# Práctica 2. Tipología y ciclo de vida de los datos

María Sánchez y Cayetano Bautista

01/05/2021

## Contents



## 1. Descripcion del dataset. ¿Por qué es importante y que pregunta/problema pretende responder?

El conjunto de datos objeto de análisis está compuesto por 53 variables y 1700 observaciones, las cuáles contienen el estilo de juego del videojuego de consola Fifa 2017, así como estadísticas reales de los jugadores de futbol.

La descripción de las principales variables es la siguiente:

- Name: Nombre del jugador
- Club: Equipo en el que juega
- Rating: Valoración global del jugador, entre 0 y 100
- Height: Altura
- Weight: Peso
- Preffered_Foot: Pie preferido para jugar
- Age: Edad
- Ball_Control: Control de la pelota, entre 0 y 100
- Club_position: Si es portero o jugador
- Dribbling : Control de regateo, entre 0 y 100

La descripción de los atributos se puede consultar en https://www.fifplay.com/encyclopedia. La descripción de las abreviaturas de la posición del jugador en el campo se puede consultar en https://www.dtgre.com/2016/10/fifa-17-position-abbreviations-acronyms.html.

La importancia de este dataset reside en que sirve como primer acercamiento al análisis de datos futbolísitico y deportivo (el dataset es ideal en tanto que se tienen ratings impuestos a cada jugador en lugar de sus estadísticas reales). Por otro lado, a un nivel mucho más terrenal, nos podría servir para hacer el equipo "más rentable" de todo el juego.

Las preguntas que pretendemos responder con este estudio son las siguientes:

**¿Cuál es el valor promedio del Rating de los jugadores?**

**¿Los jugadores de fútbol zurdos tienen mejor control de la pelota que los diestros?**

**¿Los jugadores de fútbol zurdos tienen mejor valoración global que los diestros?**

**¿Los jugadores de fútbol zurdos tienen mejor dribblling que los diestros?**

**¿El porcentaje de jugadores con un Rating superior a 90 es diferente en el Barcelona y en el Madrid?**

**¿El peso de los porteros es mayor al peso de los jugadores de campo?**

**¿Son los porteros al menos 5 cms más altos que los jugadores de campo?**

**¿Cuál sería el rating de un jugador de campo con pie izquierdo preferido, con un peso de 70, edad de 24, control del balón de 80 y visión de 60?**

## 2. Integración y selección de los datos de interes a analizar.

Leemos el fichero de datos:

```
fifa <- read.csv("Fifa.csv")
head(fifa)
```

```
##                Name Nationality National_Position National_Kit           Club
## 1 Cristiano Ronaldo    Portugal                LS            7    Real Madrid
## 2      Lionel Messi   Argentina                RW           10   FC Barcelona
## 3            Neymar      Brazil                LW           10   FC Barcelona
## 4      Luis SuÃ¡rez     Uruguay                LS            9   FC Barcelona
## 5      Manuel Neuer     Germany                GK            1      FC Bayern
## 6            De Gea       Spain                GK            1 Manchester Utd
##   Club_Position Club_Kit Club_Joining Contract_Expiry Rating Height Weight
## 1            LW        7   07/01/2009            2021     94 185 cm  80 kg
## 2            RW       10   07/01/2004            2018     93 170 cm  72 kg
## 3            LW       11   07/01/2013            2021     92 174 cm  68 kg
## 4            ST        9   07/11/2014            2021     92 182 cm  85 kg
## 5            GK        1   07/01/2011            2021     92 193 cm  92 kg
## 6            GK        1   07/01/2011            2019     90 193 cm  82 kg
##   Preffered_Foot Birth_Date Age Preffered_Position       Work_Rate Weak_foot
## 1          Right 02/05/1985  32              LW/ST      High / Low         4
## 2           Left 06/24/1987  29                 RW Medium / Medium         4
## 3          Right 02/05/1992  25                 LW   High / Medium         5
## 4          Right 01/24/1987  30                 ST   High / Medium         4
## 5          Right 03/27/1986  31                 GK Medium / Medium         4
## 6          Right 11/07/1990  26                 GK Medium / Medium         3
```

```
##   Skill_Moves Ball_Control Dribbling Marking Sliding_Tackle Standing_Tackle
## 1           5           93        92      22             23              31
## 2           4           95        97      13             26              28
## 3           5           95        96      21             33              24
## 4           4           91        86      30             38              45
## 5           1           48        30      10             11              10
## 6           1           31        13      13             13              21
##   Aggression Reactions Attacking_Position Interceptions Vision Composure
## 1         63        96                 94            29     85        86
## 2         48        95                 93            22     90        94
## 3         56        88                 90            36     80        80
## 4         78        93                 92            41     84        83
## 5         29        85                 12            30     70        70
## 6         38        88                 12            30     68        60
##   Crossing Short_Pass Long_Pass Acceleration Speed Stamina Strength Balance
## 1       84         83        77           91    92      92       80      63
## 2       77         88        87           92    87      74       59      95
## 3       75         81        75           93    90      79       49      82
## 4       77         83        64           88    77      89       76      60
## 5       15         55        59           58    61      44       83      35
## 6       17         31        32           56    56      25       64      43
##   Agility Jumping Heading Shot_Power Finishing Long_Shots Curve
## 1      90      95      85         92        93         90    81
## 2      90      68      71         85        95         88    89
## 3      96      61      62         78        89         77    79
## 4      86      69      77         87        94         86    86
## 5      52      78      25         25        13         16    14
## 6      57      67      21         31        13         12    21
##   Freekick_Accuracy Penalties Volleys GK_Positioning GK_Diving GK_Kicking
## 1                76        85      88             14         7         15
## 2                90        74      85             14         6         15
## 3                84        81      83             15         9         15
## 4                84        85      88             33        27         31
## 5                11        47      11             91        89         95
## 6                19        40      13             86        88         87
##   GK_Handling GK_Reflexes
## 1          11          11
## 2          11           8
## 3           9          11
## 4          25          37
## 5          90          89
## 6          85          90
```

```
summary(fifa)
```

```
##      Name           Nationality        National_Position   National_Kit
##  Length:17588       Length:17588       Length:17588       Min.   : 1.00
##  Class :character   Class :character   Class :character   1st Qu.: 6.00
##  Mode  :character   Mode  :character   Mode  :character   Median :12.00
##                                                           Mean   :12.22
##                                                           3rd Qu.:18.00
##                                                           Max.   :36.00
##                                                           NA's   :16513
##      Club           Club_Position         Club_Kit     Club_Joining
```

```
##  Length:17588       Length:17588       Min.   : 1.00   Length:17588
##  Class :character   Class :character   1st Qu.: 9.00   Class :character
##  Mode  :character   Mode  :character   Median :18.00   Mode  :character
##                                        Mean   :21.29
##                                        3rd Qu.:27.00
##                                        Max.   :99.00
##                                        NA's   :1
##  Contract_Expiry     Rating         Height             Weight
##  Min.   :2017    Min.   :45.00   Length:17588       Length:17588
##  1st Qu.:2017    1st Qu.:62.00   Class :character   Class :character
##  Median :2019    Median :66.00   Mode  :character   Mode  :character
##  Mean   :2019    Mean   :66.17
##  3rd Qu.:2020    3rd Qu.:71.00
##  Max.   :2023    Max.   :94.00
##  NA's   :1
##  Preffered_Foot      Birth_Date             Age        Preffered_Position
##  Length:17588       Length:17588       Min.   :17.00   Length:17588
##  Class :character   Class :character   1st Qu.:22.00   Class :character
##  Mode  :character   Mode  :character   Median :25.00   Mode  :character
##                                        Mean   :25.46
##                                        3rd Qu.:29.00
##                                        Max.   :47.00
##
##   Work_Rate           Weak_foot      Skill_Moves     Ball_Control
##  Length:17588       Min.   :1.000   Min.   :1.000   Min.   : 5.00
##  Class :character   1st Qu.:3.000   1st Qu.:2.000   1st Qu.:53.00
##  Mode  :character   Median :3.000   Median :2.000   Median :63.00
##                     Mean   :2.934   Mean   :2.303   Mean   :57.97
##                     3rd Qu.:3.000   3rd Qu.:3.000   3rd Qu.:69.00
##                     Max.   :5.000   Max.   :5.000   Max.   :95.00
##
##    Dribbling       Marking      Sliding_Tackle  Standing_Tackle   Aggression
##  Min.   : 4.0   Min.   : 3.00   Min.   : 5.00   Min.   : 3.00   Min.   : 2.00
##  1st Qu.:47.0   1st Qu.:22.00   1st Qu.:23.00   1st Qu.:26.00   1st Qu.:44.00
##  Median :60.0   Median :48.00   Median :51.00   Median :54.00   Median :59.00
##  Mean   :54.8   Mean   :44.23   Mean   :45.57   Mean   :47.44   Mean   :55.92
##  3rd Qu.:68.0   3rd Qu.:64.00   3rd Qu.:64.00   3rd Qu.:66.00   3rd Qu.:70.00
##  Max.   :97.0   Max.   :92.00   Max.   :95.00   Max.   :92.00   Max.   :96.00
##
##    Reactions     Attacking_Position Interceptions       Vision
##  Min.   :29.00   Min.   : 2.00      Min.   : 3.00   Min.   :10.00
##  1st Qu.:55.00   1st Qu.:37.00      1st Qu.:26.00   1st Qu.:43.00
##  Median :62.00   Median :54.00      Median :52.00   Median :54.00
##  Mean   :61.77   Mean   :49.59      Mean   :46.79   Mean   :52.71
##  3rd Qu.:68.00   3rd Qu.:64.00      3rd Qu.:64.00   3rd Qu.:64.00
##  Max.   :96.00   Max.   :94.00      Max.   :93.00   Max.   :94.00
##
##    Composure        Crossing       Short_Pass      Long_Pass     Acceleration
##  Min.   : 5.00   Min.   : 6.00   Min.   :10.00   Min.   : 7.0   Min.   :11.00
##  1st Qu.:47.00   1st Qu.:38.00   1st Qu.:52.00   1st Qu.:42.0   1st Qu.:57.00
##  Median :57.00   Median :54.00   Median :62.00   Median :56.0   Median :68.00
##  Mean   :55.85   Mean   :49.74   Mean   :58.12   Mean   :52.4   Mean   :65.29
##  3rd Qu.:66.00   3rd Qu.:64.00   3rd Qu.:68.00   3rd Qu.:64.0   3rd Qu.:75.00
##  Max.   :94.00   Max.   :91.00   Max.   :92.00   Max.   :93.0   Max.   :96.00
```

```
##
##      Speed           Stamina         Strength        Balance
##  Min.   :11.00   Min.   :10.00   Min.   :20.00   Min.   :10.00
##  1st Qu.:58.00   1st Qu.:57.00   1st Qu.:57.00   1st Qu.:56.00
##  Median :68.00   Median :66.00   Median :66.00   Median :65.00
##  Mean   :65.48   Mean   :63.48   Mean   :65.09   Mean   :64.01
##  3rd Qu.:75.00   3rd Qu.:74.00   3rd Qu.:74.00   3rd Qu.:74.00
##  Max.   :96.00   Max.   :95.00   Max.   :98.00   Max.   :97.00
##
##     Agility         Jumping         Heading        Shot_Power
##  Min.   :11.00   Min.   :15.00   Min.   : 4.00   Min.   : 3.00
##  1st Qu.:55.00   1st Qu.:58.00   1st Qu.:45.00   1st Qu.:45.00
##  Median :65.00   Median :65.00   Median :56.00   Median :59.00
##  Mean   :63.21   Mean   :64.92   Mean   :52.39   Mean   :55.58
##  3rd Qu.:74.00   3rd Qu.:73.00   3rd Qu.:65.00   3rd Qu.:69.00
##  Max.   :96.00   Max.   :95.00   Max.   :94.00   Max.   :93.00
##
##    Finishing       Long_Shots       Curve       Freekick_Accuracy
##  Min.   : 2.00   Min.   : 4.0   Min.   : 6.00   Min.   : 4.00
##  1st Qu.:29.00   1st Qu.:32.0   1st Qu.:34.00   1st Qu.:31.00
##  Median :48.00   Median :52.0   Median :48.00   Median :42.00
##  Mean   :45.16   Mean   :47.4   Mean   :47.18   Mean   :43.38
##  3rd Qu.:61.00   3rd Qu.:63.0   3rd Qu.:62.00   3rd Qu.:57.00
##  Max.   :95.00   Max.   :91.0   Max.   :92.00   Max.   :93.00
##
##    Penalties        Volleys      GK_Positioning    GK_Diving
##  Min.   : 7.00   Min.   : 3.00   Min.   : 1.00   Min.   : 1.00
##  1st Qu.:39.00   1st Qu.:30.00   1st Qu.: 8.00   1st Qu.: 8.00
##  Median :50.00   Median :44.00   Median :11.00   Median :11.00
##  Mean   :49.17   Mean   :43.28   Mean   :16.61   Mean   :16.82
##  3rd Qu.:61.00   3rd Qu.:57.00   3rd Qu.:14.00   3rd Qu.:14.00
##  Max.   :96.00   Max.   :93.00   Max.   :91.00   Max.   :89.00
##
##    GK_Kicking     GK_Handling     GK_Reflexes
##  Min.   : 1.00   Min.   : 1.00   Min.   : 1.0
##  1st Qu.: 8.00   1st Qu.: 8.00   1st Qu.: 8.0
##  Median :11.00   Median :11.00   Median :11.0
##  Mean   :16.46   Mean   :16.56   Mean   :16.9
##  3rd Qu.:14.00   3rd Qu.:14.00   3rd Qu.:14.0
##  Max.   :95.00   Max.   :91.00   Max.   :90.0
##
```

Como podemos ver, las variables 'Height' y 'Weight' están entendidas como categóricas cunado en realidad deberían ser continuas. Por tanto, vamos a corregir esto.

```
aNum_H_W <- function(numstr) {
  if (grepl("kg", numstr)) {r = as.numeric(sub("kg", "", numstr))}
  else {r = as.numeric(sub("cm", "", numstr))}
  return(r)}

fifa$Height <- sapply(fifa$Height, aNum_H_W)
fifa$Weight <- sapply(fifa$Weight, aNum_H_W)

head(fifa$Height)
```

```
## [1] 185 170 174 182 193 193
```

```
class(fifa$Height)
```

```
## [1] "numeric"
```

```
head(fifa$Weight)
```

```
## [1] 80 72 68 85 92 82
```

```
class(fifa$Weight)
```

```
## [1] "numeric"
```

Solucionado.

Vamos a transformar a factor las variables que son caracter.

```
str(mutate_if(fifa, is.character, as.factor))
```

```
## 'data.frame':    17588 obs. of  53 variables:
##  $ Name               : Factor w/ 17341 levels "Ã-gmundur Kristinsson",..: 3365 9997 12509 10338 10615
##  $ Nationality        : Factor w/ 160 levels "Afghanistan",..: 122 6 20 155 59 139 121 158 143 14 ...
##  $ National_Position : Factor w/ 28 levels "","CAM","CB",..: 14 25 15 14 6 6 14 24 1 6 ...
##  $ National_Kit       : num  7 10 10 9 1 1 9 11 NA 1 ...
##  $ Club               : Factor w/ 634 levels "1. FC Heidenheim",..: 461 207 207 207 209 364 209 461 3
##  $ Club_Position      : Factor w/ 30 levels "","CAM","CB",..: 16 27 16 29 7 7 29 27 29 7 ...
##  $ Club_Kit           : num  7 10 11 9 1 1 9 11 9 13 ...
##  $ Club_Joining       : Factor w/ 1678 levels "","01/01/1993",..: 848 843 852 927 850 850 853 1247 85
##  $ Contract_Expiry    : num  2021 2018 2021 2021 2021 ...
##  $ Rating             : int  94 93 92 92 92 90 90 90 90 89 ...
##  $ Height             : num  185 170 174 182 193 193 185 183 195 199 ...
##  $ Weight             : num  80 72 68 85 92 82 79 74 95 91 ...
##  $ Preffered_Foot     : Factor w/ 2 levels "Left","Right": 2 1 2 2 2 2 2 1 2 1 ...
##  $ Birth_Date         : Factor w/ 6063 levels "01/01/1982","01/01/1983",..: 623 2991 630 412 1490 521
##  $ Age                : int  32 29 25 30 31 26 28 27 35 24 ...
##  $ Preffered_Position: Factor w/ 292 levels "CAM","CAM/CDM",..: 172 237 157 266 113 113 266 237 266
##  $ Work_Rate          : Factor w/ 9 levels "High / High",..: 2 9 3 3 9 9 3 3 8 9 ...
##  $ Weak_foot          : int  4 4 5 4 4 3 4 3 4 3 ...
##  $ Skill_Moves        : int  5 4 5 4 1 1 3 4 4 1 ...
##  $ Ball_Control       : int  93 95 95 91 48 31 87 88 90 23 ...
##  $ Dribbling          : int  92 97 96 86 30 13 85 89 87 13 ...
##  $ Marking            : int  22 13 21 30 10 13 25 51 15 11 ...
##  $ Sliding_Tackle     : int  23 26 33 38 11 13 19 52 27 16 ...
##  $ Standing_Tackle    : int  31 28 24 45 10 21 42 55 41 18 ...
##  $ Aggression         : int  63 48 56 78 29 38 80 65 84 23 ...
##  $ Reactions          : int  96 95 88 93 85 88 88 87 85 81 ...
##  $ Attacking_Position: int  94 93 90 92 12 12 89 86 86 13 ...
##  $ Interceptions      : int  29 22 36 41 30 30 39 59 20 15 ...
##  $ Vision             : int  85 90 80 84 70 68 78 79 83 44 ...
##  $ Composure          : int  86 94 80 83 70 60 87 85 91 52 ...
```

```
##  $ Crossing          : int  84 77 75 77 15 17 62 87 76 14 ...
##  $ Short_Pass        : int  83 88 81 83 55 31 83 86 84 32 ...
##  $ Long_Pass         : int  77 87 75 64 59 32 65 80 76 31 ...
##  $ Acceleration      : int  91 92 93 88 58 56 79 93 69 46 ...
##  $ Speed             : int  92 87 90 77 61 56 82 95 74 52 ...
##  $ Stamina           : int  92 74 79 89 44 25 79 78 75 38 ...
##  $ Strength          : int  80 59 49 76 83 64 84 80 93 70 ...
##  $ Balance           : int  63 95 82 60 35 43 79 65 41 45 ...
##  $ Agility           : int  90 90 96 86 52 57 78 77 86 61 ...
##  $ Jumping           : int  95 68 61 69 78 67 84 85 72 68 ...
##  $ Heading           : int  85 71 62 77 25 21 85 86 80 13 ...
##  $ Shot_Power        : int  92 85 78 87 25 31 86 91 93 36 ...
##  $ Finishing         : int  93 95 89 94 13 13 91 87 90 14 ...
##  $ Long_Shots        : int  90 88 77 86 16 12 82 90 88 17 ...
##  $ Curve             : int  81 89 79 86 14 21 77 86 82 19 ...
##  $ Freekick_Accuracy : int  76 90 84 84 11 19 76 85 82 11 ...
##  $ Penalties         : int  85 74 81 85 47 40 81 76 91 27 ...
##  $ Volleys           : int  88 85 83 88 11 13 86 76 93 12 ...
##  $ GK_Positioning    : int  14 14 15 33 91 86 8 5 9 86 ...
##  $ GK_Diving         : int  7 6 9 27 89 88 15 15 13 84 ...
##  $ GK_Kicking        : int  15 15 15 31 95 87 12 11 10 69 ...
##  $ GK_Handling       : int  11 11 9 25 90 85 6 15 15 91 ...
##  $ GK_Reflexes       : int  11 8 11 37 89 90 10 6 12 89 ...
```

Solucionado.

Ahora vamos a crear una nueva variable categórica 'clasificacion' a partir de la numérica 'Rating'. Los niveles serán los siguientes:

- 'Rating': 90-99 ⇒ 'clasificacion' = "Excelente".
- 'Rating': 80-89 ⇒ 'clasificacion' = "Muy bueno".
- 'Rating': 70-89 ⇒ 'clasificacion' = "Bueno".
- 'Rating': 50-69 ⇒ 'clasificacion' = "Regular".
- 'Rating': 40-49 ⇒ 'clasificacion' = "Malo".
- 'Rating': 0-39 ⇒ 'clasificacion' = "Muy malo".

Lo haremos a través de la función *cut()* de R.

```r
# Intervalos (cerrados por la izquierda).
b = c(0, 40, 50, 70, 80, 90, 100)

# Etiquetas.
lab = c("Muy malo", "Malo", "Regular", "Bueno", "Muy bueno", "Excelente")

# Incluimos el valor mínimo '0': include.lowest = T
# Intervalos cerrados por la izquierda y abiertos por la derecha: right = F
# Los niveles están ordenados: ordered_result = T
fifa$clasificacion <- cut(fifa$Rating, breaks = b, labels = lab,
                          include.lowest = T, right = F, ordered_result = T)

summary(fifa$clasificacion)
```

```
##  Muy malo      Malo   Regular     Bueno Muy bueno Excelente
##         0       121     11921      5017       520         9
```

Vamos a crear una nueva variable 'portero' que indique si el jugador juega de portero ('GK' en 'Club_Position').

```
fifa$portero <- (fifa$Club_Position == 'GK')
table(fifa$portero)
```

```
##
## FALSE  TRUE
## 16956   632
```

Es decir, tenemos 632 porteros y 16956 jugadores de campo.

Pasamos ahora a limpiar los datos.

# 3. Limpieza de los datos.

Vamos a comprobar si los datos contienen valores perdidos:

```
colSums(is.na(fifa))
```

```
##               Name        Nationality  National_Position       National_Kit
##                  0                  0                  0              16513
##               Club      Club_Position           Club_Kit       Club_Joining
##                  0                  0                  1                  0
##    Contract_Expiry             Rating             Height             Weight
##                  1                  0                  0                  0
##     Preffered_Foot         Birth_Date                Age Preffered_Position
##                  0                  0                  0                  0
##          Work_Rate          Weak_foot        Skill_Moves       Ball_Control
##                  0                  0                  0                  0
##          Dribbling            Marking     Sliding_Tackle    Standing_Tackle
##                  0                  0                  0                  0
##         Aggression          Reactions Attacking_Position      Interceptions
##                  0                  0                  0                  0
##             Vision          Composure           Crossing         Short_Pass
##                  0                  0                  0                  0
##          Long_Pass       Acceleration              Speed            Stamina
##                  0                  0                  0                  0
##           Strength            Balance            Agility            Jumping
##                  0                  0                  0                  0
##            Heading         Shot_Power          Finishing         Long_Shots
##                  0                  0                  0                  0
##              Curve  Freekick_Accuracy          Penalties            Volleys
##                  0                  0                  0                  0
##     GK_Positioning          GK_Diving         GK_Kicking        GK_Handling
##                  0                  0                  0                  0
##        GK_Reflexes      clasificacion            portero
##                  0                  0                  0
```

Como vemos, existen algunos. Analicémoslos.

```
head(fifa$National_Position[is.na(fifa$National_Kit)])
```

```
## [1] "" "" "" "" "" ""
```

Los valores perdidos de la variable 'National_Kit' corresponden a jugadores que no son internacionales (cadena vacía en 'National_Position'), como no podía ser de otra manera. Para indicarlo, cambiaremos las cadenas vacías de 'National_Position' por "NO international". También cambiaremos los NA de 'National_Kit' por 0, ya que no es un número que pueda llevar ningún jugador, por lo que es un perfecto valor *centinela*.

```
fifa$National_Kit[is.na(fifa$National_Kit)] = 0
fifa$National_Position[fifa$National_Position == ""] = "NO international"
sum(is.na(fifa$National_Kit))
```

```
## [1] 0
```

```
sum(fifa$National_Position == "")
```

```
## [1] 0
```

Solucionado. Comprobemos igualmente si hay más variables con cadenas vacías.

```
colSums(fifa == "")
```

```
##               Name        Nationality  National_Position       National_Kit
##                  0                  0                  0                  0
##               Club      Club_Position           Club_Kit       Club_Joining
##                  0                  1                 NA                  1
##    Contract_Expiry             Rating             Height             Weight
##                 NA                  0                  0                  0
##     Preffered_Foot         Birth_Date                Age Preffered_Position
##                  0                  0                  0                  0
##          Work_Rate          Weak_foot        Skill_Moves       Ball_Control
##                  0                  0                  0                  0
##          Dribbling            Marking     Sliding_Tackle    Standing_Tackle
##                  0                  0                  0                  0
##         Aggression          Reactions Attacking_Position      Interceptions
##                  0                  0                  0                  0
##             Vision          Composure           Crossing         Short_Pass
##                  0                  0                  0                  0
##          Long_Pass       Acceleration              Speed            Stamina
##                  0                  0                  0                  0
##           Strength            Balance            Agility            Jumping
##                  0                  0                  0                  0
##            Heading         Shot_Power          Finishing         Long_Shots
##                  0                  0                  0                  0
##              Curve  Freekick_Accuracy          Penalties            Volleys
##                  0                  0                  0                  0
##     GK_Positioning          GK_Diving         GK_Kicking        GK_Handling
##                  0                  0                  0                  0
##        GK_Reflexes      clasificacion            portero
##                  0                  0                  0
```

Hay también 1 en las variables 'Club_Position' y 'Club_Joining', y devuelve NA en las variables donde tdavía no hemos tratado los 2 NA que quedan.

Veamos qué ocurre con los valores perdido de 'Contract_Expiry' y 'Club_Kit', y a ver si tienen relación entre ellos y con las cadenas vacías que todavía tenemos.

```
fifa[which(is.na(fifa$Contract_Expiry)), ]
```

```
##              Name Nationality National_Position National_Kit       Club
## 384 Didier Drogba Ivory Coast  NO international            0 Free agent
##     Club_Position Club_Kit Club_Joining Contract_Expiry Rating Height Weight
## 384                    NA                           NA     81    189     80
##     Preffered_Foot Birth_Date Age Preffered_Position    Work_Rate Weak_foot
## 384          Right 03/11/1978  39                 ST Medium / Low         4
##     Skill_Moves Ball_Control Dribbling Marking Sliding_Tackle Standing_Tackle
## 384           3           80        74      22             29              32
##     Aggression Reactions Attacking_Position Interceptions Vision Composure
## 384         80        80                 81            42     76        80
##     Crossing Short_Pass Long_Pass Acceleration Speed Stamina Strength Balance
## 384       67         60        60           64    64      62       86      56
##     Agility Jumping Heading Shot_Power Finishing Long_Shots Curve
## 384      63      76      85         85        82         79    78
##     Freekick_Accuracy Penalties Volleys GK_Positioning GK_Diving GK_Kicking
## 384                84        84      76              6        10          8
##     GK_Handling GK_Reflexes clasificacion portero
## 384          11          14     Muy bueno   FALSE
```

Ya sabemos lo que ocurre: tenemos que el jugador Didier Drogba es agente libre, por lo que no tiene posición en club ('Club_Position' vacío), número ('Club_Kit' NA) ni fechas de inicio ('Club_Joining' vacío) y fin de contrato ('Contract_Expiry' NA).

Lo solucionamos copiando "Free agent" a 'Club_Position', 'Club_Kit' y 'Club_Joining'. En 'Contract_Expiry' y'Club_Kit' colocaremos un 0, actuando como *centinela* de nuevo.

```
fifa$Club_Position[fifa$Club_Position == ""] = "Free agent"
fifa$Club_Joining[fifa$Club_Joining == ""] = "Free agent"
fifa$Club_Kit[is.na(fifa$Club_Kit)] = 0
fifa$Contract_Expiry[is.na(fifa$Contract_Expiry)] = 0
```

Comprobemos que lo hemos solucionado todo.

```
sum(colSums(is.na(fifa)))
```

```
## [1] 0
```

```
sum(colSums(fifa == ""))
```

```
## [1] 0
```

```
fifa[which(fifa$Club == "Free agent"), ]
```

```
##              Name Nationality National_Position National_Kit       Club
## 384 Didier Drogba Ivory Coast  NO international            0 Free agent
##     Club_Position Club_Kit Club_Joining Contract_Expiry Rating Height Weight
## 384    Free agent        0   Free agent               0     81    189     80
##     Preffered_Foot Birth_Date Age Preffered_Position    Work_Rate Weak_foot
## 384          Right 03/11/1978  39                ST Medium / Low         4
##     Skill_Moves Ball_Control Dribbling Marking Sliding_Tackle Standing_Tackle
## 384           3           80        74      22             29              32
##     Aggression Reactions Attacking_Position Interceptions Vision Composure
## 384         80        80                 81            42     76        80
##     Crossing Short_Pass Long_Pass Acceleration Speed Stamina Strength Balance
## 384       67         60        60           64    64      62       86      56
##     Agility Jumping Heading Shot_Power Finishing Long_Shots Curve
## 384      63      76      85         85        82         79    78
##     Freekick_Accuracy Penalties Volleys GK_Positioning GK_Diving GK_Kicking
## 384                84        84      76              6        10          8
##     GK_Handling GK_Reflexes clasificacion portero
## 384          11          14     Muy bueno   FALSE
```

Solucionado.

Con los datos preparados, procedemos al análisis de los datos.

# 4. Análisis de los datos.

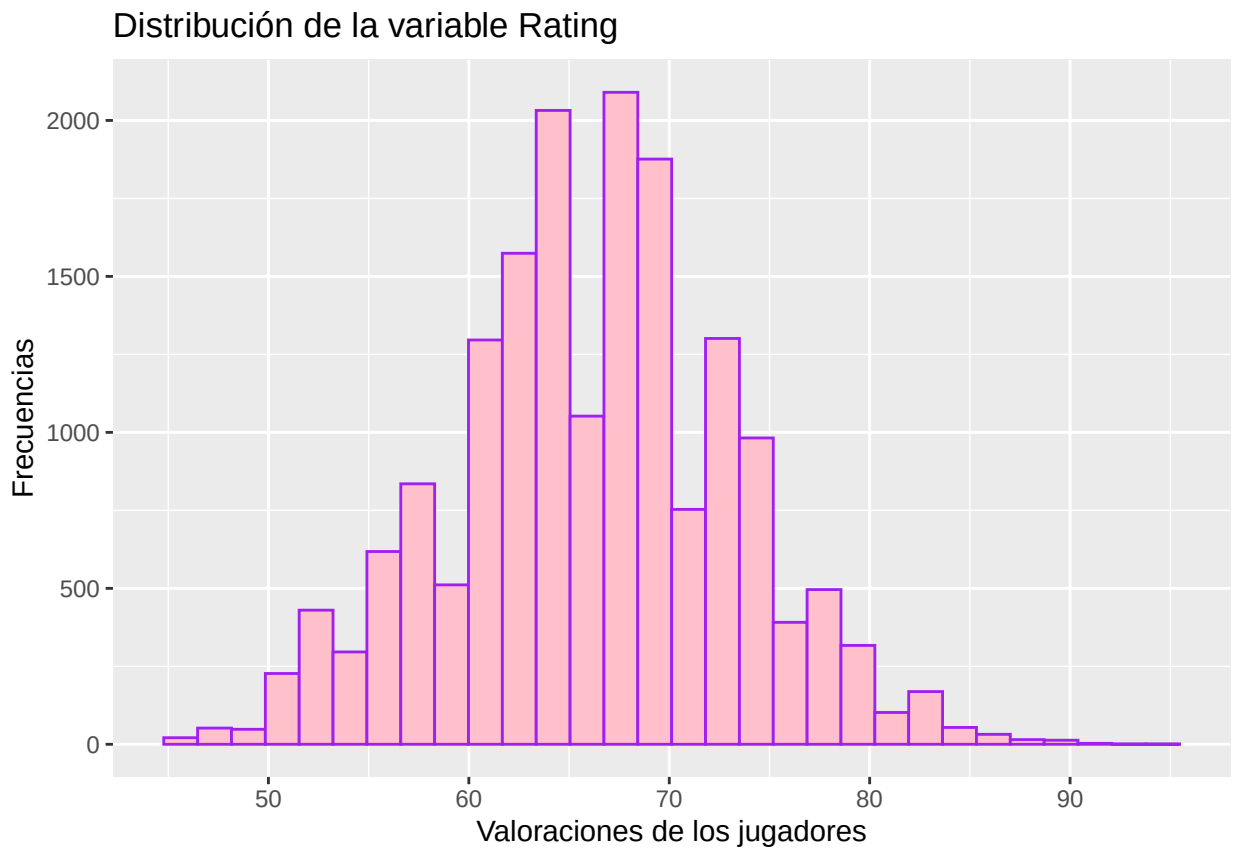
**Selección de los grupos de datos que se quieren analizar/comparar (planificacion de los analisis a aplicar),representación de los resultados a partir de tablas y gráficas, comprobación de la normalidad y homogeneidad de la varianza, aplicación de pruebas estadisticas para comparar los grupos de datos. En funcion de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipotesis, correlaciones, regresiones, etc. Aplicar al menos tres metodos de analisis diferentes.**

Vamos a responder a cada una de las preguntas planteadas representando los resultados en tablas y gráficas y aplicando pruebas estadísticas como intervalos de Confianza, contraste de hipótesis, regresión, etc, previa comprobación de la normalidad y homegeneidad.

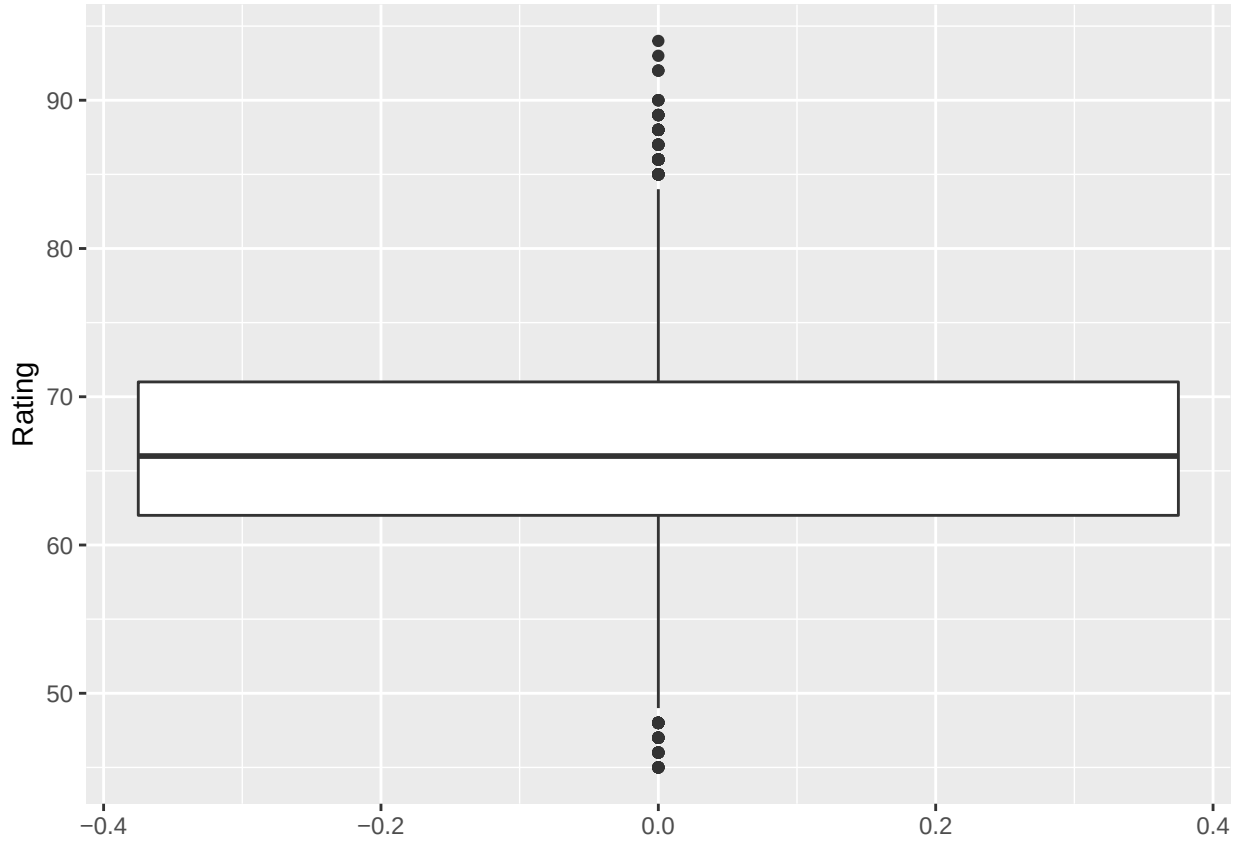
### ¿Cuál es el valor promedio del Rating de los jugadores?

Representamos visualmente esta variable:

```
# Histograma de Rating
library(ggplot2)
ggplot(fifa, aes(Rating)) + geom_histogram(fill="pink",col="purple") +
 xlab("Valoraciones de los jugadores") + ylab("Frecuencias") +
  ggtitle("Distribución de la variable Rating")
```




```
# Boxplot de Rating
library(ggplot2)
ggplot(fifa, aes(y=Rating)) + geom_boxplot()
```




La variable **Rating** se distribuye **simétricamente**, aproximadamente como una **distribución Normal**, con un valor mínimo de 45 y un máximo de 95, siendo muy poco frecuentes los valores menores de 50 y los mayores de 85.

Vamos a calcularlo de manera análitica a través de un **intervalo de confianza de la media poblacional de la variable 'Rating'**

Construimos primero una función que dada una muestra y un nivel de confianza dado, calcule el intervalo de confianza asociado. Esto facilitará los cálculos posteriores.

```
IC <- function(x, alfa=0.05){
  n <- length(x)
  errorT <- sd(x)/sqrt(n)
  errorT
  t<-qnorm(1-alfa/2)
  t
  error<- t*errorT
  error
  intervalo = c(mean(x) - error, mean(x) + error)
  return(intervalo)
}
```

Intervalo de confianza al 95% de la media poblacional de la variable Rating:

```
IC(fifa$Rating, alfa=0.05)
```

```
## [1] 66.06151 66.27087
```

Vamos a comprobar con la función `t.test` que realiza este cálculo automáticamente:

```
# Comprobamos con la función ya implementada
t.test(fifa$Rating, sigma.df=sd(fifa$Rating))
```

```
##
##  One Sample t-test
##
## data:  fifa$Rating
## t = 1238.9, df = 17587, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  66.06151 66.27088
## sample estimates:
## mean of x
##  66.16619
```

Vemos que efectivamente nos sale el mismo resultado que con la función implementada.

La **interprestación** de este intervalo de confianza es: si obtenemos infinitas muestras de las valoraciones globales (Rating) de la población de jugadores de fútbol, el 95 % de los intervalos de confianza calculados a partir de estas muestras contendrían al valor real de la media poblacional.

**¿Los jugadores de fútbol zurdos tienen mejor control de la pelota que los diestros?**

**¿Los jugadores de fútbol zurdos tienen mejor valoración global que los diestros?**

**¿Los jugadores de fútbol zurdos tienen mejor dribblling que los diestros?**

Vamos a seleccionar los jugadores que **no son porteros**:

```
# Jugadores no porteros:
fifa_jug <- filter(fifa, portero == F)

# Jugadores porteros:
fifa_port <- filter(fifa, portero == T)
```

Creamos un dataframe para los jugadores Zurdos (**Z**) y otro para los jugadores diestros (**D**):

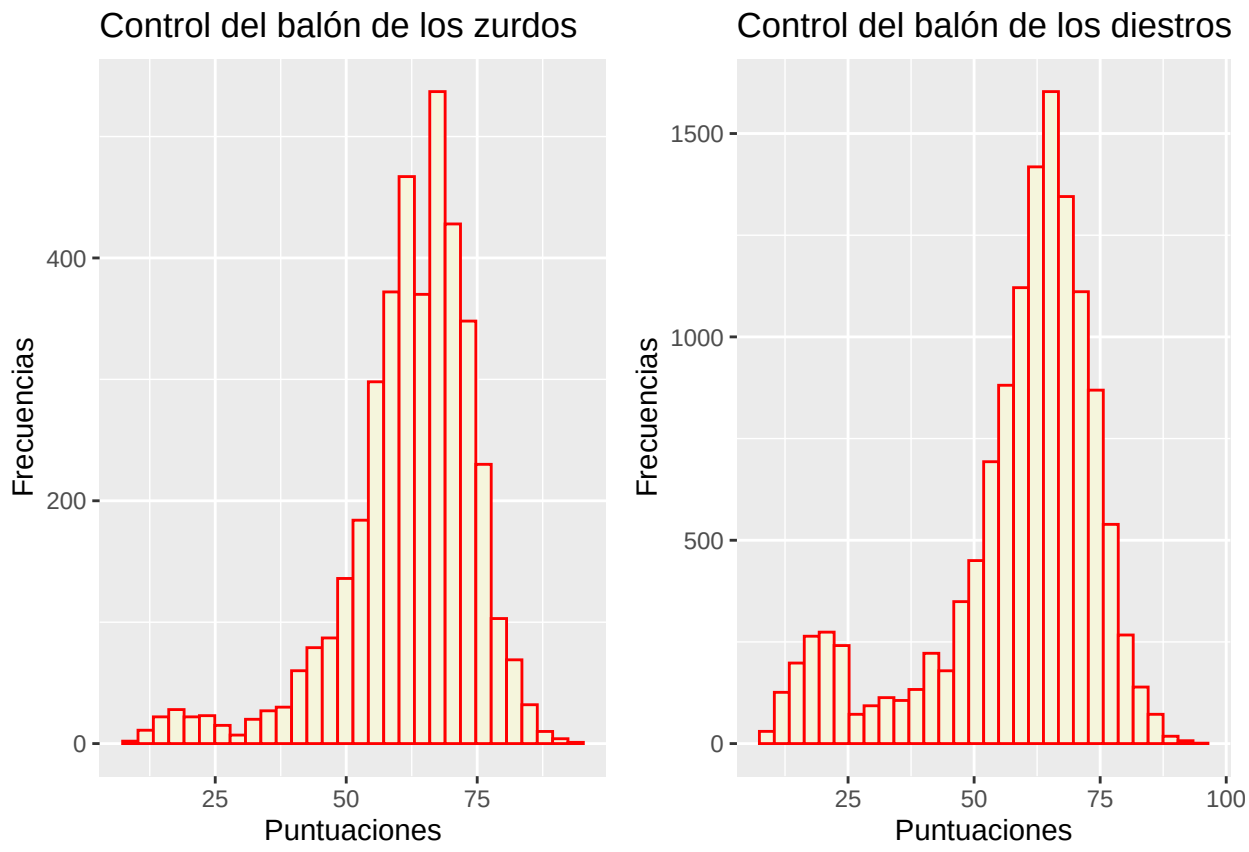
```
# fifa para zurdos y diestros
Z = fifa_jug[fifa_jug$Preffered_Foot == "Left",]
D = fifa_jug[fifa_jug$Preffered_Foot == "Right",]
```

Para hacernos una primera idea, vamos a representar visualmente mediante dintintos gráficos los datos del control de pelota, la valoración global y el dribling, de forma comparativa para zurdos y diestros:

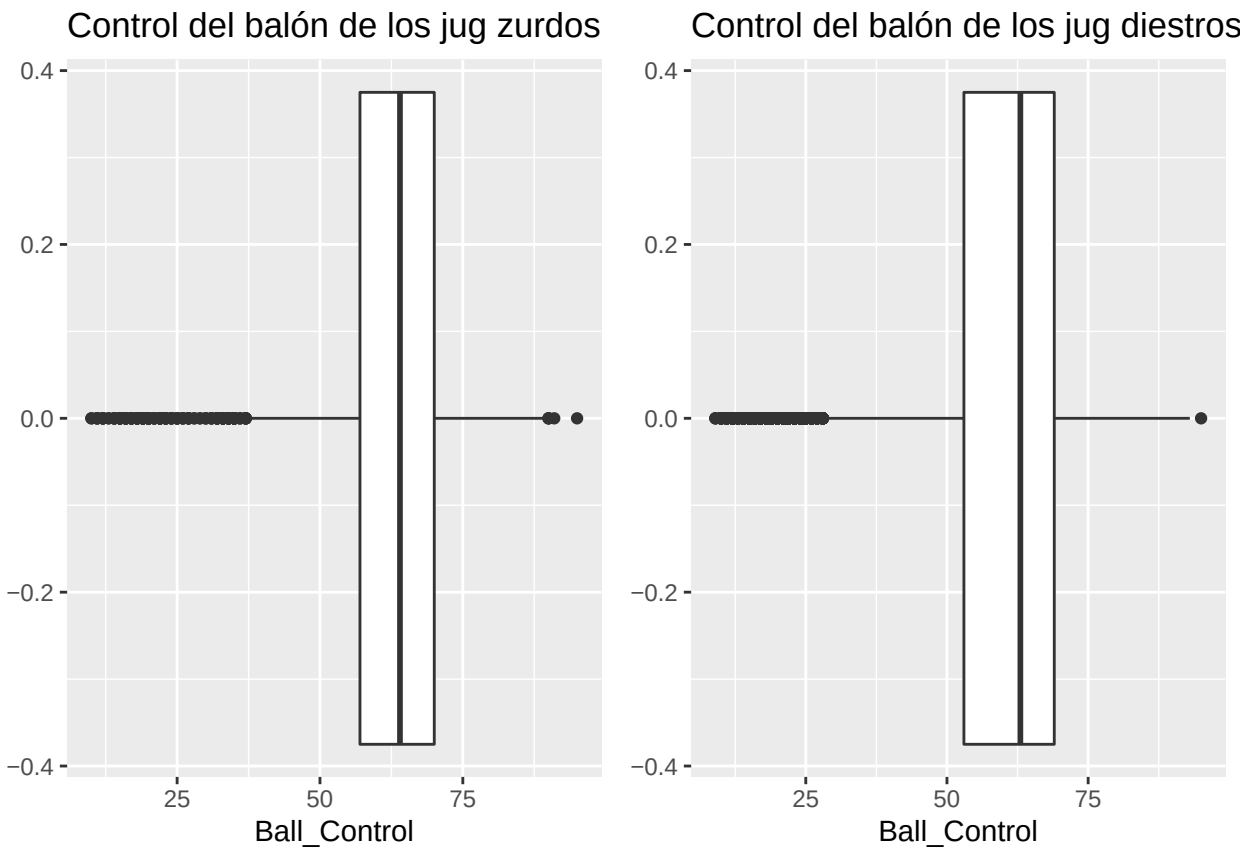
```
library(gridExtra)
library(ggplot2)
g1 <- ggplot( Z, aes(Ball_Control)) + geom_histogram(fill="beige",col="red") +
 xlab("Puntuaciones") + ylab("Frecuencias") +
  ggtitle("Control del balón de los zurdos")
```

```
g2 <- ggplot( D, aes(Ball_Control)) + geom_histogram(fill="beige",col="red") +
 xlab("Puntuaciones") + ylab("Frecuencias") +
  ggtitle("Control del balón de los diestros")

grid.arrange(g1,g2, nrow=1)
```




```
g3 <- ggplot( Z, aes(x=Ball_Control)) + geom_boxplot() +  ggtitle("Control del balón de los jug zurdos")
g4 <- ggplot( D, aes(x=Ball_Control)) + geom_boxplot() +  ggtitle("Control del balón de los jug diestros
grid.arrange(g3,g4, nrow=1)
```

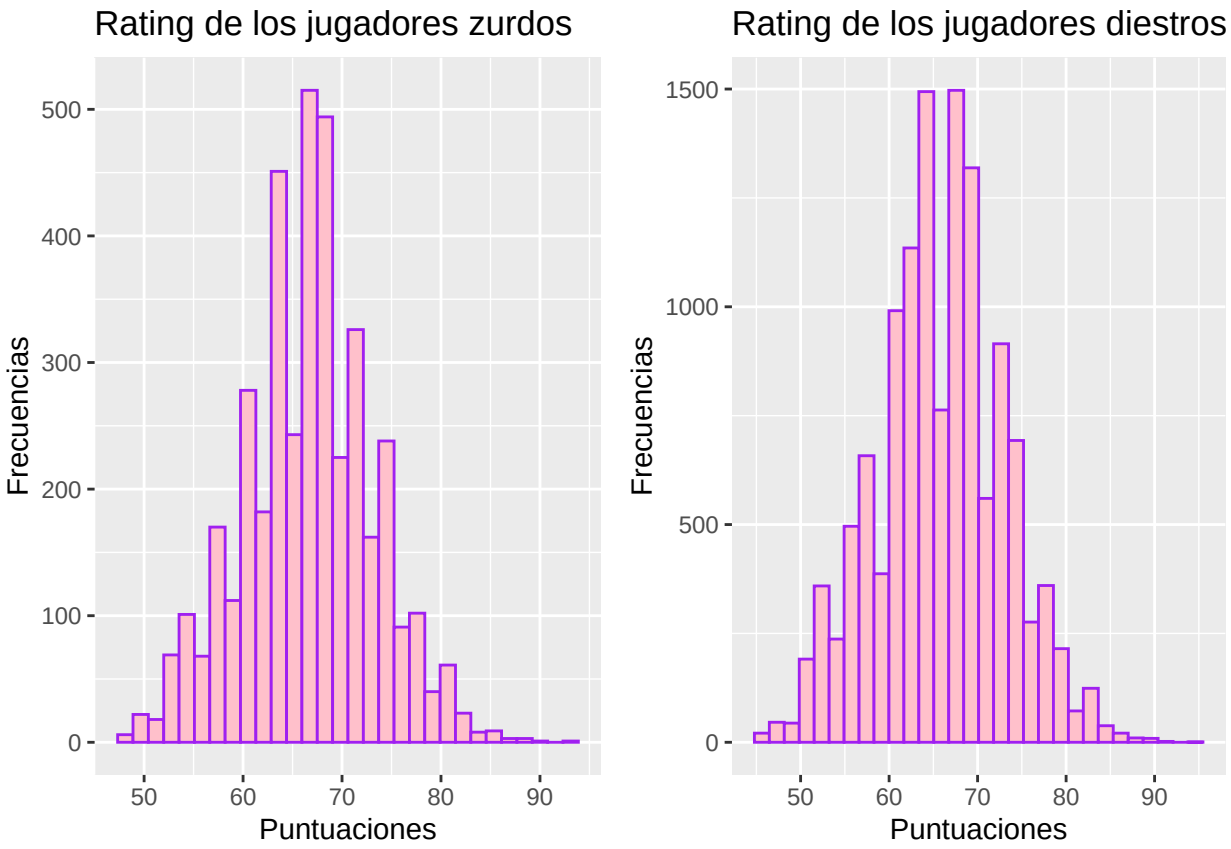



Con respecto al **control de la pelota** podemos apreciar que el volumen de jugadores con puntuaciones inferiores a 25 es mayor en los jugadores diestros que en los zurdos, y que las puntuaciones que más jugadores tienen, en ambos tipos, se encuentran en torno a 60 y 70.

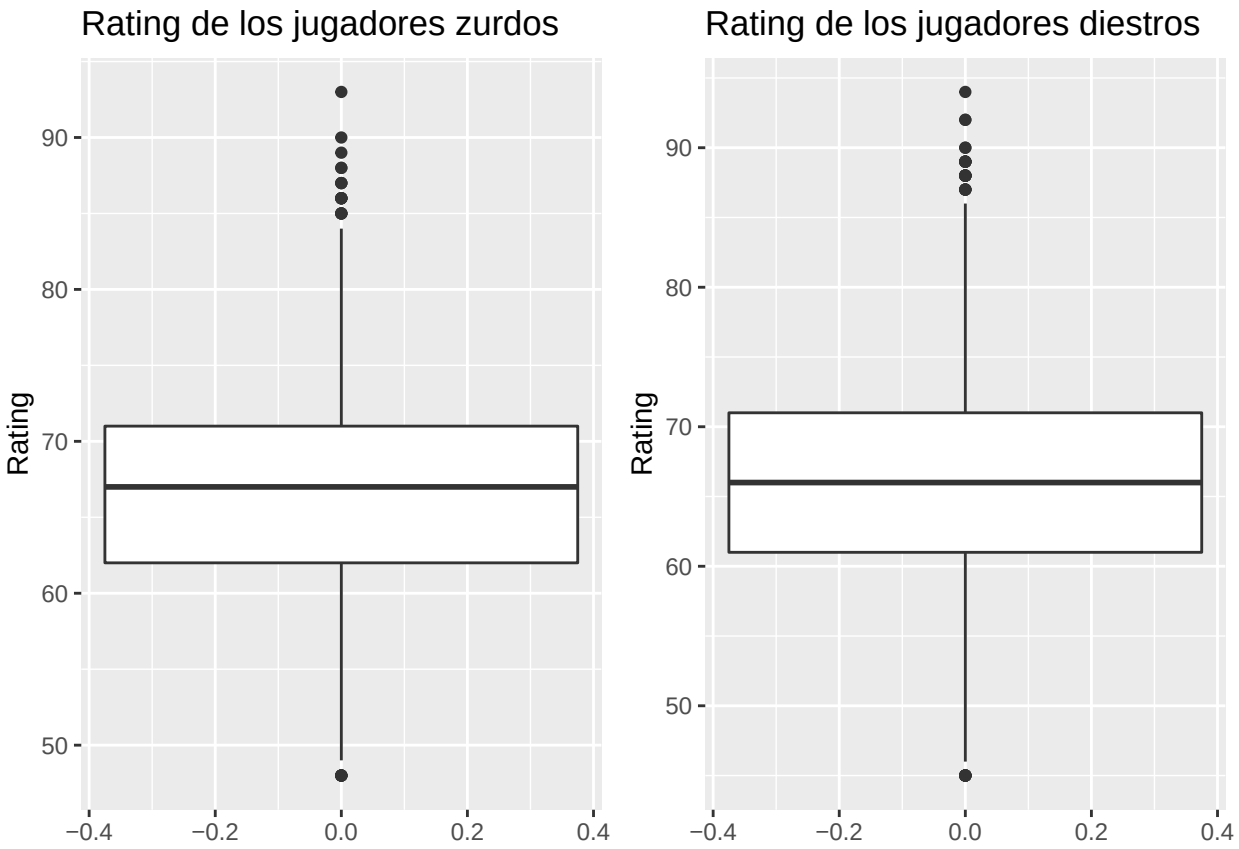
```
library(gridExtra)
library(ggplot2)
g5 <- ggplot( Z, aes(Rating)) + geom_histogram(fill="pink",col="purple") +
 xlab("Puntuaciones") + ylab("Frecuencias") +
  ggtitle("Rating de los jugadores zurdos")

g6 <- ggplot( D, aes(Rating)) + geom_histogram(fill="pink",col="purple") +
 xlab("Puntuaciones") + ylab("Frecuencias") +
  ggtitle("Rating de los jugadores diestros")

grid.arrange(g5,g6, nrow=1)
```




```
g7 <- ggplot( Z, aes(y=Rating)) + geom_boxplot() +  ggtitle("Rating de los jugadores zurdos")
g8 <- ggplot( D, aes(y=Rating)) + geom_boxplot() +  ggtitle("Rating de los jugadores diestros")
grid.arrange(g7,g8, nrow=1)
```

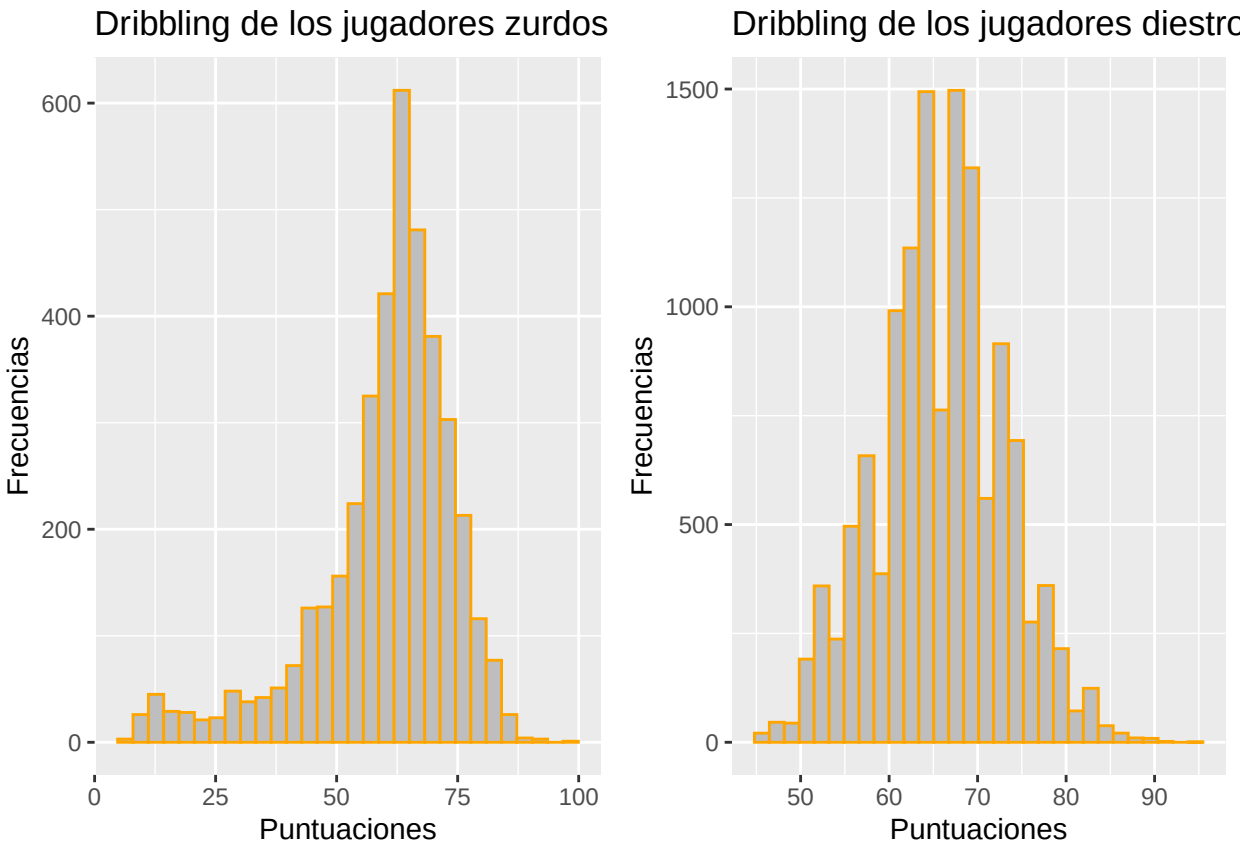



Para la variable **Rating**, ambas muestras se presentan muy igualadas, siendo la distribución a partir del tercer cuartil, prácticamente similares.

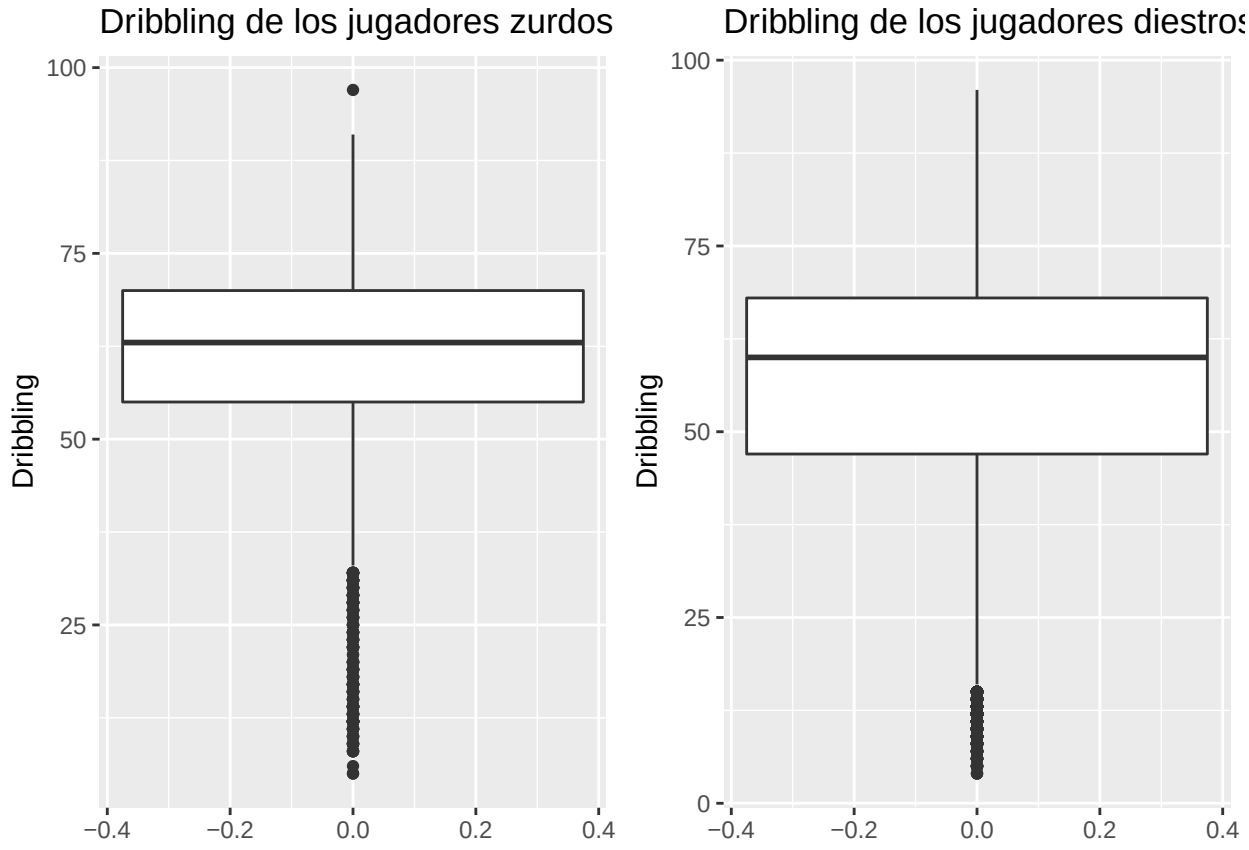
```
library(gridExtra)
library(ggplot2)
g9 <- ggplot( Z, aes(Dribbling)) + geom_histogram(fill="grey",col="orange") +
 xlab("Puntuaciones") + ylab("Frecuencias") +
  ggtitle("Dribbling de los jugadores zurdos")

g10 <- ggplot( D, aes(Rating)) + geom_histogram(fill="grey",col="orange") +
 xlab("Puntuaciones") + ylab("Frecuencias") +
  ggtitle("Dribbling de los jugadores diestros")

grid.arrange(g9,g10, nrow=1)
```




```
g11 <- ggplot( Z, aes(y=Dribbling)) + geom_boxplot() +  ggtitle("Dribbling de los jugadores zurdos")
g12 <- ggplot( D, aes(y=Dribbling)) + geom_boxplot() +  ggtitle("Dribbling de los jugadores diestros")
grid.arrange(g11,g12, nrow=1)
```




Con respecto a la variable **Dribbling** vemos que para los jugadores diestros, el rango intercuartílico es claramente superior que para los jugadores zurdos, siendo el volumen en éstos mayor en cuanto a las puntuaciones inferiores, menores de 50.

Calculemos ahora las respuestas a las preguntas planteadas de manera analítica a través del siguiente **Contrastes de Hipótesis**. Lo escribimos de manera genérica, particularizando luego para la $\mu$ de cada variable:

$$\begin{cases} H_0 : \mu_{zurdos} = \mu_{diestros} \\ H_1 : \mu_{zurdos} > \mu_{diestros} \end{cases}$$

siendo $\mu$ la **media** de **Ball_Control, Rating, Dribbling** para la primera, segunda y tercera pregunta, respectivamente.

Estamos ante un contraste de **dos muestras independientes**, ya que no tienen un relación directa o inversamente proporcional unas con otras.

Por el teorema central del límite podemos **asumir normalidad**, puesto que tenemos una muestra de tamaño grande (n=33 > 30) y hemos asumido que la población original es normal, por lo tanto el test es **paramétrico**.

Fijándonos en cómo hemos planteado la Hipótesis Alternativa, es un test **unilateral por la derecha**

Para comprobar si podemos asumir que las varianzas sean iguales (**homocedasticidad**) aplicamos el test `var.test` de R:

```
# Test de varianzas para Ball_Control
var.test(Z$Ball_Control, D$Ball_Control)
```

```
##
```

```
##  F test to compare two variances
##
## data:  Z$Ball_Control and D$Ball_Control
## F = 0.59095, num df = 4021, denom df = 12933, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5622844 0.6214864
## sample estimates:
## ratio of variances
##          0.5909475
```

```
# Test de varianzas para Rating
var.test(Z$Rating, D$Rating)
```

```
##
##  F test to compare two variances
##
## data:  Z$Rating and D$Rating
## F = 0.84569, num df = 4021, denom df = 12933, p-value = 1.037e-10
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.8046699 0.8893922
## sample estimates:
## ratio of variances
##          0.8456888
```

```
# Test de varianzas para Dribbling
var.test(Z$Dribbling, D$Dribbling)
```

```
##
##  F test to compare two variances
##
## data:  Z$Dribbling and D$Dribbling
## F = 0.62698, num df = 4021, denom df = 12933, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5965684 0.6593800
## sample estimates:
## ratio of variances
##          0.6269791
```

Vemos que para las 3 variables el p-valor es menor que $\alpha = 0,05$, por lo tanto, **rechazamos** en los 3 casos la $Ho$ de que las varianzas sean iguales en las dos poblaciones.

Creamos una función para calcular el estadístico de contraste, el valor crítico y el valor p:

```
library(dplyr)
library(kableExtra)

testCH <- function(var, x1, x2, CL=0.95,equalvar=TRUE, alternative="bilateral" ){

  mean1<-mean(x1)
  n1<-length(x1)
```

| var | mean_Left | mean_Right | n_Left | n_Right | obs_value | critical | pvalue |
|---|---|---|---|---|---|---|---|
| Bcontrol | 62.16335 | 58.47727 | 4022 | 12934 | 15.18192 | 1.64503 | 0 |

```
  sd1<-sd(x1)
  mean2<-mean(x2)
  n2<-length(x2)
  sd2<-sd(x2)

  if (equalvar==TRUE){
  s <-sqrt( ((n1-1)*sd1^2 + (n2-1)*sd2^2 )/(n1+n2-2) )
  Sb <- s*sqrt(1/n1 + 1/n2)
  df<-n1+n2-2
  }
  else{
  Sb <- sqrt( sd1^2/n1 + sd2^2/n2 )
  denom <- ( (sd1^2/n1)^2/(n1-1) + (sd2^2/n2)^2/(n2-1))
  df <- ( (sd1^2/n1 + sd2^2/n2)^2 ) / denom
  }
  alfa <- (1-CL)
  t<- (mean1-mean2) / Sb

  if (alternative=="bilateral"){
  tcritical <- qt( alfa/2, df, lower.tail=FALSE ) #two sided
  pvalue<-pt( abs(t), df, lower.tail=FALSE )*2 #two sided
  }
  else if (alternative=="less"){
  tcritical <- qt( alfa, df, lower.tail=TRUE )
  pvalue<-pt( t, df, lower.tail=TRUE )
  }
  else{ #(alternative=="greater")
  tcritical <- qt( alfa, df, lower.tail=FALSE )
  pvalue<-pt( t, df, lower.tail=FALSE )
  }

  #Guardamos el resultado en un data frame

  resultado<-data.frame(var, mean1, mean2, n1, n2,t,tcritical,pvalue)
  return (resultado)
  }
```

Aplicamos la función a Ball_Control:

```
testBControl<-testCH('Bcontrol',Z$Ball_Control, D$Ball_Control, equalvar=FALSE, alternative = "greater"

# Cambiamos los nombres del data frame para nuestro ejercicio
nombres_col <- c("var", "mean_Left", "mean_Right", "n_Left", "n_Right", "obs_value", "critical", "pvalu
colnames(testBControl) <- nombres_col

testBControl  %>% kable() %>% kable_styling()
```

Comprobamos con la función implementada:

| var | mean_Left | mean_Right | n_Left | n_Right | obs_value | critical | pvalue |
|---|---|---|---|---|---|---|---|
| Rating | 66.58155 | 65.8582 | 4022 | 12934 | 5.933765 | 1.645065 | 0 |

```
t.test( Z$Ball_Control, D$Ball_Control, var.equal=FALSE, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  Z$Ball_Control and D$Ball_Control
## t = 15.182, df = 8623.8, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  3.286679      Inf
## sample estimates:
## mean of x mean of y
##  62.16335  58.47727
```

Vemos que efectivamente coincide.

Aplicamos la función a Rating:

```
testRating<-testCH('Rating',Z$Rating, D$Rating, equalvar=FALSE, alternative = "greater")
colnames(testRating) <- nombres_col
testRating  %>% kable() %>% kable_styling()
```

Comprobamos con la función implementada:

```
t.test( Z$Rating, D$Rating, var.equal=FALSE, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  Z$Rating and D$Rating
## t = 5.9338, df = 7218.3, p-value = 1.549e-09
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.5228087       Inf
## sample estimates:
## mean of x mean of y
##  66.58155  65.85820
```

Coincide.

Aplicamos la función a Dribbling:

```
testDribbling<-testCH('Dribbling', Z$Dribbling, D$Dribbling, equalvar=FALSE, alternative = "greater")
colnames(testDribbling) <- nombres_col
testDribbling  %>% kable() %>% kable_styling()
```

Comprobamos con la función implementada:

| var | mean_Left | mean_Right | n_Left | n_Right | obs_value | critical | pvalue |
|---|---|---|---|---|---|---|---|
| Dribbling | 60.15266 | 55.09688 | 4022 | 12934 | 18.13756 | 1.645036 | 0 |

| var | mean_Left | mean_Right | n_Left | n_Right | obs_value | critical | pvalue |
|---|---|---|---|---|---|---|---|
| Rating | 66.58155 | 65.85820 | 4022 | 12934 | 5.933765 | 1.645065 | 0 |
| Dribbling | 60.15266 | 55.09688 | 4022 | 12934 | 18.137562 | 1.645036 | 0 |
| Bcontrol | 62.16335 | 58.47727 | 4022 | 12934 | 15.181923 | 1.645030 | 0 |

```
t.test( Z$Dribbling, D$Dribbling, var.equal=FALSE, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  Z$Dribbling and D$Dribbling
## t = 18.138, df = 8359.4, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  4.597236      Inf
## sample estimates:
## mean of x mean of y
##  60.15266  55.09688
```

Coincide.

La tabla de los resultados de los tests anteriores es la siguiente:

```
testall = rbind(testRating,testDribbling,testBControl)
testall %>% kable() %>% kable_styling()
```

Para los tres casos el valor observado es mayor que el valor crítico y el p-valor es menor que $\alpha = 0.05$, por lo tanto se **rechazan en los 3 casos las Hipótesis Nulas**, es decir, que el Control de la pelota, el Rating y el Dribbling es superior en los jugadores zurdos que en los diestros, al 95% de confianza.

**¿El porcentaje de jugadores con un Rating superior a 90 es diferente en el Barcelona y en el Madrid?**

Vamos a contestar a esta pregunta a través del siguiente **contraste de hipótesis**:

$$\begin{cases} H_0 : p_{Bcn} = p_{Md} \\ H_1 : p_{Bcn} \neq p_{Md} \end{cases}$$

Siendo:

$p_{Bcn}$ = proporción de jugadores con Rating mayor a 90 del equipo del FC Barcelona y

$p_{Md}$ = proporción de jugadores con Rating mayor a 90 del equipo del Madrid.

Vamos a aplicar el Test para la diferencia de dos prorporciones con muestras grandes.Vamos a calcular la proporción de jugadores con un Rating mayor a 90 para el equipo del Barcelona, y lo mismo para el equipo del Madrid. Vamos a obtener dos proporciones y vamos a comparar si la primera es significativamente diferente de la segunda, es decir, estamos ante un test **bilateral**.

Definimos las muestras y las proporciones:

```
Bcn <- fifa[fifa$Club=="FC Barcelona",]
Md <- fifa[fifa$Club=="Real Madrid",]
fifa_Rat90Bcn = Md[Bcn$Rating > 90,]
fifa_Rat90Md = Md[Md$Rating > 90,]
n1 <-nrow(Bcn)
n2<-nrow(Md)
p1 <- sum(Bcn$Rating>90)/n1
p2 <- sum(Md$Rating>90)/n2

paste("El número de jugadores del FC Barcelona es",n1, "y la proporción de jugadores con un Rating mayo
```

```
## [1] "El número de jugadores del FC Barcelona es 33 y la proporción de jugadores con un Rating mayor a
```

```
paste("El número de jugadores del Madrid es",n2, "y la proporción de jugadores con un Rating mayor a 90
```

```
## [1] "El número de jugadores del Madrid es 33 y la proporción de jugadores con un Rating mayor a 90, 0
```

Implementamos el test:

```
alpha<-0.03
p<- (n1*p1 + n2*p2) /(n1+n2)
zobs<- (p1-p2) / sqrt( p*(1-p)*(1/n1 + 1/n2))
zcrit <- qnorm(alpha/2, lower.tail=FALSE)
pvalue<- 2*pnorm(zobs, lower.tail=FALSE)
c(zobs, zcrit, pvalue)
```

```
## [1] 1.0317539 2.1700904 0.3021874
```

Comprobamos con la función ya implementada en R:

```
success <- c(p1*n1,p2*n2)
n <- c(n1,n2)
prop.test( success, n, alternative="two.sided", correct=FALSE, conf.level = 0.97)
```

```
##
##  2-sample test for equality of proportions without continuity
##  correction
##
## data:  success out of n
## X-squared = 1.0645, df = 1, p-value = 0.3022
## alternative hypothesis: two.sided
## 97 percent confidence interval:
##  -0.06583462  0.18704674
## sample estimates:
##     prop 1     prop 2
## 0.09090909 0.03030303
```

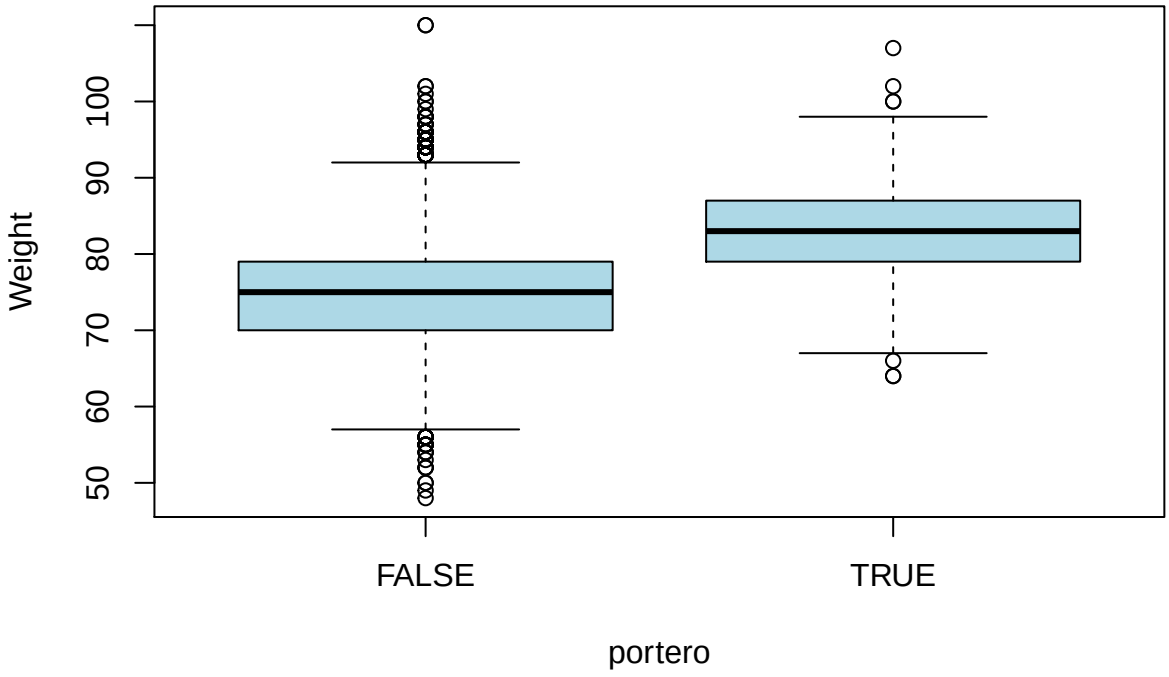
Vemos que efectivamente coincide.

El p-valor=0.3022 $>$ $\alpha$=0.03 por lo tanto, **aceptamos la Hipótesis nula** $H_0$, es decir, no podemos afirmar que las diferencias de las proporciones del Barcelona y el Madrid sean significativamente diferentes con un nivel de confianza del 97%.

**¿El peso de los porteros es mayor al peso de los jugadores de campo?**

Veamos la distribución del peso entre porteros y jugadores gráficamente:

```
boxplot(Weight ~ portero, fifa, col = "lightblue")
```




Podemos ver claramente que los porteros suelen pesar más que los jugadores. Comprobémoslo de manera analítica a través de un intervalo de cofianza al 95% de la media poblacional de la variable Weight para porteros y jugadores:

```
IC(fifa_jug$Weight, alfa=0.05)
```

```
## [1] 74.86264 75.06612
```

Comprobamos con la función *t.test()*.

```
t.test(fifa_jug$Weight, conf.level=0.95)$conf.int
```

```
## [1] 74.86263 75.06613
## attr(,"conf.level")
## [1] 0.95
```

Coinciden.

Intervalo de confianza al 95% de la media poblacional para los porteros:

```r
IC(fifa_port$Weight, alfa=0.05)
```

```
## [1] 82.53818 83.47448
```

Comprobamos con la función *t.test()*.

```r
t.test(fifa_port$Weight, conf.level=0.95)$conf.int
```

```
## [1] 82.53728 83.47538
## attr(,"conf.level")
## [1] 0.95
```

Coinciden.

Además, como los intervalos son disjuntos, **podemos asegurar al 95% que la media del peso de los porteros de la última década es mayor que la de los jugadores de campo de la última década**.

```r
IC(fifa_port$Weight, alfa=0.05)[2] - IC(fifa_jug$Weight, alfa=0.05)[1]
```

```
## [1] 8.611841
```

```r
IC(fifa_port$Weight, alfa=0.05)[1] - IC(fifa_jug$Weight, alfa=0.05)[2]
```

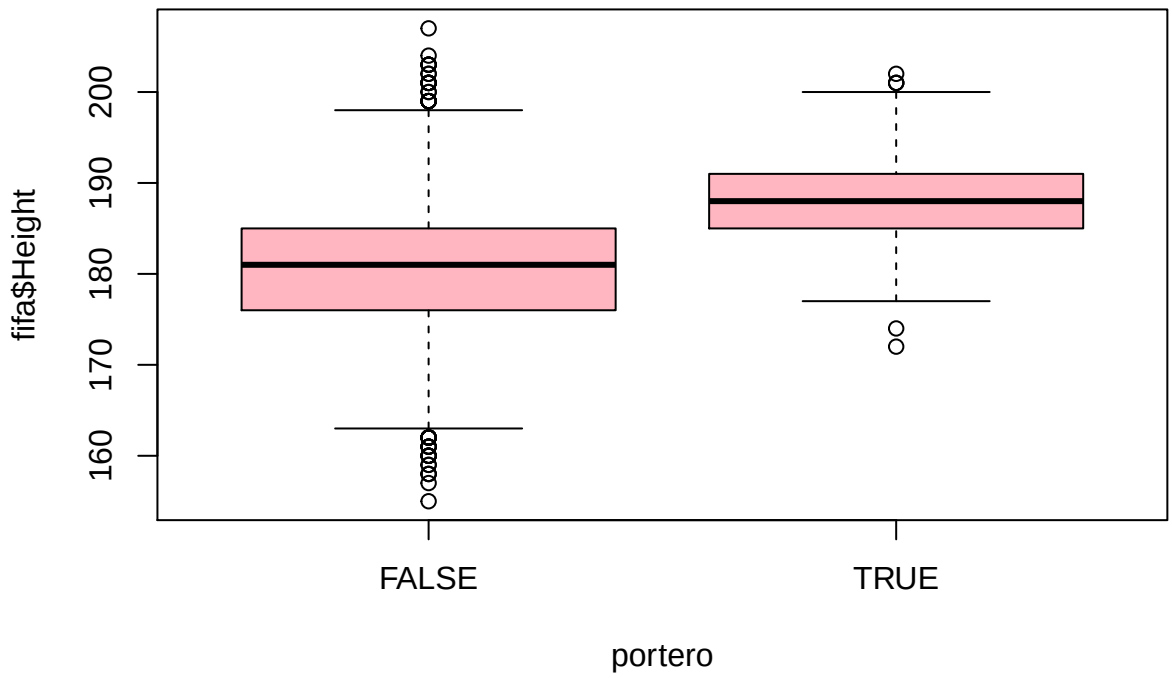
```
## [1] 7.472061
```

De hecho, fijándonos en los límites inferiores y superiores de los intervalos, podemos asegurar, al 95%, que la media de peso de los porteros de la última década es entre 7.47 y 8.61 kg superior a la de los jugadores de campo.

### ¿Son los porteros al menos 5 cms más altos que los jugadores de campo?

Representamos visualmente la altura con respecto a porteros y jugadores:

```r
boxplot(fifa$Height ~ portero, fifa, col = "lightpink")
```




Vemos claramente que la altura de los porteros es mayor que la de los jugadores.

Ahora plantearemos si podemos aceptar que la altura de los porteros supera en más de 5 centímetros la altura de los jugadores de campo, a través del siguiente **contraste de hipótesis**:

- $H_0$: $\mu_{portero} - \mu_{jugador} \leq 5$
- $H_1$: $\mu_{portero} - \mu_{jugador} > 5$

donde $mu$ es la media poblacional de la altura de los porteros/jugadores.

Estamos ante un test de dos muestras sobre la media con varianzas desconocidas. Por el teorema del límite central, podemos asumir normalidad, pues las dos muestras tienen un tamaño muy superior a 30.

Aplicaremos la distribución t, dado que no se conocen la varianzas de la población. Es necesario comprobar si podemos suponer varianzas iguales. Para ello, aplicamos el test var.test de R:

```
var.test(fifa_port$Weight, fifa_jug$Weight)
```

```
##
##  F test to compare two variances
##
## data:  fifa_port$Weight and fifa_jug$Weight
## F = 0.78918, num df = 631, denom df = 16955, p-value = 7.348e-05
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.7073328 0.8857451
```

```
## sample estimates:
## ratio of variances
##          0.7891781
```

El p_value del test es p-value = 0.7891781 > 0.05. Por tanto, descartamos igualdad de varianzas en las dos poblaciones. Por tanto, asumimos igualdad de varianzas.

En consecuencia, el test se corresponde con un test de dos muestras independientes sobre la media con varianzas desconocidas iguales. El test es unilateral.

Generamos una función que calcule el t-test: valor del estadístico de contraste, valor crítico y p-value.

```
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 4.0.5
```

```
library(kableExtra)
myttest <- function(x1, x2, d=0, CL=0.95, equalvar=TRUE, alternative="bilateral"){
  mean1<-mean(x1)
  n1<-length(x1)
  sd1<-sd(x1)
  mean2<-mean(x2)
  n2<-length(x2)
  sd2<-sd(x2)
  if (equalvar==TRUE){
    s <-sqrt( ((n1-1)*sd1^2 + (n2-1)*sd2^2 )/(n1+n2-2) )
    Sb <- s*sqrt(1/n1 + 1/n2)
    df<-n1+n2-2
    }
  else{ #equalvar==FALSE
    Sb <- sqrt( sd1^2/n1 + sd2^2/n2 )
    denom <- ( (sd1^2/n1)^2/(n1-1) + (sd2^2/n2)^2/(n2-1))
    df <- ( (sd1^2/n1 + sd2^2/n2)^2 ) / denom
    }
  alfa <- (1-CL)
  t<- (mean1-mean2- d) / Sb
  if (alternative=="bilateral"){
    tcritical <- qt( alfa/2, df, lower.tail=FALSE ) #two sided
    pvalue<-pt( abs(t), df, lower.tail=FALSE )*2 #two sided
    }
  else if (alternative=="less"){
    tcritical <- qt( alfa, df, lower.tail=TRUE )
    pvalue<-pt( t, df, lower.tail=TRUE )
    }
  else{ #(alternative=="greater")
    tcritical <- qt( alfa, df, lower.tail=FALSE )
    pvalue<-pt( t, df, lower.tail=FALSE )
    }
  #Guardamos el resultado en un data frame
  info<-data.frame(t,tcritical,pvalue)
  info %>% kable() %>% kable_styling()
  return (info)
}
```

Lo evaluamos para las variables correspondientes, d=5, equalvar=TRUE y alternative="greater".

```
info<-myttest(fifa_port$Weight, fifa_jug$Weight, equalvar=TRUE,
              alternative = "greater")
info
```

```
##          t tcritical        pvalue
## 1 29.47915   1.64494 8.920927e-187
```

El valor crítico para un nivel de confianza del 95% es 1.64494 y el valor observado es 29.47915. Por tanto, nos encontramos en la zona de rechazo de la hipótesis nula y podemos concluir que **la altura de los porteros supera en más de 5 centímetros la altura de los jugadores de campo**. Se concluye lo mismo con el valor p, que da un valor de 8.920927e-187, muy inferior a alfa=0.05. Notar que a un nivel de confianza mucho mayor, como 99.9% también seguiría siendo muy inferior y podríamos seguir aceptando.

**¿Cuál sería el rating de un jugador de campo con pie izquierdo preferido, con un peso de 70, edad de 24, control del balón de 80 y visión de 60?**

Para responder a esta pregunta vamos a estimar un **modelo de regresión lineal múltiple** que tenga como variables explicativas: 'Age', 'portero', 'Weight', 'Preffered_Foot', 'Vision' y 'Ball_Control', y como variable dependiente el 'Rating' de los jugadores.

Tomaremos como nivel de referencia para 'portero' la categoría "o valor lógico"Portero" y para 'Preffered_Foot' la categoría "Left".

Para ello, primeramente convertiremos en factor las variables que R las ha entendido como carácter. También cambiaremos la variable 'portero' de la siguiente forma: TRUE → "Portero", FALSE → "Jugador de campo".

```
fifa$portero[fifa$portero == TRUE] = "Portero"
fifa$portero[fifa$portero == FALSE] = "Jugador de campo"

fifa <- mutate_if(fifa, is.character, as.factor)

contrasts(fifa$portero)
```

```
##                  Portero
## Jugador de campo       0
## Portero                1
```

```
contrasts(fifa$Preffered_Foot)
```

```
##       Right
## Left      0
## Right     1
```

La variable 'Preffered_Foot' tiene la categoría de referencia elegida, pero 'portero' no.

```
fifa$portero <- relevel(fifa$portero, ref="Portero")
contrasts(fifa$portero)
```

```
##                  Jugador de campo
## Portero                         0
## Jugador de campo                1
```

Generamos el modelo.

```
attach(fifa)

r = lm(Rating ~ Age + portero + Weight + Preffered_Foot + Vision + Ball_Control)
r
```

```
##
## Call:
## lm(formula = Rating ~ Age + portero + Weight + Preffered_Foot +
##     Vision + Ball_Control)
##
## Coefficients:
##             (Intercept)                      Age  porteroJugador de campo
##                28.81591                  0.44692                  -9.35312
##                  Weight      Preffered_FootRight                    Vision
##                 0.24443                 -0.04720                   0.08981
##            Ball_Control
##                 0.20522
```

```
detach(fifa)
```

El modelo estimado es el siguiente:

*Rating = 28.81591 + 0.44692Age -9.35312Jugador de campo + 0.24443Weight -0.04720Right_Foot + 0.08981Vision +0.20522Ball_Control*

Interpretemos ahora el modelo obtenido.

```
summary(r)
```

```
##
## Call:
## lm(formula = Rating ~ Age + portero + Weight + Preffered_Foot +
##     Vision + Ball_Control)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.0546  -3.3565  -0.2425   3.0590  26.0989
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              28.815910   0.561504  51.319   <2e-16 ***
## Age                       0.446922   0.008599  51.976   <2e-16 ***
## porteroJugador de campo -9.353119   0.231154 -40.463   <2e-16 ***
## Weight                    0.244431   0.006036  40.495   <2e-16 ***
## Preffered_FootRight      -0.047195   0.089209  -0.529    0.597
## Vision                    0.089808   0.003946  22.758   <2e-16 ***
## Ball_Control              0.205216   0.003697  55.502   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.96 on 17581 degrees of freedom
## Multiple R-squared:  0.5097, Adjusted R-squared:  0.5096
## F-statistic:  3046 on 6 and 17581 DF,  p-value: < 2.2e-16
```

- **Calidad del ajuste**: Multiple R-squared = 0.5097, Adjusted R-squared = 0.5096. Son práticamente iguales. Tomando el segundo, que penaliza el número de covariables utilizado, se interpreta del siguiente modo: el conjunto de covariables permiten explicar un 50.96% de la variable respuesta 'Rating'. Por tanto, es un modelo de mala calidad. Nota: cabe señalar, además, que es una medida de calidad o precisión *indirecta* en cuanto a que, para tener una medida más *directa* o real, deberíamos evaluarlo sobre una muestra de prueba. Para ello deberíamos haber utilizado, al menos, una muestra de entrenamiento y otra de validación. O incluso utilizar técnicas más completas como la validación cruzada. Esto lo dejamos para un análisis más exhaustivo, pero conviene comentarlo.
- **Contraste fundamental**: p-value: $<$ 2.2e-16. Por tanto, el conjunto de covariables *sirve* para describir la variable respuesta 'Rating'. Es decir, el vector de coeficientes poblacionales no es el vector nulo.
- **Contrastes particulares**: p-value(s) $<$ 0.05 todos, excepto para 'Preffered_Foot'. Esto indica que todas las covariables son significativas excepto ella. O lo que es lo mismo, que a un 95% de confianza podemos asegurar que todos los coeficientes poblacionales no nulos, excepción del de 'Preffered_Foot' (su categroría no base), que no podemos asgurarlo.
- **Interpretación de los coeficientes**: las variables 'Age', 'Weight', 'vision' y 'Ball_Control' tienen coeficientes postivos. Esto quiere decir que el aumento de la edad, peso o visión del jugador influyen en un aumento de su 'Rating'. Dicho de otro modo, a igualdad de condiciones, si un jugador es mayor que otro tendrá mayor 'Rating'. Análogo para mayor peso o visión de juego. Por su parte, las variables categóricas 'portero' y 'Preffered_foot' tienen los respectivos coeficientes negativos -3.9001 y -0.4618. Atendiendo a la categoría base de cada una, la interpretación es la siguiente: a igualdad de condiciones, un jugador de campo tienen un 'Rating' 3.9 menor que un portero. Análogamente, a igualdad de condiciones, un jugador diestro tiene un 'Rating' 0.46 menor que uno zurdo. Hay que tener en cuenta que, por ejemplo, un portero con misma visión que otro jugador de campo es lógico que tenga notablemente mayor 'Rating' que el jugador. Es coherente con la realidad futbolística.

Ahora aplicaremos el modelo de regresión para contestar a la pregunta planteada: predecir el rating de un jugador de campo con pie izquierdo preferido, con un peso de 70, edad de 24, control del balón de 80 y visión de 60.

```
(newdata = data.frame(Age=24, portero="Jugador de campo", Weight=70,
                      Preffered_Foot="Left", Vision=60, Ball_Control=80))
```

```
##   Age          portero Weight Preffered_Foot Vision Ball_Control
## 1  24 Jugador de campo     70           Left     60           80
```

```
predict(r, newdata)
```

```
##       1
## 69.1048
```

**El 'Rating' predecido por el modelo, para un jugador de campo con pie izquierdo preferido, con un peso de 70, edad de 24, control del balón de 80 y visión de 60 es 69.1048.**

Por último, plateemos un caso un tanto curioso. l

Veamos qué 'Rating' predice de un jugador con una vision de juego y control de balón perfectos. Es decir ambos valores serán 100, con el resto de características igual que las anteriores.

```
(newdata = data.frame(Age=24, portero="Jugador de campo", Weight=70,
                      Preffered_Foot="Left", Vision=100, Ball_Control=100))
```

```
##   Age          portero Weight Preffered_Foot Vision Ball_Control
## 1  24 Jugador de campo     70           Left    100          100
```

```
predict(r, newdata)
```

```
##        1
## 76.80142
```

Como vemos, nos predice un valor de 'Rating' bastante mediocre para los valores de visión de juego y control de balón indicados. Esto puede tener dos causas: la mala calidad del modelo, o que las dos variables, visión de juego y control de balón, no sean tan influyentes realmente como podríamos pensar. ¿Por cuál te decidirías?

Damos una pista.

```
cor(fifa$Rating, fifa$Vision)
```

```
## [1] 0.4893705
```

```
cor(fifa$Rating, fifa$Ball_Control)
```

```
## [1] 0.4632865
```

```
cor(select_if(fifa, is.numeric))[,4]
```

```
##       National_Kit           Club_Kit    Contract_Expiry             Rating
##         0.24122974        -0.17281479        -0.01043570         1.00000000
##             Height             Weight                Age          Weak_foot
##         0.04707022         0.13976567         0.45827627         0.22641114
##        Skill_Moves       Ball_Control          Dribbling            Marking
##         0.25199996         0.46328646         0.36862950         0.23668321
##     Sliding_Tackle    Standing_Tackle         Aggression          Reactions
##         0.21526329         0.24903709         0.40451379         0.82836859
## Attacking_Position      Interceptions             Vision          Composure
##         0.35462351         0.31943614         0.48937051         0.61369322
##           Crossing         Short_Pass          Long_Pass       Acceleration
##         0.40190217         0.49619190         0.48321098         0.20635527
##              Speed            Stamina           Strength            Balance
##         0.22421237         0.35527898         0.36916877         0.08772942
##            Agility            Jumping            Heading         Shot_Power
##         0.28327156         0.28991107         0.34341050         0.44188142
##          Finishing         Long_Shots              Curve  Freekick_Accuracy
##         0.32872806         0.41962801         0.42090852         0.39973895
##          Penalties            Volleys     GK_Positioning          GK_Diving
##         0.34007082         0.38662798        -0.01865670        -0.02765696
##         GK_Kicking        GK_Handling        GK_Reflexes
##        -0.03175227        -0.02137892        -0.02299458
```

Ahí vemos que el coeficiente de correlación de la edad 'Age' con 'Rating' es incluso menor que el de 'Vision', mientras que el coeficiente estimado en el modelo de esta útimo es considerablemente menor. Por tanto nos decidimos que la razón es la mala calidad de nuestro modelo. Seguramente se deba al conocido problema de la *multicolinealidad*, que se refiere a la fuerte relación de las covariables involucradas.

Y con esto hemos terminado el análisis de nuestros datos.

# 5. Resolución del problema. A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

Hemos conseguido a través de las representaciones gráficas y de manera analítica a través de los distintos análisis estadísticos responder a todas las preguntas. En base a esto, podemos enumerar las siguientes conclusiones, con un 95% de confianza:

- **La media de la valoración global de los jugadores se encuentra alrededor de 66 puntos.**
- **Los jugadores zurdos son superiores a los diestros en cuanto a control de la pelota, valoración global, y control de regateo.**
- **En el caso de los equipos del Real Madrid y el Barcelona, no podemos afirmar que el número de jugadores con una valoración global superior a 90 sea distinto en alguno de los dos.**
- **Los porteros pesan más que los jugadores de campo**.
- **Los porteros son 5 cms más altos que los jugadores de campo**
- **El 'Rating' para un jugador de campo con pie izquierdo preferido, con un peso de 70, edad de 24, control del balón de 80 y visión de 60 es 69.1048.**

# 6. Dataset final

Exportamos el dataset final procesado.

```
write.csv(fifa, "fifa_clean.csv")
```

# 7. Contribuciones

- Investigación previa: María Sánchez y Cayetano Bautista
- Redacción de las respuestas: María Sánchez y Cayetano Bautista
- Desarrollo código: María Sánchez y Cayetano Bautista