Corrección Examen Parcial 1

ByteMiners

09/03/2020

Análisis exploratorio de datos

Las librerías usadas en el examen fueron las siguientes:

```
library(ggplot2)
library(cowplot)
library(dplyr)
library(grid)
library(FactoMineR)
library(factoextra)
library(corrplot)
library(ggpubr)
library(scatterplot3d)
library(reshape2)
library(knitr)
library(MVN)
library(biotools)
```

```
## ---
## biotools version 3.1
```

```
library(klaR)
library(janitor)
library(psych)
library(gridExtra)
library(GGally)
library(funModeling)
library(CCA)
library(vegan)
library(fields)
```

1. Exploración y visualización de variables

1.1 Explorar la base de datos, realizar los cambios necesarios y convenientes para poder trabajar con ella.

```r
df <- read.csv("Ejercicio 1.1.csv", sep = ",")
df
```

```
##                   ID        X1        X2        X3        X4        X5
## 1               Peso     53.74     61.84     51.64     75.95      63.3
## 2          Estatura       166       154       151       180       165
## 3              Edad      23.5      22.4        27       2.3      17.2
## 4        Complexión Mesomorfo Mesomorfo Mesomorfo Endomorfo Mesomorfo
## 5    Papás_separados        SI        SI        NO        NO        NO
## 6  Cantidad_hermanos         3         0         2         2         2
## 7            Trabaja        SI        SI        NO        SI        NO
## 8     Horas_trab_dia         5        10      <NA>        10      <NA>
## 9     Horas_trab_mes        25        80      <NA>        80      <NA>
## 10       Salario_mes      1750      3200      <NA>      3200      <NA>
##           X6        X7        X8        X9       X10       X11       X12
## 1       51.8     64.87     67.38     65.76     56.95     75.12      63.9
## 2        149       145       159       177       164       179       169
## 3       15.7        20      27.3      17.5      18.8      21.8      23.2
## 4  Mesomorfo Mesomorfo Endomorfo Endomorfo Mesomorfo Endomorfo Mesomorfo
## 5         NO        NO        NO        NO        SI        NO        NO
## 6          5         1         4         0         2         1         1
## 7         SI        NO        SI        NO        SI        SI        SI
## 8          6      <NA>         4      <NA>        10         6         8
## 9         30      <NA>        20      <NA>        80        30        40
## 10      1200      <NA>      2400      <NA>     12000      3600      2800
##          X13       X14       X15       X16       X17       X18       X19
## 1      53.79     37.85     71.25     59.55     59.84     69.44     68.21
## 2        154       157       146       141       169       144       158
## 3       29.4        30      26.3      17.7      23.2      23.7      18.4
## 4  Mesomorfo Ectomorfo Endomorfo Mesomorfo Mesomorfo Endomorfo Endomorfo
## 5         NO        SI        SI        NO        NO        SI        NO
## 6          4         4         1         2         0         2         5
## 7         NO        SI        SI        NO        SI        NO        SI
## 8       <NA>         8         7      <NA>         6      <NA>         9
## 9       <NA>        40        35      <NA>        30      <NA>        45
## 10      <NA>      3600      5250      <NA>      3600      <NA>      3150
##          X20       X21       X22       X23       X24       X25       X26
## 1      65.94     69.19     67.82     60.75     40.11      66.2     59.44
## 2        166       180       160       159       147       170       158
## 3       21.4      17.5      25.6      18.5      19.7      17.2        30
## 4  Endomorfo Endomorfo Endomorfo Mesomorfo Ectomorfo Endomorfo Mesomorfo
## 5         NO        NO        NO        NO        NO        NO        NO
## 6          3         3         2         2         5         4         5
## 7         SI        SI        SI        NO        SI        SI        SI
## 8          7         7         8      <NA>         6         6         5
## 9         35        35        40      <NA>        30        48        25
## 10      5250      5250      1600      <NA>      2100      1200      1000
##          X27       X28       X29       X30       X31       X32       X33
## 1      58.44     45.29     55.22     64.18     73.59     58.97     63.88
## 2        160       148       149       164       163       143       141
## 3       18.5      25.6      26.7        21      18.9      16.3      25.2
## 4  Mesomorfo Ectomorfo Mesomorfo Mesomorfo Endomorfo Mesomorfo Mesomorfo
## 5         NO        NO        NO        SI        NO        NO        SI
```

```
## 6          4         4         1         4         5         1         2
## 7         SI        NO        SI        SI        SI        SI        SI
## 8         10      <NA>         9        10         4         6         7
## 9         80      <NA>        45        70        20        30        35
## 10      9600      <NA>      3150      5600      1800      1200      4200
##           X34       X35       X36       X37       X38       X39       X40
## 1       59.46     46.23     55.85     56.06     59.41        71     67.63
## 2         166       177       164       162       161       179       160
## 3        15.4      22.2      18.4      15.3      21.3      26.7      19.8
## 4  Mesomorfo Ectomorfo Mesomorfo Mesomorfo Mesomorfo Endomorfo Endomorfo
## 5         NO        NO        NO        NO        NO        NO        NO
## 6          4         0         2         2         1         3         5
## 7         NO        SI        SI        NO        NO        SI        NO
## 8       <NA>         9         8      <NA>      <NA>        10      <NA>
## 9       <NA>        48        40      <NA>      <NA>        70      <NA>
## 10      <NA>      5400      6000      <NA>      <NA>      3200      <NA>
##           X41       X42       X43       X44       X45       X46       X47
## 1       58.35     57.47     66.97     65.57     53.11     52.93     63.65
## 2         167       164       150       150       169       158       147
## 3        26.4      33.2      14.3      25.3      23.4        19      31.4
## 4  Mesomorfo Mesomorfo Endomorfo Endomorfo Mesomorfo Mesomorfo Mesomorfo
## 5         NO        NO        SI        NO        NO        NO        NO
## 6          5         1         3         2         3         1         2
## 7         SI        SI        NO        SI        SI        SI        SI
## 8          6         7      <NA>         6         7         6        10
## 9         30        35      <NA>        30        35        30        80
## 10      4500      1400      <NA>      3600      2450      2100     12000
##           X48       X49       X50
## 1       67.69     58.88     68.81
## 2         170       144       175
## 3        24.3      18.2        22
## 4  Endomorfo Mesomorfo Endomorfo
## 5         NO        NO        SI
## 6          1         2         4
## 7         SI        SI        SI
## 8          5         4        10
## 9         25        20        70
## 10      3750       800      9600
```

Como podemos observar esta base de datos cuenta con 10 variables o características, las cuales son: Peso, Estatura, Edad, Complexión, Papas separados, Cantidad hermanos, Trabaja, Horas trabajadas por día, Horas trabajadas por mes y salario mensual.

Además, es posible notar que se conforma de 50 observaciones, sin embargo es necesario reordenar los datos, ya que cada observación debe estar contenida en una fila y no en una columna (como se nos fue otorgada la información inicialmente).

Lo primero que realizamos fue la trasposición de las columnas del dataframe para que cada observación fuera vista en una fila y no por columnas como inicialmente estaba acomodada la información.

Otra cosa que debemos arreglar son las observaciones en donde el valor de la característica "Trabaja" es "NO", debido a que contienen variables relacionadas con la anteriormente mencionada (las cuales son: "Horas_trab_dia", "Horas_trab_mes" y "Salario_mes") y que se encuentran vacías. Esto último puede afectar nuestro análisis, es por eso que decidimos colocar ceros en aquellos campos en donde existen NA.

```r
# Trasponer las columnas del df para que las características de cada observacion
# estén ordenadas por columnas
df_transposed <- as.data.frame(t(as.matrix(df)))

# Colocar nombre a las columnas y eliminar la primera fila
names(df_transposed) <- c("Peso", "Estatura", "Edad", "Complexion", "Papas_separados",
    "Cantidad_hermanos", "Trabaja", "Horas_trab_dia", "Horas_trab_mes", "Salario_mes")

datos <- df_transposed[-c(1), ]

# Reemplazar valores NA con 0 solo en las columnas seleccionadas
vars_to_replace = c("Horas_trab_dia", "Horas_trab_mes", "Salario_mes")

datos = datos %>% mutate_at(.vars = vars_to_replace, .funs = funs(ifelse(is.na(.),
    0, .)))

datos
```

```
##     Peso Estatura Edad Complexion Papas_separados Cantidad_hermanos Trabaja
## 1  53.74      166 23.5  Mesomorfo              SI                 3      SI
## 2  61.84      154 22.4  Mesomorfo              SI                 0      SI
## 3  51.64      151   27  Mesomorfo              NO                 2      NO
## 4  75.95      180  2.3  Endomorfo              NO                 2      SI
## 5   63.3      165 17.2  Mesomorfo              NO                 2      NO
## 6   51.8      149 15.7  Mesomorfo              NO                 5      SI
## 7  64.87      145   20  Mesomorfo              NO                 1      NO
## 8  67.38      159 27.3  Endomorfo              NO                 4      SI
## 9  65.76      177 17.5  Endomorfo              NO                 0      NO
## 10 56.95      164 18.8  Mesomorfo              SI                 2      SI
## 11 75.12      179 21.8  Endomorfo              NO                 1      SI
## 12  63.9      169 23.2  Mesomorfo              NO                 1      SI
## 13 53.79      154 29.4  Mesomorfo              NO                 4      NO
## 14 37.85      157   30  Ectomorfo              SI                 4      SI
## 15 71.25      146 26.3  Endomorfo              SI                 1      SI
## 16 59.55      141 17.7  Mesomorfo              NO                 2      NO
## 17 59.84      169 23.2  Mesomorfo              NO                 0      SI
## 18 69.44      144 23.7  Endomorfo              SI                 2      NO
## 19 68.21      158 18.4  Endomorfo              NO                 5      SI
## 20 65.94      166 21.4  Endomorfo              NO                 3      SI
## 21 69.19      180 17.5  Endomorfo              NO                 3      SI
## 22 67.82      160 25.6  Endomorfo              NO                 2      SI
## 23 60.75      159 18.5  Mesomorfo              NO                 2      NO
## 24 40.11      147 19.7  Ectomorfo              NO                 5      SI
## 25  66.2      170 17.2  Endomorfo              NO                 4      SI
## 26 59.44      158   30  Mesomorfo              NO                 5      SI
## 27 58.44      160 18.5  Mesomorfo              NO                 4      SI
## 28 45.29      148 25.6  Ectomorfo              NO                 4      NO
## 29 55.22      149 26.7  Mesomorfo              NO                 1      SI
## 30 64.18      164   21  Mesomorfo              SI                 4      SI
## 31 73.59      163 18.9  Endomorfo              NO                 5      SI
## 32 58.97      143 16.3  Mesomorfo              NO                 1      SI
## 33 63.88      141 25.2  Mesomorfo              SI                 2      SI
## 34 59.46      166 15.4  Mesomorfo              NO                 4      NO
## 35 46.23      177 22.2  Ectomorfo              NO                 0      SI
```

```
## 36 55.85      164 18.4  Mesomorfo             NO                2      SI
## 37 56.06      162 15.3  Mesomorfo             NO                2      NO
## 38 59.41      161 21.3  Mesomorfo             NO                1      NO
## 39    71      179 26.7  Endomorfo             NO                3      SI
## 40 67.63      160 19.8  Endomorfo             NO                5      NO
## 41 58.35      167 26.4  Mesomorfo             NO                5      SI
## 42 57.47      164 33.2  Mesomorfo             NO                1      SI
## 43 66.97      150 14.3  Endomorfo             SI                3      NO
## 44 65.57      150 25.3  Endomorfo             NO                2      SI
## 45 53.11      169 23.4  Mesomorfo             NO                3      SI
## 46 52.93      158   19  Mesomorfo             NO                1      SI
## 47 63.65      147 31.4  Mesomorfo             NO                2      SI
## 48 67.69      170 24.3  Endomorfo             NO                1      SI
## 49 58.88      144 18.2  Mesomorfo             NO                2      SI
## 50 68.81      175   22  Endomorfo             SI                4      SI
##    Horas_trab_dia Horas_trab_mes Salario_mes
## 1               3              2           6
## 2               1              9          13
## 3               0              0           0
## 4               1              9          13
## 5               0              0           0
## 6               4              3           2
## 7               0              0           0
## 8               2              1           9
## 9               0              0           0
## 10              1              9           3
## 11              4              3          14
## 12              6              5          11
## 13              0              0           0
## 14              6              5          14
## 15              5              4          18
## 16              0              0           0
## 17              4              3          14
## 18              0              0           0
## 19              7              6          12
## 20              5              4          18
## 21              5              4          18
## 22              6              5           5
## 23              0              0           0
## 24              4              3           8
## 25              4              7           2
## 26              3              2           1
## 27              1              9          23
## 28              0              0           0
## 29              7              6          12
## 30              1              8          20
## 31              2              1           7
## 32              4              3           2
## 33              5              4          16
## 34              0              0           0
## 35              7              7          19
## 36              6              5          21
## 37              0              0           0
## 38              0              0           0
```

```
## 39              1              8          13
## 40              0              0           0
## 41              4              3          17
## 42              5              4           4
## 43              0              0           0
## 44              4              3          14
## 45              5              4          10
## 46              4              3           8
## 47              1              9           3
## 48              3              2          15
## 49              2              1          22
## 50              1              8          23
```

Para continuar con el análisis exploratorio, es momento de distinguir el tipo de dato correspondiente para cada variable que conforma esta base de datos. Esto con el fin de poder determinar cómo explorar la base de datos y si es necesario modificar algún tipo de dato para realizar tablas o gráficas relevantes.

```
# Analizar la base de datos
df_status(datos)
```

```
##             variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1               Peso       0       0    0    0     0     0  factor     50
## 2           Estatura       0       0    0    0     0     0  factor     28
## 3               Edad       0       0    0    0     0     0  factor     42
## 4         Complexion       0       0    0    0     0     0  factor      3
## 5    Papas_separados       0       0    0    0     0     0  factor      2
## 6  Cantidad_hermanos       4       8    0    0     0     0  factor      6
## 7            Trabaja       0       0    0    0     0     0  factor      2
## 8     Horas_trab_dia      14      28    0    0     0     0 numeric      8
## 9     Horas_trab_mes      14      28    0    0     0     0 numeric     10
## 10       Salario_mes      14      28    0    0     0     0 numeric     24
```

Existen 7 variables de tipo “factor” el cual es usado para variables categóricas o nominales, lo curioso es que de esas 7 variables, solo 3 (Complexión, Papas separados y Trabaja) son categóricas y las restantes (Peso, Estatura, Edad y Cantidad hermanos) deberían ser numéricas.

Para trabajar con las variables Horas_trab_dia, Horas_trab_mes y Salario_mes también se debe realizar un ajuste a su tipo de dato porque, como se pudo observar, al momento de cambiar los NA por ceros los valores almacenados en esos campos fueron alterados.

```
# Convertir en valores números aquellos almacenados en las columnas
# seleccionadas
vars_to_replace = c("Peso", "Estatura", "Edad", "Cantidad_hermanos")

datos[vars_to_replace] <- lapply(datos[vars_to_replace], function(x) as.numeric(as.character(x)))

# Cargamos csv con los valores previamente almacenados en los campos
# Horas_trab_dia, Horas_trab_mes y Salario_mes
df2 <- read.csv("Ejercicio 1.csv", sep = ",")

# Trasponer las columnas del df para que las características de cada observacion
# estén ordenadas por columnas
df2 <- as.data.frame(t(as.matrix(df2)))
```

```r
# Colocar nombre a las columnas y eliminar la primera fila
names(df2) <- c("Peso", "Estatura", "Edad", "Complexion", "Papas_separados", "Cantidad_hermanos",
    "Trabaja", "Horas_trab_dia", "Horas_trab_mes", "Salario_mes")

df2 <- df2[-c(1), ]

datos$Horas_trab_dia <- as.numeric(as.numeric_version(df2$Horas_trab_dia))
datos$Horas_trab_mes <- as.numeric(as.numeric_version(df2$Horas_trab_mes))
datos$Salario_mes <- as.numeric(as.numeric_version(df2$Salario_mes))

status(datos)
```

```
##             variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1               Peso       0    0.00    0    0     0     0 numeric     50
## 2           Estatura       0    0.00    0    0     0     0 numeric     28
## 3               Edad       0    0.00    0    0     0     0 numeric     42
## 4         Complexion       0    0.00    0    0     0     0  factor      3
## 5    Papas_separados       0    0.00    0    0     0     0  factor      2
## 6  Cantidad_hermanos       4    0.08    0    0     0     0 numeric      6
## 7            Trabaja       0    0.00    0    0     0     0  factor      2
## 8     Horas_trab_dia      14    0.28    0    0     0     0 numeric      8
## 9     Horas_trab_mes      14    0.28    0    0     0     0 numeric     10
## 10       Salario_mes      14    0.28    0    0     0     0 numeric     24
```

Otra corrección que se debe realizar es hacer función piso a los valores de Edad, ya que estos fueron ingresados como decimales y eso es erróneo porque deben ser considerados como discretos.

```r
datos["Edad"] <- floor(datos["Edad"])
```

1.2 Expliquen por medio de tablas esta información. En la tabla debe aparecer simultáneamente la frecuencia y el porcentaje

Es importante recalcar que continuamos realizando análisis exploratorio de la base de datos y estamos respetando las instrucciones de cada subinciso, es por esto que ahora elaboraremos tablas que contengan información sobre la frecuencia y porcentaje de aquellas variables que lo requieren, como lo son: Complexión, Papás separados, Cantidad hermanos, Trabaja y Horas diarias trabajadas.

```r
# Tabla de frecuencias de complexion
frequency_as_df <- as.data.frame(tabyl(datos$Complexion, sort = TRUE))

names(frequency_as_df) <- c("Complexion", "Frecuencia", "Porcentaje")
final_constitution_frequency <- frequency_as_df[-c(1), ]

final_constitution_frequency
```

```
##   Complexion Frecuencia Porcentaje
## 2  Ectomorfo          4       0.08
## 3  Endomorfo         18       0.36
## 4  Mesomorfo         28       0.56
```

A partir de esta tabla podemos inferir que el 56% de las personas que se encuentran registradas en la base de datos tienen un cuerpo mesomorfo, es decir, su complexión es intermedia (ni muy delgada ni muy gruesa) y se caracterizan por tener un metabolismo regular.

Por otro lado se observa que el 36% de las observaciones tienen un cuerpo endomorfo, lo que significa que desgraciadamente sufren de sobrepeso.

Y aquellos con complexión ectomorfa (muy delgada) solo cubren el 8% de nuestras observaciones.

En conclusión se puede decir que más de la mitad de las personas se encuentran saludables y con una buena complexión, sin embargo el 44% restante debería acudir con un médico y estar al pendiente de su salud.

```
# Tabla de frecuencias de papás_separados
frequency_as_df <- as.data.frame(tabyl(datos$Papas_separados, sort = TRUE))

names(frequency_as_df) <- c("Papás_separados", "Frecuencia", "Porcentaje")
final_separated_parents_frequency <- frequency_as_df[-c(2), ]

final_separated_parents_frequency
```

```
##   Papás_separados Frecuencia Porcentaje
## 1              NO         40        0.8
## 3              SI         10        0.2
```

Sabemos que solo el 20% de los individuos registrados en la base de datos tiene padres separados, esto puede ser por diversas razones, las cuales desconocemos. Esta información puede ser útil para evaluar si el desempeño de una persona o nivel de carga de trabajo se ve influenciado por la relación de sus progenitores.

```
# Tabla de frecuencias de cantidad_hermanos
frequency_as_df <- as.data.frame(tabyl(datos$Cantidad_hermanos, sort = TRUE))

names(frequency_as_df) <- c("Cantidad_hermanos", "Frecuencia", "Porcentaje")
final_siblings_frequency <- frequency_as_df[-c(7), ]

final_siblings_frequency
```

```
##   Cantidad_hermanos Frecuencia Porcentaje
## 1                 0          4       0.08
## 2                 1         10       0.20
## 3                 2         14       0.28
## 4                 3          6       0.12
## 5                 4          9       0.18
## 6                 5          7       0.14
```

Con esta tabla se puede inferir que en esta base de datos predominan las observaciones con menos de 3 hermanos, mientras que aquellas observaciones con más de 3 hermanos son menos frecuentes. El número de hermanos es un dato relevante para estudiar si el nivel socioeconómico de una persona se ve afectado por la cantidad de hijos que deciden criar sus padres.

```
# Tabla de frecuencias de los que trabajan
frequency_as_df <- as.data.frame(tabyl(datos$Trabaja, sort = TRUE))

names(frequency_as_df) <- c("Trabaja", "Frecuencia", "Porcentaje")
final_iswork_frequency <- frequency_as_df[-c(3), ]

final_iswork_frequency
```

```
##   Trabaja Frecuencia Porcentaje
## 1      NO         14       0.28
## 2      SI         36       0.72
```

De igual forma es importante destacar que el 72% de las personas registradas en la base de datos sí trabaja, mientras que el 28% restante no lo hace, esto puede ser debido a que esos individuos son menores de edad o por factores que no son posibles determinar con la información otorgada por la base de datos.

```
# Filtro de las observaciones que SI trabajan

trabajan <- filter(datos, Trabaja == "SI")

# Tabla de frecuencias de horas trabajadas x dia

frequency_as_df <- as.data.frame(tabyl(trabajan$Horas_trab_dia, sort = TRUE))

names(frequency_as_df) <- c("Horas_trabajo x dia", "Frecuencia", "Porcentaje")

frequency_as_df
```

```
##   Horas_trabajo x dia Frecuencia Porcentaje
## 1                   4          3 0.08333333
## 2                   5          3 0.08333333
## 3                   6          9 0.25000000
## 4                   7          6 0.16666667
## 5                   8          4 0.11111111
## 6                   9          3 0.08333333
## 7                  10          8 0.22222222
```

Con esta tabla podemos observar que las personas registradas en la base de datos que sí trabajan, lo hacen en promedio entre 1 a 7 horas al día y que el 25% de ellas prefiere laborar medio día (4 horas).

Continuando con el análisis, es momento de enfocarnos en las otras variables de la base de datos (Peso, Estatura, Edad, Horas trabajadas al mes y Salario mensual), aquellas que no colocamos en tablas pues la frecuencia de sus datos es de 1 y sus tablas de frecuencia y porcentajes resultaban poco agradables visualmente.

Lo que haremos con ellas es visualizar un resumen de las mismas.

```
summary(datos$Peso)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   37.85   56.28   61.30   61.01   67.28   75.95
```

```
sd(datos$Peso)
```

```
## [1] 8.313902
```

Podemos notar que en la base de datos el peso más bajo registrado es de 37.85 Kg (que es probable que corresponda a una persona con complexión ectomorfa) y el máximo es de 75.95 Kg. Además, existe una desviación estándar de 8.31 Kg en el peso de las observaciones.

```r
summary(datos$Estatura)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   141.0   150.0   160.0   160.0   166.8   180.0
```

```r
sd(datos$Estatura)
```

```
## [1] 11.13179
```

La persona con la menor estatura de la base de datos mide 141 cm, mientras que la más alta mide 180 cm. Se cuenta con una desviación estándar de 11.13 cm, lo que significa que hay mucha variación en las estaturas registradas.

```r
summary(datos$Edad)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     2.0    18.0    21.0    21.3    25.0    33.0
```

```r
sd(datos$Edad)
```

```
## [1] 5.406912
```

La persona más joven de la base de datos tiene 2 años de edad, mientras que la más grande tiene 33 años. La variable de Edad tiene una desviación estándar de 5 años.

```r
summary(trabajan$Horas_trab_mes)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   20.00   30.00   35.00   42.81   48.00   80.00
```

```r
sd(trabajan$Horas_trab_mes)
```

```
## [1] 19.58107
```

El mínimo de horas laboradas mensualmente es de 20 y el máximo registrado es de 80 horas. Se cuenta con una desviación estándar de casi 20 horas, la cual es muy amplia.

```r
summary(trabajan$Salario_mes)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     800    2025    3200    3958    5250   12000
```

```r
sd(trabajan$Salario_mes)
```

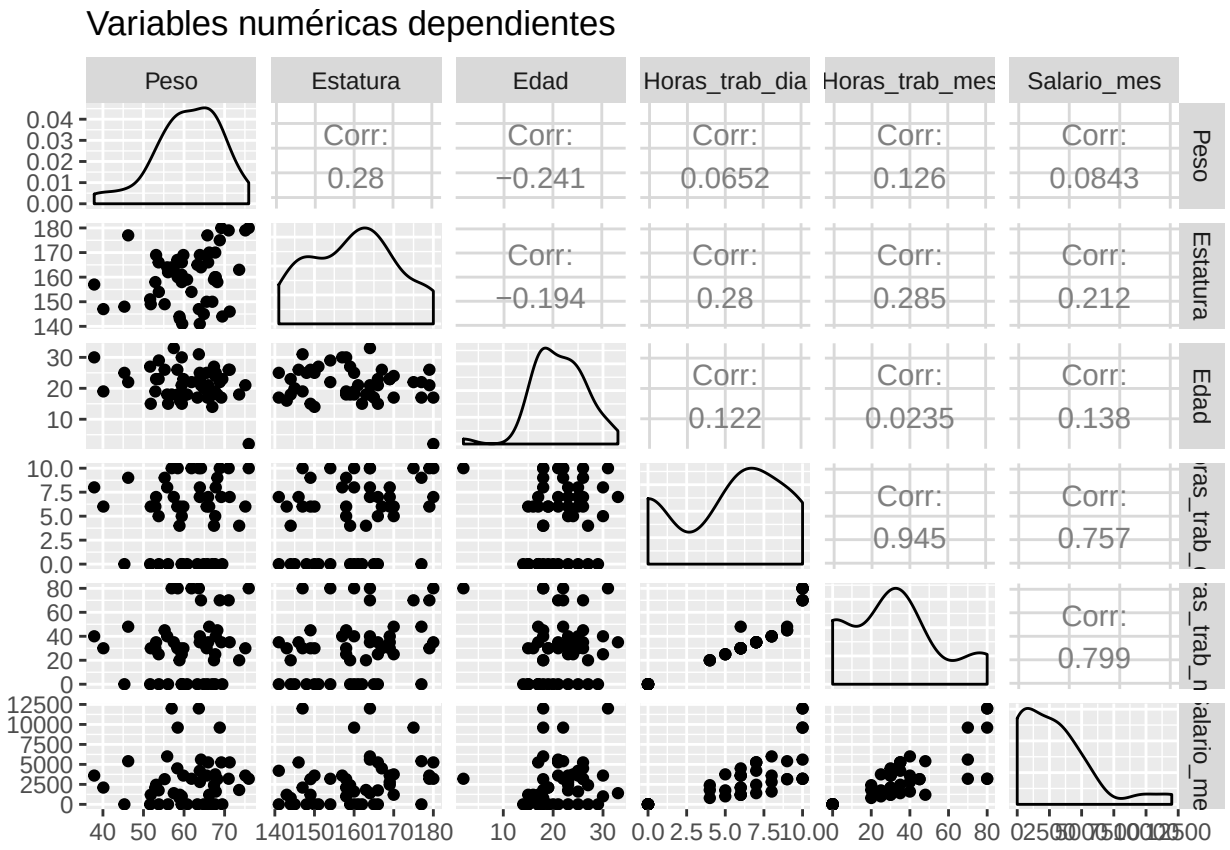
```
## [1] 2863.701
```

El salario mensual mínimo de las observaciones registradas es de 800 pesos, mientras que el máximo es de 12000 pesos. La variable de salario al mes tiene una desviación estándar de 2863.7 pesos, que tomando en cuenta nuestro rango de valores (800 - 12000) se puede decir que es muy amplia.

El promedio de 3958 pesos es probable que esté influenciado por datos atípicos

1.3 Hagan un gráfico donde pueda explicar las variables dependientes.

```
numericas <- datos %>% dplyr::select(Peso, Estatura, Edad, Horas_trab_dia, Horas_trab_mes,
    Salario_mes)
ggpairs(numericas, title = "Variables numéricas dependientes")
```



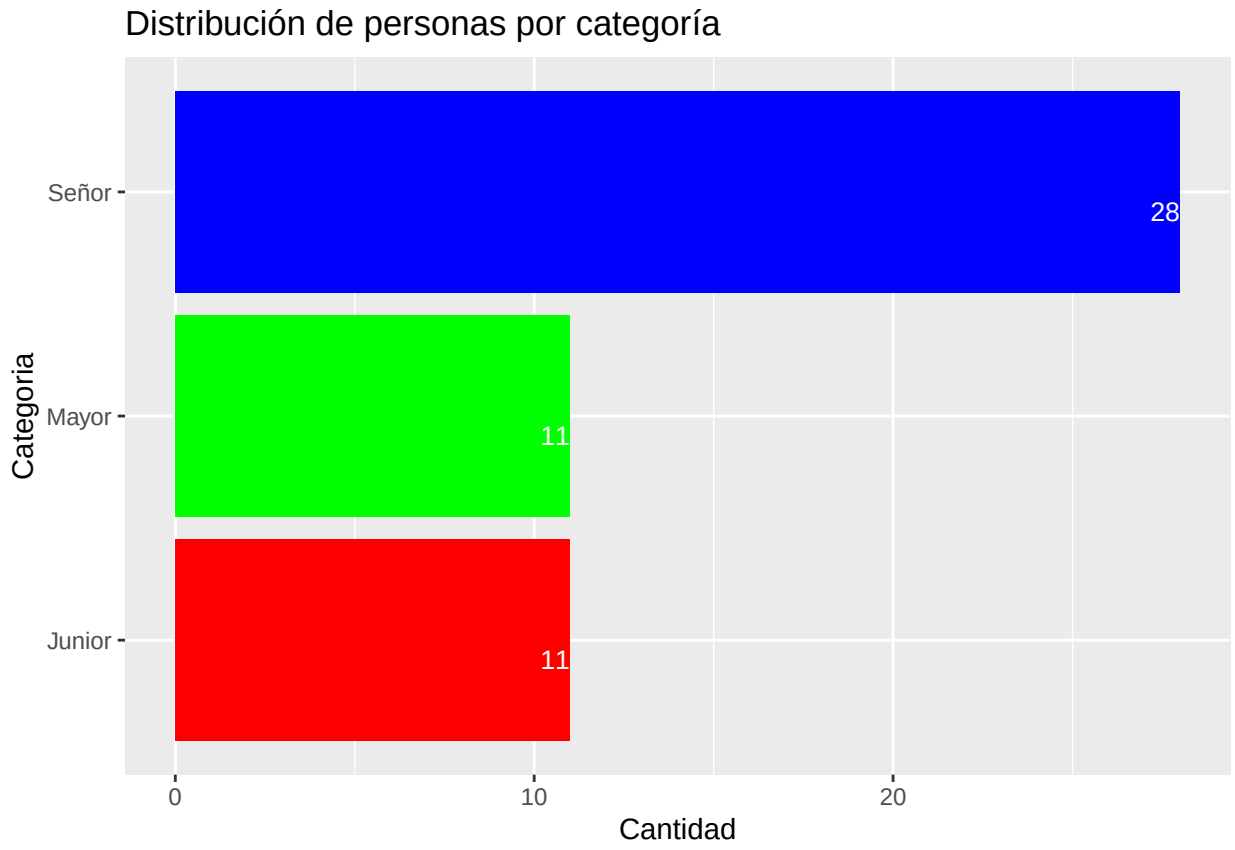
Con esta gráfica podemos notar que la correlación entre las variables numéricas que conforman la base de datos es, en su mayoría, un tanto débil. En el caso del salario mensual con respecto a la variable de horas laboradas al mes se observa una correlación de 0.799, la cual es de las más altas vistas en la gráfica, pero no es sorpresa porque sabemos que en ocasiones el sueldo mensual depende de las horas que se laboran en ese periodo, como es en el caso de esta base de datos.

Y otra correlación que desde un principio era evidente que sería muy fuerte es la de las horas trabajadas al día con respecto a las horas trabajadas al mes, la cual tien un valor de 0.945.

1.4 Construyan diferentes gráficos donde puedan mostrar patrones y detalles de la base de datos con respecto a una nueva variable que podemos denominar: "categoría"

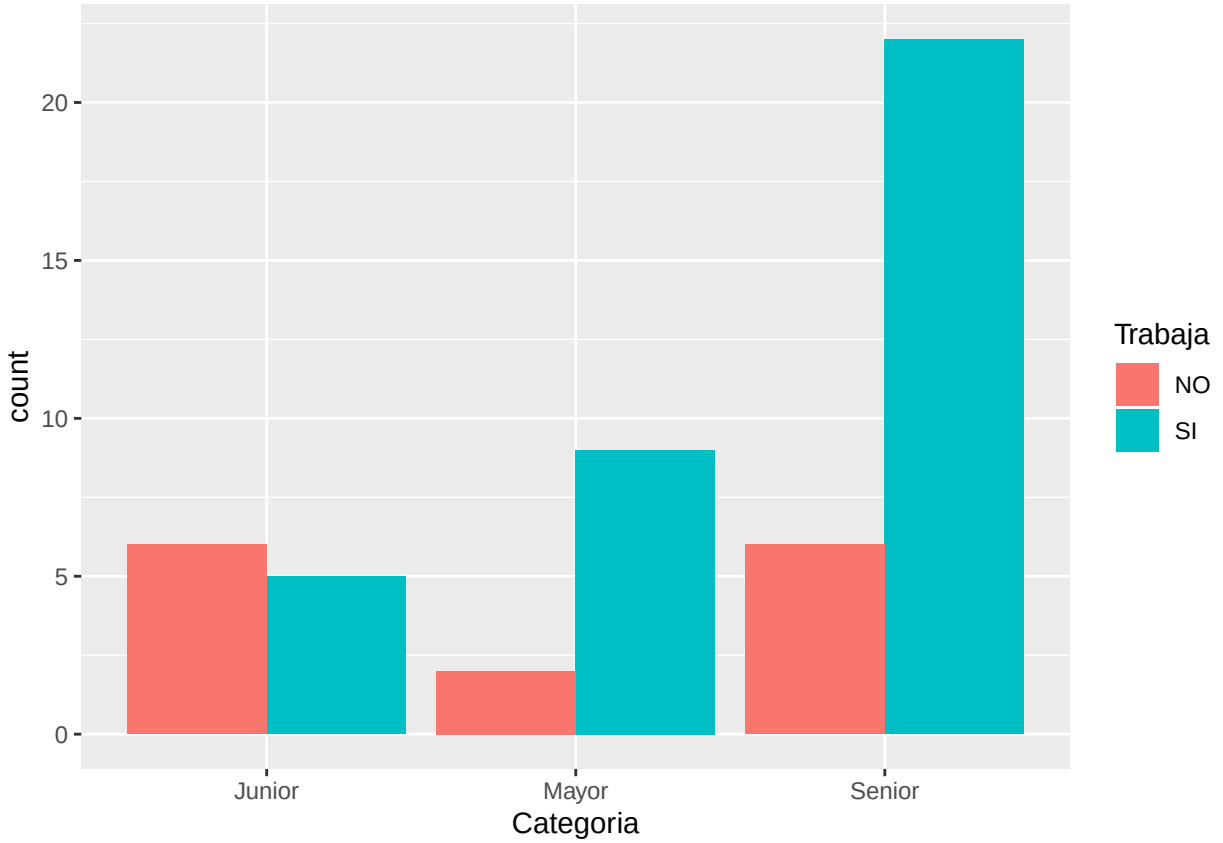
```
# Número de personas por categoría
cantidad_personas <- c(sum(with(datos, Edad < 18)), sum(with(datos, Edad >= 18 &
    Edad <= 25)), sum(with(datos, Edad > 25)))
```

```r
dist_personas <- data.frame(Categoria = c("Junior", "Señor", "Mayor"), Cantidad = cantidad_personas)
ggplot(dist_personas, aes(x = Categoria, y = Cantidad)) + geom_bar(stat = "identity",
    fill = rainbow(3)) + coord_flip() + labs(title = "Distribución de personas por categoría") +
    geom_text(aes(y = Cantidad, label = cantidad_personas), vjust = 1.5, color = "white",
        size = 3.5, hjust = "right")
```



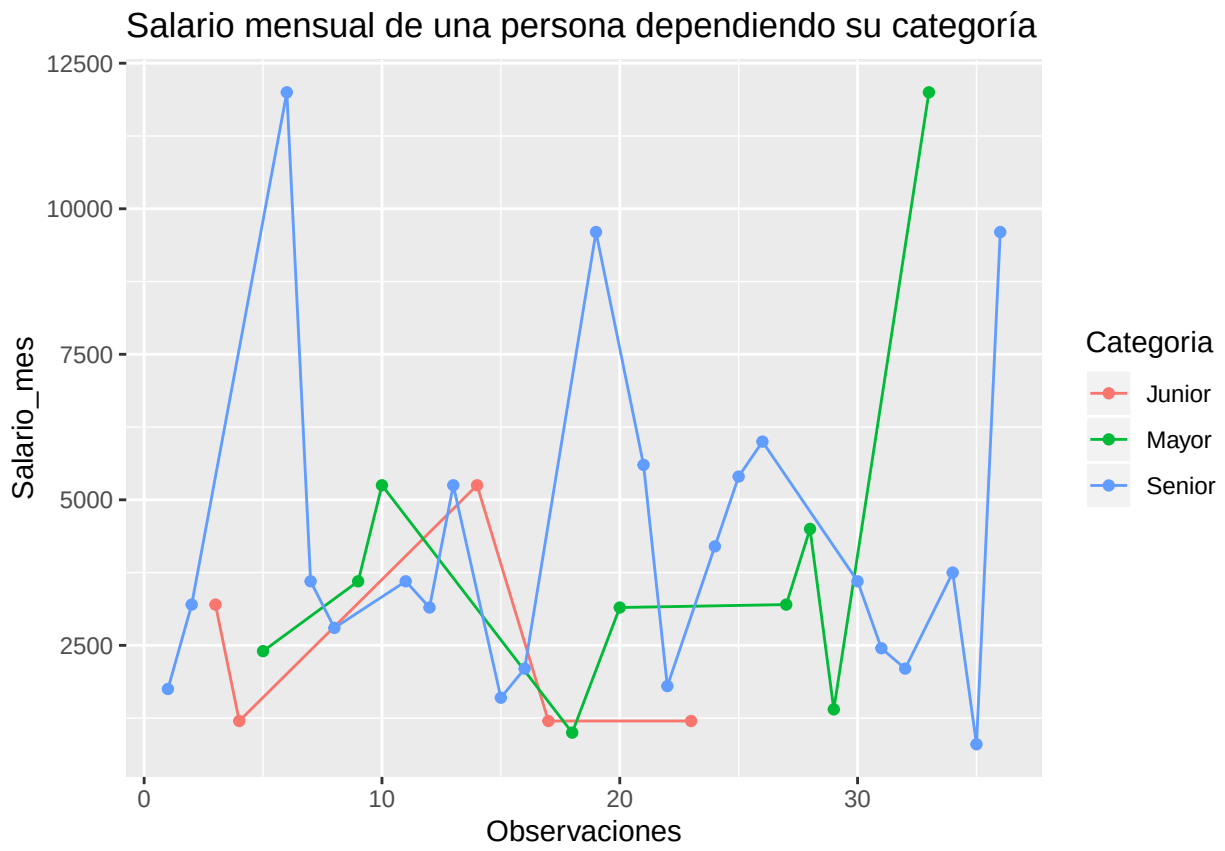
Como se puede observar el grupo con más personas es el de "Señor", lo que significa que más del 50% de las personas registradas en la base de datos tienen edades entre 18 a 25 años. En cambio, en las otras 2 categorías existen 22 personas, 11 respectivamente.

```r
# Personas que trabajan dependiendo su categoría
datos$Categoria[datos$Edad < 18] = "Junior"
datos$Categoria[datos$Edad >= 18 & datos$Edad <= 25] = "Senior"
datos$Categoria[datos$Edad > 25] = "Mayor"


ggplot(datos, aes(Categoria, ..count..)) + geom_bar(aes(fill = Trabaja), position = "dodge")
```



Como era de esperarse, en la categoría junior hay más personas que no trabajan. En el caso de la categoría Señor las personas que sí trabajan triplican a la cantidad que no hace.

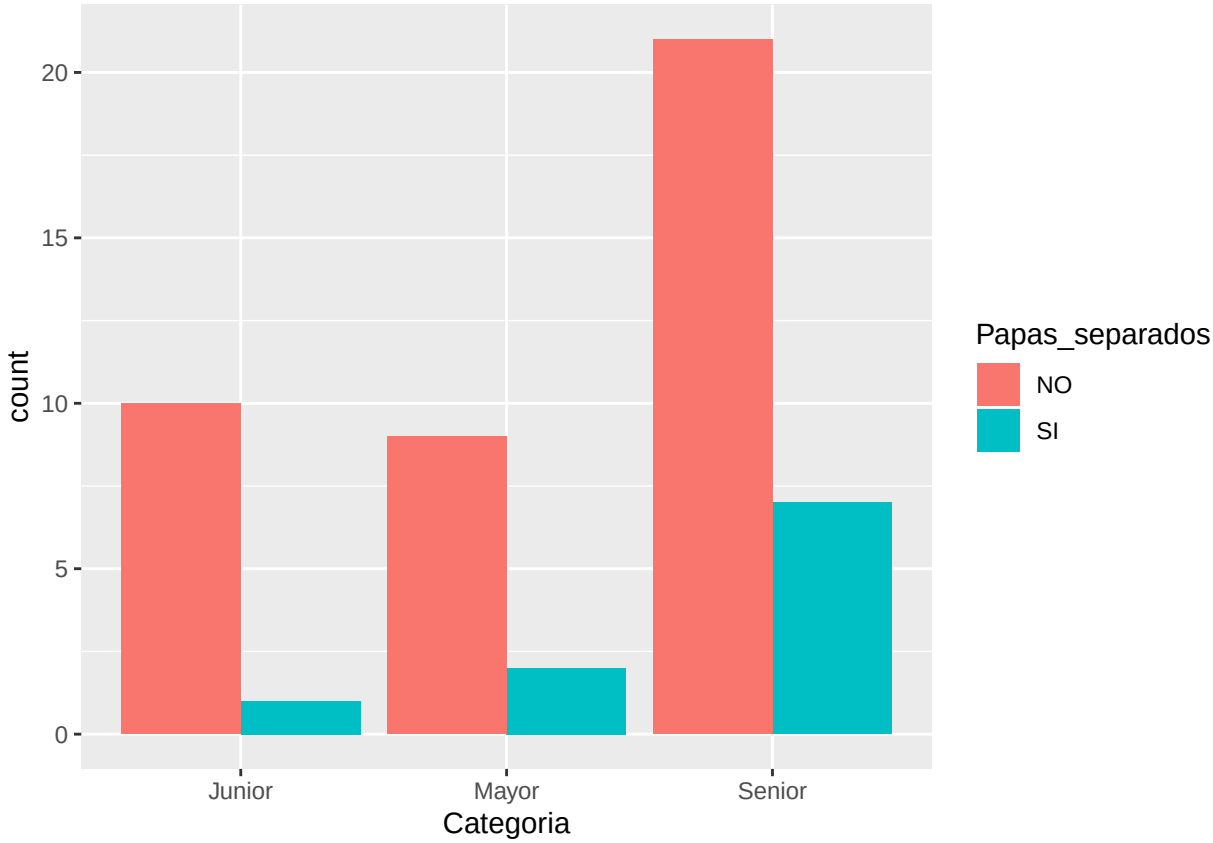
```
# Salario mensual de una persona dependiendo su categoría
trabajan <- filter(datos, Trabaja == "SI")

trabajan$Categoria[trabajan$Edad < 18] = "Junior"
trabajan$Categoria[trabajan$Edad >= 18 & trabajan$Edad <= 25] = "Senior"
trabajan$Categoria[trabajan$Edad > 25] = "Mayor"

Observaciones <- c(1:36)

ggplot(trabajan, aes(x = Observaciones, y = Salario_mes, col = Categoria)) + labs(title = "Salario mensu
    geom_line() + geom_point(aes(x = Observaciones, y = Salario_mes))
```



Para realizar esta gráfica omitimos aquellas personas que no trabajan de cada categoría. Lo anterior hace posible notar que el grupo con salarios más bajos es el de Junior, mientras que en el de Mayor existe un dato atipico cuyo salario asciende a los 12000 pesos cuando el resto de observaciones de dicha categoría no superan los 5500 pesos.

Se podría decir que la categoría con mejores salarios es la de Señor, pues supera (en algunos casos) a las otras dos agrupaciones.

```
# Número de personas con padres separados por categoría

ggplot(datos, aes(Categoria, ..count..)) + geom_bar(aes(fill = Papas_separados),
    position = "dodge")
```



Con esta gráfica es posible notar que todas las categorías tienen algo en común, y eso es que predominan personas cuyos padres no están separados.

2. Metodo de reducción de dimensión (PCA)

Antes de aplicar el método PCA para reducir la dimensión de un conjunto de datos, vamos a realizar un breve análisis exploratorio a la base de datos del inciso 2.1.

```
df2 <- read.csv("Ejercicio 2.1.csv", sep = ",")
status(df2)
```

```
##   variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1     tipo       0       0    0    0     0     0  factor      3
## 2    largo       0       0    0    0     0     0 integer     18
## 3    ancho       0       0    0    0     0     0 integer     24
## 4     alto       0       0    0    0     0     0 integer     10
```

La base de datos contiene 4 variables, las cuales son: tipo, largo, ancho y alto. Además, cabe destacar que esta constituida por 300 observaciones.

```
frecuency_as_df <- as.data.frame(tabyl(df2$tipo, sort = TRUE))

names(frecuency_as_df) <- c("Tipo", "Frecuencia", "Porcentaje")

frecuency_as_df
```

```
##     Tipo Frecuencia Porcentaje
## 1   cama        100  0.3333333
## 2 mueble        100  0.3333333
## 3  silla        100  0.3333333
```

Como podemos observar existen 3 tipos: cama, mueble y silla. Y de cada grupo hay 100 observaciones, es decir su porcentaje de frecuencia es el mismo, 33%.

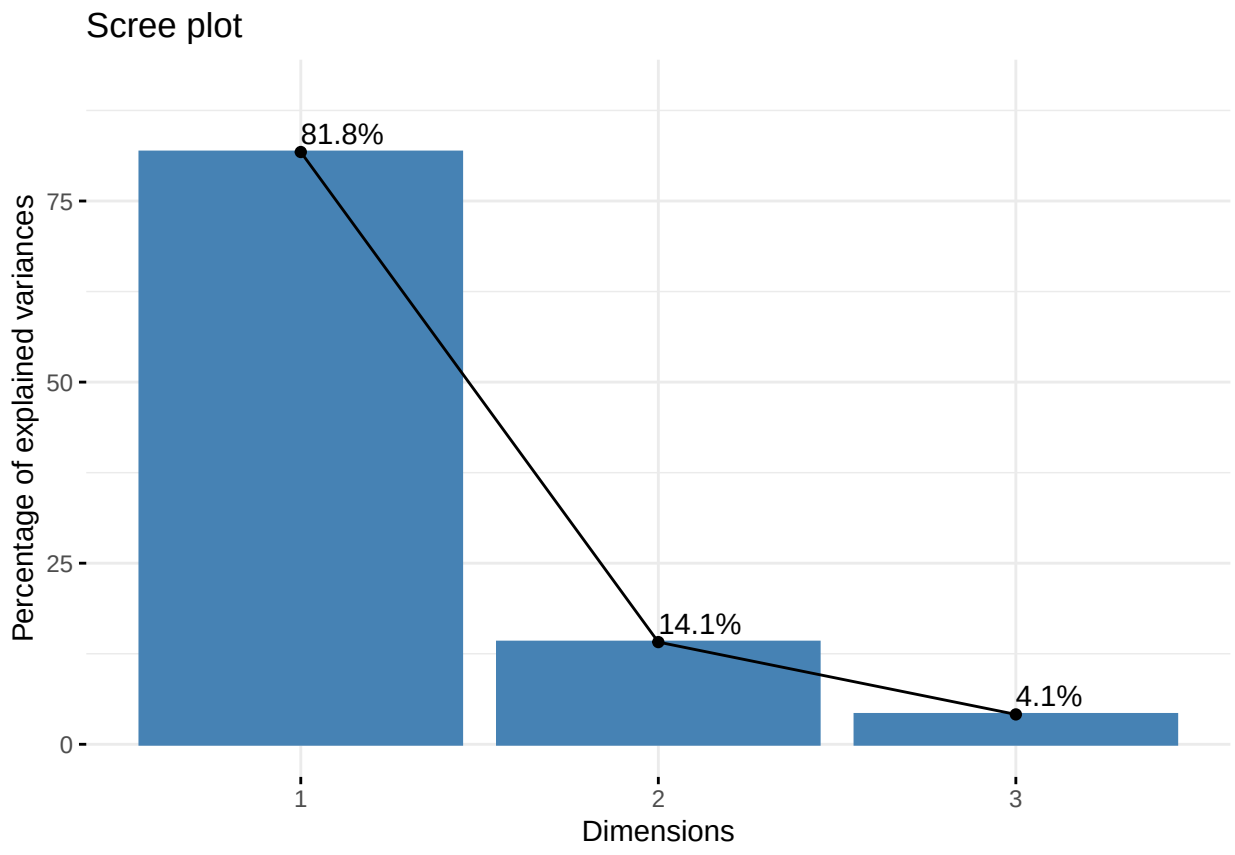
2.1 PCA base de datos PCA1

Para esta base de datos decidimos aplicar el modelo PCA Como primer paso sacamos el porcentaje de varianza y la graficamos. Una vez ya obtenido este punto proseguimos a graficar la correlación.

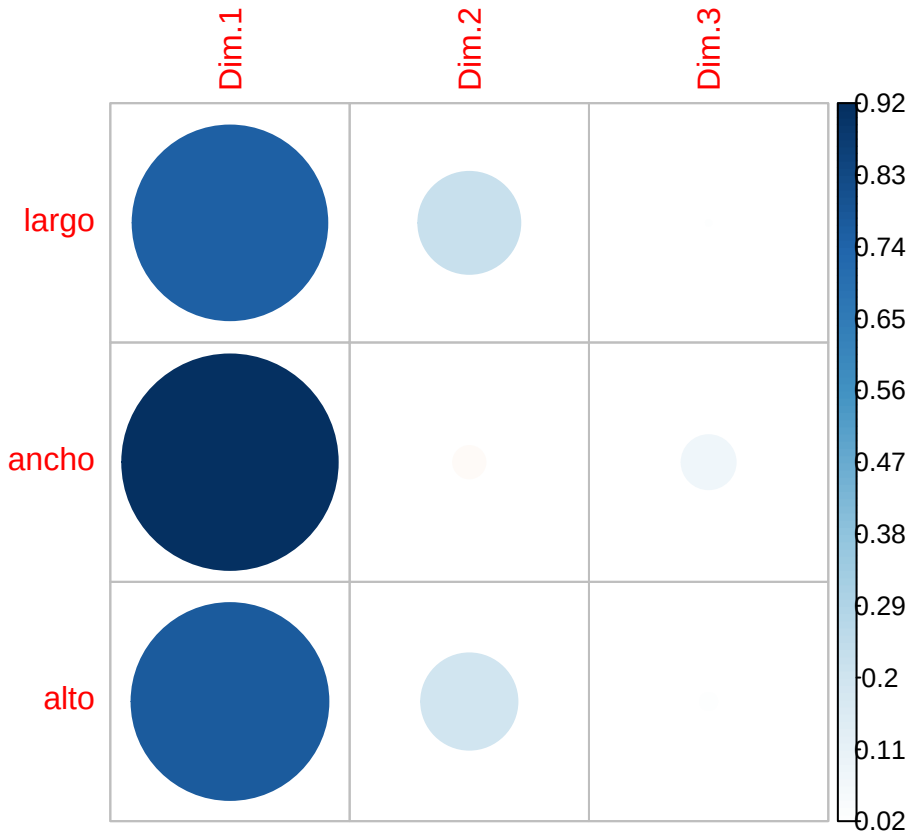
```
res.pca <- PCA(df2[, -1], graph = F)

eig.val <- get_eigenvalue(res.pca)
eig.val
```

```
##       eigenvalue variance.percent cumulative.variance.percent
## Dim.1  2.4529147        81.763823                    81.76382
## Dim.2  0.4232700        14.109001                    95.87282
## Dim.3  0.1238153         4.127176                   100.00000
```

```
fviz_eig(res.pca, addlabels = TRUE, ylim = c(0, 90))
```

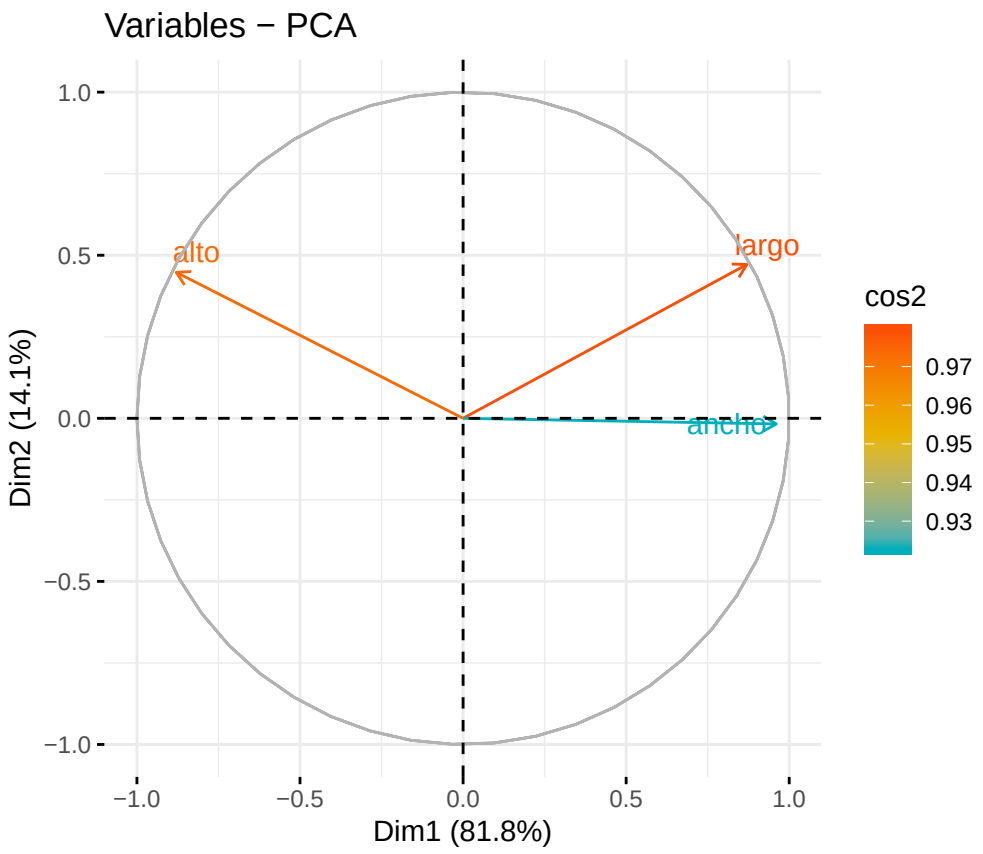


```
var <- get_pca_var(res.pca)

corrplot(var$cos2, is.corr = FALSE)
```

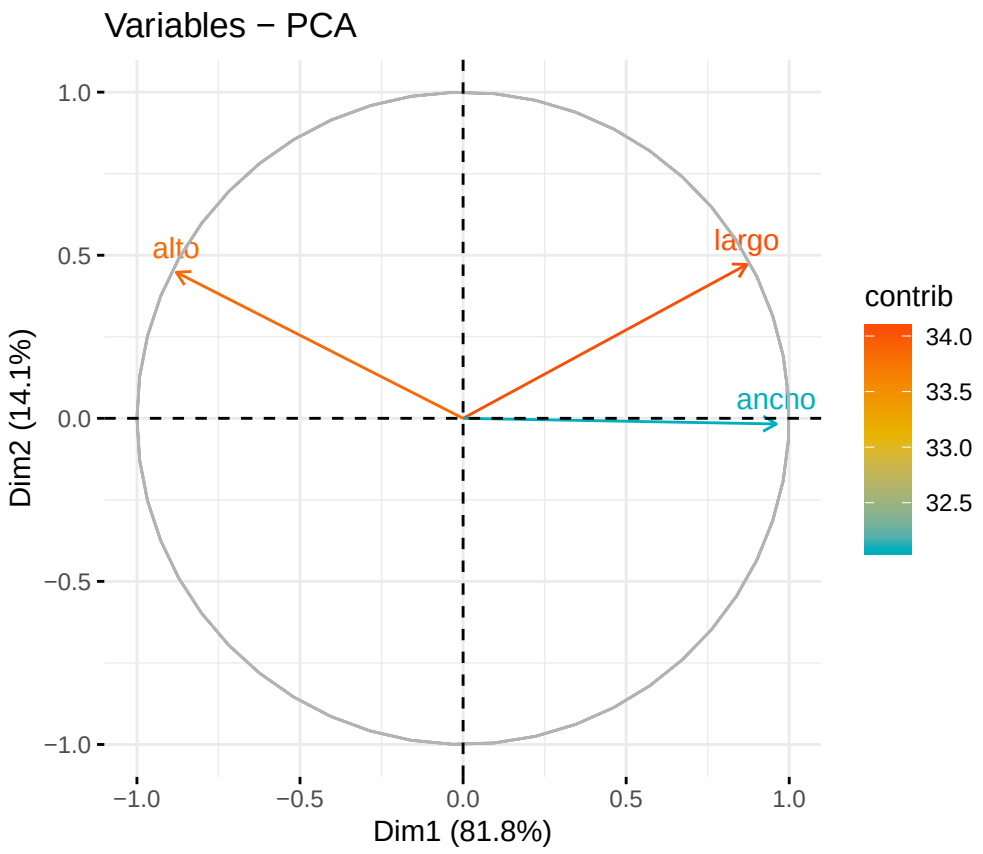


Proseguimos a graficar las variables y de igual manera graficamos las observaciones.

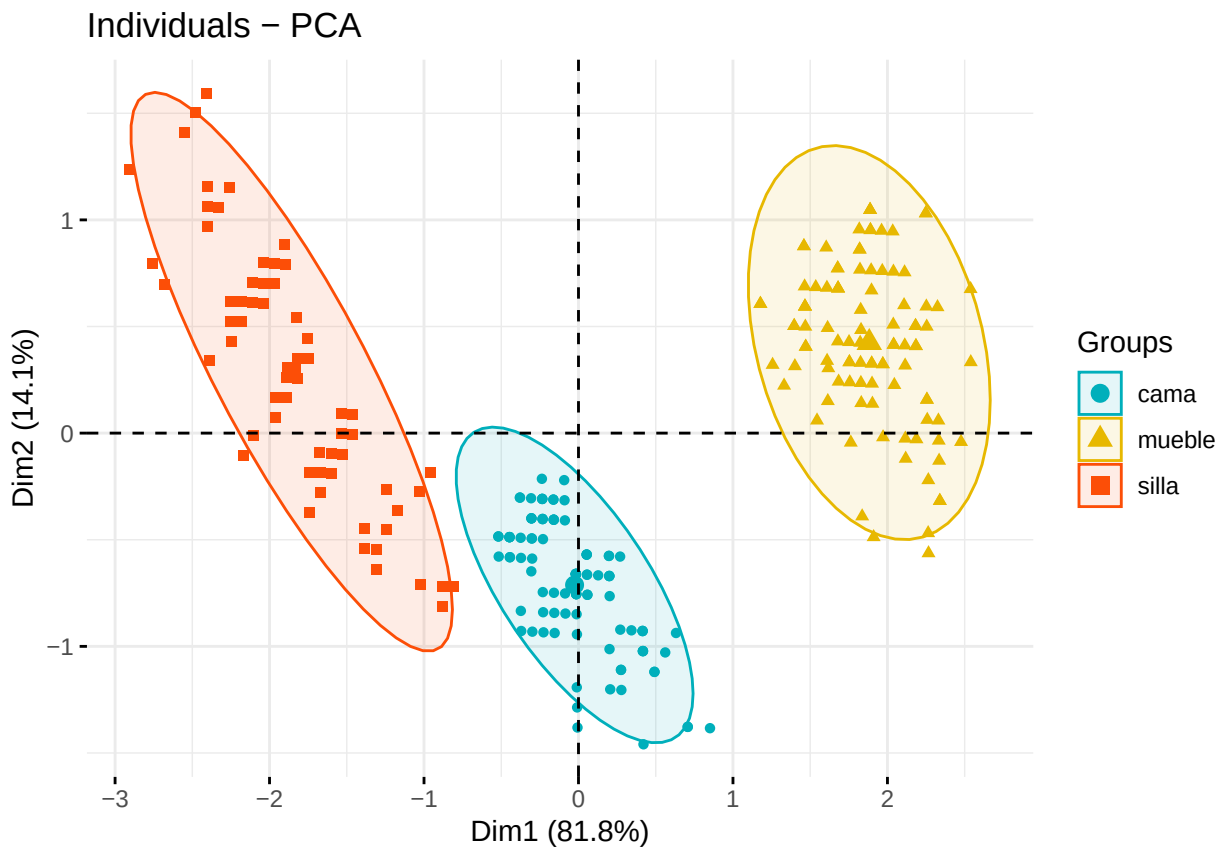
```
fviz_pca_var(res.pca, col.var = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE # Avoid text overlapping
)
```




```
fviz_pca_var(res.pca, col.var = "contrib",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07")
)
```




```
fviz_pca_ind(res.pca,
             geom.ind = "point", # show points only (nbut not "text")
             col.ind =df2$tipo, # color by groups
             palette = c("#00AFBB", "#E7B800", "#FC4E07"),
             addEllipses = TRUE, # Concentration ellipses
             legend.title = "Groups")
```



El modelo logró agrupar correctamente las observaciones de la base de datos.

2.2 PCA base de datos PCA2

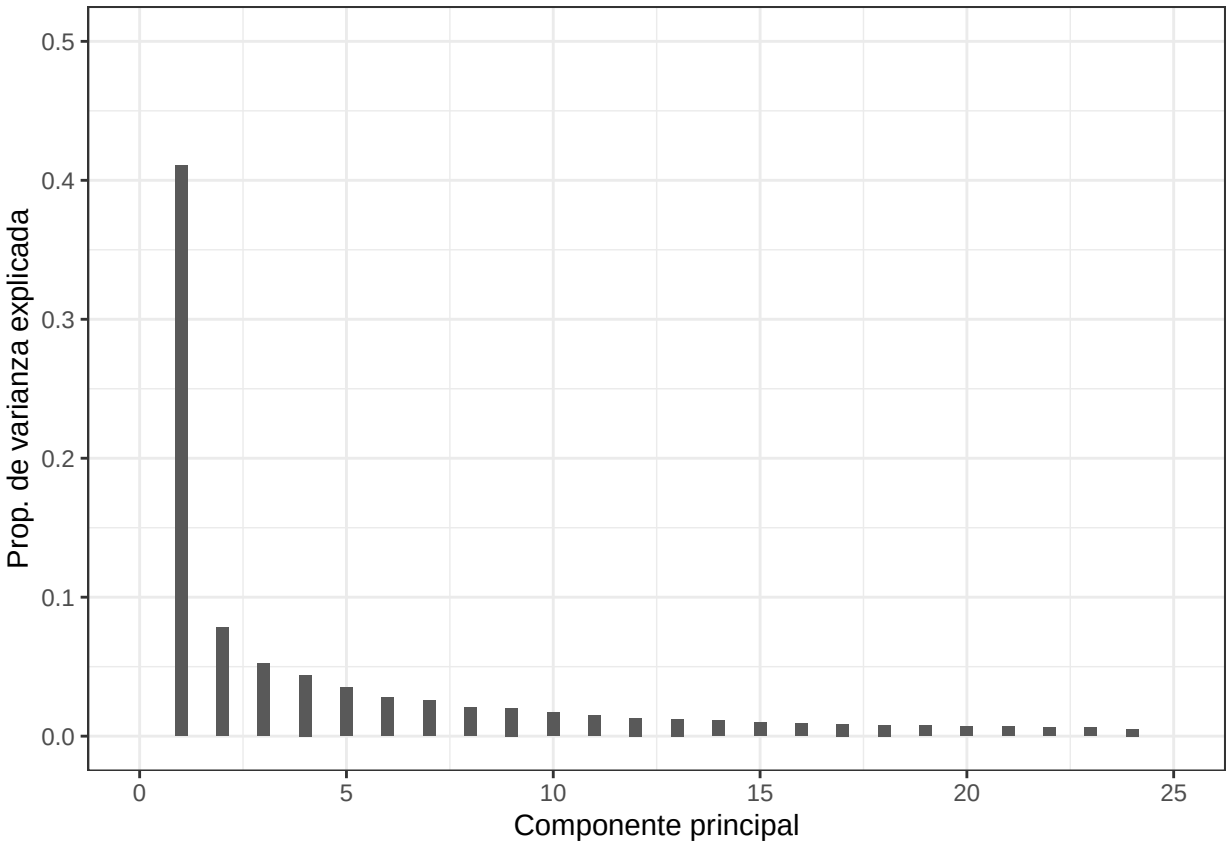
Antes de aplicar el método PCA para reducir la dimensión de un conjunto de datos, vamos a realizar un breve análisis exploratorio a la base de datos del inciso 2.1.

```
url <- "http://web.stanford.edu/~hastie/ElemStatLearn//datasets/zip.digits/train.2"
data <- read.csv(url)
```

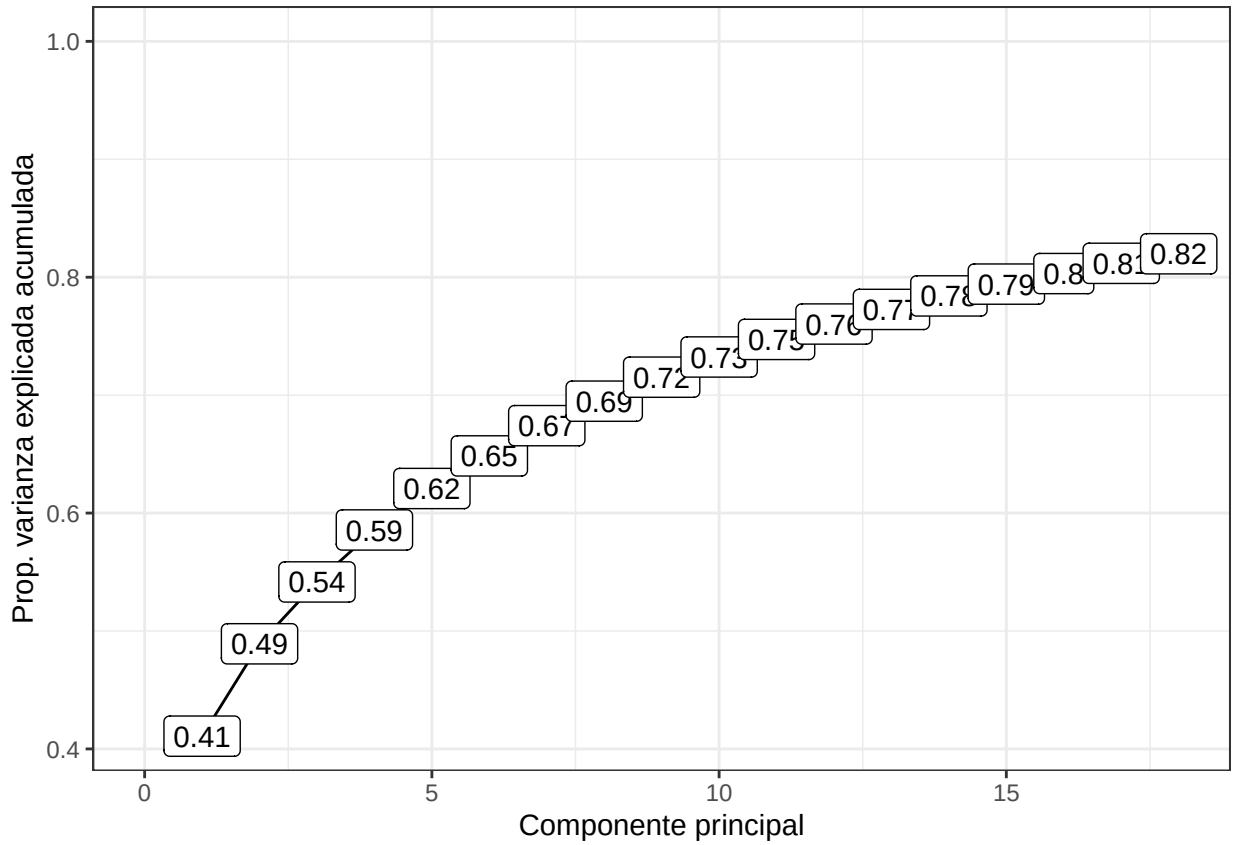
Esta base de datos es más grande que las anteriores, cuenta con 730 observaciones y 256 variables, mismas que se pretende reducir a través del modelo PCA, con el cual obtendremos el número óptimo de componentes principales que describen la mayor cantidad de información del conjunto de datos sin necesidad de utilizar una enorme cantidad de predictores.

Graficando la varianza acumulada, y guiados por el método del codo, podremos obtener lo siguiente:

```
data.pca = scale(data)
data.pca <- prcomp(data, center = F, scale = T)
prop_varianza <- data.pca$sdev^2/sum(data.pca$sdev^2)
prop_varianza_acum <- cumsum(prop_varianza)

ggplot(data = data.frame(prop_varianza, pc = 1:256), aes(x = pc, y = prop_varianza)) +
    xlim(0, 25) + geom_col(width = 0.3) + scale_y_continuous(limits = c(0, 0.5)) +
    theme_bw() + labs(x = "Componente principal", y = "Prop. de varianza explicada")
```




```
ggplot(data = data.frame(prop_varianza_acum, pc = 1:256), aes(x = pc, y = prop_varianza_acum,
    group = 1)) + xlim(0, 18) + geom_point() + geom_line() + geom_label(aes(label = round(prop_varianza_
    2))) + theme_bw() + labs(x = "Componente principal", y = "Prop. varianza explicada acumulada")
```



Guiados por la primera gráfica se podría pensar que el número más optimo es de 10 aproximadamente, sin embargo al observar la gráfica 2 es posible notar que un valor más acertado es 18, con el cual se describe un 82% de los datos originales y con lo cual estaríamos reduciendo bastante la dimensionalidad de la base de datos que inicialmente era de 256.

3. Metodo de reducción de dimensión (LDA)

Primeramente debemos realizar un breve análsis exploratorio de la base de datos para entender mejor la información con la que trabajaremos.

```
datos <- read.csv("Ejercicio 3.csv", sep = ",")
df_status(datos)
```

```
##   variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1     tipo       0       0    0    0     0     0  factor      3
## 2    largo       0       0    0    0     0     0 integer     22
## 3    ancho       0       0    0    0     0     0 integer     30
## 4     alto       0       0    0    0     0     0 integer      5
```

Es posible notar que, al igual que la base de datos del inciso 2.1, contiene 4 variables. Las cuales son: tipo, largo, ancho y alto. También se observó que cuenta con 300 observaciones.

```r
frecuency_as_df <- as.data.frame(tabyl(datos$tipo, sort = TRUE))

names(frecuency_as_df) <- c("Tipo", "Frecuencia", "Porcentaje")

frecuency_as_df
```

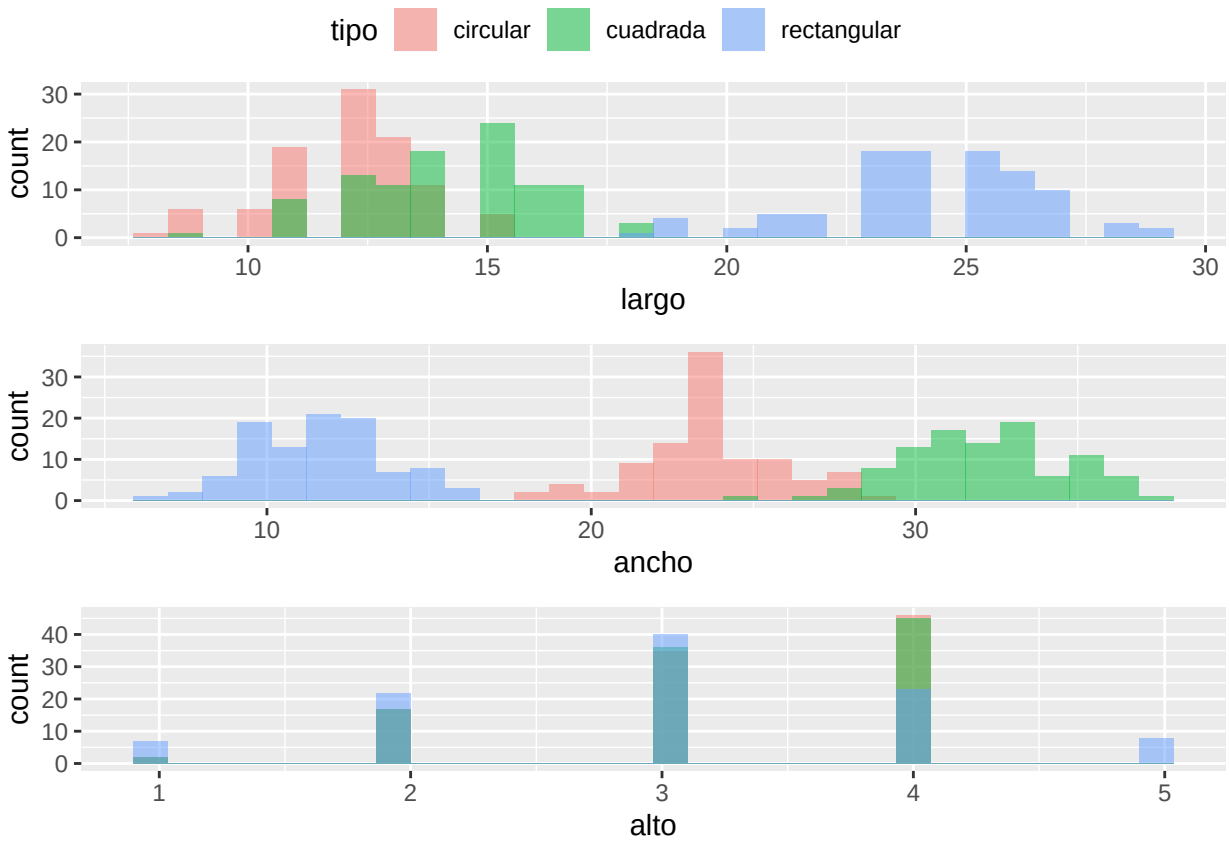
```
##          Tipo Frecuencia Porcentaje
## 1    circular        100  0.3333333
## 2    cuadrada        100  0.3333333
## 3 rectangular        100  0.3333333
```

Tipo es una variable nominal que contiene 3 valores: circular, cuadrada y rectangular. Dichos tipos o categorias se encuentran presentes en la base de datos por partes iguales, es decir, existen 100 observaciones que correspondern a cada tipo.

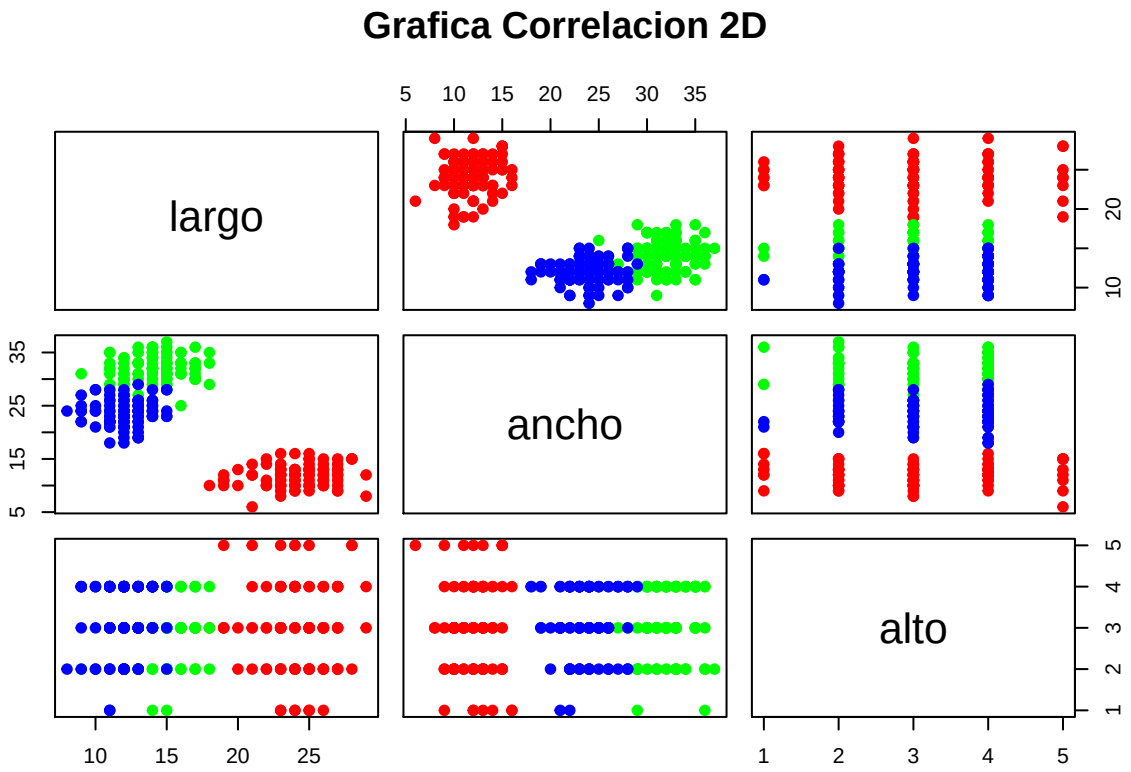
3.1 Exploren los datos y encuentren gráficamente sus tendencias y sus posibles correlaciones en gráficos 2D y 3D.

Graficamos los datos y sus correlaciones mediante un modelo 2D y 3D.En la grafica 2D podemos apreciar que las variables tienen una distribución normal y la relación entre el largo y el ancho generan agrupaciones las cuales están bien definidos.

```r
datos <- read.csv("Ejercicio 3.csv", sep = ",")
p1 <- ggplot(data = datos, aes(x = largo, fill = tipo)) + geom_histogram(position = "identity",
    alpha = 0.5)
p2 <- ggplot(data = datos, aes(x = ancho, fill = tipo)) + geom_histogram(position = "identity",
    alpha = 0.5)
p3 <- ggplot(data = datos, aes(x = alto, fill = tipo)) + geom_histogram(position = "identity",
    alpha = 0.5)
ggarrange(p1, p2, p3, nrow = 3, common.legend = TRUE)
```

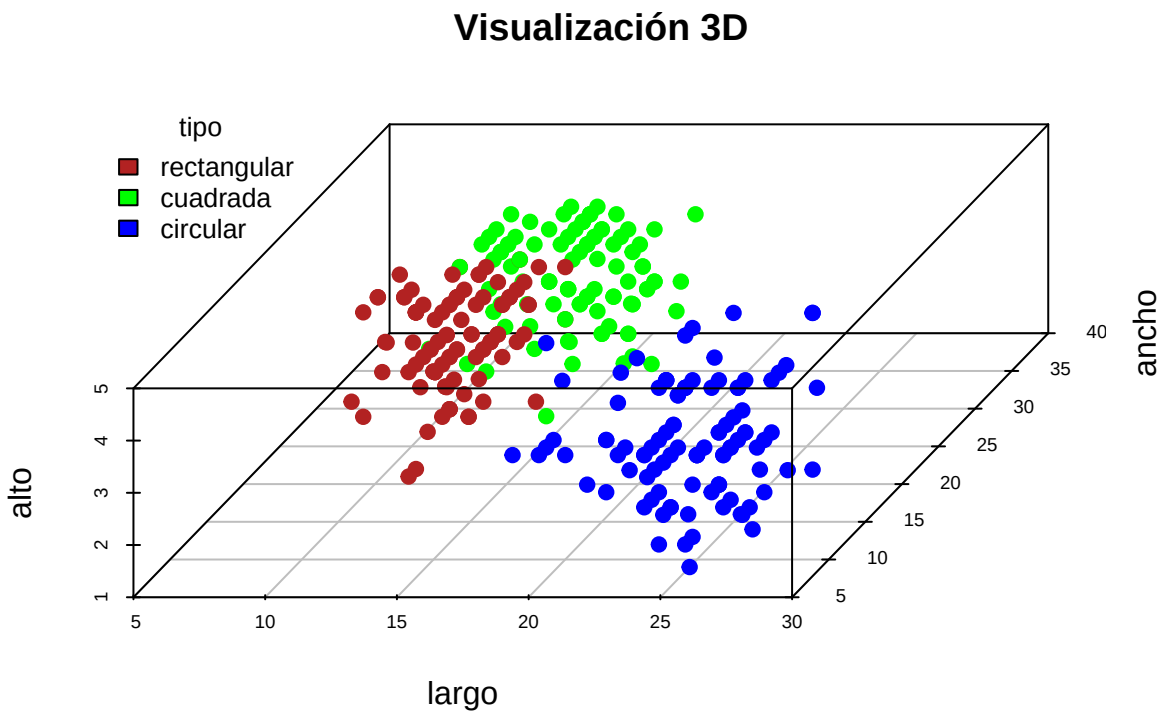



```
pairs(x = datos[, c("largo", "ancho", "alto")], col = c("blue", "green", "red")[datos$tipo],
    pch = 19, main = "Grafica Correlacion 2D")
```



Es importante destacar en las gráficas de dispersiones se puede apreciar que la relación de largo con respecto al ancho (y viceversa) crea unos gráficos en donde se ven perfectamente la separación de los tipos (circular, rectangular y cuadrada).

```
scatterplot3d(datos$largo, datos$ancho, datos$alto, color = c("firebrick", "green",
    "blue")[datos$tipo], pch = 19, grid = TRUE, xlab = "largo", ylab = "ancho", zlab = "alto",
    angle = 65, cex.axis = 0.6, main = "Visualización 3D")
legend("topleft", bty = "n", cex = 0.8, title = "tipo", c("rectangular", "cuadrada",
    "circular"), fill = c("firebrick", "green", "blue"))
```



Gracias a la graficación realizada en 3D se puede apreciar que el grupo aislado es el de tipo circular y aquellos que casi se traslapan (desde el ángulo en el que están siendo vistos) son el tipo rectangular y cuadrada.

3.2 Realicen la verificación de supuestos.

Aplicamos las técnicas vistas en clase para evaluar la normalidad multivariable y para identificar outlayers que influyen en el comportamiento los datos, mediante los test Mardia, Henze-Zirkler y Royston

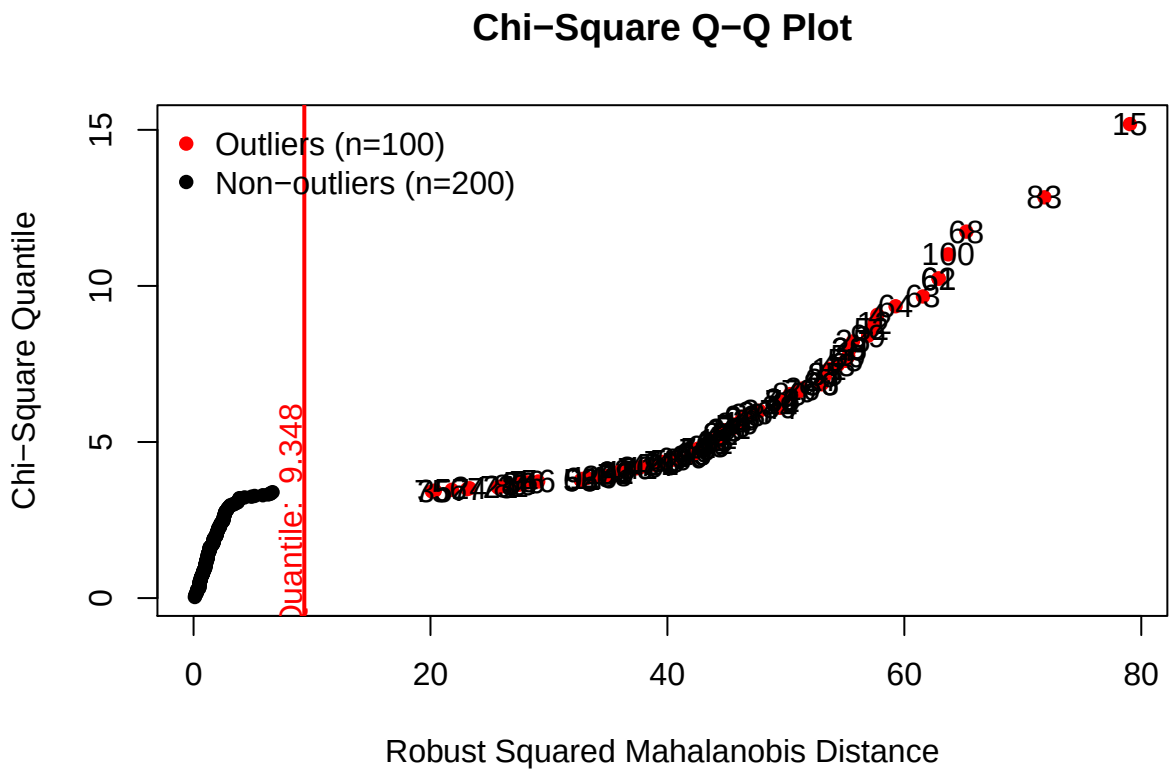
```
datos_tidy <- melt(datos, value.name = "valor")
```

```
## Using tipo as id variables
```

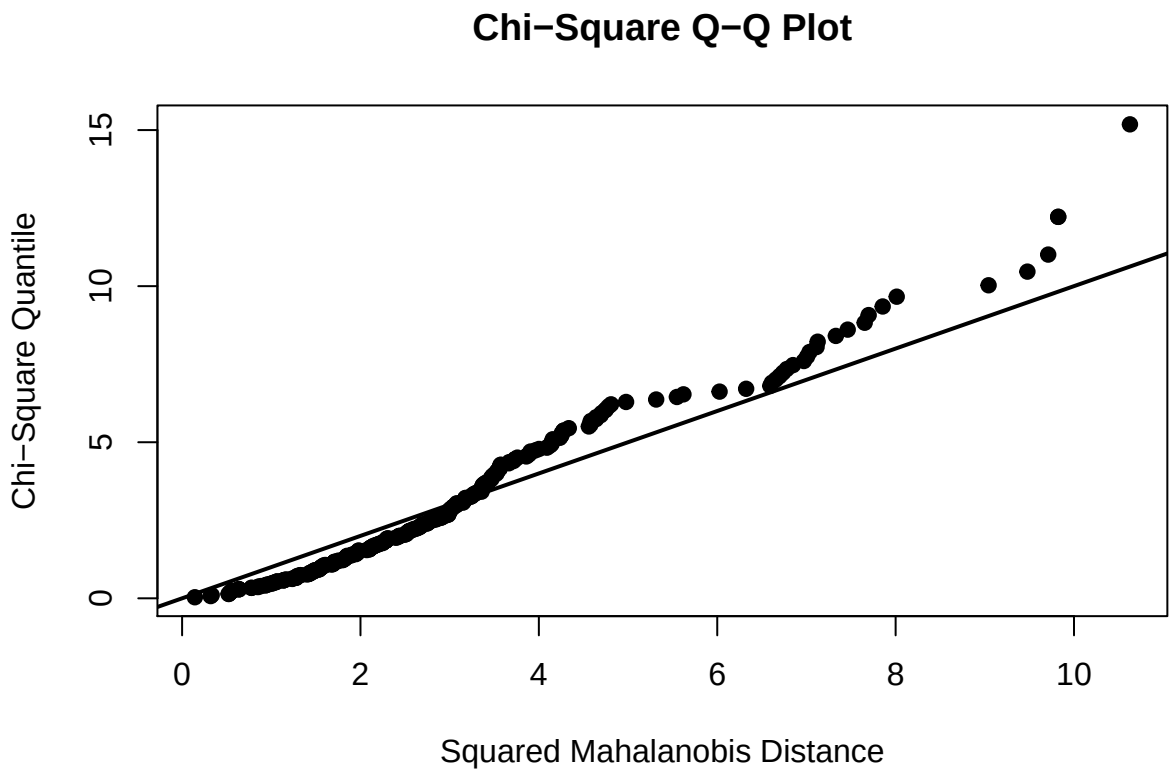
```
kable(datos_tidy %>% group_by(tipo, variable) %>% summarise(p_value_Shapiro.test = shapiro.test(valor)$
```

tipo	variable	p_value_Shapiro.test
circular	largo	0.0008715
circular	ancho	0.0516593
circular	alto	0.0000000
cuadrada	largo	0.0048815
cuadrada	ancho	0.0432692
cuadrada	alto	0.0000000
rectangular	largo	0.0075805
rectangular	ancho	0.0135727
rectangular	alto	0.0000054

```
outliers <- mvn(data = datos[, -1], mvnTest = "hz", multivariateOutlierMethod = "quan")
```




```
royston_test <- mvn(data = datos[, -1], mvnTest = "royston", multivariatePlot = "qq")
```




```
royston_test$multivariateNormality
```

```
##      Test        H      p value MVN
## 1 Royston 130.9617 8.058205e-29  NO
```

```
hz_test <- mvn(data = datos[, -1], mvnTest = "hz")
hz_test$multivariateNormality
```

```
##            Test       HZ p value MVN
## 1 Henze-Zirkler 9.658802       0  NO
```

```
boxM(data = datos[, 2:4], grouping = datos[, 1])
```

```
##
##  Box's M-test for Homogeneity of Covariance Matrices
##
## data:  datos[, 2:4]
## Chi-Sq (approx.) = 32.9, df = 12, p-value = 0.001003
```

3.3 Apliquen el método LDA o QDA según consideren conveniente paso a paso.

Para esta base de datos decidimos aplicar el modelo LDA, ya que los datos como vimos anteriormente tienen una distribución normal.

```r
modelo_lda <- lda(formula = tipo ~ largo + ancho + alto, data = datos)
```

3.4 Grafiquen como quedaría la clasificación y calculen su error para nuevos datos.

Al aplicar el modelo de predicción para una nueva observación y al evaluarla nos dio un error del 0% lo cual indica que el modelo funciona correctamente.

```r
nuevas_observaciones <- data.frame(largo = 30, ancho = 15, alto = 20)
predict(object = modelo_lda, newdata = nuevas_observaciones)
```

```
## $class
## [1] rectangular
## Levels: circular cuadrada rectangular
##
## $posterior
##       circular     cuadrada rectangular
## 1 1.202897e-21 2.299888e-28           1
##
## $x
##         LD1      LD2
## 1 -7.045358 2.995213
```

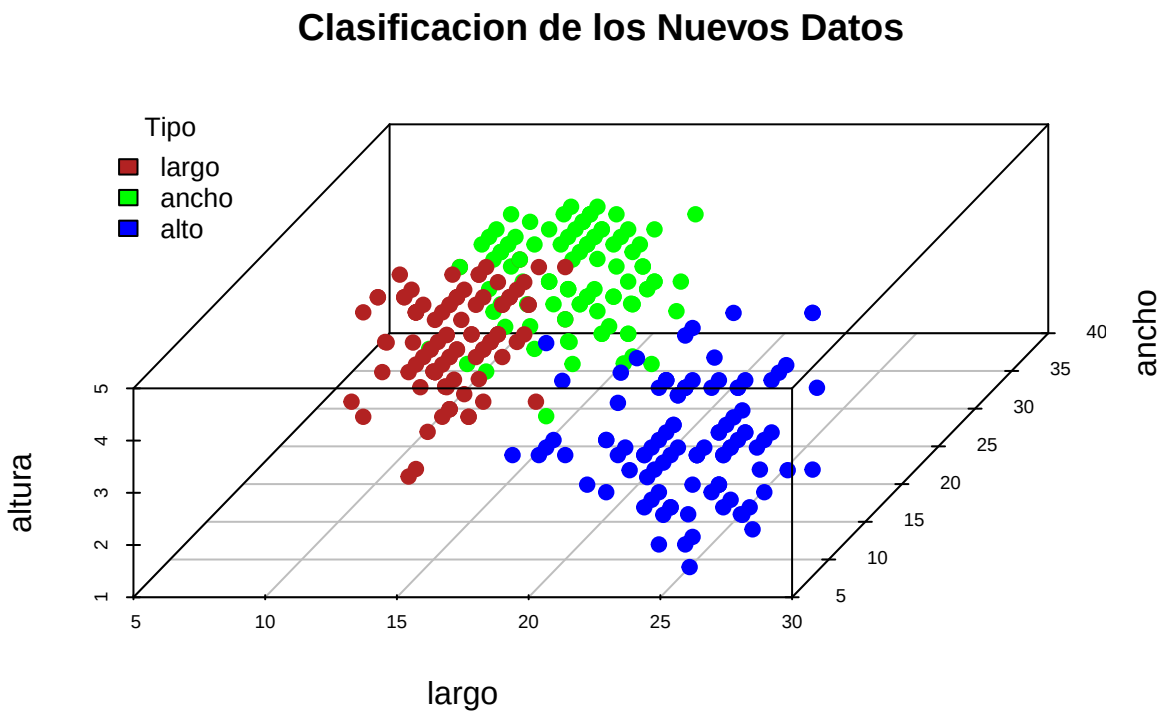
```r
predicciones <- predict(object = modelo_lda, newdata = datos[, -1], method = "predictive")
table(datos$tipo, predicciones$class, dnn = c("Clase real", "Clase predicha"))
```

```
##              Clase predicha
## Clase real    circular cuadrada rectangular
##   circular          96        4           0
##   cuadrada           4       96           0
##   rectangular        0        0         100
```

```r
trainig_error <- mean(datos$especie != predicciones$class) * 100
paste("trainig_error=", trainig_error, "%")
```

```
## [1] "trainig_error= NaN %"
```

```r
with(datos, {
    s3d <- scatterplot3d(largo, ancho, alto, color = c("firebrick", "green", "blue")[datos$tipo],
        pch = 19, grid = TRUE, xlab = "largo", ylab = "ancho", zlab = "altura", angle = 65,
        cex.axis = 0.6, main = "Clasificacion de los Nuevos Datos")

    s3d.coords <- s3d$xyz.convert(largo, ancho, alto)
    # convierte coordenadas 3D en proyecciones 2D

    legend("topleft", bty = "n", cex = 0.8, title = "Tipo", c("largo", "ancho", "alto"),
        fill = c("firebrick", "green", "blue"))
})
```



Empleando las mismas observaciones con las que se ha generado el modelo discriminante (trainig data), la precisión de clasificación es del 100%.

4. Método de reducción de dimensión (CCA)

4.1 Exploren los datos y encuentren gráficamente sus tendencias y sus posibles correlaciones.

```
data <- read.csv("Ejercicio 4.csv", sep = ",")
df_status(data)
```

```
##   variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1     peso       0       0    0    0     0     0 numeric     97
## 2   altura       0       0    0    0     0     0 integer     61
## 3     edad       0       0    0    0     0     0 integer     40
## 4    salud       0       0    0    0     0     0  factor      3
## 5 ansiedad      26      26    0    0     0     0 integer      4
## 6  deporte       0       0    0    0     0     0  factor      3
```

La base de datos a explorar se conforma de 6 variables, las cuales son: Peso, Altura, Edad, Salud, Ansiedad y Deporte. Además, contiene 100 observaciones.

```r
frecuency_as_df <- as.data.frame(tabyl(data$ansiedad, sort = TRUE))

names(frecuency_as_df) <- c("Tipo", "Frecuencia", "Porcentaje")

frecuency_as_df
```

```
##   Tipo Frecuencia Porcentaje
## 1    0         26       0.26
## 2    1         12       0.12
## 3    2         30       0.30
## 4    3         32       0.32
```

Como se puede apreciar en la tabla, existen 4 niveles de ansiedad que se enumeran del 0 al 3, siendo 0 la ausencia de esta condición y 3 la presencia máxima de la misma. Tan solo el 26% de las personas registradas en la base de datos se encuentran libres de sufrir de ansidad, mientras que el 74% la presenta en distintos niveles.

En general, predominan las observaciones con ansiedad de tipo 3, ya que son el 32% de la información total almacenada en la base de datos.

```r
frecuency_as_df <- as.data.frame(tabyl(data$deporte, sort = TRUE))

names(frecuency_as_df) <- c("Tipo", "Frecuencia", "Porcentaje")

frecuency_as_df
```

```
##    Tipo Frecuencia Porcentaje
## 1  bajo         63       0.63
## 2 Medio         12       0.12
## 3  nulo         25       0.25
```

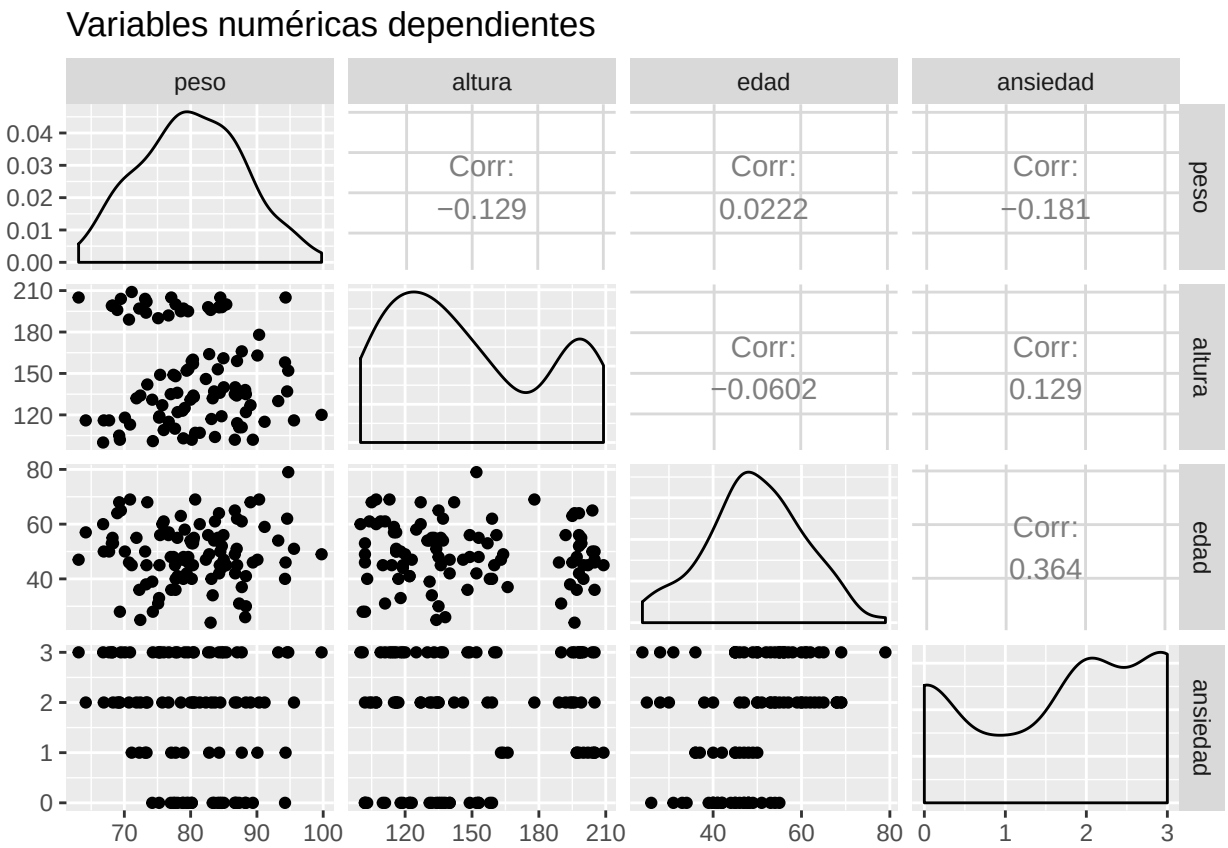
La variable “Deporte” podría referirse al nivel de dificultad del deporte practicado por cierta persona o el nivel de hábito que tiene un individuo para ejercitarse. Con los datos de la tabla es posible inferir que predomina la categoria “Bajo” en la variable deporte con una presencia del 63% en los datos del conjunto con el que se está trabajando.

```r
frecuency_as_df <- as.data.frame(tabyl(data$salud, sort = TRUE))

names(frecuency_as_df) <- c("Tipo", "Frecuencia", "Porcentaje")

frecuency_as_df
```

```
##      Tipo Frecuencia Porcentaje
## 1   buena         39       0.39
## 2    mala         31       0.31
## 3 regular         30       0.30
```

En el caso de la variable salud, lamentablemente se puede observar que solo el 39% de las personas registradas en la base de datos cuentan con una buena salud, mientras que el 61% restante presentan una salud regular o mala. Estos resultados pueden ser reflejo a sus hábitos deportivos, nivel de ansidad, edad y peso.

```
numericas <- data %>% dplyr::select(peso, altura, edad, ansiedad)
ggpairs(numericas, title = "Variables numéricas dependientes")
```



Se puede apreciar que la correlación entre las variables numéricas de esta base de datos es demasiado débil. La más alta es de 0.364 y es la presente en la relación entre la ansiedad y la edad.

4.2 Apliquen el método CCA paso a paso encontrando todas la matrices de correlación.

Calculamos La matriz de covarianza:

```
dataa <- c("peso", "altura", "edad", "ansiedad")
datos <- data[dataa]
cov(datos)
```

```
##                peso      altura       edad  ansiedad
## peso      60.549749  -35.310937   1.917487 -1.656921
## altura   -35.310937 1228.211717 -23.402424  5.349899
## edad       1.917487  -23.402424 123.175354  4.762020
## ansiedad  -1.656921    5.349899   4.762020  1.391515
```

Matriz de covarianzas de X

```
X <- datos[, (1:2)]
cov(X)
```

```
##             peso     altura
## peso    60.54975  -35.31094
## altura -35.31094 1228.21172
```

Matriz de covarianzas de Y

```
Y <- datos[, (3:4)]
cov(Y)
```

```
##               edad ansiedad
## edad     123.17535 4.762020
## ansiedad   4.76202 1.391515
```

Matriz de covarianzas de X e Y

```
cov(X, Y)
```

```
##              edad  ansiedad
## peso     1.917487 -1.656921
## altura -23.402424  5.349899
```

Matriz de covarianzas de Y y X

```
cov(Y, X)
```

```
##               peso     altura
## edad      1.917487 -23.402424
## ansiedad -1.656921   5.349899
```

El siguiente paso es encontrar los autovalores y autovectores de las matrices cuadradas

```
A <- solve(cov(X, X)) %*% cov(X, Y) %*% solve(cov(Y, Y)) %*% cov(Y, X)
B <- solve(cov(Y, Y)) %*% cov(Y, X) %*% solve(cov(X, X)) %*% cov(X, Y)
eigen(A)$values
```

```
## [1] 0.062490391 0.001196716
```

```
eigen(B)$values
```

```
## [1] 0.062490391 0.001196716
```

```
r <- sqrt(eigen(A)$values)
r
```

```
## [1] 0.24998078 0.03459358
```

```r
a <- eigen(A)$vector[, 1]
b <- eigen(B)$vector[, 1]
a
```

```
## [1]  0.9844316 -0.1757684
```

```r
b
```

```
## [1]  0.05953744 -0.99822607
```

```r
t(a) %*% cov(X, X) %*% a
```

```
##          [,1]
## [1,] 108.8439
```

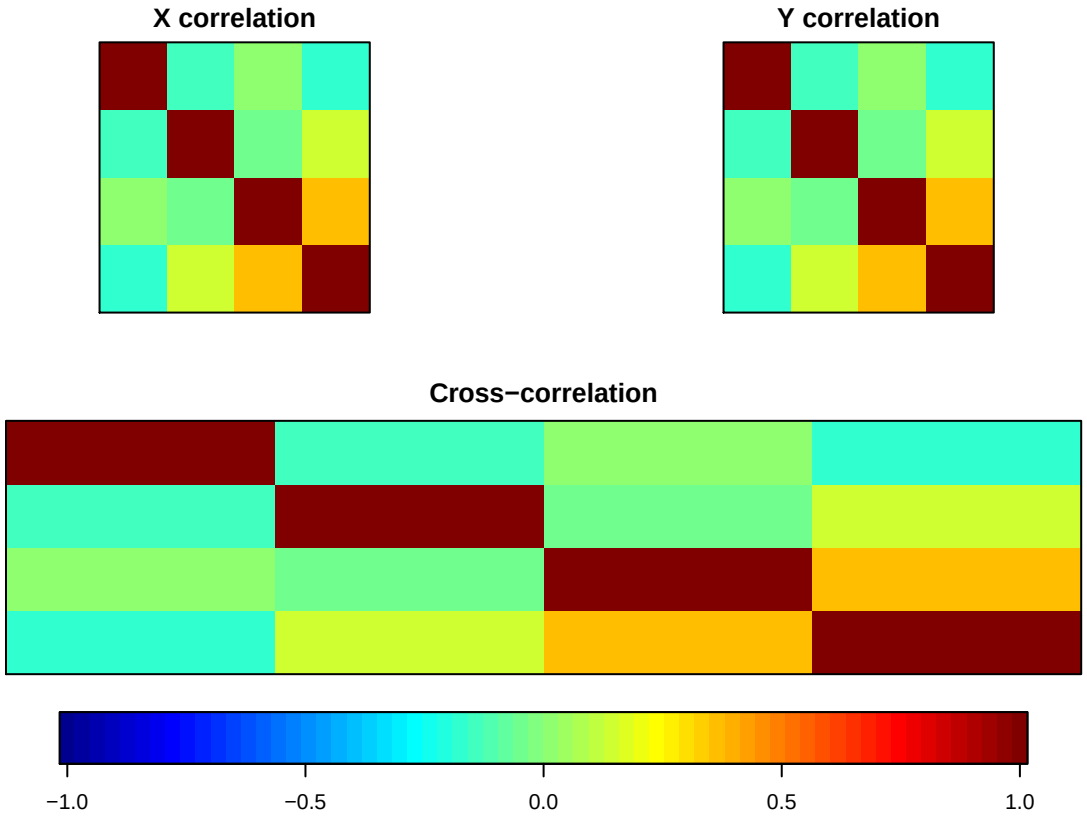
```r
mat_cor <- matcor(datos, datos)
mat_cor
```

```
## $Xcor
##                 peso      altura        edad   ansiedad
## peso       1.00000000 -0.12948410  0.02220314 -0.1805102
## altura    -0.12948410  1.00000000 -0.06016755  0.1294092
## edad       0.02220314 -0.06016755  1.00000000  0.3637352
## ansiedad -0.18051019  0.12940915  0.36373522  1.0000000
##
## $Ycor
##                 peso      altura        edad   ansiedad
## peso       1.00000000 -0.12948410  0.02220314 -0.1805102
## altura    -0.12948410  1.00000000 -0.06016755  0.1294092
## edad       0.02220314 -0.06016755  1.00000000  0.3637352
## ansiedad -0.18051019  0.12940915  0.36373522  1.0000000
##
## $XYcor
##                 peso      altura        edad   ansiedad        peso      altura
## peso       1.00000000 -0.12948410  0.02220314 -0.1805102  1.00000000 -0.12948410
## altura    -0.12948410  1.00000000 -0.06016755  0.1294092 -0.12948410  1.00000000
## edad       0.02220314 -0.06016755  1.00000000  0.3637352  0.02220314 -0.06016755
## ansiedad -0.18051019  0.12940915  0.36373522  1.0000000 -0.18051019  0.12940915
## peso       1.00000000 -0.12948410  0.02220314 -0.1805102  1.00000000 -0.12948410
## altura    -0.12948410  1.00000000 -0.06016755  0.1294092 -0.12948410  1.00000000
## edad       0.02220314 -0.06016755  1.00000000  0.3637352  0.02220314 -0.06016755
## ansiedad -0.18051019  0.12940915  0.36373522  1.0000000 -0.18051019  0.12940915
##                 edad   ansiedad
## peso       0.02220314 -0.1805102
## altura    -0.06016755  0.1294092
## edad       1.00000000  0.3637352
## ansiedad   0.36373522  1.0000000
## peso       0.02220314 -0.1805102
## altura    -0.06016755  0.1294092
## edad       1.00000000  0.3637352
## ansiedad   0.36373522  1.0000000
```

4.3 Expliquen de forma gráfica como se relacionan estas variables

Podemos apreciar que las variables tienen una alta correlación en la matriz, y de igual manera el diagrama de correlación cruzada es practicamente la misma que las otras 2 graficas.

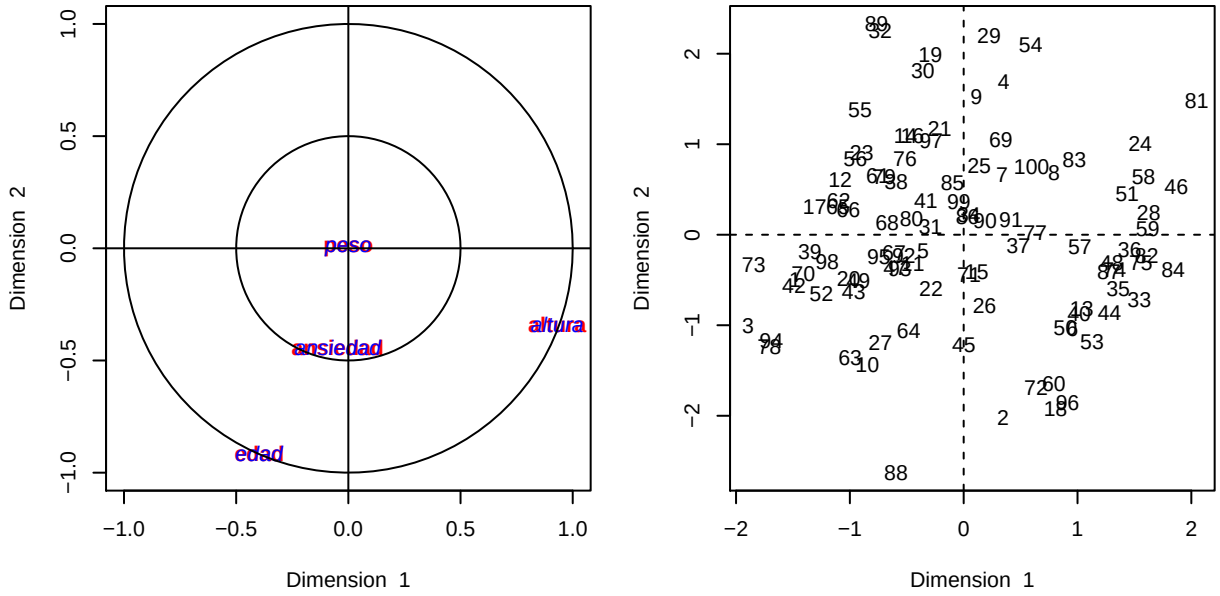
```
cca1 <- cc(datos, datos)
img.matcor(mat_cor, type = 2)
```



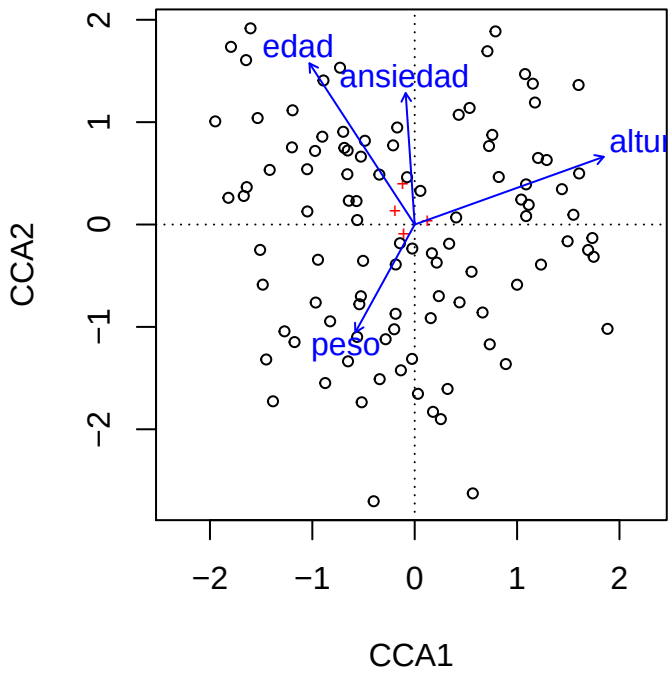
4.4 Construyan como quedaría una posible clasificación de estas observaciones según estas variables.

Hacemos un diagrama de correspondencia lo cual permite agrupar las variables, y podemos apreciar que las variables ansiedad y edad se encuentran muy cercanas

```
plt.cc(cca1, var.label = T)
```




```
cca2 <- cca(datos, datos)
plot(cca2)
```


edad
ansiedad
altur
peso
CCA2
CCA1
−2
−1
0
1
2