A4-Componentes Principales

Ricardo Salinas

2024-10-08

Analisis Descriptivo

```
#Primero se realiza un análisis descriptivo para conocer las variables.
Incluye las medidas que vienen en el summary() y la desviación estándar.
Describe las correlaciones que se establecen entre las variables.

print("Estos datos contienen informacion sobre dimensiones principales del
cuerpo, los cuales se pueden utilizar para medir diferentes aspectos del
cuerpo, contiene 6 columnas con diferentes medidas.")

## [1] "Estos datos contienen informacion sobre dimensiones principales del
cuerpo, los cuales se pueden utilizar para medir diferentes aspectos del
cuerpo, contiene 6 columnas con diferentes medidas."

corporal = read.csv("corporal.csv")
corporal1 = corporal[, -which(names(corporal) == "sexo")]
print(corporal)

##    edad peso altura   sexo muneca biceps
## 1    43 87.3  188.0 Hombre   12.2   35.8
## 2    65 80.0  174.0 Hombre   12.0   35.0
## 3    45 82.3  176.5 Hombre   11.2   38.5
## 4    37 73.6  180.3 Hombre   11.2   32.2
## 5    55 74.1  167.6 Hombre   11.8   32.9
## 6    33 85.9  188.0 Hombre   12.4   38.5
## 7    25 73.2  180.3 Hombre   10.6   38.3
## 8    35 76.3  167.6 Hombre   11.3   35.0
## 9    28 65.9  183.0 Hombre   10.2   32.1
## 10   26 90.9  183.0 Hombre   12.0   40.4
## 11   43 89.1  179.1 Hombre   11.3   36.5
## 12   30 62.3  170.2 Hombre   11.5   34.2
## 13   26 82.7  177.8 Hombre   11.5   35.2
## 14   51 79.1  179.1 Hombre   11.8   34.0
## 15   30 98.2  190.5 Hombre   10.7   34.8
## 16   24 84.1  177.8 Hombre   11.5   38.6
## 17   35 83.2  180.3 Hombre   11.1   36.4
## 18   37 83.2  180.3 Hombre   10.5   34.0
## 19   22 51.6  161.2  Mujer    9.2   24.3
## 20   20 59.0  167.5  Mujer    9.9   27.8
## 21   19 49.2  159.5  Mujer    8.9   24.0
## 22   25 63.0  157.0  Mujer    9.5   28.0
## 23   21 53.6  155.8  Mujer    9.1   26.9
```

```
## 24   23 59.0  170.0  Mujer   10.0   26.5
## 25   26 47.6  159.1  Mujer    9.4   24.1
## 26   22 69.8  166.0  Mujer   10.7   29.2
## 27   28 66.8  176.2  Mujer    9.8   29.0
## 28   40 75.2  160.2  Mujer   11.5   33.6
## 29   32 55.2  172.5  Mujer    8.6   24.8
## 30   25 54.2  170.9  Mujer    9.7   25.4
## 31   25 62.5  172.9  Mujer    9.2   25.9
## 32   29 42.0  153.4  Mujer    8.3   24.0
## 33   22 50.0  160.0  Mujer    8.6   25.6
## 34   25 49.8  147.2  Mujer    9.0   26.0
## 35   23 49.2  168.2  Mujer    9.6   23.5
## 36   37 73.2  175.0  Mujer   11.0   31.0
```

```
n = ncol(corporal1)
d = matrix(NA, ncol=7, nrow=n)
colnames(d) = c("Min", "Q1", "Median", "Mean", "Q3", "Max", "SD")

for (i in 1:n) {
  resumen = summary(corporal1[, i])
  desviacion = sd(corporal1[, i], na.rm = TRUE)
  d[i, ] = c(resumen[1], resumen[2], resumen[3], resumen[4], resumen[5],
resumen[6], desviacion)
}

print(d)
```

```
##         Min      Q1 Median      Mean     Q3   Max        SD
## [1,]  19.0  24.750  28.00  31.44444  37.00  65.0 10.554469
## [2,]  42.0  54.950  71.50  68.95278  82.40  98.2 14.868999
## [3,] 147.2 164.800 172.70 171.55556 179.40 190.5 10.520170
## [4,]   8.3   9.475  10.65  10.46667  11.50  12.4  1.175463
## [5,]  23.5  25.975  32.15  31.16667  35.05  40.4  5.234392
```

PARTE 1

```
#Realiza el análisis de los valores y vectores propios con la matriz de
covarianzas y con la de correlación. Analiza la varianza explicada por cada
componente en cada caso e interpreta dentro del contexto del problema.

#1. Calcule las matrices de varianza-covarianza S con cov(X) y la matriz de
correlaciones R con cor(X) y realice los siguientes pasos con cada una:

s = cov(corporal1)

r = cor(corporal1)

#-Calcule los valores y vectores propios de cada matriz.La función en R es:
eigen().
```

```r
eigen_s = eigen(s)
print(eigen_s)

## eigen() decomposition
## $values
## [1] 359.3980243  80.3757858  27.6229011   4.3074318   0.2343571
##
## $vectors
##             [,1]       [,2]        [,3]         [,4]         [,5]
## [1,] -0.34871002  0.9075501 -0.23248825 -0.001589466  0.026473941
## [2,] -0.76617586 -0.1616581  0.52166894 -0.338508602  0.010707863
## [3,] -0.47632405 -0.3851755 -0.78905759  0.046160807  0.003543154
## [4,] -0.05386189  0.0155423  0.02785902  0.126103480 -0.990039959
## [5,] -0.24817367 -0.0402221  0.22455005  0.931330496  0.137814357

eigen_r = eigen(r)
print(eigen_r)

## eigen() decomposition
## $values
## [1] 3.75749733 0.72585665 0.32032981 0.12461873 0.07169749
##
## $vectors
##            [,1]       [,2]        [,3]       [,4]       [,5]
## [1,] -0.3359310  0.8575601 -0.34913780 -0.1360111  0.1065123
## [2,] -0.4927066 -0.1647821  0.06924561 -0.5249533 -0.6706087
## [3,] -0.4222426 -0.4542223 -0.73394453  0.2070673  0.1839617
## [4,] -0.4821923  0.1082775  0.36690716  0.7551547 -0.2255818
## [5,] -0.4833139 -0.1392684  0.44722747 -0.3046138  0.6739511

#2. Calcule la proporción de varianza explicada por cada componente en ambas
matrices. Se sugiere dividir cada lambda entre la varianza total (las lambdas
están en eigen(S)$values). La varianza total es la suma de las varianzas de
la diagonal de S. Una forma es sum(diag(S)). La varianza total de los
componentes es la suma de los valores propios (es decir, la suma de la
varianza de cada componente), sin embargo, si sumas la diagonal de S (es
decir, la varianza de cada x), te da el mismo valor (¡compruébalo!). Recuerda
que las combinaciones lineales buscan reproducir la varianza de X.

varianza_total_s <- sum(eigen_s$values)

proporcion_varianza_s <- eigen_s$values / varianza_total_s
print(proporcion_varianza_s)

## [1] 0.7615357176 0.1703098726 0.0585307219 0.0091271040 0.0004965839

varianza_total_r <- sum(eigen_r$values)

proporcion_varianza_r <- eigen_r$values / varianza_total_r
print(proporcion_varianza_r)
```

```
## [1] 0.75149947 0.14517133 0.06406596 0.02492375 0.01433950

#3. Acumule los resultados anteriores (cumsum() puede servirle) para obtener la varianza acumulada en cada componente.

varianza_acumulada_s <- cumsum(proporcion_varianza_s)
print("Varianza S")

## [1] "Varianza S"

print(varianza_acumulada_s)

## [1] 0.7615357 0.9318456 0.9903763 0.9995034 1.0000000

varianza_acumulada_r <- cumsum(proporcion_varianza_r)
print("Varianza R")

## [1] "Varianza R"

print(varianza_acumulada_r)

## [1] 0.7514995 0.8966708 0.9607368 0.9856605 1.0000000

#4. Según los resultados anteriores, ¿qué componentes son los más importantes?

print("Con los datos vistos, podemos definir que las primeros dos componentes son los mas utiles, ya que nos dan un valor de 0.9318456, lo cual es optimo para tratar de reducir los componentes y mantener un buen valor.")

## [1] "Con los datos vistos, podemos definir que las primeros dos componentes son los mas utiles, ya que nos dan un valor de 0.9318456, lo cual es optimo para tratar de reducir los componentes y mantener un buen valor."

#5. Escriba la ecuación de la combinación lineal de los Componentes principales CP1 y CP2 (eiX, donde ei está en eigen(S)$vectors[1], e2X para obtener CP2, donde X = c(X1, X2, ...)) ¿qué variables son las que más contribuyen a la primera y segunda componentes principales? (observe los coeficientes en valor absoluto de las combinaciones lineales). Justifique su respuesta.

etiquetas = colnames(corporal1)

CP1_coeficientes_S = eigen_s$vectors[,1]
CP1_combinacion_S = data.frame(Variable = etiquetas, Coeficiente_CP1_S = CP1_coeficientes_S)

CP2_coeficientes_S = eigen_s$vectors[,2]
CP2_combinacion_S = data.frame(Variable = etiquetas, Coeficiente_CP2_S = CP2_coeficientes_S)

print("Combinación lineal de CP1 para la matriz de covarianza (S):")
```

```
## [1] "Combinación lineal de CP1 para la matriz de covarianza (S):"

print(CP1_combinacion_S)

##   Variable Coeficiente_CP1_S
## 1     edad       -0.34871002
## 2     peso       -0.76617586
## 3   altura       -0.47632405
## 4   muneca       -0.05386189
## 5   biceps       -0.24817367

print("Combinación lineal de CP2 para la matriz de covarianza (S):")

## [1] "Combinación lineal de CP2 para la matriz de covarianza (S):"

print(CP2_combinacion_S)

##   Variable Coeficiente_CP2_S
## 1     edad         0.9075501
## 2     peso        -0.1616581
## 3   altura        -0.3851755
## 4   muneca         0.0155423
## 5   biceps        -0.0402221

CP1_coeficientes_R = eigen_r$vectors[,1]
CP1_combinacion_R = data.frame(Variable = etiquetas, Coeficiente_CP1_R =
CP1_coeficientes_R)

CP2_coeficientes_R = eigen_r$vectors[,2]
CP2_combinacion_R = data.frame(Variable = etiquetas, Coeficiente_CP2_R =
CP2_coeficientes_R)

print("Combinación lineal de CP1 para la matriz de correlación (R):")

## [1] "Combinación lineal de CP1 para la matriz de correlación (R):"

print(CP1_combinacion_R)

##   Variable Coeficiente_CP1_R
## 1     edad        -0.3359310
## 2     peso        -0.4927066
## 3   altura        -0.4222426
## 4   muneca        -0.4821923
## 5   biceps        -0.4833139

print("Combinación lineal de CP2 para la matriz de correlación (R):")

## [1] "Combinación lineal de CP2 para la matriz de correlación (R):"

print(CP2_combinacion_R)

##   Variable Coeficiente_CP2_R
## 1     edad         0.8575601
```

```
## 2     peso        -0.1647821
## 3   altura        -0.4542223
## 4   muneca         0.1082775
## 5   biceps        -0.1392684
```

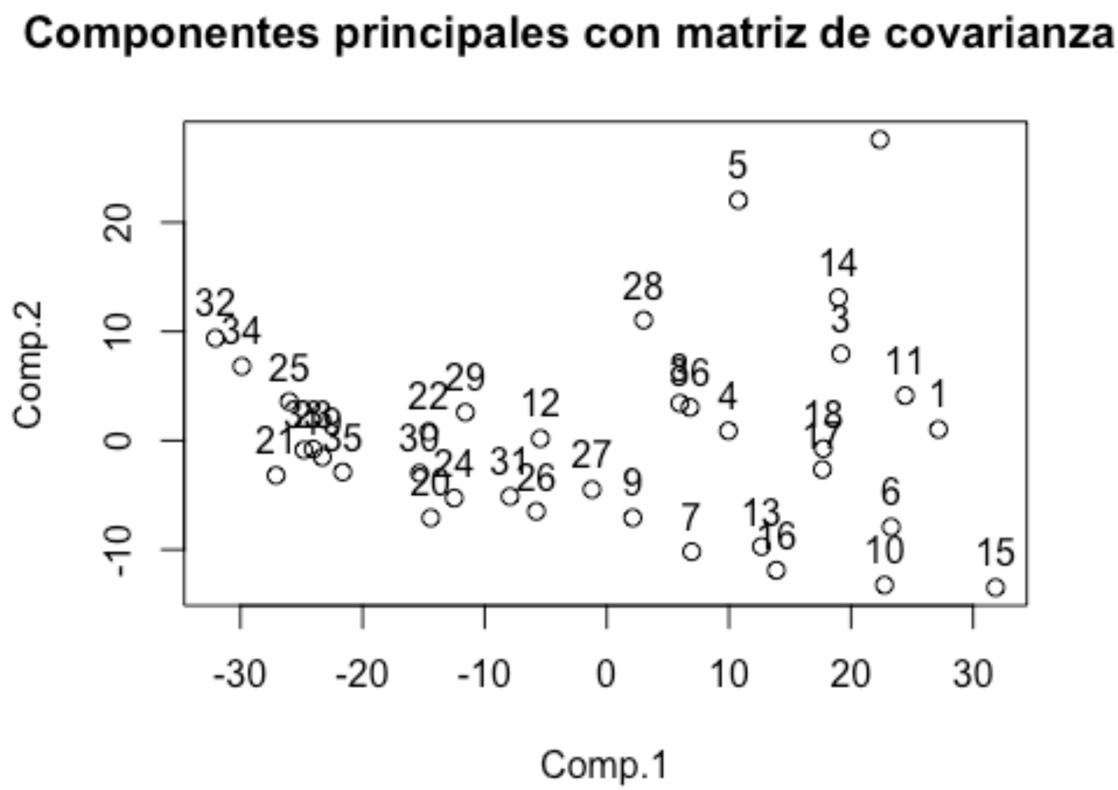
```
#¡No te olvides de seguir los mismos pasos con la matriz de correlaciones (se
obtiene con cor(x) si x está compuesto por variables numéricas)
```

Parte 2

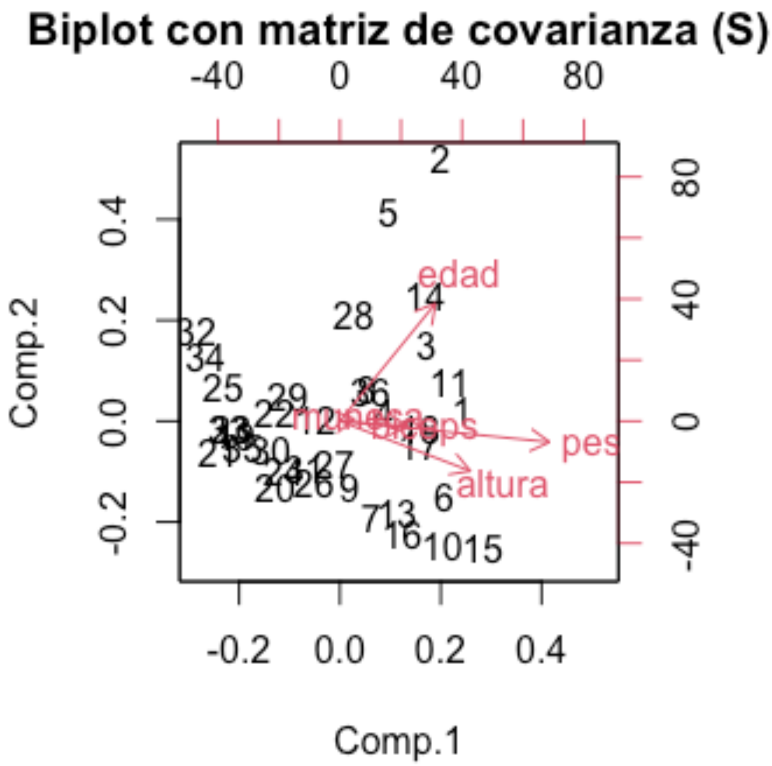
```
#Obtenga las gráficas respectivas con S (matriz de varianzas-covarianzas) y
con R (matriz de correlaciones) de las dos primeras componentes.

cpS = princomp(corporal1, cor=FALSE)
summary(cpS)
```

```
## Importance of components:
##                            Comp.1    Comp.2     Comp.3      Comp.4
Comp.5
## Standard deviation     18.6926388 8.8398600 5.18223874 2.046406827
0.4773333561
## Proportion of Variance  0.7615357 0.1703099 0.05853072 0.009127104
0.0004965839
## Cumulative Proportion   0.7615357 0.9318456 0.99037631 0.999503416
1.0000000000
```

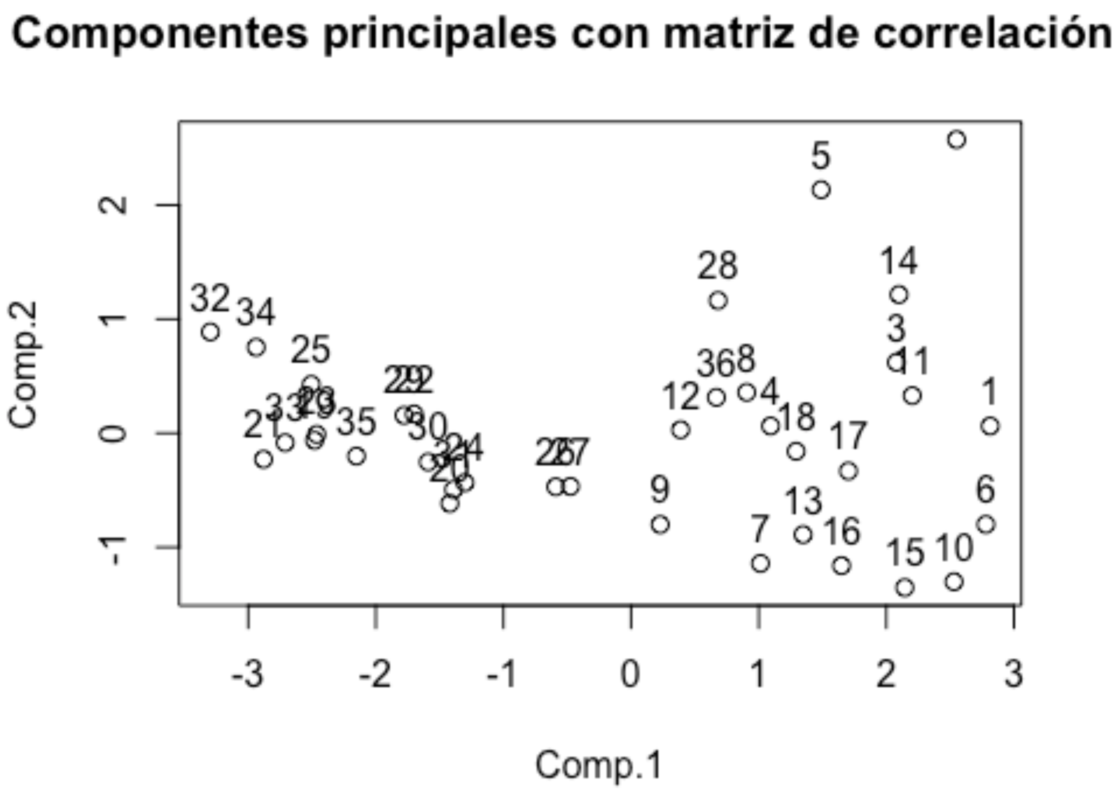
```
plot(cpS$scores[, 1:2], type="p", main="Componentes principales con matriz de
covarianza (S)")
text(cpS$scores[, 1], cpS$scores[, 2], labels=1:nrow(cpS$scores), pos=3)
```



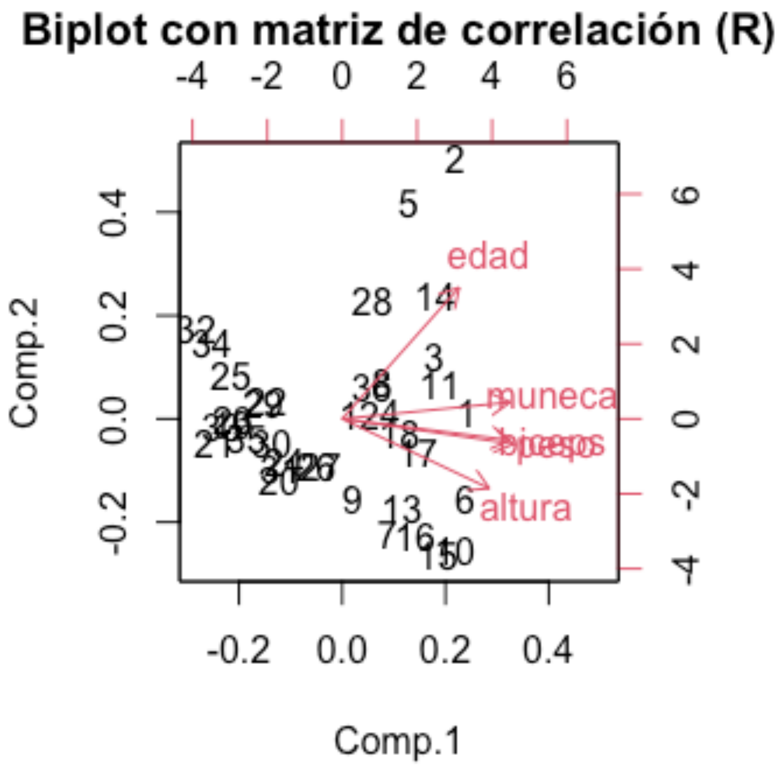

```r
biplot(cpS, main="Biplot con matriz de covarianza (S)")
```



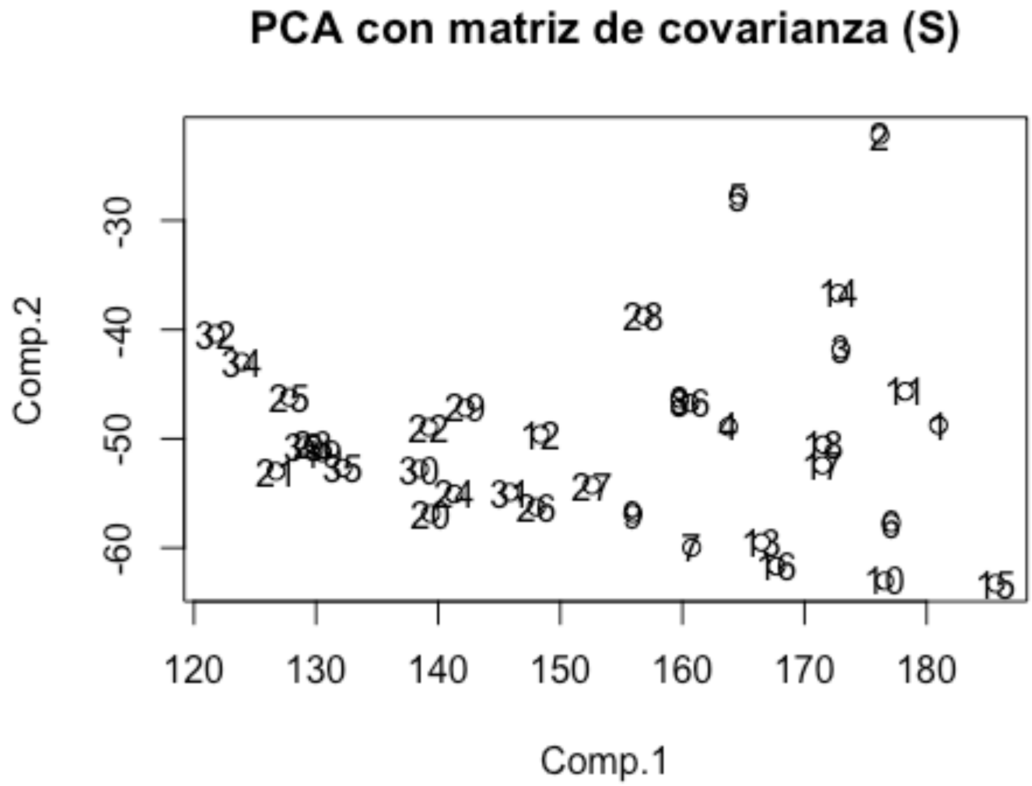

```
cpR = princomp(corporal1, cor=TRUE)
summary(cpR)

## Importance of components:
##                           Comp.1    Comp.2     Comp.3     Comp.4    Comp.5
## Standard deviation     1.9384265 0.8519722 0.56597686 0.35301378 0.2677639
## Proportion of Variance 0.7514995 0.1451713 0.06406596 0.02492375 0.0143395
## Cumulative Proportion  0.7514995 0.8966708 0.96073676 0.98566050 1.0000000

plot(cpR$scores[, 1:2], type="p", main="Componentes principales con matriz de
correlación (R)")
text(cpR$scores[, 1], cpR$scores[, 2], labels=1:nrow(cpR$scores), pos=3)
```



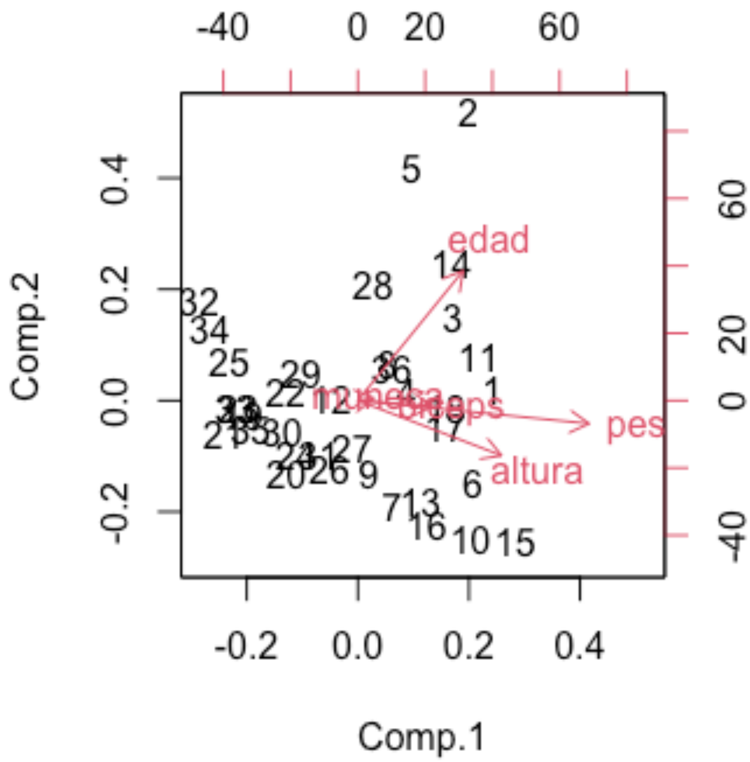

```r
biplot(cpR, main="Biplot con matriz de correlación (R)")
```



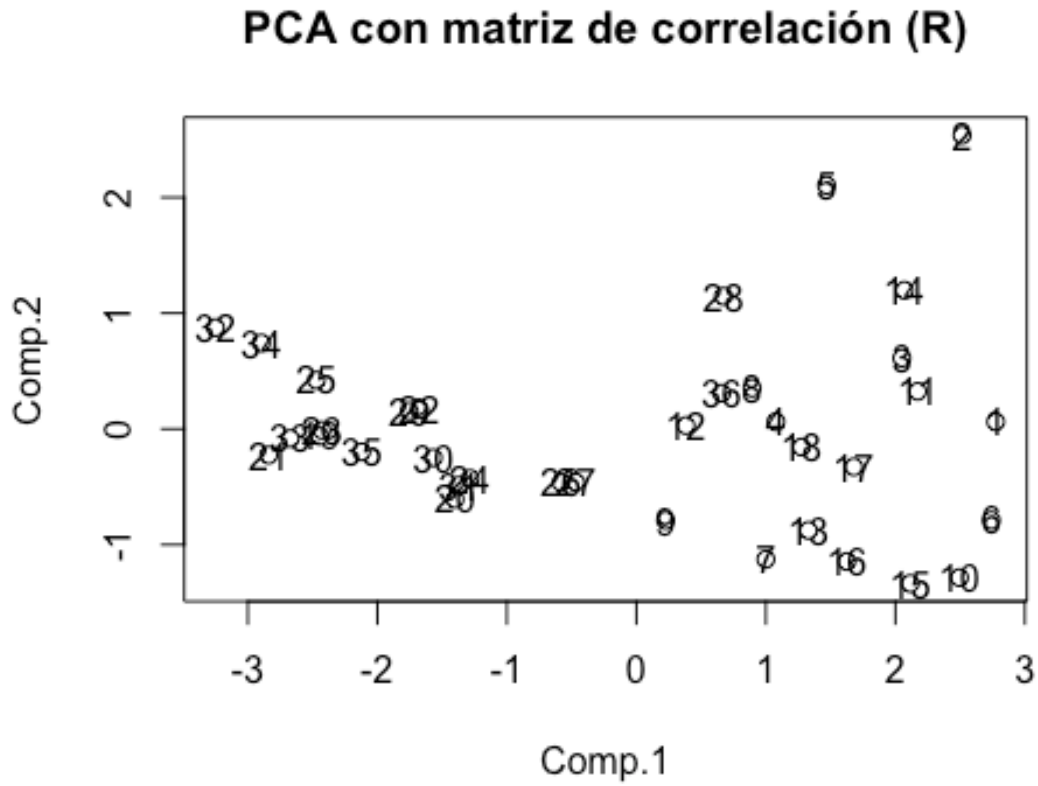

```
#Calcule las puntuaciones (scores) de las observaciones para los componentes obtenidos con la matriz de varianzas-covarianzas

cpS = princomp(corporal1, cor = FALSE)
cpaS = as.matrix(corporal1) %*% cpS$loadings

plot(cpaS[,1:2], type = "p", main = "PCA con matriz de covarianza (S)")
text(cpaS[,1], cpaS[,2], labels = 1:nrow(cpaS))
```


PCA con matriz de covarianza (S)
Comp.2
Comp.1
-30
-40
-50
-60
120
130
140
150
160
170
180

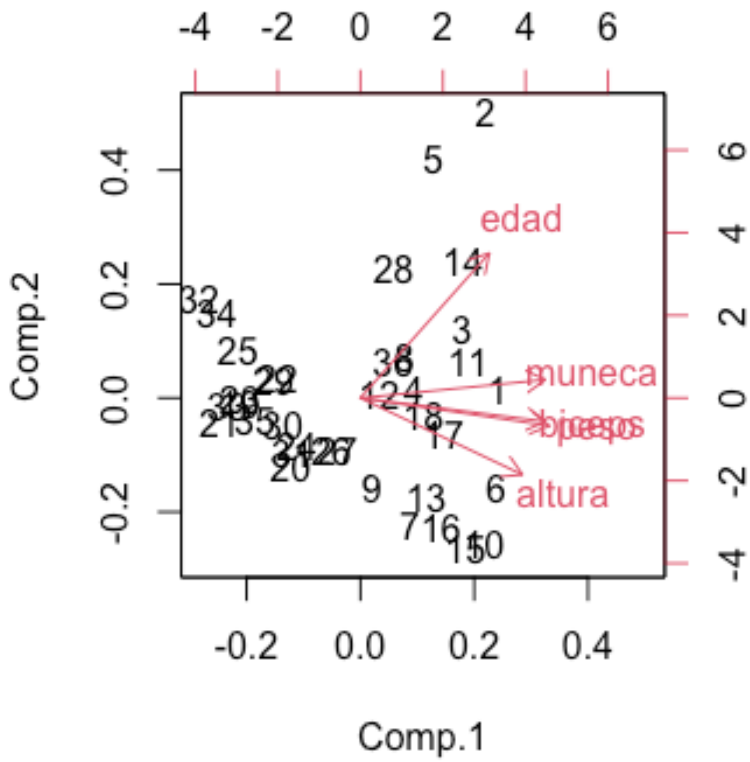
```r
biplot(cpS)
```




```r
#Calcule las puntuaciones (scores) de las observaciones para los componentes
obtenidos con la matriz de correlaciones. Recuerde que en la matriz de
correlaciones las variables tienen que estar estandarizadas.

cpR <- princomp(corporal1, cor = TRUE)
cpaR <- as.matrix(scale(corporal1)) %*% cpR$loadings

plot(cpaR[,1:2], type = "p", main = "PCA con matriz de correlación (R)")
text(cpaR[,1], cpaR[,2], labels = 1:nrow(cpaR))
```




```r
biplot(cpR)
```




```
#Explora el: princomp() en library(stats). Puedes poner help(princomp) en la
consola o buscarlo en la ventana de ayuda. Indaga: ¿qué otras opciones tiene
para facilitarte el análisis? En particular, explora los comandos y
subcomandos: summary(cpS), cpaS$loading, cpaS$scores. ¿Cómo se interpreta el
resultado?

summary(cpS)

## Importance of components:
##                           Comp.1    Comp.2     Comp.3      Comp.4
Comp.5
## Standard deviation     18.6926388 8.8398600 5.18223874 2.046406827
0.4773333561
## Proportion of Variance  0.7615357 0.1703099 0.05853072 0.009127104
0.0004965839
## Cumulative Proportion   0.7615357 0.9318456 0.99037631 0.999503416
1.0000000000

cpS$loadings

##
## Loadings:
##         Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## edad     0.349  0.908  0.232
```

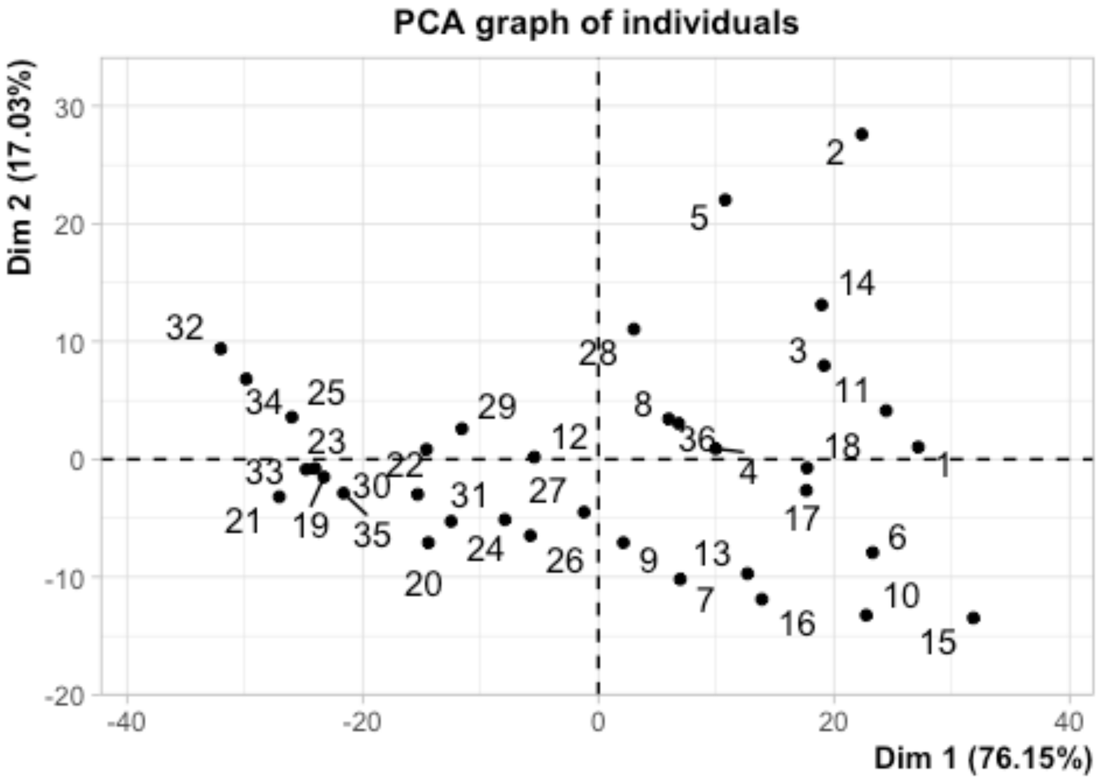
```
## peso    0.766 -0.162 -0.522  0.339
## altura  0.476 -0.385  0.789
## muneca                      -0.126 -0.990
## biceps  0.248        -0.225 -0.931  0.138
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## SS loadings       1.0    1.0    1.0    1.0    1.0
## Proportion Var    0.2    0.2    0.2    0.2    0.2
## Cumulative Var    0.2    0.4    0.6    0.8    1.0
```

```
head(cpS$scores)
```

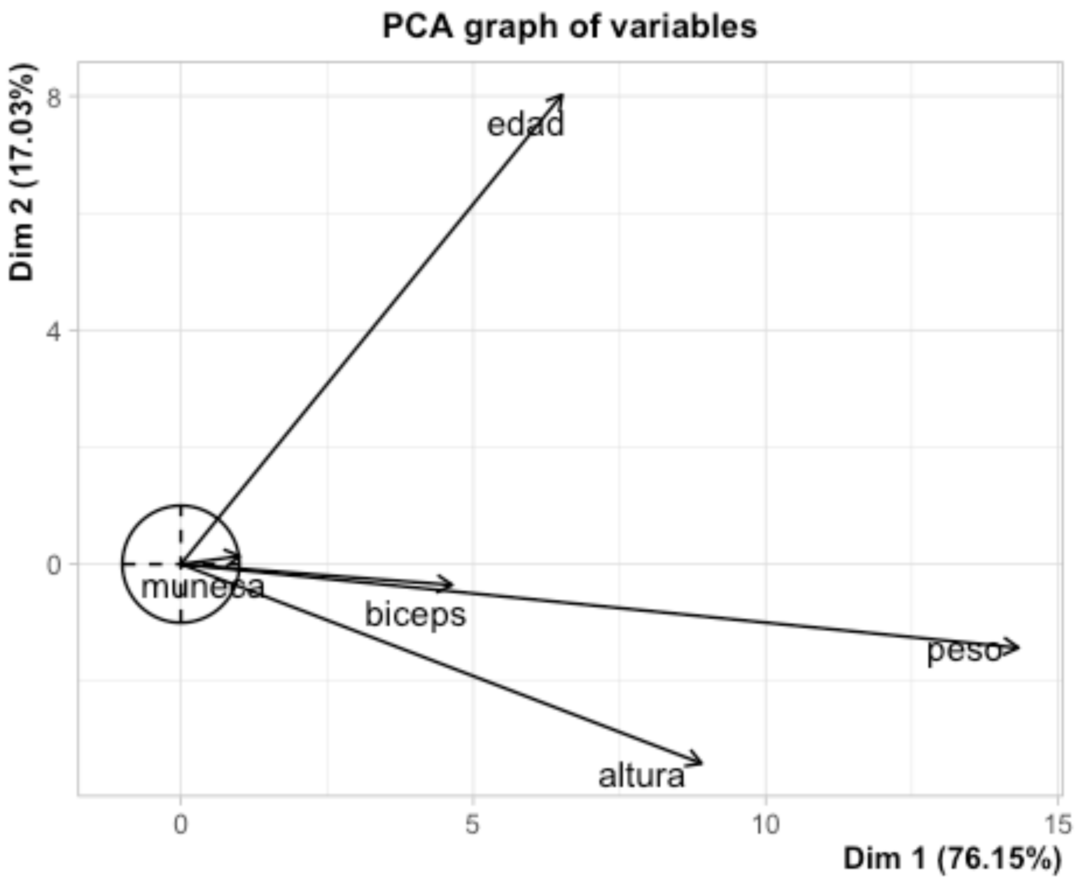
```
##         Comp.1     Comp.2     Comp.3      Comp.4      Comp.5
## [1,] 27.162853  1.0278492  5.0022646  0.93622690 -0.51688356
## [2,] 22.363542 27.5955807  3.0635949 -0.08338126  0.02552809
## [3,] 19.167874  7.9566157 -1.5770026 -2.61077676  0.80391745
## [4,]  9.959001  0.8923731  5.5146952  0.12345373 -0.35579895
## [5,] 10.775593 22.0203437 -0.7562826  0.17996723 -0.41646606
## [6,] 23.283948 -7.9268214  2.7958617 -2.09339284 -0.62252321
```

Parte 3

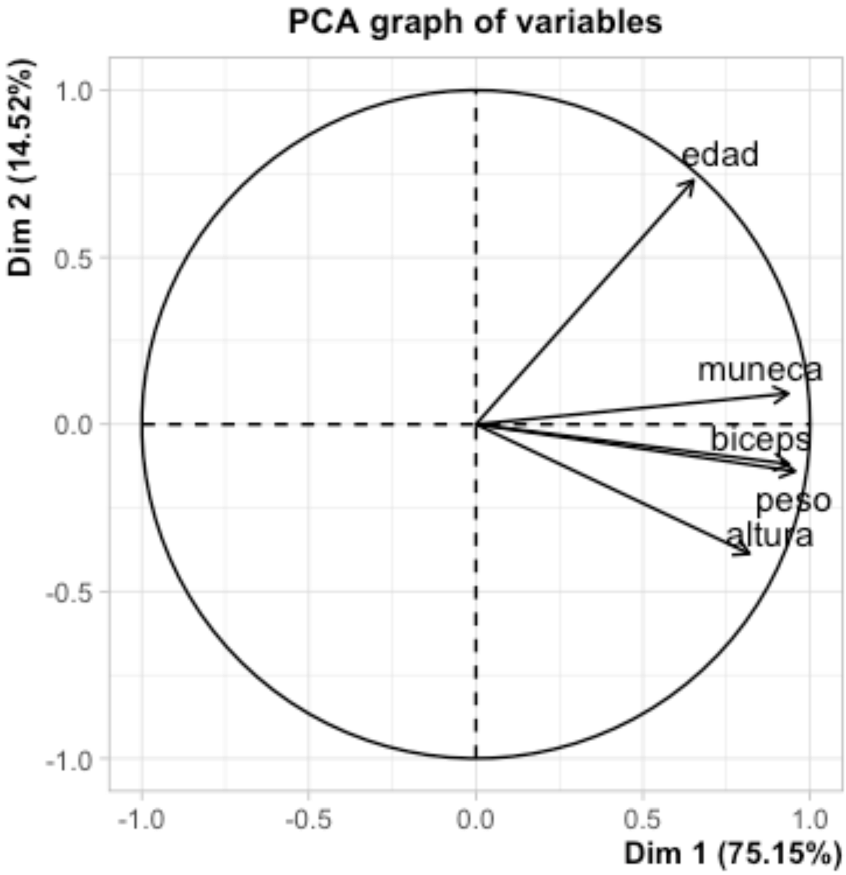
```
#Explore los siguientes gráficos relativos a Componentes Principales.

library(FactoMineR)
library(ggplot2)
datos=corporal1
cpS = PCA(datos,scale.unit=FALSE)
```



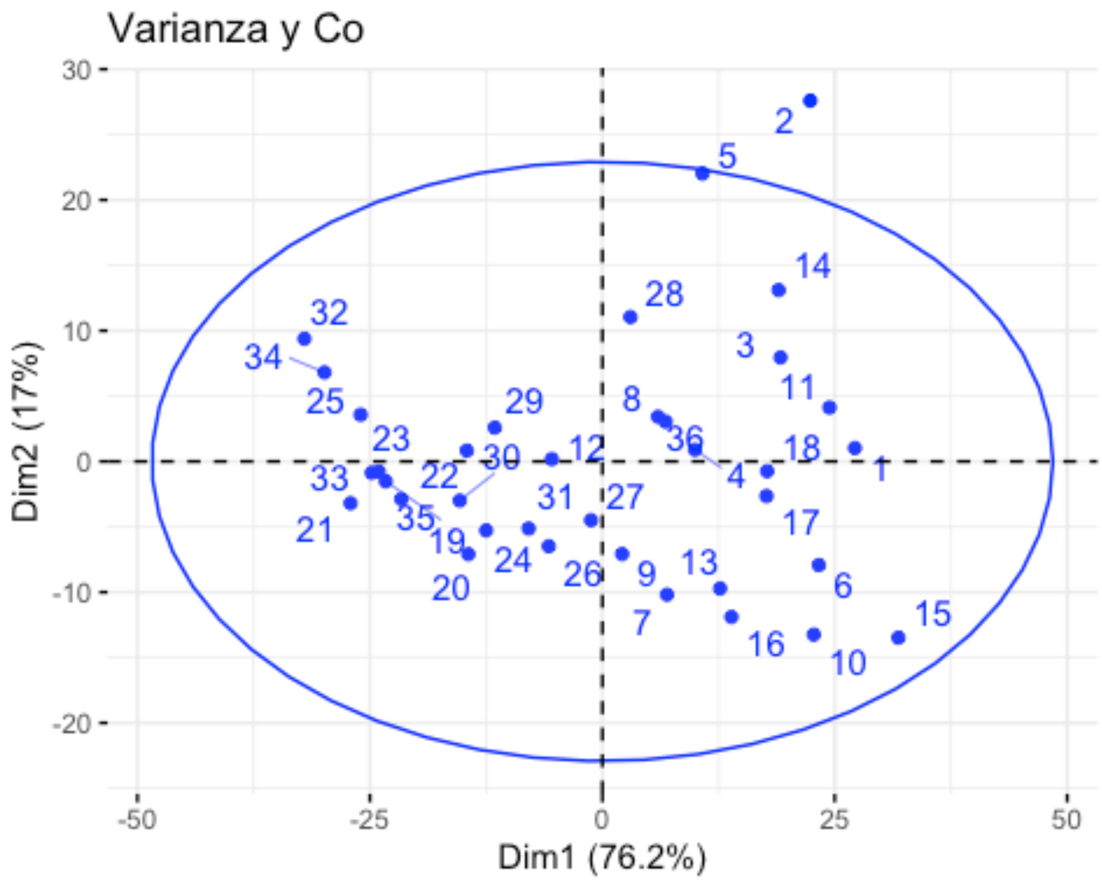
PCA graph of individuals
Dim 2 (17.03%)
Dim 1 (76.15%)
30
20
10
0
-10
-20
-40
-20
0
20
40
2
5
14
3
32
28
11
34
25
29
8
12
36
18
1
23
33
22
30
4
31
27
17
21
19
35
6
24
26
9
13
20
10
7
16
15



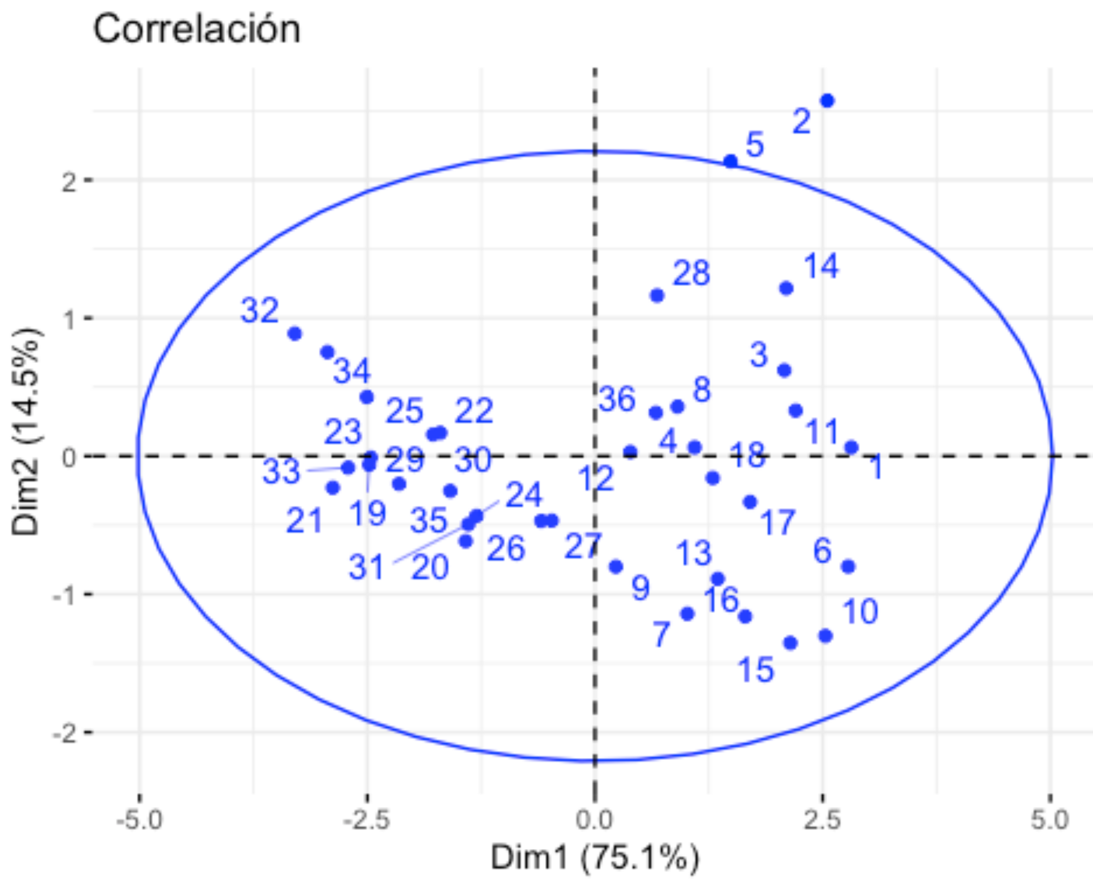

```
cpR = PCA(datos,scale.unit=TRUE)
```



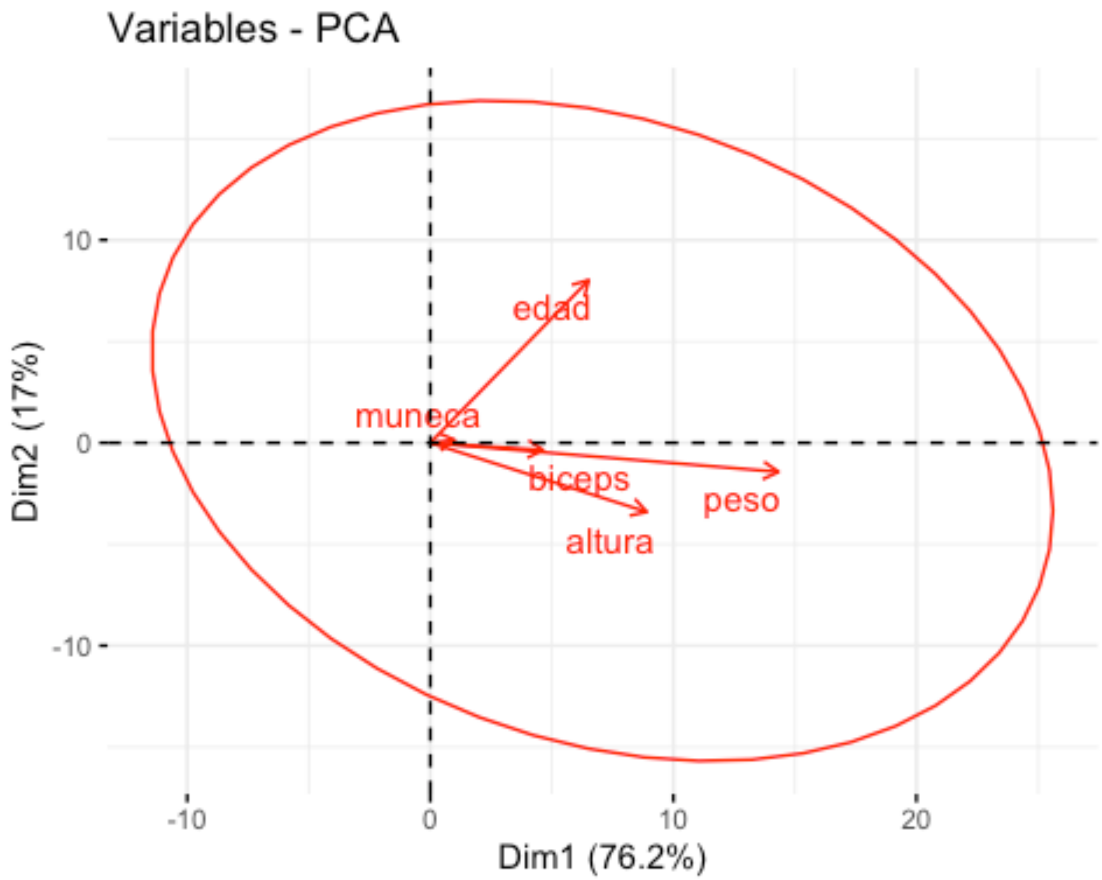

```
library(factoextra)

## Welcome! Want to learn more? See two factoextra-related books at
https://goo.gl/ve3WBa

fviz_pca_ind(cpS, col.ind = "blue", addEllipses = TRUE, repel = TRUE,
title="Varianza y Co")
```



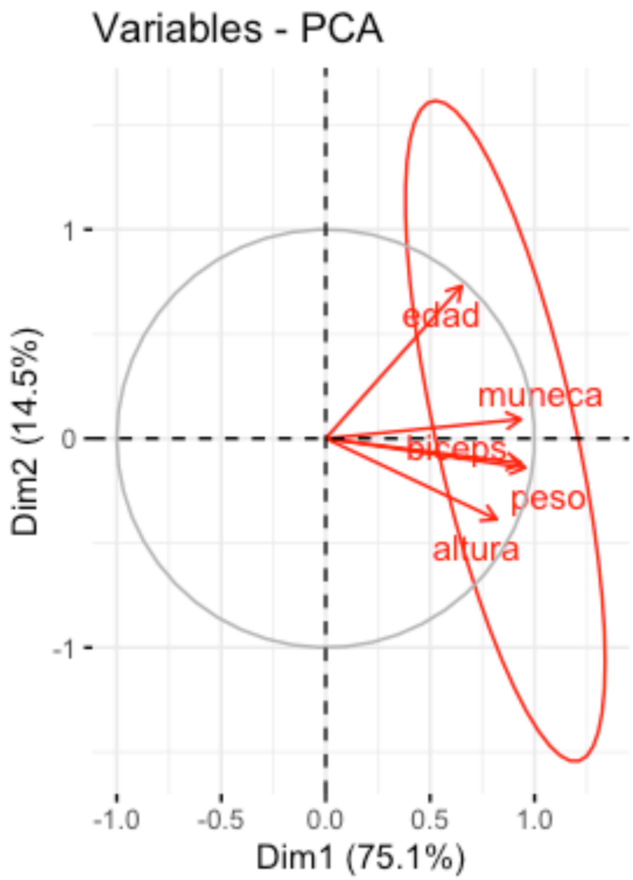

```r
fviz_pca_ind(cpR, col.ind = "blue", addEllipses = TRUE, repel = TRUE,
title="Correlación")
```



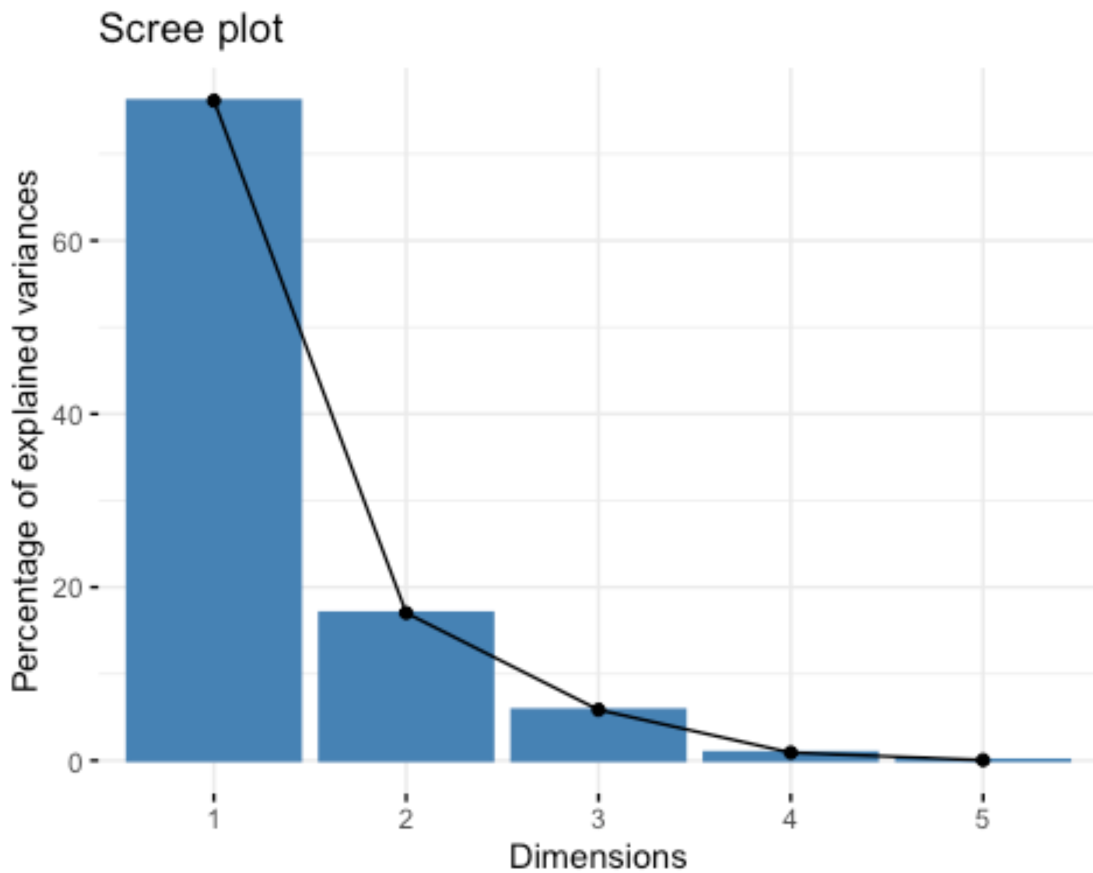

```r
fviz_pca_var(cpS, col.var = "red", addEllipses = TRUE, repel = TRUE)
```



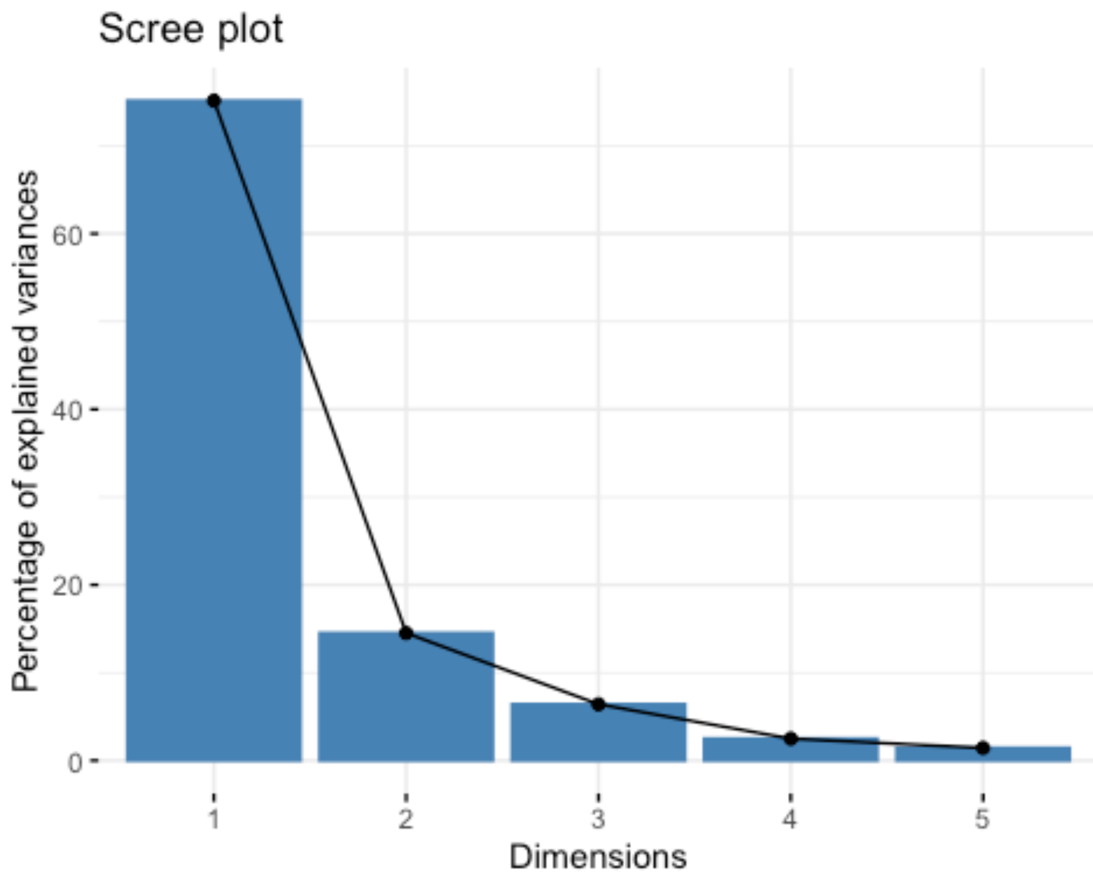

```r
fviz_pca_var(cpR, col.var = "red", addEllipses = TRUE, repel = TRUE)
```



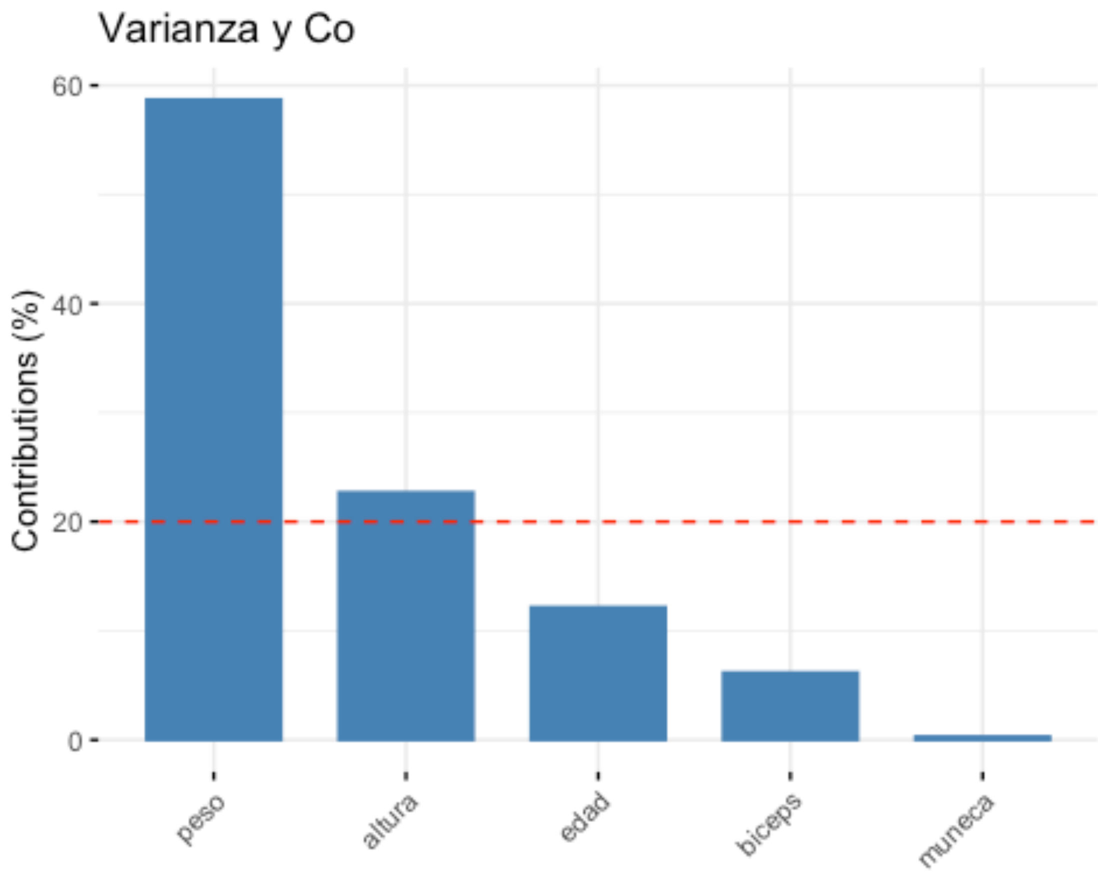

```r
fviz_screeplot(cpS)
```



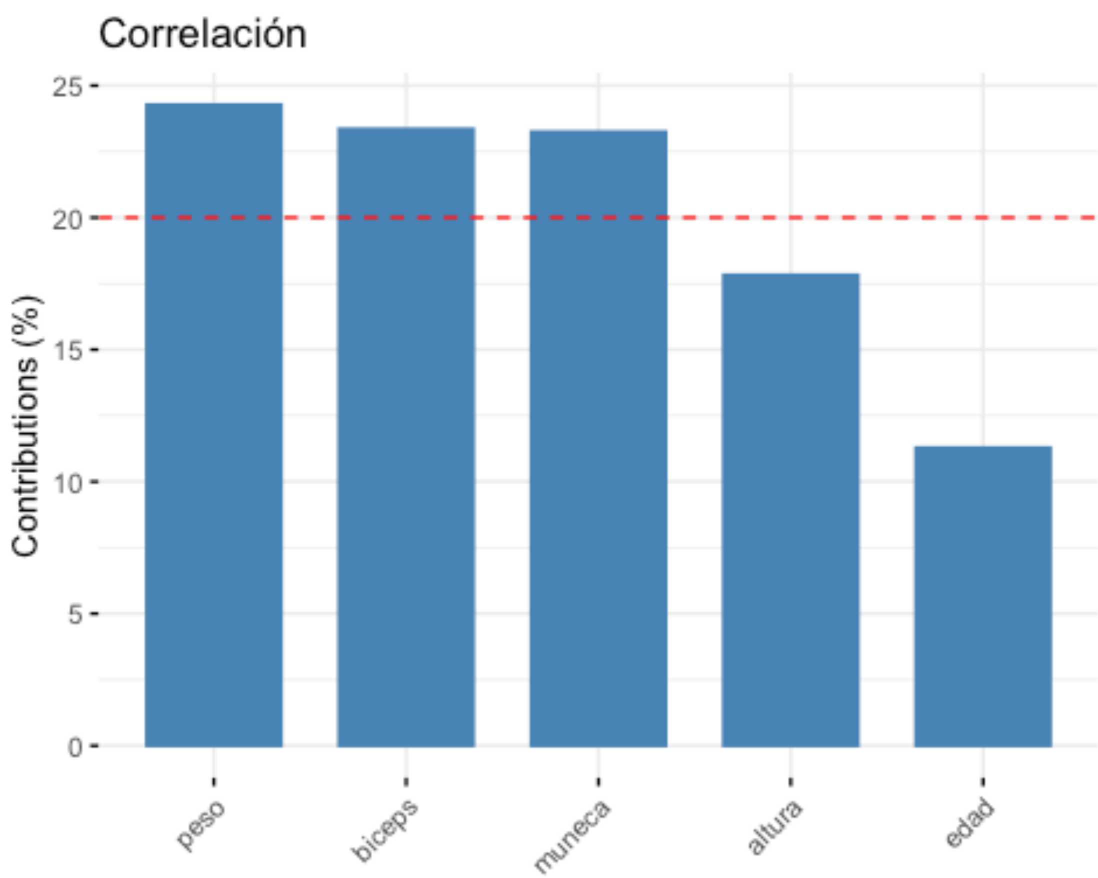

```
fviz_screeplot(cpR)
```



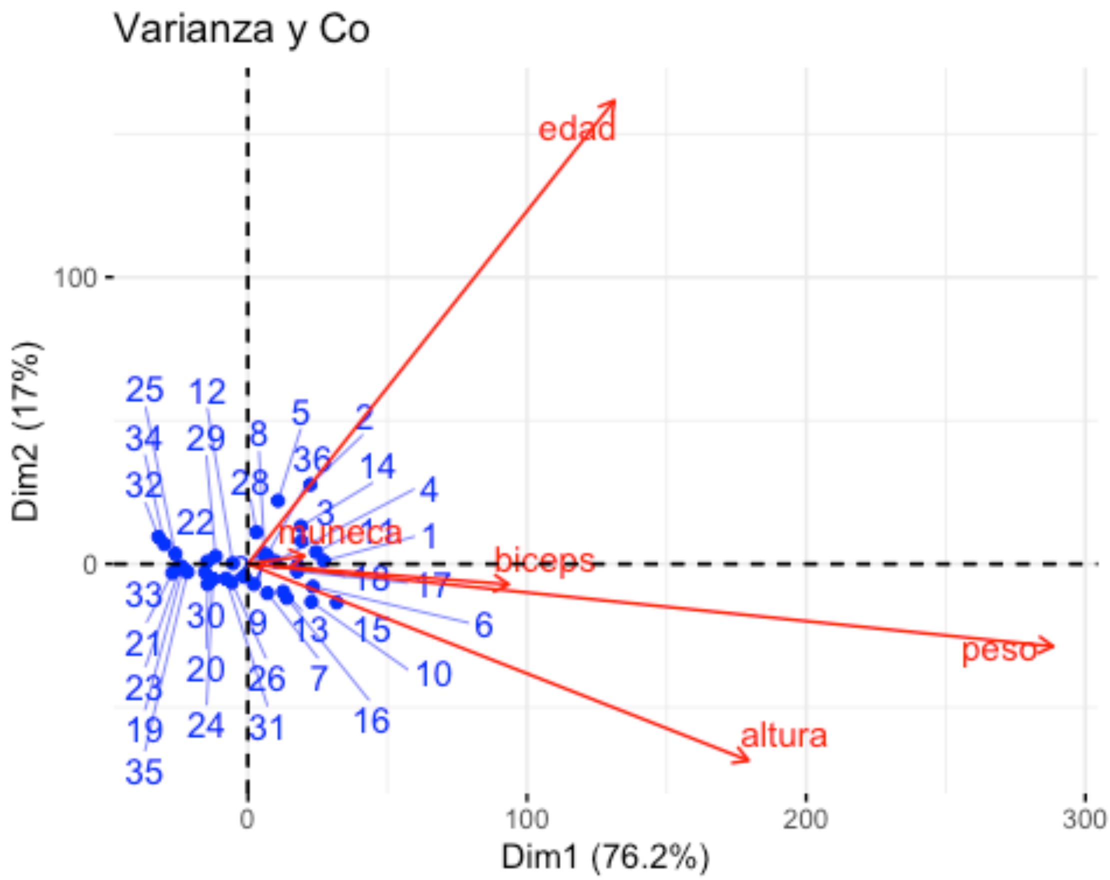

```r
fviz_contrib(cpS, choice = c("var"), title="Varianza y Co")
```



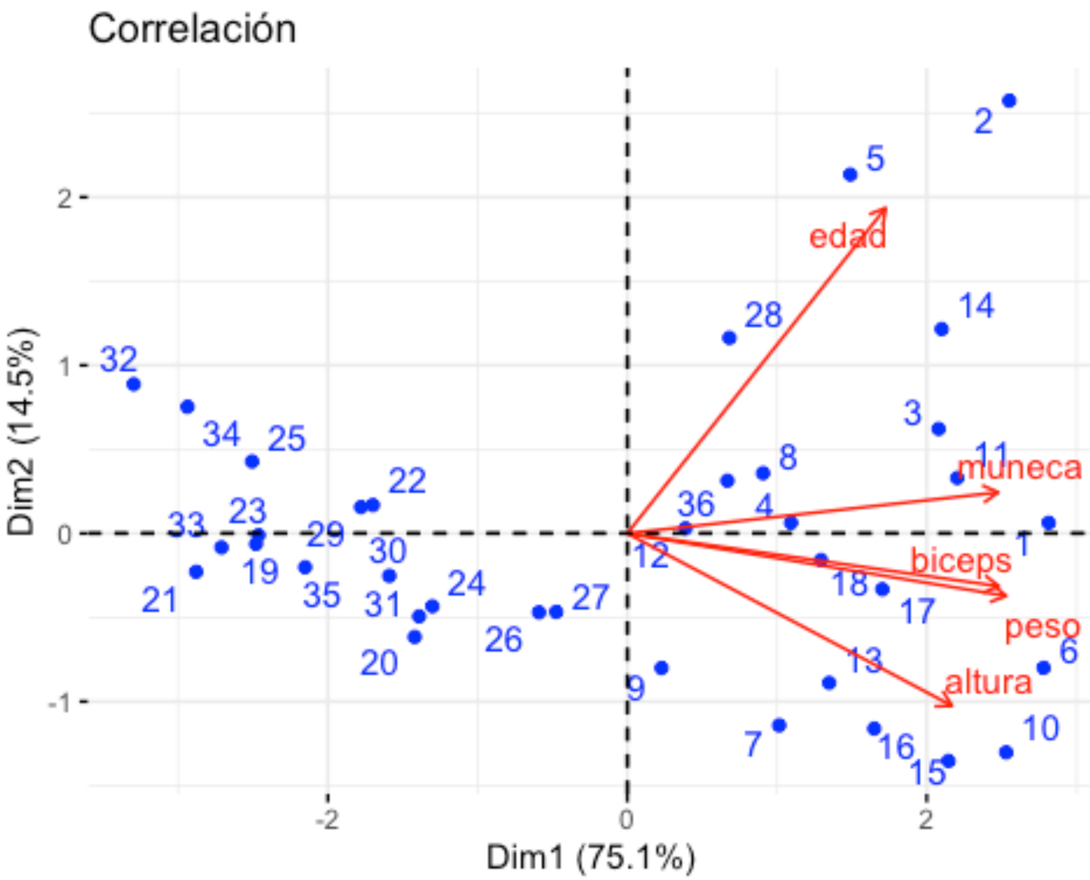

```r
fviz_contrib(cpR, choice = c("var"), title="Correlación")
```




```
fviz_pca_biplot(cpS, repel=TRUE, col.var="red", col.ind="blue",
title="Varianza y Co")
```




```
fviz_pca_biplot(cpR, repel=TRUE, col.var="red", col.ind="blue",
title="Correlación")
```




```
#Interprete cada gráfico e identifica qué es lo que se está graficando en
cada uno. Realiza el análisis con la matriz de varianzas y covarianzas y
correlación.
print("Observaciones:")

## [1] "Observaciones:"

print("La primera grafica nos define cuales variables llegan a ser similares,
casi todos los datos tienen caracteristicas similares, pero en cuanto a datos
atipicos tenemos el numero 2")

## [1] "La primera grafica nos define cuales variables llegan a ser
similares, casi todos los datos tienen caracteristicas similares, pero en
cuanto a datos atipicos tenemos el numero 2"

print("La segunda grafica muestra la cantidad de correlacion que tienen las
variables dependiendo de su cercania.")

## [1] "La segunda grafica muestra la cantidad de correlacion que tienen las
variables dependiendo de su cercania."

print("El tercer grafico contiene tanto la varianza como la correlacion
explicadas para cada componente.")
```

```
## [1] "El tercer grafico contiene tanto la varianza como la correlacion
explicadas para cada componente."
```

```r
print("El cuarto grafico nos muestra como el peso y la altura tienen un gran
peso en la varianza.")
```

```
## [1] "El cuarto grafico nos muestra como el peso y la altura tienen un gran
peso en la varianza."
```

```r
print("El quinto grafico nos muestra la relacion entre los datos y las
variables, esto definido por las lineas azules, las lineas rojas muestran el
peso que tienen estas variables en los componentes principales.")
```

```
## [1] "El quinto grafico nos muestra la relacion entre los datos y las
variables, esto definido por las lineas azules, las lineas rojas muestran el
peso que tienen estas variables en los componentes principales."
```

```r
print("Conclusion:")
```

```
## [1] "Conclusion:"
```

```r
print("Como todas las variables hablan de una diferente medida del cuerpo muy
diferente, el metodo de correlacion permite que cada una tenga su respectivo
peso en el analisis, asi teniendo un mejor analisis en general.")
```

```
## [1] "Como todas las variables hablan de una diferente medida del cuerpo
muy diferente, el metodo de correlacion permite que cada una tenga su
respectivo peso en el analisis, asi teniendo un mejor analisis en general."
```