  0.43
33 longjump longjump  1.00
34  javelin longjump -0.18
35  run800m longjump  0.23
36  hurdles  javelin -0.27
37 highjump  javelin -0.03
38     shot  javelin  0.62
39  run200m  javelin -0.35
40 longjump  javelin -0.18
41  javelin  javelin  1.00
42  run800m  javelin -0.03
43  hurdles  run800m  0.05
44 highjump  run800m  0.13
45     shot  run800m -0.13
46  run200m  run800m  0.34
47 longjump  run800m  0.23
48  javelin  run800m -0.03
49  run800m  run800m  1.00
```

```
> names(datos.lista) <- c("Variable_1","Variable_2","Correlacion")
> names(datos.lista)
[1] "Variable_1"  "Variable_2"  "Correlacion"
```

```
> escala <- seq(-1,1,0.1)
> escala
 [1] -1.0 -0.9 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2 -0.1  0.0  0.1  0.2  0.3
[15]  0.4  0.5  0.6  0.7  0.8  0.9  1.0
```

```
> (p <- ggplot(datos.lista, aes(Variable_1, Variable_2, fill=Correlacion))
+   + geom_tile(aes(fill=Correlacion))
+   + scale_fill_continuous(low = "white", high = "steelblue" ,breaks=escala)
+   + labs(title="Análisis de correlaciones",
+          plot.title = element_text(face="bold", size=14)))
```



Conclusiones

Atletas

Los dos primeros ejes representan el 62% de la variabilidad y los tres primeros el 77% por lo tanto con la representación bidimensional no tenemos la explicación completa del comportamiento de los atletas.

Los atletas están ordenados según los 2 primeros ejes en el siguiente sentido, cuanto más a la derecha del origen y hacia abajo (cuarto cuadrante), mejor clasificación y cuanto más a la izquierda y hacia arriba (segundo cuadrante), peor clasificación. Los primeros puestos los obtienen las atletas N_Thiam (BEL), J_Ennis-Hill (GBR), B_Theisen Eaton (CAN), L_Ikauniece (LAT), C_Schafer (GER), situadas en el cuarto.

En cambio S_Yfantidou (GRE), A_Felix (PUR), H_Kasyanova (UKR), K_Cachova (CZE), Vanessa_Chefer(BRA) que ocupan los últimos puestos de la clasificación se encuentran en el segundo cuadrante y algunas cerca del primero. Las atletas E_Aguilar y Xenia_Frizsan que están entorno al origen se encuentran en puestos intermedios (15 y 16).

Pruebas

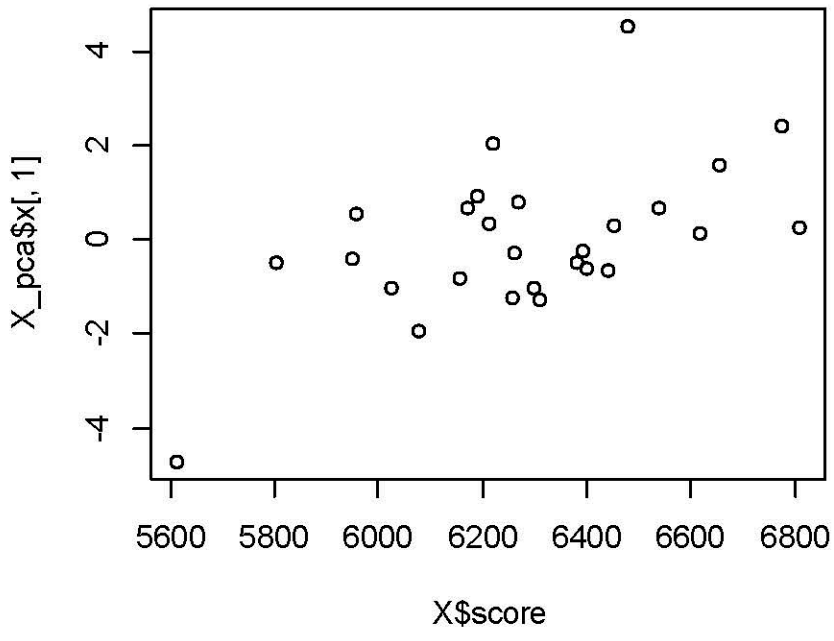
Todas las pruebas se dirigen desde el origen hacia la derecha hacia abajo que es donde están los atletas mejor clasificados, excepto las pruebas de jabalina y peso que se van hacia la izquierda y hacia abajo por estar cerca de la ganadora, N_Thiam (BEL) con buenas clasificaciones en ambas pruebas.

En general las carreras se encuentran correlacionadas positivamente y negativamente con el lanzamiento de javalina y peso.

Atletas y pruebas

En general las pruebas van dirigidas hacia los atletas con mejor puntuación en ellas, por ejemplo 100 metros vallas se dirige hacia Ennis-Hill (GBR), 200 m apunta a K_Johnson-Thompson, salto de altura y lanzamiento de peso hacia la ganadora N_Thiam (BEL), 800 m se dirige hacia C_Rath_(GER), salto de longitud y altura se sitúan en medio de K_Johnson_(GBR) y N_Thiam (BEL).

```
> X <- read.table("2acp-16.txt" ,header=T) #hay que volver a leer los datos para tomar
##los valores de score

> X <- replace(X, X==0, NA)

> X <- subset(X,complete.cases(X))

> cor(X$score, X_pca$x[,1])
[1] 0.5594179

> plot(X$score, X_pca$x[,1])
```



2º parte de la Práctica (ACP, paso a paso)

```
> X <- read.table("2acp-16.txt" ,header=T)

> X <- replace(X, X==0, NA)
> X <- subset(X,complete.cases(X))

> X$hurdles <- max(X$hurdles) -X$hurdles
> X$run200m <- max(X$run200m) -X$run200m
> X$run800m <- max(X$run800m) -X$run800m

> X <- X[,-8]
> X
```

	hurdles	highjump	shot	run200m	longjump	javelin	run800m
N_Thiam_(BEL)	0.68	1.98	14.91	1.22	6.58	53.13	24.58
J_Ennis-Hill_(GBR)	1.40	1.89	13.86	2.83	6.34	46.06	32.05
B_Theisen_Eaton_(CAN)	1.06	1.86	13.45	2.14	6.48	47.36	31.62
L_Ikauniece_(LAT)	0.91	1.77	13.52	2.56	6.12	55.93	31.69
C_Schafer_(GER)	1.12	1.83	14.57	2.33	6.20	47.99	24.60
K_Johnson_(GBR)	1.59	1.98	11.68	3.06	6.51	36.36	30.65
Y_Rodriguez_(CUB)	0.63	1.86	13.69	2.06	6.25	48.89	26.47
G_Zsivoczkyarkas_(HUN)	0.45	1.86	14.39	0.94	6.31	48.07	29.36
J_Oeser_(GER)	0.55	1.86	14.28	1.33	6.10	47.22	27.30
A_Vetter_(NED)	0.77	1.77	14.78	2.39	6.10	48.42	23.41
A_N_Djimou_Ida_(FRA)	0.87	1.77	14.88	1.25	6.43	48.76	20.76
B_Nwaba_(USA)	0.43	1.83	14.81	1.55	5.81	46.85	29.51
N_Broersen_(NED)	0.68	1.77	14.04	1.38	6.15	50.80	23.57
C_Rath_(GER)	0.61	1.74	12.83	1.84	6.55	39.39	33.90
E_Aguilar_(COL)	0.40	1.74	13.60	2.20	6.23	46.90	26.80
Xenia_Krizsan_(HUN)	0.58	1.77	13.78	1.08	6.08	49.78	27.66
K_Williams_(USA)	1.20	1.83	11.21	2.23	6.31	40.93	24.88
H_Miller-Koch_(USA)	0.68	1.80	12.91	1.35	6.16	40.25	34.30
N_Visser_(NED)	1.22	1.68	12.84	1.98	6.35	42.48	26.65
A_Jones_(BAR)	1.24	1.89	14.09	1.97	6.30	42.00	0.00
Ivona_Dadic_(AUT)	0.40	1.77	13.43	1.72	6.05	46.08	25.48
E_Klucinova_(CZE)	0.17	1.80	14.41	0.95	6.08	46.73	18.31
Vanessa_Chefer_(BRA)	0.00	1.68	13.06	2.21	6.10	45.05	26.92
K_Cachova_(CZE)	1.05	1.77	12.38	2.00	5.91	37.77	22.17
H_Kasyanova_(UKR)	0.58	1.77	13.25	1.72	5.88	38.10	24.54
A_Felix_(PUR)	0.17	1.68	11.36	1.58	6.22	40.17	25.80
S_Yfantidou_(GRE)	0.25	1.65	12.97	0.00	5.51	54.57	11.04

```
> Z=round(cor(X),2)
> Z
```

	hurdles	highjump	shot	run200m	longjump	javelin	run800m
hurdles	1.00	0.52	-0.15	0.62	0.46	-0.27	0.03
highjump	0.52	1.00	0.25	0.29	0.50	-0.03	0.08
shot	-0.15	0.25	1.00	-0.25	-0.03	0.62	-0.16
run200m	0.62	0.29	-0.25	1.00	0.43	-0.35	0.33
longjump	0.46	0.50	-0.03	0.43	1.00	-0.18	0.30
javelin	-0.27	-0.03	0.62	-0.35	-0.18	1.00	-0.10
run800m	0.03	0.08	-0.16	0.33	0.30	-0.10	1.00

```
##Cálculo de la matriz de correlaciones, el cálculo de sd se hace dividiendo por n-1
 > X <- as.matrix(X)

 > Xc <- sweep(X,2, apply(X,2,mean), FUN="-")#centramos la matriz

 > Xd <- sweep(Xc,2, apply(X,2,sd), FUN="/")#tipificamos las variables
 > n <- dim(X)[1]
 > Y <- Xd*(1/sqrt(n))
```

$$\mathbb{R} = Y' \cdot Y = \left(\frac{r_{ij} - \bar{r}_j}{s_j\sqrt{n}}\right)' \left(\frac{r_{ij} - \bar{r}_j}{s_j\sqrt{n}}\right) \qquad s_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}(r_{ij} - \bar{r}_j)^2 \text{ en R, por lo tanto } \mathbb{R} = Y' \cdot Y \cdot \frac{n}{n-1}$$

Ya que el denominador de s_j^2 es $n-1$ y s_j está en el denominador de Y, el término $n-1$ sube al numerador y para contrarrestar ese valor y cambiarlo por n, tendríamos que multiplicar por $\frac{n}{n-1}$.

```
> Z <- (t(Y)%*%Y)*n/(n-1) #matriz de correlaciones
> round(Z,2)
```

	hurdles	highjump	shot	run200m	longjump	javelin	run800m
hurdles	1.00	0.52	-0.15	0.62	0.46	-0.27	0.03
highjump	0.52	1.00	0.25	0.29	0.50	-0.03	0.08
shot	-0.15	0.25	1.00	-0.25	-0.03	0.62	-0.16

```
run200m     0.62     0.29 -0.25    1.00     0.43   -0.35    0.33
longjump    0.46     0.50 -0.03    0.43     1.00   -0.18    0.30
javelin    -0.27    -0.03  0.62   -0.35    -0.18    1.00   -0.10
run800m     0.03     0.08 -0.16    0.33     0.30   -0.10    1.00
```

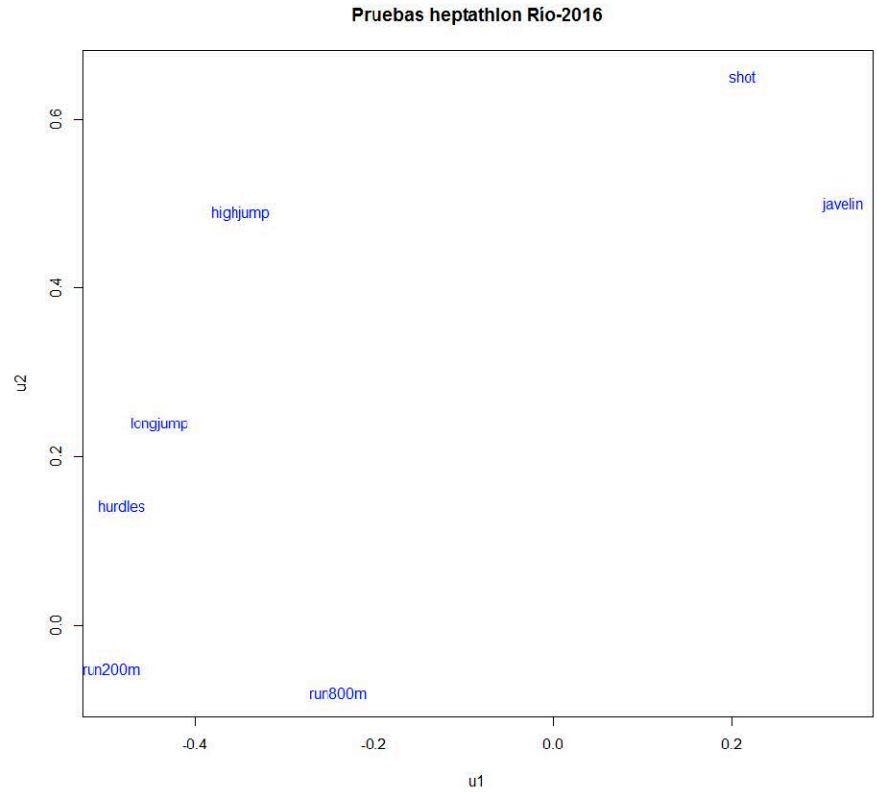
```
> svd(Z)
$d
[1] 2.6969102 1.6576940 1.0181963 0.6008860 0.4406781 0.3523312 0.2333042

$u
           [,1]        [,2]        [,3]        [,4]        [,5]        [,6]
[1,] -0.4823786  0.14319190 -0.36173020  0.34266244 -0.05182669 -0.3868867
[2,] -0.3490533  0.48915795 -0.14785050 -0.32317736  0.56007084 -0.2072972
[3,]  0.2126808  0.65171727  0.06296514  0.08932881  0.12918701  0.5852192
[4,] -0.4932162 -0.05060990  0.05984647  0.61295457 -0.03493213  0.4355074
[5,] -0.4398372  0.23931968  0.17113888 -0.48329701 -0.68795159  0.1130882
[6,]  0.3255236  0.49942083  0.22804117  0.40002830 -0.33059101 -0.4855649
[7,] -0.2399060 -0.07892245  0.87089264  0.02910766  0.28837009 -0.1632056
           [,7]
[1,] -0.58836496
[2,]  0.39488155
[3,] -0.39864318
[4,]  0.42894581
[5,]  0.01839809
[6,]  0.29584479
[7,] -0.25907858

$v
           [,1]        [,2]        [,3]        [,4]        [,5]        [,6]
[1,] -0.4823786  0.14319190 -0.36173020  0.34266244 -0.05182669 -0.3868867
[2,] -0.3490533  0.48915795 -0.14785050 -0.32317736  0.56007084 -0.2072972
[3,]  0.2126808  0.65171727  0.06296514  0.08932881  0.12918701  0.5852192
[4,] -0.4932162 -0.05060990  0.05984647  0.61295457 -0.03493213  0.4355074
[5,] -0.4398372  0.23931968  0.17113888 -0.48329701 -0.68795159  0.1130882
[6,]  0.3255236  0.49942083  0.22804117  0.40002830 -0.33059101 -0.4855649
[7,] -0.2399060 -0.07892245  0.87089264  0.02910766  0.28837009 -0.1632056
           [,7]
[1,] -0.58836496
[2,]  0.39488155
[3,] -0.39864318
[4,]  0.42894581
[5,]  0.01839809
[6,]  0.29584479
[7,] -0.25907858
```

```
> eigen(Z)
eigen() decomposition
$values
[1] 2.6969102 1.6576940 1.0181963 0.6008860 0.4406781 0.3523312 0.2333042

$vectors
           [,1]        [,2]        [,3]        [,4]        [,5]        [,6]        [,7]
[1,] -0.4823786 -0.14319190 -0.36173020  0.34266244 -0.05182669  0.3868867  0.58836496
[2,] -0.3490533 -0.48915795 -0.14785050 -0.32317736  0.56007084  0.2072972 -0.39488155
[3,]  0.2126808 -0.65171727  0.06296514  0.08932881  0.12918701 -0.5852192  0.39864318
[4,] -0.4932162  0.05060990  0.05984647  0.61295457 -0.03493213 -0.4355074 -0.42894581
[5,] -0.4398372 -0.23931968  0.17113888 -0.48329701 -0.68795159 -0.1130882 -0.01839809
[6,]  0.3255236 -0.49942083  0.22804117  0.40002830 -0.33059101  0.4855649 -0.29584479
[7,] -0.2399060  0.07892245  0.87089264  0.02910766  0.28837009  0.1632056  0.25907858
```

```
> u1 <- svd(Z)$u[,1] ##primera componente de las pruebas
> u1
[1] -0.4823786 -0.3490533  0.2126808 -0.4932162 -0.4398372  0.3255236
[7] -0.2399060
```

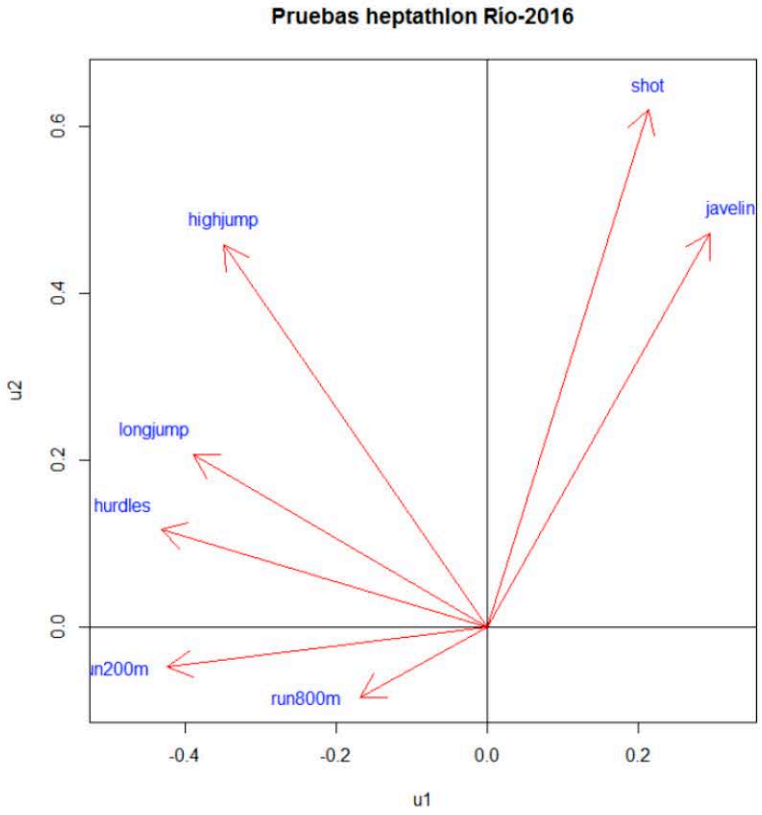
```
> u2 <- svd(Z)$u[,2] ##segunda componente de las pruebas
> u2
[1]  0.14319190  0.48915795  0.65171727 -0.05060990  0.23931968  0.49942083
[7] -0.07892245
```

```
> plot(u1,u2,col = "blue", type='n')
> text(u1,u2, col='blue', labels=colnames(X), title(main='Pruebas heptathlon Río-
2016'))
```



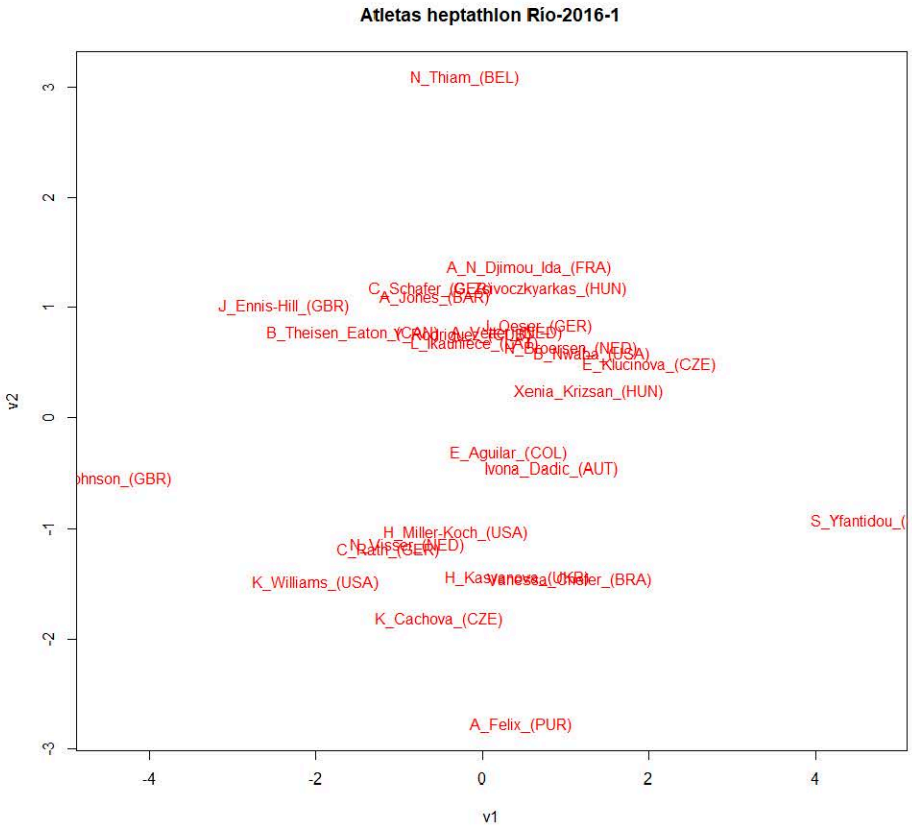
Podemos poner las variables como vectores, según el gráfico biplot, con las órdenes:

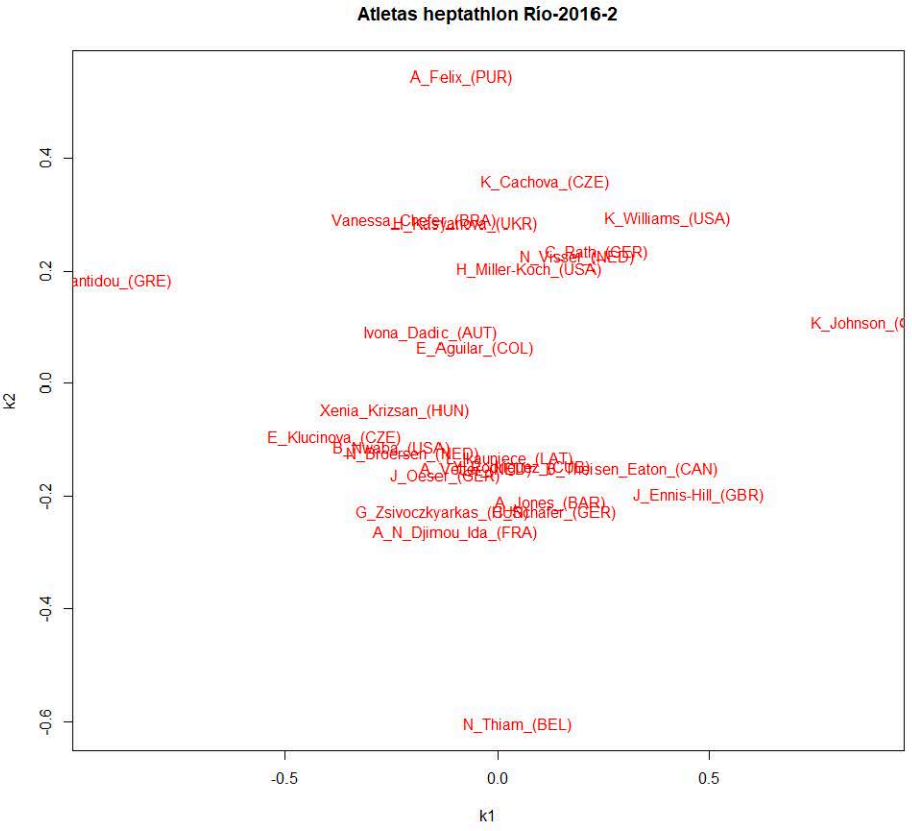
```
> abline(h=0)
> abline(v=0)

> arrows(0,0,u1[1]+0.05,u2[1]-0.03, col = "red")
> arrows(0,0,u1[2],u2[2]-0.03, col = "red")
> arrows(0,0,u1[3],u2[3]-0.03, col = "red")
> arrows(0,0,u1[4]+0.07,u2[4], col = "red")
> arrows(0,0,u1[5]+0.05,u2[5]-0.03, col = "red")
> arrows(0,0,u1[6]-0.03,u2[6]-0.03, col = "red")
> arrows(0,0,u1[7]+0.07,u2[7], col = "red")
```



```
> h1 <- Xd%*%u1 ##coordenadas de las atletas
> h1
                              [,1]
 N_Thiam_(BEL)           -0.2260969
 J_Ennis-Hill_(GBR)      -2.3951904
 B_Theisen_Eaton_(CAN)   -1.5887567
 L_Ikauniece_(LAT)       -0.1244757
 C_Schafer_(GER)         -0.6505642
 K_Johnson_(GBR)         -4.5064455
 Y_Rodriguez_(CUB)       -0.2672145
 G_Zsivoczkyarkas_(HUN)   0.6727212
 J_Oeser_(GER)            0.6408868
 A_Vetter_(NED)           0.2723651
 A_N_Djimou_Ida_(FRA)     0.5320837
 B_Nwaba_(USA)            1.2915869
 N_Broersen_(NED)         1.0395230
 C_Rath_(GER)            -1.1589018
 E_Aguilar_(COL)          0.2899863
 Xenia_Krizsan_(HUN)      1.2562643
 K_Williams_(USA)        -2.0248050
 H_Miller-Koch_(USA)     -0.3508171
 N_Visser_(NED)          -0.9281708
 A_Jones_(BAR)           -0.6039843
 Ivona_Dadic_(AUT)        0.8137975
 E_Klucinova_(CZE)        1.9730021
 Vanessa_Chefer_(BRA)     1.0217630
 K_Cachova_(CZE)         -0.5486902
 H_Kasyanova_(UKR)        0.3998344
 A_Felix_(PUR)            0.4474626
 S_Yfantidou_(GRE)        4.7228361

> h2 <- Xd%*%u2 ##coordenadas de las atletas
> h2
                              [,1]
 N_Thiam_(BEL)            3.0849320
 J_Ennis-Hill_(GBR)       1.0086259
 B_Theisen_Eaton_(CAN)    0.7765884
 L_Ikauniece_(LAT)        0.6761041
 C_Schafer_(GER)          1.1724130
 K_Johnson_(GBR)         -0.5455710
 Y_Rodriguez_(CUB)        0.7572118
 G_Zsivoczkyarkas_(HUN)   1.1696640
 J_Oeser_(GER)            0.8351378
 A_Vetter_(NED)           0.7762536
 A_N_Djimou_Ida_(FRA)     1.3568810
 B_Nwaba_(USA)            0.5811524
 N_Broersen_(NED)         0.6348761
 C_Rath_(GER)            -1.1928045
 E_Aguilar_(COL)         -0.3189258
 Xenia_Krizsan_(HUN)      0.2427588
 K_Williams_(USA)        -1.4944914
 H_Miller-Koch_(USA)     -1.0372801
 N_Visser_(NED)          -1.1533669
 A_Jones_(BAR)            1.0862362
 Ivona_Dadic_(AUT)       -0.4578808
 E_Klucinova_(CZE)        0.4896377
 Vanessa_Chefer_(BRA)    -1.4697058
 K_Cachova_(CZE)         -1.8246252
 H_Kasyanova_(UKR)       -1.4451542
 A_Felix_(PUR)           -2.7770334
 S_Yfantidou_(GRE)       -0.9316335

> plot(h1,h2,col = "red", type='n')
> text(h1,h2, col='red', labels=rownames(X), title(main='Atletas heptathlon Río-2016'-
1))
####otra forma
> k1 <- svd(Y*(sqrt(n)/sqrt(n-1)))$u[,1]*svd(Y*(sqrt(n)/sqrt(n-1)))$d[1]
> k1 ##coordenadas de las filas, raíz de lamda subalpha por v subalpha
 [1]  0.04434124  0.46973548  0.31158082  0.02441170  0.12758613  0.88378667  0.05240507
 [8] -0.13193148 -0.12568825 -0.05341519 -0.10435019 -0.25330103 -0.20386723  0.22727935
[15] -0.05687099 -0.24637370  0.39709693  0.06880089  0.18202927  0.11845106 -0.15959882
[22] -0.38693754 -0.20038421  0.10760699 -0.07841397 -0.08775464 -0.92622437
> k2 <- svd(Y*(sqrt(n)/sqrt(n-1)))$u[,2]*svd(Y*(sqrt(n)/sqrt(n-1)))$d[2]
> k2
 [1] -0.60500494 -0.19780781 -0.15230153 -0.13259492 -0.22992910  0.10699527 -0.14850144
```

```
 [8] -0.22938998 -0.16378399 -0.15223586 -0.26610626 -0.11397336 -0.12450944  0.23392821
[15]  0.06254650 -0.04760891  0.29309387  0.20342737  0.22619385 -0.21302845  0.08979782
[22] -0.09602586  0.28823301  0.35783845  0.28341806  0.54462105  0.18270837
> plot(k1,k2,col = "red", type='n')
> text(k1,k2, col='red', labels=rownames(X), title(main='Atletas heptathlon Río-2016-2'))
```






```
> X1 <- Y*(sqrt(n)/sqrt(n-1)) #otra forma de representar a las atletas
> svd(X1)
$d
[1] 1.6422272 1.2875146 1.0090571 0.7751684 0.6638359 0.5935750 0.4830158
```

```
$u
             [,1]        [,2]         [,3]        [,4]        [,5]         [,6]
[1,]  0.02700068 -0.46990141 -0.052819864  0.34449571  0.06045064 -0.156529480
[2,]  0.28603562 -0.15363539 -0.059038676 -0.24787747 -0.09769040 -0.028276150
[3,]  0.18973064 -0.11829110 -0.132205823  0.01419378  0.10297368 -0.138937585
[4,]  0.01486499 -0.10298518 -0.223463360 -0.49698848  0.13669058 -0.236887322
[5,]  0.07769091 -0.17858367  0.053320514 -0.24646313 -0.01512375  0.108434850
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
$v
           [,1]        [,2]        [,3]        [,4]        [,5]       [,6]        [,7]
[1,]  0.4823786 -0.14319190  0.36173020 -0.34266244  0.05182669 -0.3868867  0.58836496
[2,]  0.3490533 -0.48915795  0.14785050  0.32317736 -0.56007084 -0.2072972 -0.39488155
[3,] -0.2126808 -0.65171727 -0.06296514 -0.08932881 -0.12918701  0.5852192  0.39864318
[4,]  0.4932162  0.05060990 -0.05984647 -0.61295457  0.03493213  0.4355074 -0.42894581
[5,]  0.4398372 -0.23931968 -0.17113888  0.48329701  0.68795159  0.1130882 -0.01839809
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
> h1 <- Xd%*%svd(X1)$v[,1]
> h1
                              [,1]
N_Thiam_(BEL)            0.2260969
J_Ennis-Hill_(GBR)       2.3951904
B_Theisen_Eaton_(CAN)    1.5887567
L_Ikauniece_(LAT)        0.1244757
C_Schafer_(GER)          0.6505642
K_Johnson_(GBR)          4.5064455
Y_Rodriguez_(CUB)        0.2672145
G_Zsivoczkyarkas_(HUN)  -0.6727212
J_Oeser_(GER)           -0.6408868
--------------------------------
--------------------------------
> h2 <- Xd%*%svd(X1)$v[,2]
> h2
                              [,1]
N_Thiam_(BEL)           -3.0849320
J_Ennis-Hill_(GBR)      -1.0086259
B_Theisen_Eaton_(CAN)   -0.7765884
L_Ikauniece_(LAT)       -0.6761041
C_Schafer_(GER)         -1.1724130
K_Johnson_(GBR)          0.5455710
Y_Rodriguez_(CUB)       -0.7572118
G_Zsivoczkyarkas_(HUN)  -1.1696640
J_Oeser_(GER)           -0.8351378

> plot(h1,h2,col = "red", type='n')
> text(h1,h2, col='red', labels=rownames(X), title(main='Atletas heptathlon Río-2016-
3'))
```