Práctica 2: Tratamiento y análisis de un dataset

Máster Universitario en Ciencia de Datos | M2.851

Alejandro Hernández Naranjo

8 de junio de 2021

1 Introducción

En este trabajo se desarrolla un caso práctico para identificar aquellos datos relevantes de un proyecto analítico donde se realizarán la integración, limpieza, validación y análisis de los datos.

Se ha realizado por Alejandro Hernández Naranjo con un dataset que recopila datos relativos a jugadores de fútbol profesional obtenido de kaggle.

1.1 Librería

Para la correcta ejecución de este trabajo se utilizan los siguientes paquetes.

```
library("stringr")
library("lubridate")
library("randomForest")
library("Hmisc")
library("nortest")
library("kableExtra")
```

2 Descripción del dataset

El dataset seleccionado se puede encontrar en https://www.kaggle.com/karangadiya/fifa19 (https://www.kaggle.com/karangadiya/fifa19). Este dataset recoge los atributos de los jugadores que figuran en el videojuego FIFA 19. La descripción de la mayoría de los atributos se puede consultar en https://www.fifplay.com/encyclopedia/ (https://www.fifplay.com/encyclopedia/).

Para el desarrollo del videojuego FIFA 19 se hace una recopilación de las características reales de los jugadores de fútbol y puntúa a su vez las habilidades que estos han demostrado durante sus partidos. De este modo le confiere cierta fidelidad a la realidad, lo que confiere a su vez al dataset unos datos de alta calidad.

Dicho esto, con él pretendo analizar diferencias entre jugadores zurdos y diestros, el valor de mercado de los jugadores y la influencia de sus atributos físicos en calidad como jugadores.

2.1 Campos

- **X.U.FEFF.**: Número de la fila
- **ID**: Identificador único del jugador.
- **Name**: Nombre del jugador
- **Age**: Edad del jugador
- **Photo**: Enlace web con la foto del jugador.
- **Nationality**: Nacionalidad del jugador.
- **Flag**: Enlace web con la bandera de su nacionalidad.
- **Overall**: Puntuación que valora en término globales qué buen jugador es respecto a sus habilidades y rendimiento.
- **Potential**: Puntuación que valora la capacidad global de un jugador de alcanzar unas habilidades o rendimientos.
- **Club**: Club de fútbol en el que está registrado el jugador en el momento en el que se constituyó el dataset.
- **Club.logo**: Enlace web donde podemos encontrar la imagen del logo del club.
- **Value**: Precio del jugador en el mercado.
- **Wage**: Salario del jugador.
- **Preferred.Foot**: Pie preferido o dominante, es decir, si es zurdo o diestro.
- **International.Reputation**: Es una puntuación de prestigio internacional.
- **Weak.Foot**: Puntuación de habilidad con el pie que no es el preferido.
- **Skill.Moves**: Puntuación de la habilidad en el movimiento.
- **Work.Rate**: Desempeño del jugador en ataque y defensa.
- **Body.Type**: A la morfología del cuerpo del jugador le han asignado el nombre de cada jugador.
- **Real.Face**: Indica si la cara es la real del jugador o no.
- **Position**: Posición que ocupa el jugador en el campo.
- **Jersey.Number**: Número de dorsal del jugador.
- **Joined**: Fecha en la que el jugador se incorporó a su club.
- **Loaned.From**: El club de donde el jugador es prestado.
- **Contact.Valid.Until**: Año hasta que es vigente el contrato del jugador.
- **Height**: Altura del jugador en pies.
- **Weight**: Peso en libras del jugador.
- **LS**: Puntuación como *Left Striker*.
- **ST**: Puntuación como *Striker*.
- **RS**: Puntuación como *Right Striker*.
- **LW**: Puntuación como *Left Wing*.
- **LF**: Puntuación como *Left Forward*.

- CF: Puntuación como *Centre Forward*.
- RF: Puntuación como *Right Forward*.
- RW: Puntuación como *Right Wing*.
- LAM: Puntuación como *Left Attacking Midfielder*.
- CAM: Puntuación como *Centre Attacking Midfielder*.
- RAM: Puntuación como *Right Attacking Midfielder*.
- LM: Puntuación como *Left Midfielder*.
- LCM: Puntuación como *Left Centre Midfielder*.
- CM: Puntuación como *Centre Midfielder*.
- RCM: Puntuación como *Right Centre Midfielder*.
- RM: Puntuación como *Right Midfielder*.
- LWB: Puntuación como *Left Wing Back*.
- LDM: Puntuación como *Left Defensive Midfielder*.
- CDM: Puntuación como *Central Defensive Midfielder*.
- RDM: Puntuación como *Right Defensive Midfielder*.
- RWB: Puntuación como *Right Wing Back*.
- LB: Puntuación como *Left Back*.
- LCB: Puntuación como *Left Centre Back*.
- CB: Puntuación como *Centre Back*.
- RCB: Puntuación como *Right Centre Back*.
- RB: Puntuación como *Right Back*.
- Crossing: Puntuación dando pases largos desde las bandas hacia áreas centrales campo cercanas a la portería del oponente.
- Finishing: Puntuación relativa a la habilidad del jugador de acabar la jugada en gol.
- HeadingAccuracy: Puntuación para valorar la precisión del jugador usando la cabeza para pasar, disparar a puerta o despejar la pelota.
- ShorPassing: Puntuación dando pasaes cortos.
- Volleys: Puntuación sobre la habilidad del jugador en las voleas.
- Dribbling: Puntuación en maniobras de *dribbling*.
- Curve: Puntuación que valora la habilidad del jugador de curvar la trayectoria que describe el balón cuando pasa o dispara a puerta.
- FKAccuracy: Puntuación en *Free Kick Accuracy*.
- LongPassing: Puntuación en pases largos.
- BallControl: Puntuación en control del balón.
- Acceleration: Puntuación que determina la capacidad de acelerar la velocidad de carrera.
- SrpintSpeed: Puntuación para valorar la velocidad en *sprint*.
- Agility: Puntuación de agilidad del jugador para controlar el balón. Tiene en cuenta la parte física y mental.
- Reactions: Puntuación de la reacción del jugador a los sucesos de su entorno.
- Balance: Puntuación del equilibrio del jugador.
- ShotPower: Puntuación en la potencia de disparo.
- Jumping: Puntuación de salto.
- Stamina: Puntuación que mide la capacidad de mantener un esfuerzo físico o mental.
- Strength: Puntuación en fuerza.
- Aggression: Puntuación en el nivel de casitgo y de presionar al contrario.
- Intercepctions: Puntuación en la habilidad de interceptar la pelota.
- LongShots: Puntuación en disparos largos.
- Positioning: Puntuación sobre lo bien que se posiciona el jugador en el campo.
- Vision: Puntuación en la habilidad mental de proyectar o entender las jugadas que se van a ejecutar.

- **Penalties**: Puntuación de la precisión en los tiros de penalti.
- **Composure**: Puntuación de la capacidad de mantener la compostura ante situaciones frustrantes o tensas.
- **Marking**: Puntuación de marcaje del oponente para prevenir que reciba la pelota.
- **StandingTackle**: Puntuación en el robo de balón estando de pie.
- **SlidingTackle**: Puntuación en el robo de balón deslizandose por el suelo.
- **GKDiving**: Puntuación del portero al lanzarse para capturar la pelota.
- **GKKicking**: Puntuación del portero para patear la bola.
- **GKPositioning**: Puntuación del portero para posicionarse correctamente.
- **GKReflexes**: Puntuación del portero en reflejos.
- **Release.Clause**: Precio a pagar para liberar el contrato de un jugador con su actual club.

3 Selección de los datos de interés

Pondré el foco en los campos que aportan información relativa a los atributos físicos de los jugadores y en los que puntúan sus habilidades.

Se tomará como referencia la posición del jugador que figura en el campo **Position**, por lo tanto, descarto aquellos campos que puntuaban al jugador en cada posición.

El atributo Overall se calcula teniendo en cuenta los atributos físicos, mentales y habilidades, con él podremos reducir la dimensionalidad del conjunto descartando el resto de puntuaciones. En la fórmula de cálculo del Overall también se tiene en cuenta la reputación iternacional.

```
# Lectura del dataset.
data_fifa <- read.csv("data.csv", header=TRUE, encoding="UTF-8")
# Selección de columnas.
columnas <- c('ID', 'Age', 'Nationality', 'Overall', 'Club', 'Value', 'Wage',
              'Preferred.Foot','Weak.Foot','Skill.Moves', 'Work.Rate',
              'Position', 'Height', 'Weight')
fifa_DF <- data_fifa[,columnas]
# Estructura del dateset resultante
str(fifa_DF)
```

```
## 'data.frame':    18207 obs. of  14 variables:
##  $ ID            : int  158023 20801 190871 193080 192985 183277 177003 176580 155862 200389
...
##  $ Age           : int  31 33 26 27 27 27 32 31 32 25 ...
##  $ Nationality   : chr  "Argentina" "Portugal" "Brazil" "Spain" ...
##  $ Overall       : int  94 94 92 91 91 91 91 91 91 90 ...
##  $ Club          : chr  "FC Barcelona" "Juventus" "Paris Saint-Germain" "Manchester United"
...
##  $ Value         : chr  "\200110.5M" "\20077M" "\200118.5M" "\20072M" ...
##  $ Wage          : chr  "\200565K" "\200405K" "\200290K" "\200260K" ...
##  $ Preferred.Foot: chr  "Left" "Right" "Right" "Right" ...
##  $ Weak.Foot     : int  4 4 5 3 5 4 4 4 3 3 ...
##  $ Skill.Moves   : int  4 5 5 1 4 4 4 3 3 1 ...
##  $ Work.Rate     : chr  "Medium/ Medium" "High/ Low" "High/ Medium" "Medium/ Medium" ...
##  $ Position      : chr  "RF" "ST" "LW" "GK" ...
##  $ Height        : chr  "5'7" "6'2" "5'9" "6'4" ...
##  $ Weight        : chr  "159lbs" "183lbs" "150lbs" "168lbs" ...
```

4 Limpieza de los datos

En esta etapa voy formatear los datos y prepararlos para la aplicación de los métodos de análisis de las siguientes fases.

Un vistazo a la composición del dataset revela que hay muchos campos numéricos con tipo chr porque han incluido los caracteres de las unidades de medida. Por otro lado, hay otros campos de caracteres que pueden ser factorizados, lo que permitirá cuantificar las ocurrencias de cada categoría.

4.1 Normalización de datos cuantitativos

4.1.1 Value

Este campo contiene una cadena de texto compuesto por el símbolo del € y un sufijo M o K como notación de la magnitud, si son millones o miles.

La estrategia sería, usando gsub, eliminar ambos caracteres, convertir el dato a número y posteriormente multiplicarlo por su magnitud.

```
# Eliminar símbolo del Euro € ("\200") de la cadena.
fifa_DF$Value <- gsub("€", '',fifa_DF$Value)

# Recorremos las filas del dataset
for (i in 1:nrow(fifa_DF)) {
  # Si contiene M:
  if (str_detect(fifa_DF$Value[i], "M") && !is.na(fifa_DF$Value[i])) {
    # Elimino la M del string y convierto a double.
    valor <-  as.double(gsub("M", '',fifa_DF$Value[i]))
    # Como M representa 1 millón, multiplico el valor por 1 millón.
    fifa_DF$Value[i] <- valor * 1000000
  }
  # Si contiene K:
  if (str_detect(fifa_DF$Value[i], "K") && !is.na(fifa_DF$Value[i])){
    # Elimino la K del string y convierto a double.
    valor <-  as.double(gsub("K", '',fifa_DF$Value[i]))
    # Como K representa mil, multiplico el valor por 1000.
    fifa_DF$Value[i] <- valor * 1000
  }
}
fifa_DF$Value <- as.numeric(fifa_DF$Value)

summary(fifa_DF$Value)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##         0    300000    675000   2410696   2000000 118500000
```

4.1.2 Wage

El campo Wage está expresado con el mismo formato que Value, con lo cual aplicamos las mismas instrucciones.

```
# Eliminar símbolo del Euro € ("\200") de la cadena.
fifa_DF$Wage <- gsub("€", '',fifa_DF$Wage)

# Recorremos las filas del dataset
for (i in 1:nrow(fifa_DF)) {
  # Si contiene M:
  if (str_detect(fifa_DF$Wage[i], "M") && !is.na(fifa_DF$Wage[i])) {
    # Elimino la M del string y convierto a double.
    valor <-  as.double(gsub("M", '',fifa_DF$Wage[i]))
    ## Como M representa 1 millón, multiplico el valor por 1 millón.
    fifa_DF$Wage[i] <- valor * 1000000
  }
  # Si contiene K:
  if (str_detect(fifa_DF$Wage[i], "K") && !is.na(fifa_DF$Wage[i])){
    # Elimino la K del string y convierto a double.
    valor <-  as.double(gsub("K", '',fifa_DF$Wage[i]))
    # Como K representa mil, multiplico el valor por 1000.
    fifa_DF$Wage[i] <- valor * 1000
  }
}
fifa_DF$Wage <- as.numeric(fifa_DF$Wage)

summary(fifa_DF$Wage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1000    3000    9731    9000  565000
```

4.1.3 Height

La altura está escrita con un apóstrofe como separador decimal. Uso gsub para sustituir el separador por el punto (.). El factor de conversión de pies a metros usado es 0.3048

```
# Sustituyo la (') por el punto  (.)
fifa_DF$Height <- gsub("'", '.',fifa_DF$Height)
# Convierto el tipo de la variable numeric
fifa_DF$Height <- as.numeric(fifa_DF$Height)
# Camio la unidedad de pies a metros.
fifa_DF$Height <- fifa_DF$Height*0.3048

summary(fifa_DF$Height)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   1.554   1.558   1.798   1.767   1.859   2.103      48
```

4.1.4 Weight

El peso está escrito usando el punto como separador decimal, con lo cual, solo hay que quitar los caracteres de la unidad de medida y convertir el campo a numérico. El factor de conversión de libras a Kilogramos usado es 2.2046

```
# Elimino los caracteres lbs del string
fifa_DF$Weight <- gsub("lbs", '',fifa_DF$Weight)
# Convierto el tipo de la variable numeric
fifa_DF$Weight <- as.numeric(fifa_DF$Weight)
# Camio la unidedad de libras a kilogramos
fifa_DF$Weight <- fifa_DF$Weight/2.2046

summary(fifa_DF$Weight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   49.90   69.85   74.84   75.29   79.83  110.22      48
```

4.2 Normalización de datos cualitativos

Durante el desarrollo de la actividad he visto que estos campos tienen algunas cadenas vacías. Entonces, antes de convertir a factor sustituyo los campos vacíos por el elemento NA para que el sistema reconozca que es valor nulo. De no ser así el valor vacío sería interpretado como una categoría más del conjunto.

4.2.1 Nationality

```
# Valores que puede tomar Nationality
head(unique(fifa_DF$Nationality), 15)
```

```
##  [1] "Argentina" "Portugal"  "Brazil"    "Spain"     "Belgium"   "Croatia"
##  [7] "Uruguay"   "Slovenia"  "Poland"    "Germany"   "France"    "England"
## [13] "Italy"     "Egypt"     "Colombia"
```

```
# Hacer que los strings vacíos sean nulos
fifa_DF$Nationality[fifa_DF$Nationality==''] <- NA
# Convertir a factor
fifa_DF$Nationality <- factor(fifa_DF$Nationality)

summary(fifa_DF$Nationality)
```

```
##             England             Germany               Spain           Argentina
##                1662                1198                1072                 937
##              France              Brazil               Italy            Colombia
##                 914                 827                 702                 618
##               Japan         Netherlands              Sweden            China PR
##                 478                 453                 397                 392
##               Chile Republic of Ireland              Mexico       United States
##                 391                 368                 366                 353
##              Poland              Norway        Saudi Arabia             Denmark
##                 350                 341                 340                 336
##      Korea Republic            Portugal              Turkey             Austria
##                 335                 322                 303                 298
##            Scotland             Belgium           Australia         Switzerland
##                 286                 260                 236                 220
##             Uruguay             Senegal               Wales             Croatia
##                 149                 130                 129                 126
##              Serbia             Nigeria               Ghana              Greece
##                 126                 121                 114                 102
##      Czech Republic         Ivory Coast            Cameroon             Morocco
##                 100                 100                  90                  85
##            Paraguay    Northern Ireland              Russia             Ukraine
##                  85                  80                  79                  73
##        South Africa             Finland           Venezuela              Canada
##                  71                  67                  67                  64
##  Bosnia Herzegovina             Algeria            Slovenia             Romania
##                  61                  60                  55                  54
##            Slovakia            DR Congo             Iceland         New Zealand
##                  54                  52                  47                  44
##             Ecuador                Mali             Albania             Hungary
##                  43                  43                  40                  38
##                Peru              Kosovo            Bulgaria             Jamaica
##                  37                  33                  32                  32
##             Tunisia               Egypt              Guinea             Bolivia
##                  32                  31                  31                  30
##          Costa Rica               India             Georgia               Congo
##                  30                  30                  26                  25
##          Montenegro       FYR Macedonia          Cape Verde                Iran
##                  23                  20                  19                  17
##        Burkina Faso            Honduras              Angola               Benin
##                  16                  16                  15                  15
##               Gabon              Gambia       Guinea Bissau              Panama
##                  15                  15                  15                  15
##             Curacao              Israel             Estonia            Zimbabwe
##                  14                  14                  13                  13
##          Madagascar                Togo             Armenia               Haiti
##                  12                  12                  10                  10
##               Kenya               Syria              Zambia              Cyprus
##                  10                   9                   9                   8
##           Lithuania          Luxembourg                Iraq             (Other)
##                   8                   8                   7                 176
```

4.2.2 Club

```
# Valores que puede tomar Club
head(unique(fifa_DF$Club), 15)
```

```
##  [1] "FC Barcelona"        "Juventus"            "Paris Saint-Germain"
##  [4] "Manchester United"   "Manchester City"     "Chelsea"
##  [7] "Real Madrid"         "Atlético Madrid"     "FC Bayern München"
## [10] "Tottenham Hotspur"   "Liverpool"           "Napoli"
## [13] "Arsenal"             "Milan"               "Inter"
```

```
# Hacer que los strings vacíos sean nulos
fifa_DF$Club[fifa_DF$Club==''] <- NA
# Convertir a factor
fifa_DF$Club <- factor(fifa_DF$Club)

summary(fifa_DF$Club)
```

```
##                 Arsenal                AS Monaco          Atlético Madrid
##                      33                       33                       33
##       Borussia Dortmund                  Burnley             Cardiff City
##                      33                       33                       33
##              CD Leganés                 Chelsea      Eintracht Frankfurt
##                      33                       33                       33
##                  Empoli                 Everton             FC Barcelona
##                      33                       33                       33
##      Fortuna Düsseldorf                Frosinone                Liverpool
##                      33                       33                       33
##         Manchester City        Manchester United         Newcastle United
##                      33                       33                       33
##          Rayo Vallecano                 RC Celta              Real Madrid
##                      33                       33                       33
##             Southampton        Tottenham Hotspur      TSG 1899 Hoffenheim
##                      33                       33                       33
##             Valencia CF  Wolverhampton Wanderers          1. FSV Mainz 05
##                      33                       33                       32
##             Bournemouth   Brighton & Hove Albion           Crystal Palace
##                      32                       32                       32
##               FC Nantes                  Fulham               Hertha BSC
##                      32                       32                       32
##       Huddersfield Town                   Lazio           Leicester City
##                      32                       32                       32
##             Levante UD         SV Werder Bremen            VfL Wolfsburg
##                      32                       32                       32
##           Villarreal CF          West Ham United  Athletic Club de Bilbao
##                      32                       32                       31
##             FC Augsburg FC Girondins de Bordeaux            Real Sociedad
##                      31                       31                       31
##   Toulouse Football Club                     Ajax                  Al Ahli
##                      31                       30                       30
##                Al Batin               Al Faisaly                Al Hazem
##                      30                       30                       30
##                Al Hilal               Al Ittihad             Al Qadisiyah
##                      30                       30                       30
##                 Al Raed                Al Shabab                Al Wehda
##                      30                       30                       30
##              Alanyaspor              Albacete BP               Angers SCO
##                      30                       30                       30
##             Antalyaspor        AS Nancy Lorraine                   Ascoli
##                      30                       30                       30
##             Aston Villa                AZ Alkmaar          Birmingham City
##                      30                       30                       30
##        Blackburn Rovers         Bolton Wanderers Borussia Mönchengladbach
##                      30                       30                       30
##                Brentford                  Brescia             Bristol City
##                      30                       30                       30
##                Bursaspor                 Cádiz CF                    Carpi
##                      30                       30                       30
##          Çaykur Rizespor                 CD Lugo              CD Tenerife
##                      30                       30                       30
##             Cerezo Osaka             Club América                Club León
```

```
##                       30                       30                       30
##              Club Necaxa                  Cosenza            De Graafschap
##                       30                       30                       30
##   Deportivo de La Coruña             Derby County             ESTAC Troyes
##                       30                       30                       30
##               Ettifaq FC          Extremadura UD                FC Emmen
##                       30                       30                       30
##         FC Ingolstadt 04              FC Lorient             FC St. Pauli
##                       30                       30                       30
##                FC Tokyo            Fenerbahçe SK           Galatasaray SK
##                       30                       30                       30
##              Gamba Osaka   Gimnàstic de Tarragona                  (Other)
##                       30                       30                    14913
##                     NA's
##                      241
```

4.2.3 Preferred.Foot

```
# Valores que puede tomar Preferred.Foot
unique(fifa_DF$Preferred.Foot)
```

```
## [1] "Left"  "Right" ""
```

```
# Hacer que los strings vacíos sean nulos
fifa_DF$Preferred.Foot[fifa_DF$Preferred.Foot==''] <- NA
# Convertir a factor
fifa_DF$Preferred.Foot <- factor(fifa_DF$Preferred.Foot)

summary(fifa_DF$Preferred.Foot)
```

```
##  Left Right  NA's
##  4211 13948    48
```

4.2.4 Work.Rate

```
# Valores que puede tomar Work.Rate
unique(fifa_DF$Work.Rate)
```

```
##  [1] "Medium/ Medium" "High/ Low"      "High/ Medium"   "High/ High"
##  [5] "Medium/ High"   "Medium/ Low"    "Low/ High"      "Low/ Medium"
##  [9] "Low/ Low"       ""
```

```
# Hacer que los strings vacíos sean nulos
fifa_DF$Work.Rate[fifa_DF$Work.Rate==''] <- NA
# Convertir a factor
fifa_DF$Work.Rate <- factor(fifa_DF$Work.Rate)

summary(fifa_DF$Work.Rate)
```

```
##     High/ High      High/ Low   High/ Medium      Low/ High       Low/ Low
##           1015            699           3173            439             34
##    Low/ Medium   Medium/ High    Medium/ Low Medium/ Medium           NA's
##            449           1690            850            9810             48
```

4.2.5 Position

```
## Position
unique(fifa_DF$Position)
```

```
##  [1] "RF"  "ST"  "LW"  "GK"  "RCM" "LF"  "RS"  "RCB" "LCM" "CB"  "LDM" "CAM"
## [13] "CDM" "LS"  "LCB" "RM"  "LAM" "LM"  "LB"  "RDM" "RW"  "CM"  "RB"  "RAM"
## [25] "CF"  "RWB" "LWB" ""
```

```
# Hacer que los strings vacíos sean nulos
fifa_DF$Position[fifa_DF$Position==''] <- NA
# Convertir a factor
fifa_DF$Position <- factor(fifa_DF$Position)

summary(fifa_DF$Position)
```

```
##  CAM   CB  CDM   CF   CM   GK  LAM   LB  LCB  LCM  LDM   LF   LM   LS   LW  LWB
##  958 1778  948   74 1394 2025   21 1322  648  395  243   15 1095  207  381   78
##  RAM   RB  RCB  RCM  RDM   RF   RM   RS   RW  RWB   ST NA's
##   21 1291  662  391  248   16 1124  203  370   87 2152   60
```

4.3 Valores nulos

A medida que he realizado la normalización de los campos se puede ver que contienen valores nulos o vacíos.

```
# Nulos por columna
colSums(is.na(fifa_DF))
```

```
##             ID            Age    Nationality        Overall           Club
##              0              0              0              0            241
##          Value           Wage Preferred.Foot      Weak.Foot    Skill.Moves
##              0              0             48             48             48
##      Work.Rate       Position         Height         Weight
##             48             60             48             48
```

```
# Almaceno el índice de los registros que tienen nulos en Height
idx <- which(is.na(fifa_DF$Height))
```

Salvo los campos de **Nationality** y **Club**, los 48 jugadores que tienen, los tienen en los mismos atributos físicos.

```
summary(fifa_DF[is.na(fifa_DF),])
```

```
##        ID           Age           Nationality     Overall
##  Min.   : NA   Min.   : NA   Afghanistan:  0   Min.   : NA
##  1st Qu.: NA   1st Qu.: NA   Albania    :  0   1st Qu.: NA
##  Median : NA   Median : NA   Algeria    :  0   Median : NA
##  Mean   :NaN   Mean   :NaN   Andorra    :  0   Mean   :NaN
##  3rd Qu.: NA   3rd Qu.: NA   Angola     :  0   3rd Qu.: NA
##  Max.   : NA   Max.   : NA   (Other)    :  0   Max.   : NA
##  NA's   :589   NA's   :589   NA's       :589   NA's   :589
##                     Club         Value           Wage     Preferred.Foot
##   SSV Jahn Regensburg :  0   Min.   : NA   Min.   : NA   Left :  0
##  1. FC Heidenheim 1846:  0   1st Qu.: NA   1st Qu.: NA   Right:  0
##  1. FC Kaiserslautern :  0   Median : NA   Median : NA   NA's :589
##  1. FC Köln           :  0   Mean   :NaN   Mean   :NaN
##  1. FC Magdeburg      :  0   3rd Qu.: NA   3rd Qu.: NA
##  (Other)              :  0   Max.   : NA   Max.   : NA
##  NA's                 :589   NA's   :589   NA's   :589
##    Weak.Foot    Skill.Moves         Work.Rate      Position       Height
##  Min.   : NA   Min.   : NA   High/ High  :  0   CAM    :  0   Min.   : NA
##  1st Qu.: NA   1st Qu.: NA   High/ Low   :  0   CB     :  0   1st Qu.: NA
##  Median : NA   Median : NA   High/ Medium:  0   CDM    :  0   Median : NA
##  Mean   :NaN   Mean   :NaN   Low/ High   :  0   CF     :  0   Mean   :NaN
##  3rd Qu.: NA   3rd Qu.: NA   Low/ Low    :  0   CM     :  0   3rd Qu.: NA
##  Max.   : NA   Max.   : NA   (Other)     :  0   (Other):  0   Max.   : NA
##  NA's   :589   NA's   :589   NA's        :589   NA's   :589   NA's   :589
##      Weight
##  Min.   : NA
##  1st Qu.: NA
##  Median : NA
##  Mean   :NaN
##  3rd Qu.: NA
##  Max.   : NA
##  NA's   :589
```

Estos 48 jugadores tienen la particularidad de tener 62 puntos de **Overall**, sospecho que obedece a algún patrón pero no se explicar por qué se debe. He consultado *sofifa* y he visto que en revisiones posteriores de la base de datos se han actualizado estos campos.

No opto por rescatar esos datos a posteriori e integrarlos manualmente porque los atributos de los jugadores de este dataset también pueden haber sufrido cambios en las revisiones futuras. No sería justo comparar jugadores en instantes de tiempo diferentes si son jugadores coetáneos ya que el entorno o el contexto en diferentes espacios temporales puede ser muy diferente.

Dicho esto, trato por imputar los valores nulos con un modelo **Random Forest** que me permitirá hacer una imputación multivariante contemplando todo el dataset.

Debido a la alta cardinalidad de los campos *Nationality* y *Club* (más de 53 categorías) debo excluirlos del modelo.

```
# Imputación con random forest
set.seed(123)
fifa_imputed <- rfImpute(Overall ~ ., ntree=250, iter=3, data=fifa_DF[,!names(fifa_DF) %in% c('I
         D', 'Nationality', 'Club')])
# Resultados de la imputación
summary(fifa_imputed[idx,])
```

Tenemos una varianza explicada muy pequeña, no se están imputando los valores correctamente, así que prefiero eliminar del conjunto estos jugadores 48 jugadores.

También lo intenté con el paquete Hmisc que ofrece un método bastante robusto pero arroja R^2 en las predicciones muy pequeñas, por lo que tampoco predice correctamente.

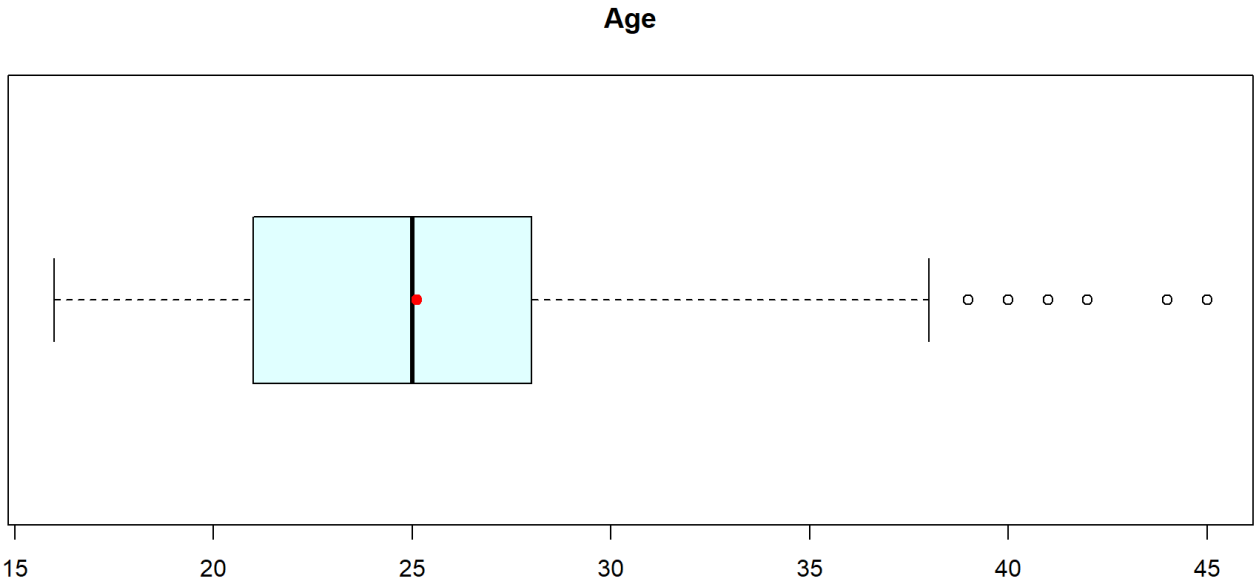
```
fifa_imputed_2 <- aregImpute(~Age + Overall + Wage + Preferred.Foot +
                               Weak.Foot + Skill.Moves + Work.Rate + Position +
                               Height + Weight + Value,
                             data = fifa_DF[,!names(fifa_DF) %in% c('ID', 'Nationality', 'Club'
         )],
                             n.impute = 5)
```

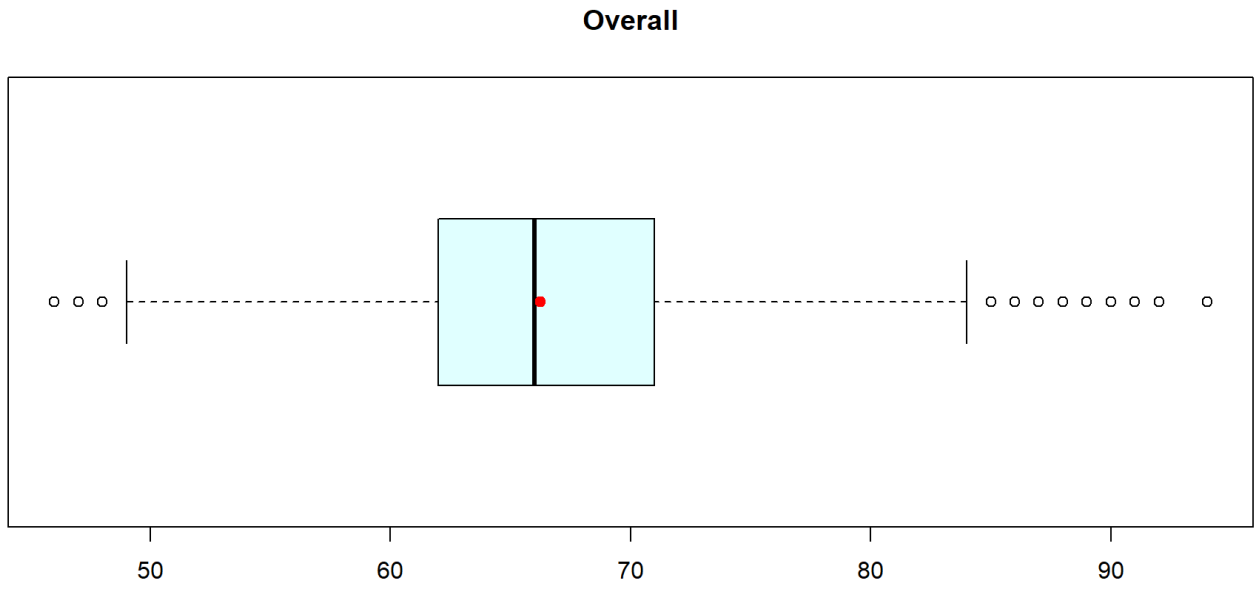
Nos quedamos con los registros que tengan sus atributos completos, es decir, eliminamos del dataset los nulos.

```
fifa_clean <- fifa_DF[complete.cases(fifa_DF[, c('Overall', 'Value', 'Wage',
                                                 'Preferred.Foot', 'Weak.Foot',
                                                 'Skill.Moves', 'Work.Rate',
                                                 'Position', 'Height', 'Weight')
                                            ]),]
```

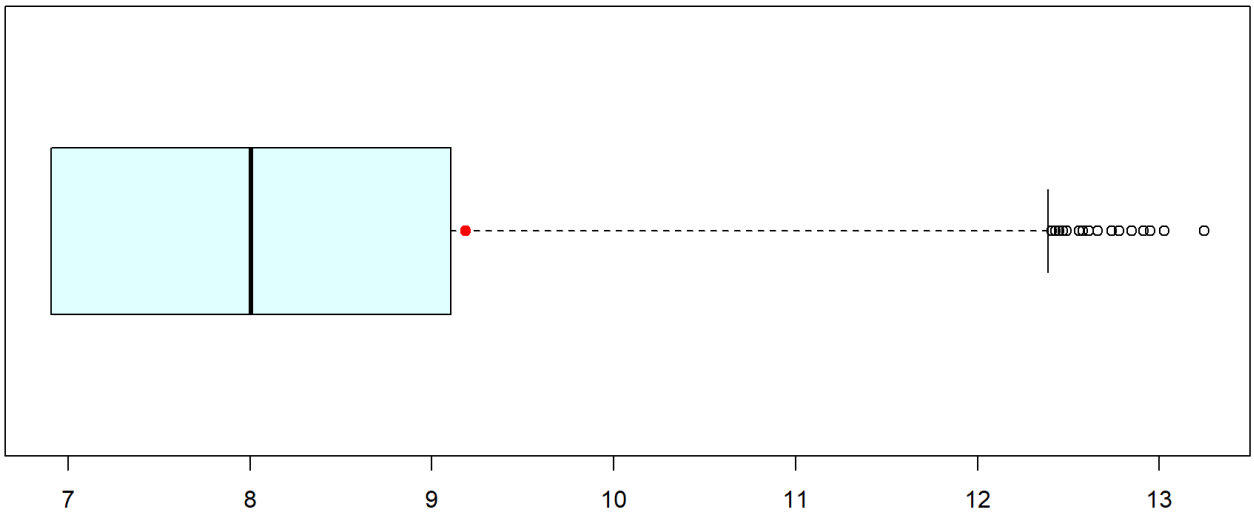
4.4 Outliers

Para los valores anómalos o atípicos en las variables cuantitativas usaré boxplots y con un indicador de la media para tener una primera visión de la dispersión y centralidad.



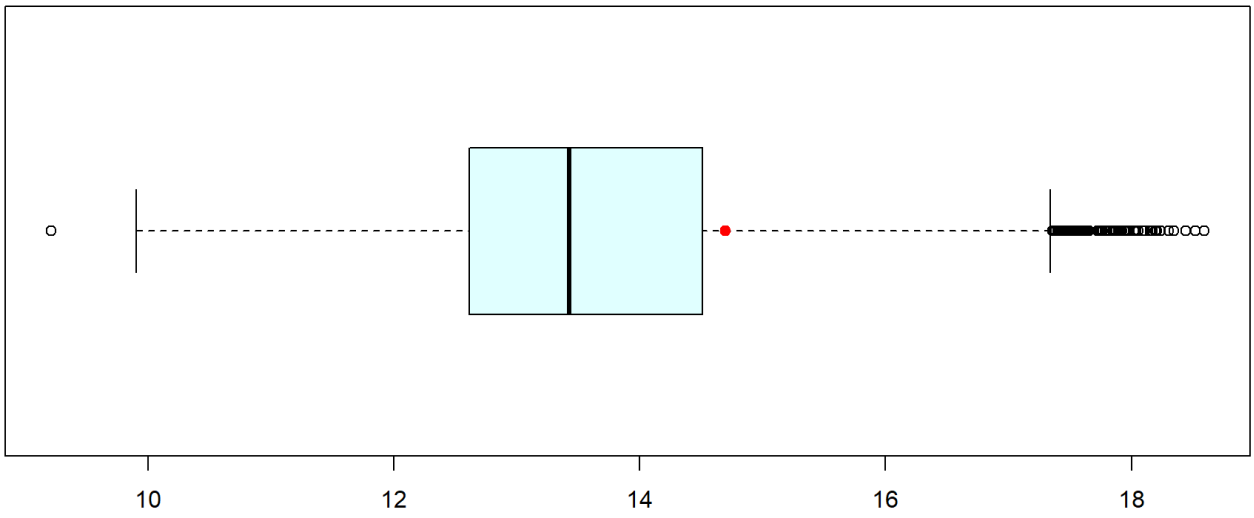



```
## Warning in bplt(at[i], wid = width[i], stats = z$stats[, i], out = z$out[z$group
## == : Outlier (-Inf) in boxplot 1 is not drawn
```

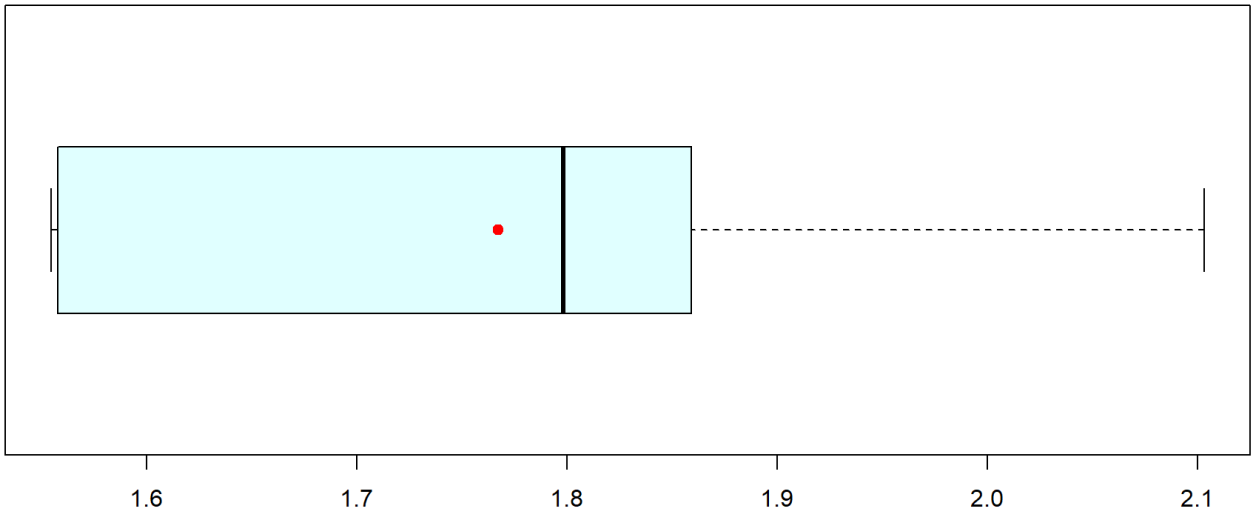



```
## Warning in bplt(at[i], wid = width[i], stats = z$stats[, i], out = z$out[z$group
## == : Outlier (-Inf) in boxplot 1 is not drawn
```

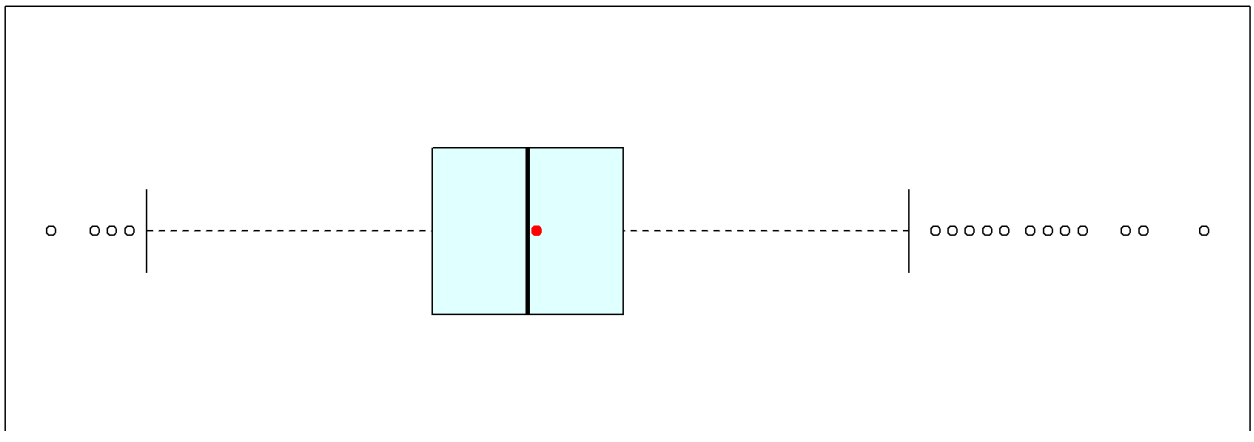
Value - Escala log

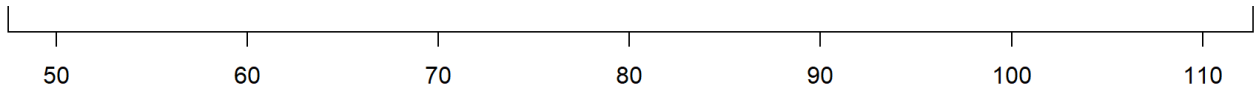


Height



Weight





No aprecio valores anómalos más allá que los salarios que cobran los jugadores más cotizados de fútbol o su valor en el mercado. Dado que se sabe que es real que se paguen esos salarios y que coticen a esos valores, no los trataré como outliers y los mantendré en el conjunto.

5 Análisis de los datos

5.1 Selección de grupos de datos

Se pretende hacer una comparación de zurdos y diestros, por lo tanto, vamos a descartar a los porteros y crear un conjunto de diestros y otro de zurdos.

```
# jugadores de campo
jCampo <- fifa_clean[fifa_clean$Position!='GK',]
# Zurdos
zurdos <- jCampo[jCampo$Preferred.Foot=='Left',]
# Diestro
diestros <- jCampo[jCampo$Preferred.Foot=='Right',]
```

5.2 Comprobación de normalidad y heterocedasticidad

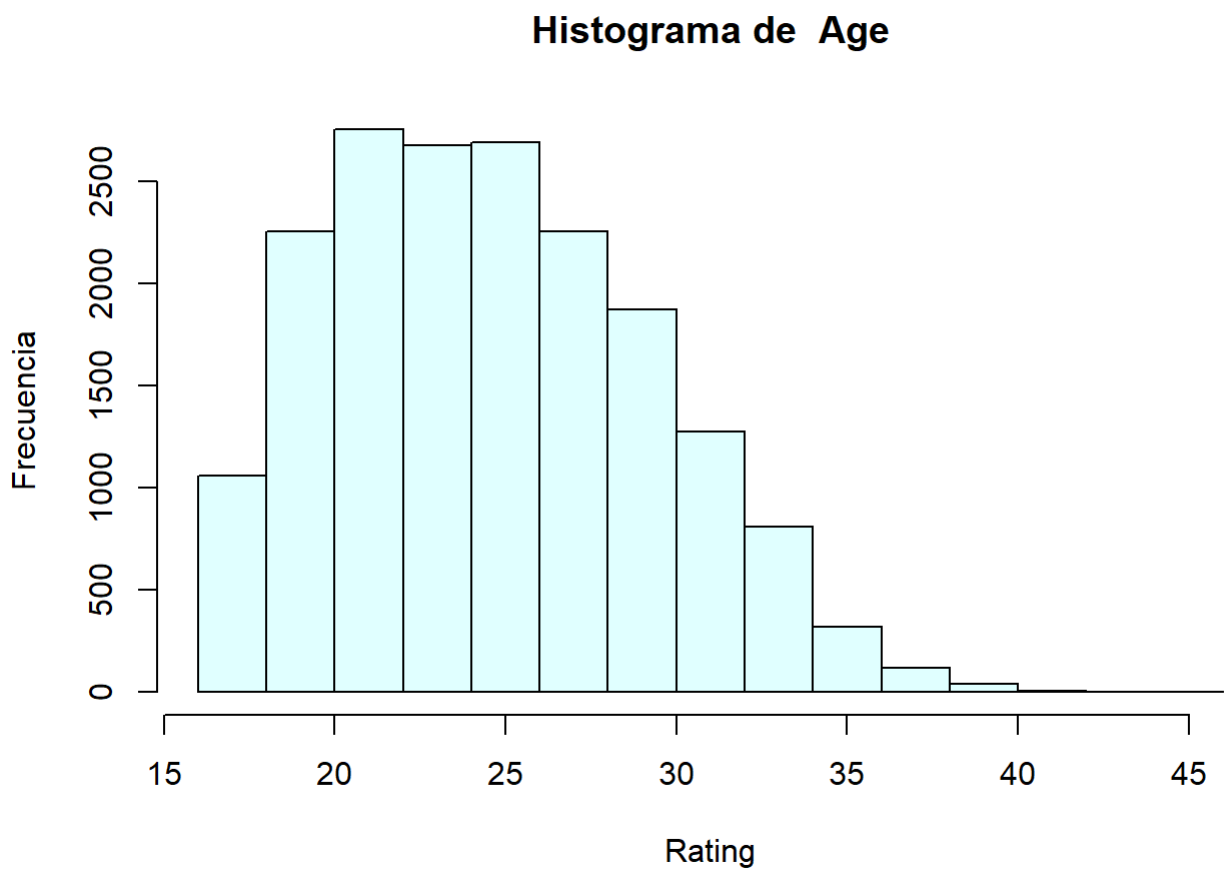
5.2.1 Normalidad

La normalidad en las variables cuantitativas la analizamos con el test Lilliefors, variante del de Kolmogorov-Smirnov. Fijamos un nivel de significación de 0.05. Aquellos resultados de p-value inferiores al nivel de significación nos indican que la distribución no es normal.

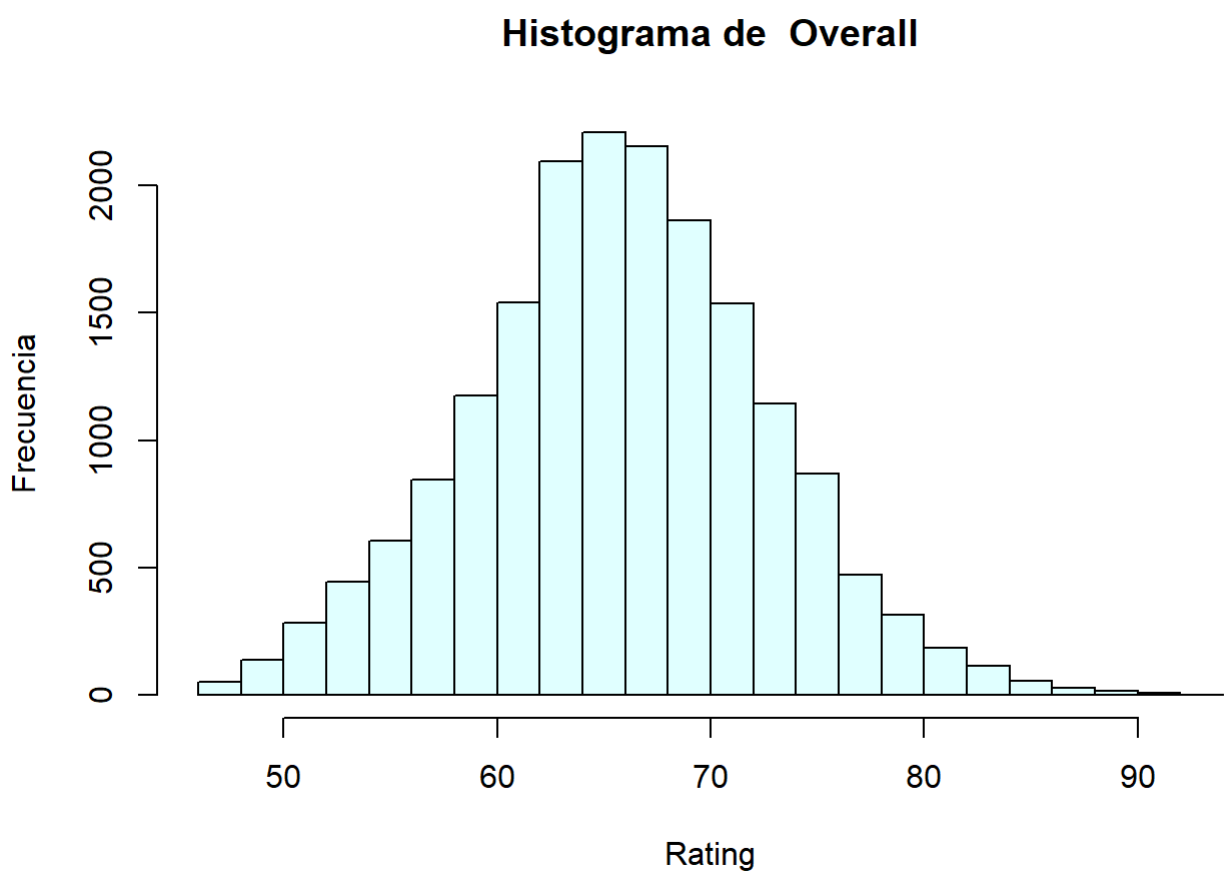
```
for (i in 2:ncol(fifa_clean)) {
  if(is.numeric(fifa_clean[,i])){
    if(lillie.test(fifa_clean[,i])$p.value<0.05){
      print(paste(colnames(fifa_clean)[i], ' No normal. p-value=',
                  lillie.test(fifa_clean[,i])$p.value, sep=' '))
    }else{
      print(paste(colnames(fifa_clean)[i], ' Distribución normal. p-value=',
                  lillie.test(fifa_clean[,i])$p.value, sep=' '))
    }

    hist(fifa_clean[,i], col="lightcyan", main=paste("Histograma de ",
                                                     colnames(fifa_clean)[i]),
         xlab="Rating", ylab="Frecuencia", breaks=20)
  }
}
```

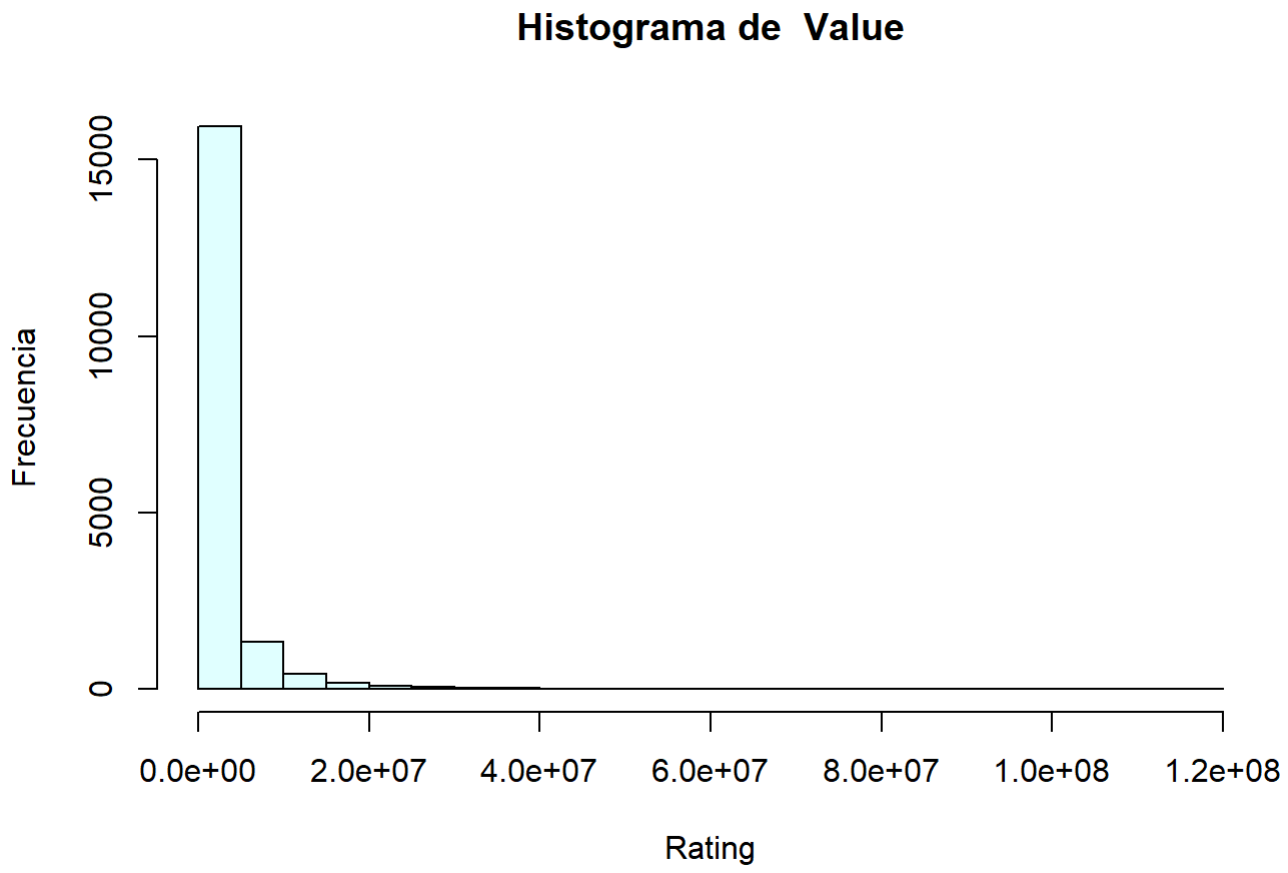
```
## [1] "Age  No normal. p-value= 0"
```



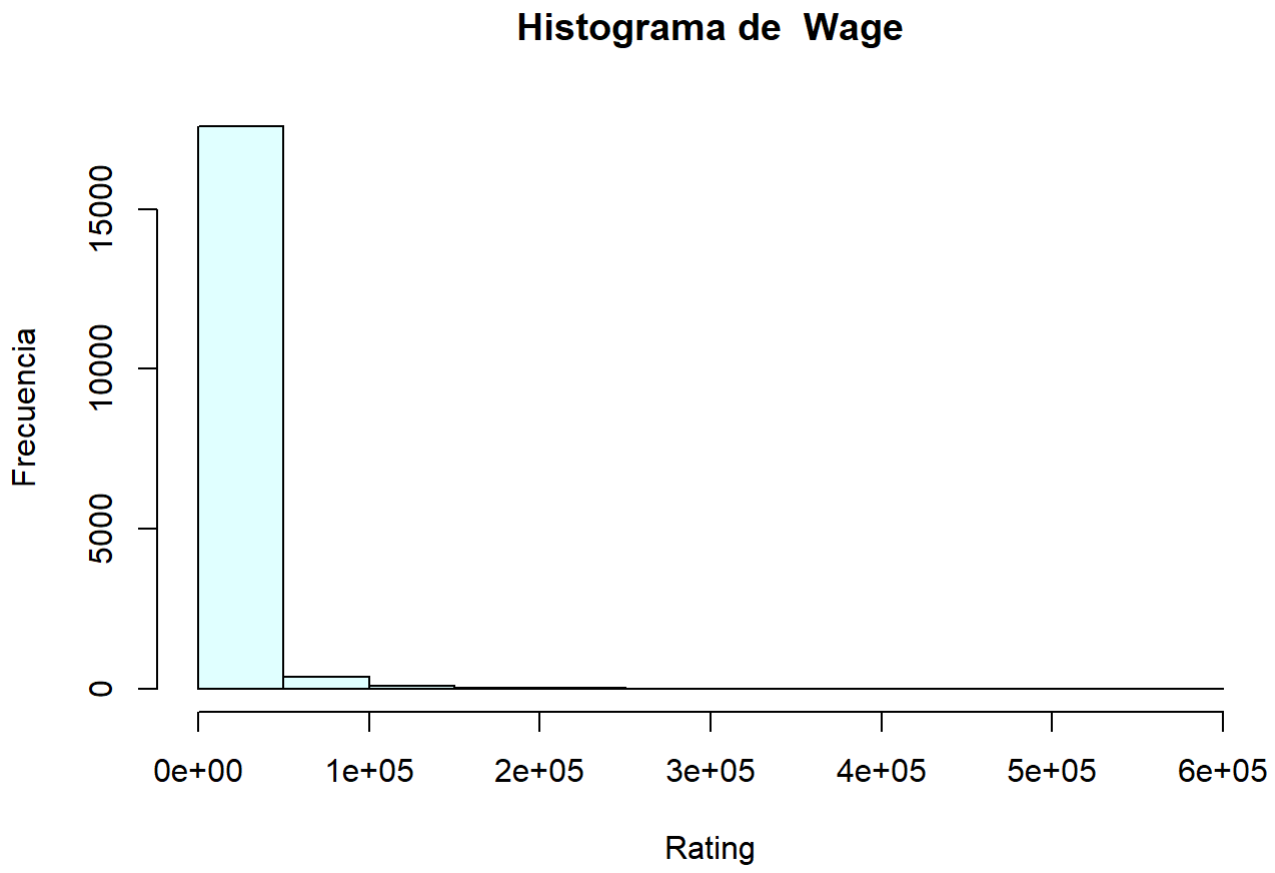

```
## [1] "Overall  No normal. p-value= 1.50975501028489e-70"
```



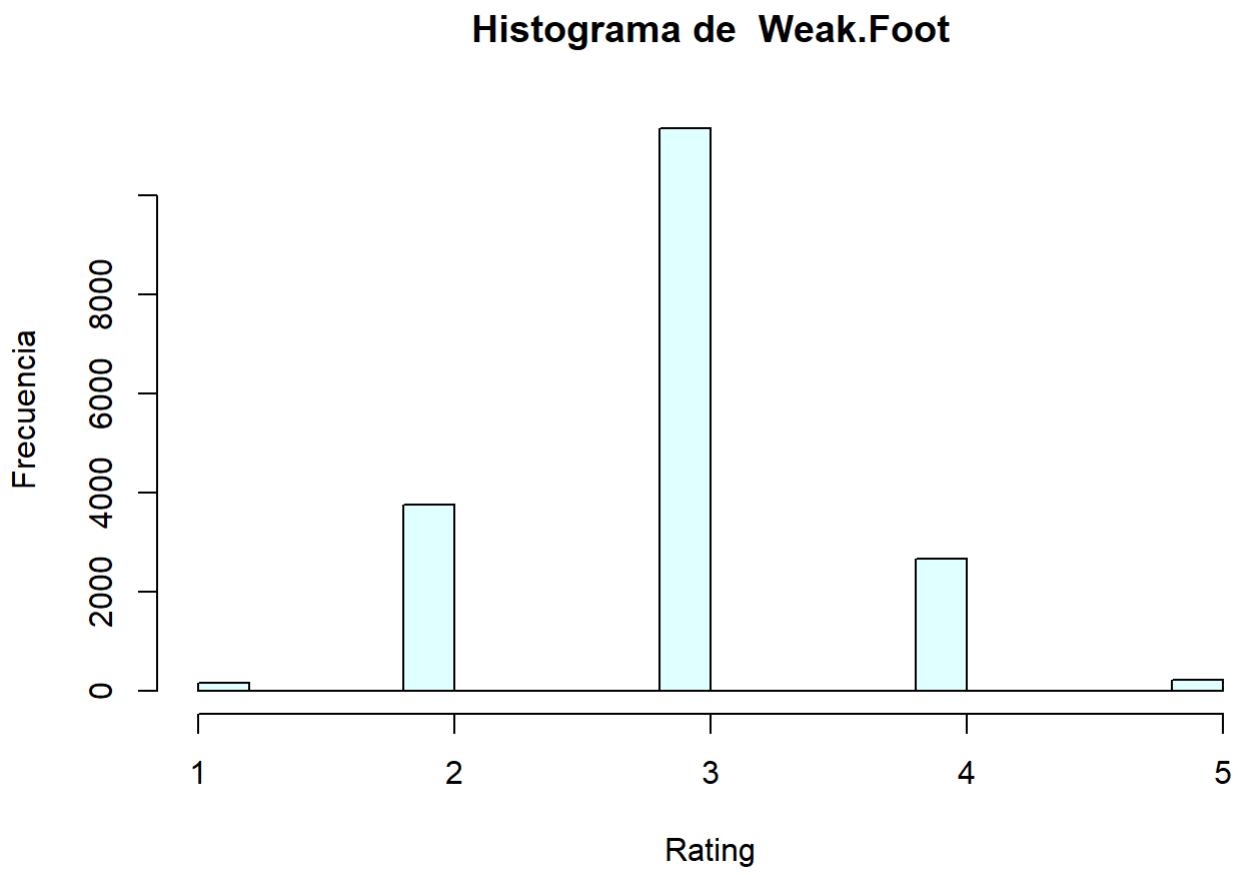

```
## [1] "Value  No normal. p-value= 0"
```




```
## [1] "Wage  No normal. p-value= 0"
```

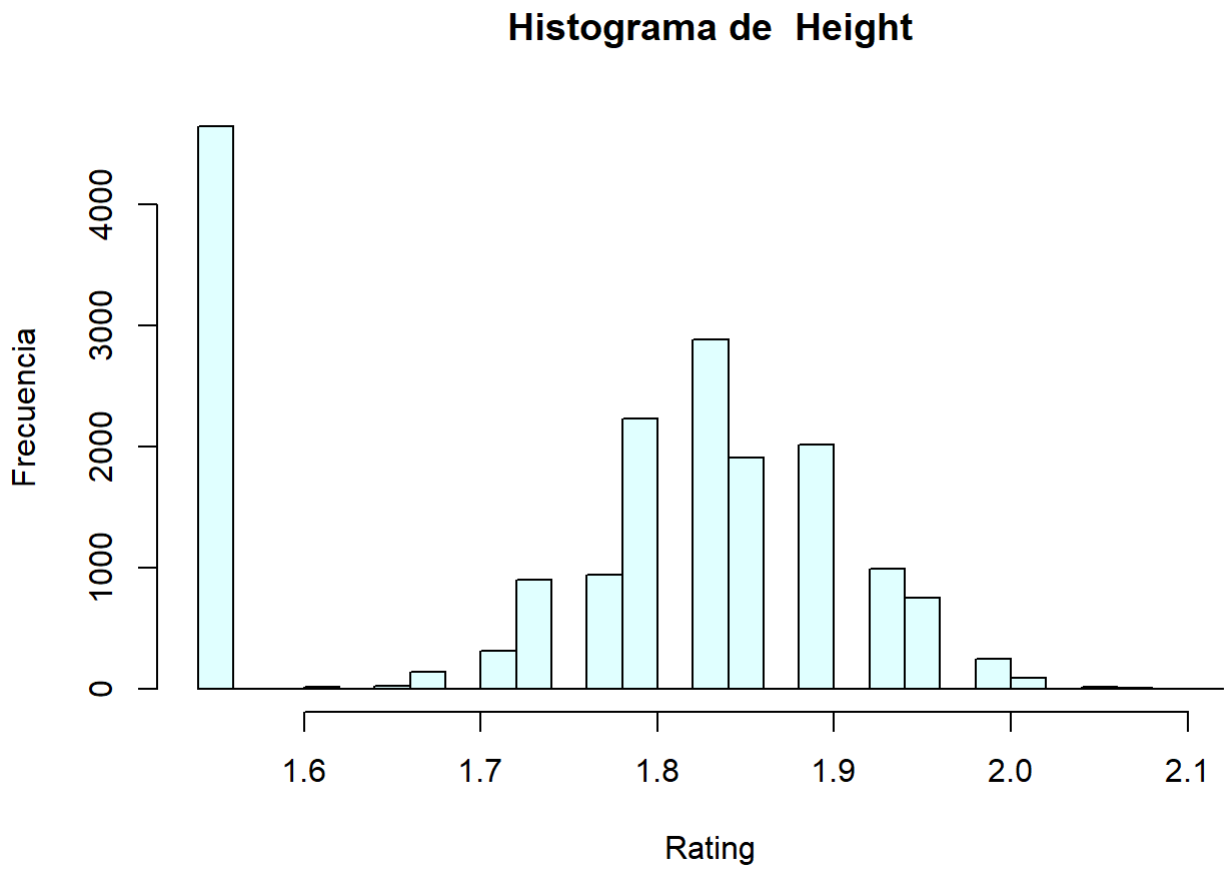



```
## [1] "Weak.Foot  No normal. p-value= 0"
```

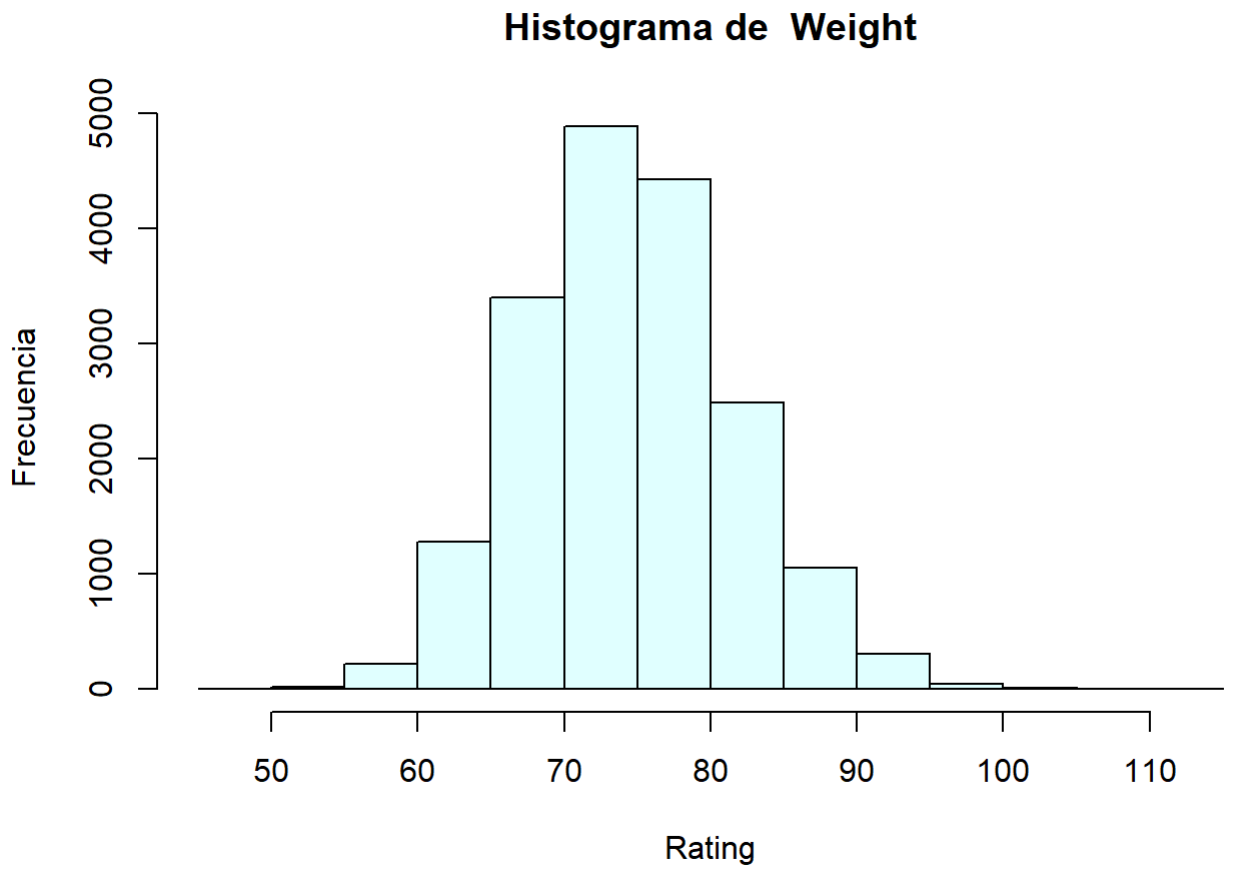



```
## [1] "Skill.Moves  No normal. p-value= 0"
```


```
## [1] "Height  No normal. p-value= 0"
```




```
## [1] "Weight  No normal. p-value= 1.88741258247453e-211"
```



Los resultados muestran que no disponemos de distribuciones normales en las variables cuantitativas seleccionadas.

5.2.2 Homogeneidad de la varianza

Con la prueba Fisher evaluamos las varianzas de las variables entre zurdos y diestros. Con un nivel de significancia de 0.001, cualquier p-valor inferior a él nos indica que debemos rechazar la hipótesis nula de igualdad de varianzas.

```
for (i in 2:ncol(fifa_clean)) {
  if(is.numeric(fifa_clean[,i])){
    if(var.test(zurdos[,i], diestros[,i])$p.value<0.001){
      print(paste(colnames(fifa_clean)[i],
                  'Varianzas disintas p-value=',
                  var.test(zurdos[,i], diestros[,i])$p.value,
                  sep=' '))
    }else{
      print(paste(colnames(fifa_clean)[i],
                  'Homogeneas. p-value=',
                  var.test(zurdos[,i], diestros[,i])$p.value,
                  sep=' '))
    }
  }
}
```

```
## [1] "Age Homogeneas. p-value= 0.0433675669102513"
## [1] "Overall Varianzas disintas p-value= 2.58640148136924e-06"
## [1] "Value Varianzas disintas p-value= 0.00016404439683404"
## [1] "Wage Varianzas disintas p-value= 1.34927233297333e-06"
## [1] "Weak.Foot Varianzas disintas p-value= 0.000367266252010223"
## [1] "Skill.Moves Varianzas disintas p-value= 0.000722468483229433"
## [1] "Height Homogeneas. p-value= 0.0362484919629019"
## [1] "Weight Homogeneas. p-value= 0.646903157419739"
```

Parece que los atributos físicos de edad, altura y peso tienen varianzas semejantes.

5.3 Pruebas estadísticas

5.3.1 Contraste de hipótesis

Esta prueba estadística consistirá en un contraste de hipótesis entre dos muestras, jugadores de campo zurdos y jugadores de campo diestros. De los datos anteriores sabemos que las varianzas son distintas entre zurdos y diestros para Overall y que no tienen distribuciones normales. Sin embargo, asumimos normalidad por el teorema del límite central ya que la muestras tienen un tamaño muy superior a 30.

Se plantea entonces el contraste de hipótesis unilateral para un T test de la siguiente manera:

- ¿Son los jugadores zurdos mejores que los diestros?
  - $H_0 : \mu_1 = \mu_2$
  - $H_1 : \mu_1 > \mu_2$

Siendo μ_1 la media muestral de los zurdos y μ_2 de los diestros. $\alpha = 0.05$.

```
# Asumimos normalidad por el teorema del límite central
t.test(zurdos$Overall, diestros$Overall,
       var.equal = FALSE,
       conf.level=0.95,
       alternative="greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  zurdos$Overall and diestros$Overall
## t = 4.0657, df = 7216.3, p-value = 2.42e-05
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.2905942       Inf
## sample estimates:
## mean of x mean of y
##  66.82813  66.34004
```

Con un resultado p-valor<0.05 debemos rechazar la hipótesis nula y, por tanto, concluir con un nivel de confianza del 95% que los jugadores zurdos tienen de media mayor Overall

5.3.2 Test de correlación

La segunda prueba es un análisis de correlación de las variables cuantitativas respecto a **Overall**. Usaré la correlación de *Spearman* ya que los datos no tienen una distribución normal.

```
# ¿ Qué atributos están más correlacionados con el Overall ?
columnas <- c('Age', 'Weak.Foot', 'Skill.Moves', 'Height', 'Weight')
corr_coef <- columnas
p_val <- columnas
for (i in 1:length(columnas)) {
  result = cor.test(jCampo[,columnas[i]],
                    jCampo[,'Overall'],
                    method = "spearman",
                    exact=FALSE)

  corr_coef[i] <- result$estimate
  p_val[i] <- result$p.value

}

out1 <- data.frame( var=columnas,
                   Correlation=corr_coef,
                   pvalue=p_val)
out1 %>% kable() %>% kable_styling()
```

var	Correlation	pvalue
Age	0.47980627838361	0
Weak.Foot	0.20202184527818	4.29859378268888e-148
Skill.Moves	0.498906366608527	0
Height	0.0669552674766174	1.73386779740641e-17
Weight	0.176509592748874	5.72830754415006e-113

Con un rango de -1 a 1 vemos la estimación de la correlación de las variables de la tabala respecto a **Overall**. También se muestra el p-valor para conocer el peso que tiene en la correlación. Destacan **Age** y **Skill.Moves** entre ellas.

5.3.3 Modelo de regresión lineal

Por último, pretendo predecir el valor de los jugadores con modelos de regresión lineal. Participarán como variables predictoras tanto las cuantitativas como las cualitativas.

Entreno tres modelos con diferente combinación de variables y construyo una tabla resumen con la bondad del ajuste, R^2, para comparar sus desempeños.

```
# Predecir el valor de los jugadores
lm1 <- lm(Value ~ Age + Overall + Club + Position, data = fifa_clean)
lm2 <- lm(Value ~ Age + Preferred.Foot + Weak.Foot +
            Skill.Moves + Height + Weight,
          data = fifa_clean)
lm3 <- lm(Value ~ Overall + Position + Work.Rate, data = fifa_clean)

coeficientes <- matrix(c(1, summary(lm1)$r.squared,
                                2, summary(lm2)$r.squared,
                                3, summary(lm3)$r.squared),
                              ncol = 2, byrow = TRUE)

out2 <- data.frame( var=coeficientes[,1],
                    R_squared=coeficientes[,2]
                    )
out2 %>% kable() %>% kable_styling()
```

var	R_squared
1	0.6260129
2	0.1351541
3	0.4081297

El modelo lm1 es mejor de los tres. No posee una bondad especialmente buena, pero nos da una idea aproximada de la valoración de un jugador de futbol.

Usando ese modelo probaremos a predecir el valor en el mercado de 3 jugadores diferentes.

```
nuevosJugadores <- data.frame(
  Age = c(22,28,18),
  Overall = c(75,85,68),
  Club = c('Borussia Dortmund', 'Borussia Dortmund', 'Chelsea'),
  Position = c('CB', 'GK', 'RF')
)

predict(lm1, nuevosJugadores)
```

```
##        1        2        3
## 12551519 16922445 21240722
```

6 Conclusión

El dataset es muy completo, contiene gran cantidad de información, necesita unos pequeños trabajos de limpieza y formateo del contenido para poder manejarlo correctamente.

Los outliers en los atributos de los jugadores podrían haber imputado con medidas de centralidad.

La comparación entre zurdos y diestros convendría hacerla entre pares para hacer una comparación más fidedigna.

Los resultados de los modelos no fueron tan satisfactorios como me hubiese gustado, hay demasiada incertidumbre. sin embargo, nos da una visión aproximada a la posible valoración de los jugadores. Con lo cual creo los resultados fueron válidos.