Actividad 4 - Análisis de varianza y repaso del curso

Francisco Javier Melchor González

10/1/2021

Contents

Paquetes

Los paquetes que se van a utilizar para el desarrollo de esta actividad, son los siguientes:

```
if(!require(dplyr)){
  install.packages("dplyr")
  library(dplyr)
}
if(!require(DataCombine)){
  install.packages("DataCombine")
  library(DataCombine)
}

if(!require(Rmisc)){
  install.packages("Rmisc")
  library(Rmisc)
}
if(!require(MLmetrics)){
  install.packages("MLmetrics")
  library(MLmetrics)
}

if(!require(agricolae)){
  install.packages("agricolae")
  library(agricolae)
}

if(!require(ggplot2)){
  install.packages("ggplot2")
  library(ggplot2)
}
```

1 Lectura del fichero y preparación de los datos

Enunciado:

Leed el fichero fifa.csv y guardad los datos en un objeto con identificador denominado fifa. A continuación, verificad el tipo de cada variable. ¿Qué variables son de tipo numérico? ¿Qué variables son de tipo categórico?

Solución: En primer lugar, realizamos la lectura del fichero **Fifa.csv**, aplicando para ello la función *read.csv*.

Los parámetros que recibe esta función son:

- **file:** ruta del archivo que se quiere leer, en este caso se indica a través de la variable **fifa_filename**

- **header:** atributo booleano que indica si el fichero a leer contiene o no cabecera, en este caso si, por lo que su valor es **TRUE**.
- **sep:** atributo que indica el separador de campos que utiliza el archivo, en este caso es la **coma (“,”)**.
- **na.strings:** atributo que indica que cadenas representan valores faltantes, en este caso, las cadenas vacías y con un espacio.
- **stringAsFactors:** atributo booleano que permite codificar todas las variables de tipo cadena como factores en vez de como cadenas si se le da el valor **TRUE** como en este caso. Esto se realiza debido a que la mayoría de variables de tipo cadena de este dataset, realmente son factores.
- **encoding:** atributo que indica la codificación del archivo, en este caso **UTF-8**.

```
fifa_filename<-"../Data/Fifa.csv"
fifa <- read.csv(file=fifa_filename, header=TRUE, sep=",",
         na.strings=c("","  "), stringsAsFactors=TRUE, encoding = 'UTF-8')
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 Suárez     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
```

A continuación, se procede a mostrar los tipos de cada una de las variables que forman el dataframe.

```
str(fifa)
```

```
## 'data.frame':    17588 obs. of  53 variables:
##  $ Name              : Factor w/ 17341 levels "A.J. DeLaGarza",..: 3270 9925 12459 10269 10555 3900 1
##  $ Nationality       : Factor w/ 160 levels "Afghanistan",..: 122 6 20 155 59 139 121 158 143 14 ...
##  $ National_Position : Factor w/ 27 levels "CAM","CB","CDM",..: 13 24 14 13 5 5 13 23 NA 5 ...
##  $ National_Kit      : num  7 10 10 9 1 1 9 11 NA 1 ...
##  $ Club              : Factor w/ 634 levels "1. FC Heidenheim",..: 460 204 204 204 206 361 206 460 3
##  $ Club_Position     : Factor w/ 29 levels "CAM","CB","CDM",..: 15 26 15 28 6 6 28 26 28 6 ...
##  $ Club_Kit          : num  7 10 11 9 1 1 9 11 9 13 ...
##  $ Club_Joining      : Factor w/ 1677 levels "01/01/1993","01/01/1998",..: 847 842 851 926 849 849 85
##  $ Contract_Expiry   : num  2021 2018 2021 2021 2021 ...
##  $ Rating            : int  94 93 92 92 92 90 90 90 90 89 ...
##  $ Height            : Factor w/ 50 levels "155 cm","157 cm",..: 30 15 19 27 38 38 30 28 40 44 ...
##  $ Weight            : Factor w/ 56 levels "100 kg","101 kg",..: 37 29 25 42 49 39 36 31 52 48 ...
##  $ 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 5211
##  $ 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 1
##  $ 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 ...
```

Como se puede observar, hay algunas variables que tienen un formato que no le corresponde. En el caso de la variable **Name**, se cambiará a un tipo de variable de cadena de caracteres, ya que no se trata de una variable de tipo cualitativa o factor. Y en el caso de las variables **National_Kit, Club_Kit y Contract_Expiry**, se cambiarán a variables enteras, ya que no contienen números con decimales distintos de 0.

```
fifa$Name<-as.character(fifa$Name)
fifa$National_Kit<-as.integer(fifa$National_Kit)
fifa$Club_Kit<-as.integer(fifa$Club_Kit)
fifa$Contract_Expiry<-as.integer(fifa$Contract_Expiry)
str(fifa)
```

```
## 'data.frame':    17588 obs. of  53 variables:
##  $ Name              : chr  "Cristiano Ronaldo" "Lionel Messi" "Neymar" "Luis Suárez" ...
##  $ Nationality       : Factor w/ 160 levels "Afghanistan",..: 122 6 20 155 59 139 121 158 143 14 ...
##  $ National_Position : Factor w/ 27 levels "CAM","CB","CDM",..: 13 24 14 13 5 5 13 23 NA 5 ...
##  $ National_Kit      : int  7 10 10 9 1 1 9 11 NA 1 ...
##  $ Club              : Factor w/ 634 levels "1. FC Heidenheim",..: 460 204 204 204 206 361 206 460 3
##  $ Club_Position     : Factor w/ 29 levels "CAM","CB","CDM",..: 15 26 15 28 6 6 28 26 28 6 ...
##  $ Club_Kit          : int  7 10 11 9 1 1 9 11 9 13 ...
##  $ Club_Joining      : Factor w/ 1677 levels "01/01/1993","01/01/1998",..: 847 842 851 926 849 849 85
##  $ Contract_Expiry   : int  2021 2018 2021 2021 2021 2019 2021 2022 2017 2019 ...
##  $ Rating            : int  94 93 92 92 92 90 90 90 90 89 ...
##  $ Height            : Factor w/ 50 levels "155 cm","157 cm",..: 30 15 19 27 38 38 30 28 40 44 ...
##  $ Weight            : Factor w/ 56 levels "100 kg","101 kg",..: 37 29 25 42 49 39 36 31 52 48 ...
##  $ 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 5211
##  $ 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 ...
```

1.1 Preparación de los datos

Enunciado:

Las variables ***Weight*** *y* ***Height*** *están clasificadas como factor. Para poder trabajar con ellas hay que convertirlas en numéricas.*

- *Convertir el peso de los jugadores en un valor numérico, eliminando el texto "kg" de los datos.*
- *Convertir la altura de los jugadores en un valor numérico, eliminando el texto "cm" de los datos.*

Solución:

A continuación, se procede a formatear las variables indicadas:

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

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

```
## 56 Levels: 100 kg 101 kg 102 kg 107 kg 110 kg 48 kg 49 kg 50 kg 52 kg ... 99 kg
```

```
fifa$Weight <- gsub("kg","",fifa$Weight)
fifa$Weight<-as.numeric(fifa$Weight)
head(fifa$Weight)
```

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

```
head(fifa$Height)
```

```
## [1] 185 cm 170 cm 174 cm 182 cm 193 cm 193 cm
## 50 Levels: 155 cm 157 cm 158 cm 159 cm 160 cm 161 cm 162 cm 163 cm ... 207 cm
```

```
fifa$Height <- gsub("cm","",fifa$Height)
fifa$Height<-as.numeric(fifa$Height)
head(fifa$Height)
```

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

1.2 Clasificación de jugadores

Enunciado: *La variable Rating indica la calidad del jugador de la siguiente forma: Excelente de 90 a 99, Muy bueno de 80 a 89, Bueno de 70 a 79, Regular de 50 a 69, Malo de 40 a 49, Muy malo de 0 a 39. Cread una variable categórica denominada clasificacion, que clasifique al jugador en una de estas categorías.*

Solución:

Para obtener la variable **clasificacion**, se procede a crear una función que devuelva los valores de las diferentes categorías de la misma en función del valor de la variable **Rating** que recibirá como parámetro de entrada:

```
get_clasificacion <- function(x){
  if (x >= 90 & x <= 99)
    return("Excelente")
  else if (x >= 80 & x<=89)
    return ("Muy bueno")
  else if (x >= 70 & x<=79)
    return("Bueno")
  else if (x >= 50 & x<=69)
    return("Regular")
  else if (x >= 40 & x <= 49)
    return("Malo")
  else if (x >= 0 & x <= 39)
    return ("Muy malo")
}
```

Una vez creada la función, se procede a realizar un lapply por cada una de las filas de la columna **Rating** del dataframe **fifa** e insertar el resultado en la nueva columna **clasificacion**:

```
fifa$clasificacion <- lapply(fifa$Rating, get_clasificacion)
fifa$clasificacion <- unlist(fifa$clasificacion)
fifa$clasificacion <- as.factor(fifa$clasificacion)

head(fifa$Rating)
```

```
## [1] 94 93 92 92 92 90
```

```
head(fifa$clasificacion)
```

```
## [1] Excelente Excelente Excelente Excelente Excelente Excelente
```

```
## Levels: Bueno Excelente Malo Muy bueno Regular
```

```
tail(fifa$Rating)
```

```
## [1] 45 45 45 45 45 45
```

```
tail(fifa$clasificacion)
```

```
## [1] Malo Malo Malo Malo Malo Malo
## Levels: Bueno Excelente Malo Muy bueno Regular
```

2 Estadística descriptiva y visualización

2.1 Análisis descriptivo

Enunciado:

Realizad un análisis descriptivo numérico de los datos (resumid los valores de las variables numéricas y categóricas). Mostrad el número de observaciones y el número de variables.

Contad cuántos clubs distintos y cuántas nacionalidades distintas hay representados en los datos.

Solución:

Se procede a continuación a obtener un análisis descriptivo de las diferentes columnas que forman al dataframe a analizar:

```
summary(fifa)
```

```
##      Name              Nationality    National_Position  National_Kit
##  Length:17588       England  : 1618   Sub    :  556     Min.   : 1.00
##  Class :character   Argentina: 1097   LCB    :   48     1st Qu.: 6.00
##  Mode  :character   Spain    : 1008   GK     :   47     Median :12.00
##                     France   :  974   RCB    :   46     Mean   :12.22
##                     Brazil   :  921   LB     :   39     3rd Qu.:18.00
##                     Italy    :  751   (Other):  339     Max.   :36.00
##                     (Other)  :11219   NA's   :16513     NA's   :16513
##                Club       Club_Position     Club_Kit         Club_Joining
##  Free Agents     :  232   Sub    :7492   Min.   : 1.00   07/01/2016: 1193
##  Angers SCO      :   33   Res    :3146   1st Qu.: 9.00   07/01/2015:  907
##  Arsenal         :   33   RCB    : 633   Median :18.00   07/01/2014:  558
##  AS Monaco       :   33   GK     : 632   Mean   :21.29   01/01/2016:  412
##  Bor. M'gladbach:   33   LCB    : 631   3rd Qu.:27.00   07/01/2013:  404
##  Bournemouth     :   33   (Other):5053   Max.   :99.00   (Other)   :14113
##  (Other)         :17191   NA's   :   1   NA's   :1       NA's      :    1
##  Contract_Expiry     Rating          Height          Weight
##  Min.   :2017    Min.   :45.00   Min.   :155.0   Min.   : 48.00
##  1st Qu.:2017    1st Qu.:62.00   1st Qu.:176.0   1st Qu.: 70.00
##  Median :2019    Median :66.00   Median :181.0   Median : 75.00
##  Mean   :2019    Mean   :66.17   Mean   :181.1   Mean   : 75.25
##  3rd Qu.:2020    3rd Qu.:71.00   3rd Qu.:186.0   3rd Qu.: 80.00
##  Max.   :2023    Max.   :94.00   Max.   :207.0   Max.   :110.00
##  NA's   :1
##  Preffered_Foot      Birth_Date         Age        Preffered_Position
##  Left : 4094    02/29/1988:  160   Min.   :17.00   CB     :2181
##  Right:13494    02/29/1984:  157   1st Qu.:22.00   GK     :2003
```

```
##                 02/29/1992:  155   Median :25.00   ST     :1825
##                 01/01/1996:   13   Mean   :25.46   CM     : 831
##                 11/11/1996:   13   3rd Qu.:29.00   LB     : 808
##                 01/08/1991:   12   Max.   :47.00   RB     : 689
##                 (Other)   :17078                   (Other):9251
##            Work_Rate      Weak_foot      Skill_Moves     Ball_Control
##  Medium / Medium:9897   Min.   :1.000   Min.   :1.000   Min.   : 5.00
##  High / Medium  :2918   1st Qu.:3.000   1st Qu.:2.000   1st Qu.:53.00
##  Medium / High  :1534   Median :3.000   Median :2.000   Median :63.00
##  Medium / Low   : 845   Mean   :2.934   Mean   :2.303   Mean   :57.97
##  High / High    : 747   3rd Qu.:3.000   3rd Qu.:3.000   3rd Qu.:69.00
##  High / Low     : 730   Max.   :5.000   Max.   :5.000   Max.   :95.00
##  (Other)        : 917
##    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     clasificacion
##  Min.   : 1.00   Min.   : 1.00   Min.   : 1.0   Bueno    : 5017
##  1st Qu.: 8.00   1st Qu.: 8.00   1st Qu.: 8.0   Excelente:    9
##  Median :11.00   Median :11.00   Median :11.0   Malo     :  121
##  Mean   :16.46   Mean   :16.56   Mean   :16.9   Muy bueno:  520
##  3rd Qu.:14.00   3rd Qu.:14.00   3rd Qu.:14.0   Regular  :11921
##  Max.   :95.00   Max.   :91.00   Max.   :90.0
##
```

Para obtener el número de clubs distintios y de nacionalidades, se procede a obtener las categorías de cada una de las variables que identifican dicha información y a contar las mismas:

```
length(levels(fifa$Club))
```

```
## [1] 634
```

```
length(levels(fifa$Nationality))
```

```
## [1] 160
```

2.2 Valores ausentes

Enunciado:

- *Eliminad los valores ausentes del conjunto de datos. Denominad al nuevo conjunto de datos fifaNet (Nota: En las variables 'National_Kit' y 'National_Position' se observan muchos casos sin valor. No eliminéis estas observaciones ya que no son verdaderos missings, sino que simplemente indican que el jugador no ha jugado nunca con el equipo nacional).*
- *Comprobad cuántas observaciones no tienen valores ausentes y sacad conclusiones sobre cómo de serio es el problema de valores ausentes en estos datos.*

Solución:

A continuación, lo primero que vamos realizar es obtener el número de valores faltantes o nulos para cada una de las variables del dataframe:

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

```
##               Name        Nationality  National_Position       National_Kit
##                  0                  0              16513              16513
##               Club      Club_Position           Club_Kit       Club_Joining
##                  0                  1                  1                  1
```

```
##    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
##                  0                  0
```

Como se puede observar, exceptuando las columnas **National_Position** y **National_Kit**, hay 1 fila con valores nulos solamente en algunas de las columnas.

Procedemos a continuación a sustituir los valores faltantes de las columnas **National_Position** y **National_Kit** por valores que nos permitan identificar que dicho jugador no ha jugado en el equipo nacional.

En el caso de la columna **National_Position**, al tratarse de una variable de tipo factor, se sustituirán los valores "NA" por "-", lo que nos permitirá identificar perfectamente que se trata de un jugador que no ha jugado en el equipo nacional.

En el caso de la columna **National_Kit**, al tratarse de una variable numérica que toma únicamente valores positivos, se le asignará el valor "-1" a aquellos jugadores que no hayan jugado en el equipo nacional.

```
fifa$National_Position<-as.character(fifa$National_Position)
fifa[is.na(fifa$National_Position),]$National_Position <- "-"
fifa$National_Position<-factor(fifa$National_Position)
head(fifa$National_Position)
```

```
## [1] LS RW LW LS GK GK
## 28 Levels: - CAM CB CDM CM GK LAM LB LCB LCM LDM LF LM LS LW LWB RAM RB ... Sub
```

```
fifa[is.na(fifa$National_Kit),]$National_Kit <- "-1"
fifa$National_Kit <- as.integer(fifa$National_Kit)
min(fifa$National_Kit)
```

```
## [1] -1
```

Una vez marcados estos valores, procedemos a eliminar todas las filas que contengan valores NA en el dataframe:

```
fifaNet = DropNA(fifa)
```

```
## No Var specified. Dropping all NAs from the data frame.
```

```
## 1 rows dropped from the data frame because of missing values.
```

```
colSums(is.na(fifaNet))
```

```
##               Name        Nationality  National_Position       National_Kit
##                  0                  0                  0                  0
##               Club      Club_Position           Club_Kit       Club_Joining
##                  0                  0                  0                  0
##    Contract_Expiry             Rating             Height             Weight
##                  0                  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
##                  0                  0
```

Como se puede observar, ya no existen valores faltantes en el dataframe.

Se podrían haber dejado los valores de las columnas **National_Position** y **National_Kit** como valores NA y hacer una subselección de columnas de las cuales eliminar valores NA no incluyéndolas, pero al haber un gran número de columnas, esto resultaría mucho más engorroso.

2.3 Visualización

Enunciado:

1. Cread una variable denominada 'portero' que indique si el jugador juega de portero en su club o juega en otra posición (categoría "GK" en 'Club_Position').

Solución:

Para obtener esta variable cualitativa, se procede a crear una función que devuelva los diferentes valores de la misma en función del valore de la variable **Club_Position** que recibe como parámetro de entrada:

```
get_portero <- function(x){
  if(x == 'GK')
    return ("Yes")
  else
    return ("No")
}
```

Una vez desarrollada la función, se procede a realizar un lapply por cada una de las filas de la columna **Club_Position** e imputar los resultados en la nueva variable **portero**:

```
fifaNet$portero <- lapply(fifaNet$Club_Position,get_portero)
fifaNet$portero <- unlist(fifaNet$portero)
fifaNet$portero <- as.factor(fifaNet$portero)
levels(fifaNet$portero)
```

```
## [1] "No"  "Yes"
```

Para comprobar que la variable ha sido creada correctamente, se procede a contar el número de porteros que hay a través de la variable **Club_Position** y a través de la variable **portero**:

```
length(fifaNet[fifaNet$Club_Position == 'GK',])
```

```
## [1] 55
```

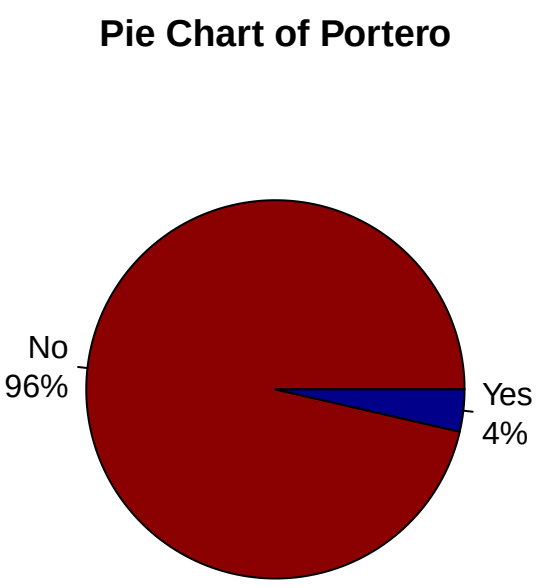
```
length(fifaNet[fifaNet$portero == 'Yes',])
```

```
## [1] 55
```

Como se puede observar, el número de filas obtenidas es el mismo en ambos casos, lo que indica que se ha realizado correctamente la imputación de la variable **portero**.

A continuación, se procede a representar la distribución de esta nueva variable a través de un Gráfico Circular o *Pie Chart*:

```
table_portero <- table(fifaNet$portero)
pct_portero <- round(table_portero/sum(table_portero)*100)
lbls_portero <- paste(names(table_portero), "\n", pct_portero, sep="")
lbls_portero <- paste(lbls_portero, '%', sep="")
pie(table_portero, labels = lbls_portero, main="Pie Chart of Portero\n", col=c("red4","darkblue"))
```



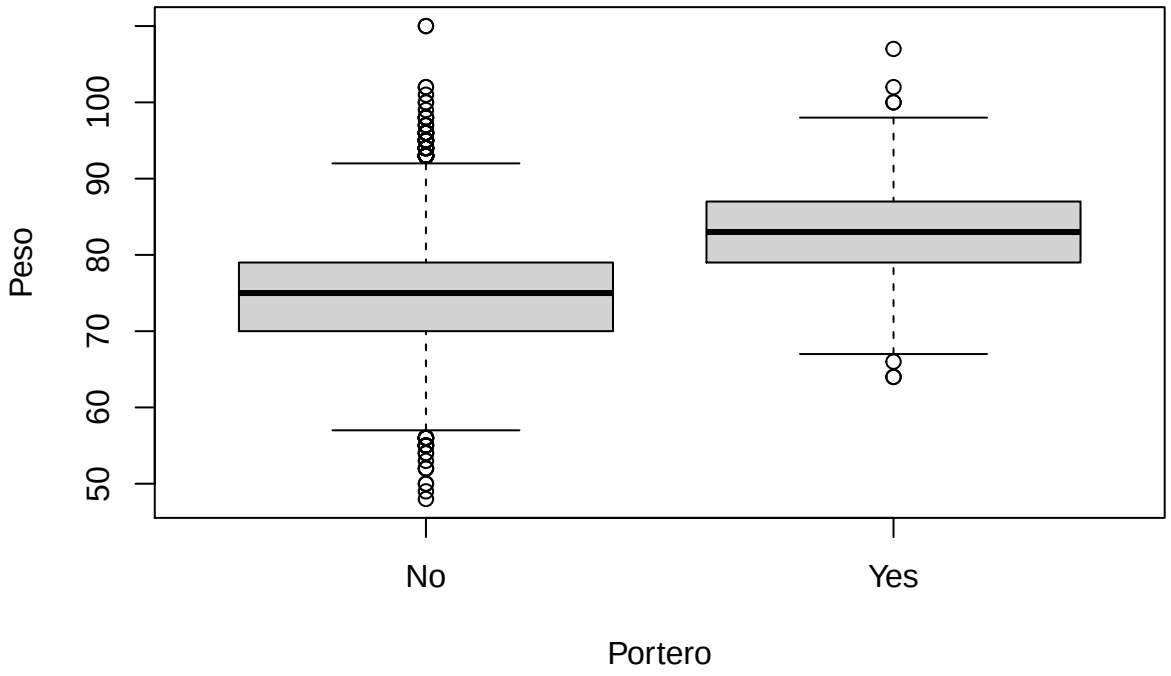
Enunciado:

2. Mostrad con diversos diagramas de caja la distribución de la variable 'Weight' según la variable 'portero', según 'Preffered_Foot', según 'clasificacion' y según 'Age'.

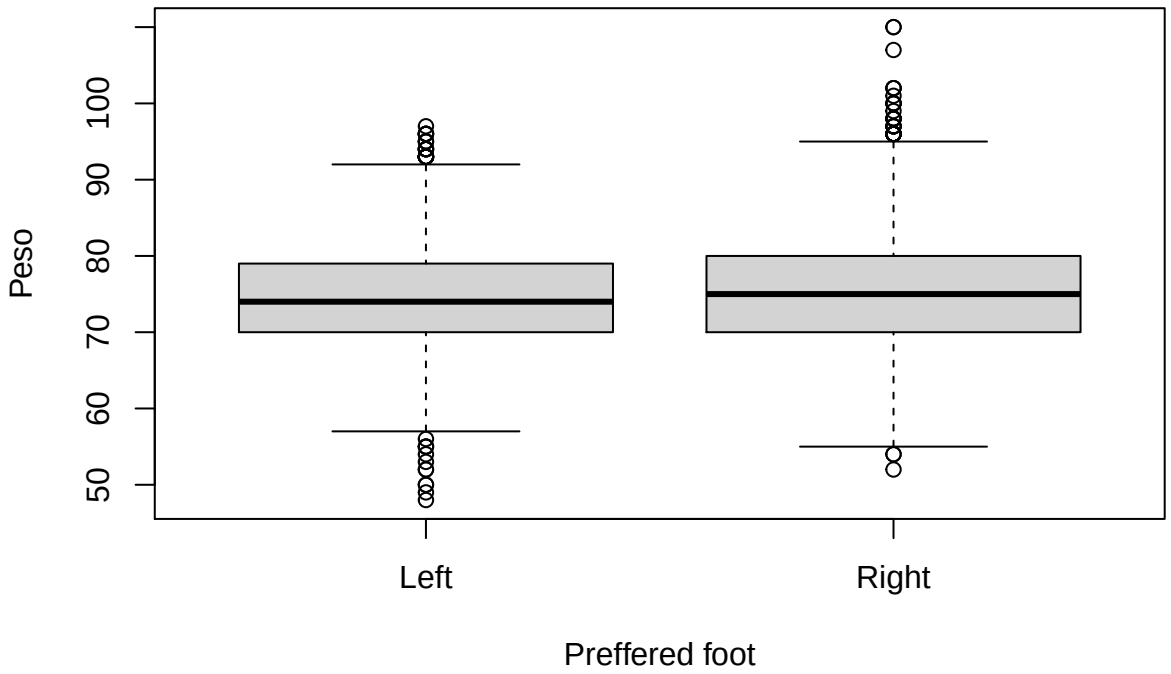
Solución:

A continuación se procede a representar mediante diagramas de cajas la variable **Weight** según las variables solicitadas en el enunciado:

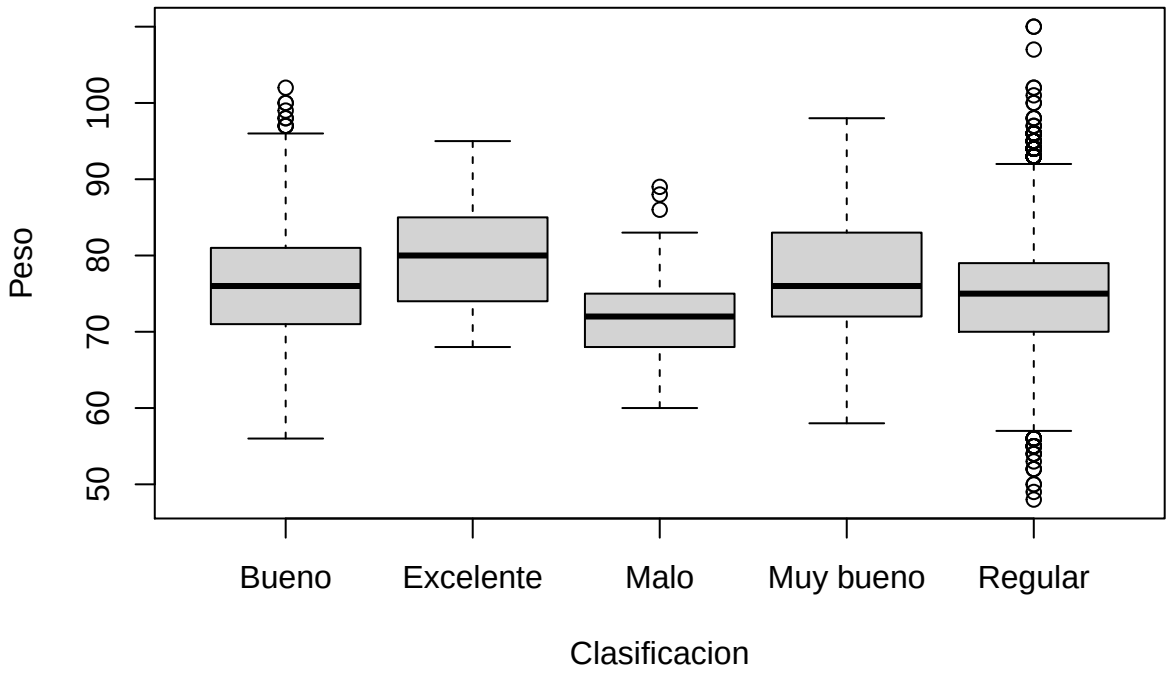
```
boxplot(Weight~portero,data=fifaNet,xlab="Portero",
        ylab="Peso")
```



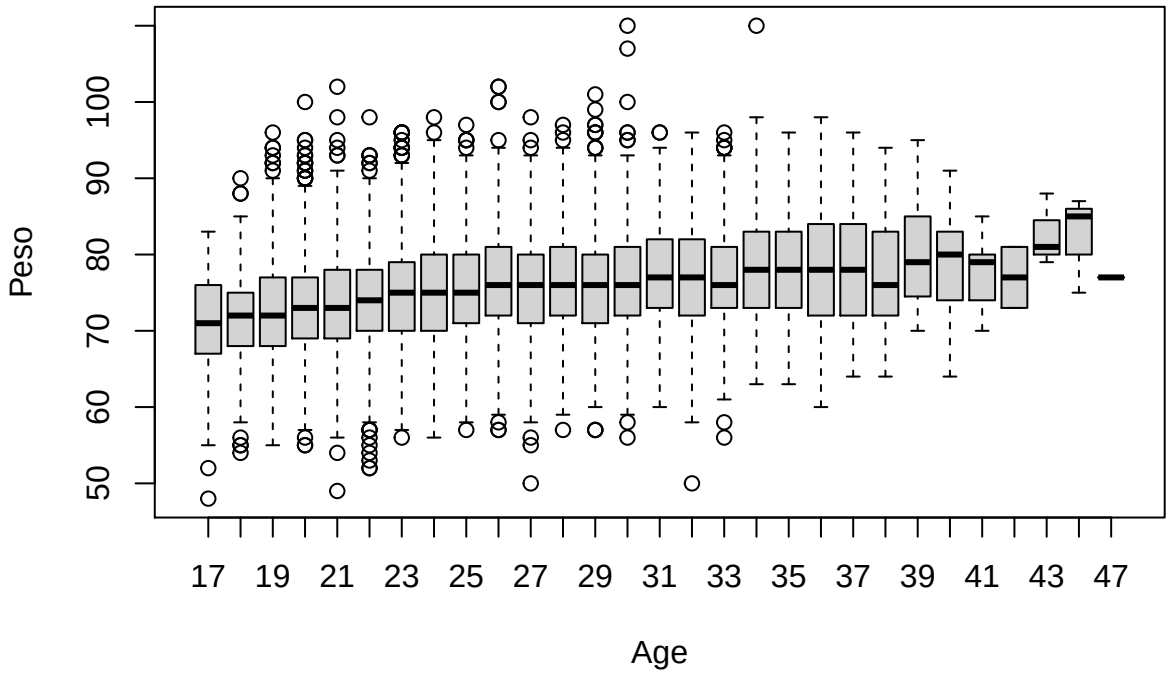

```
boxplot(Weight~Preffered_Foot,data=fifaNet,xlab="Preffered foot",
        ylab="Peso")
```




```
boxplot(Weight~clasificacion,data=fifaNet,xlab="Clasificacion",
        ylab="Peso")
```




```
boxplot(Weight~Age,data=fifaNet,xlab="Age",
        ylab="Peso")
```



Enunciado:

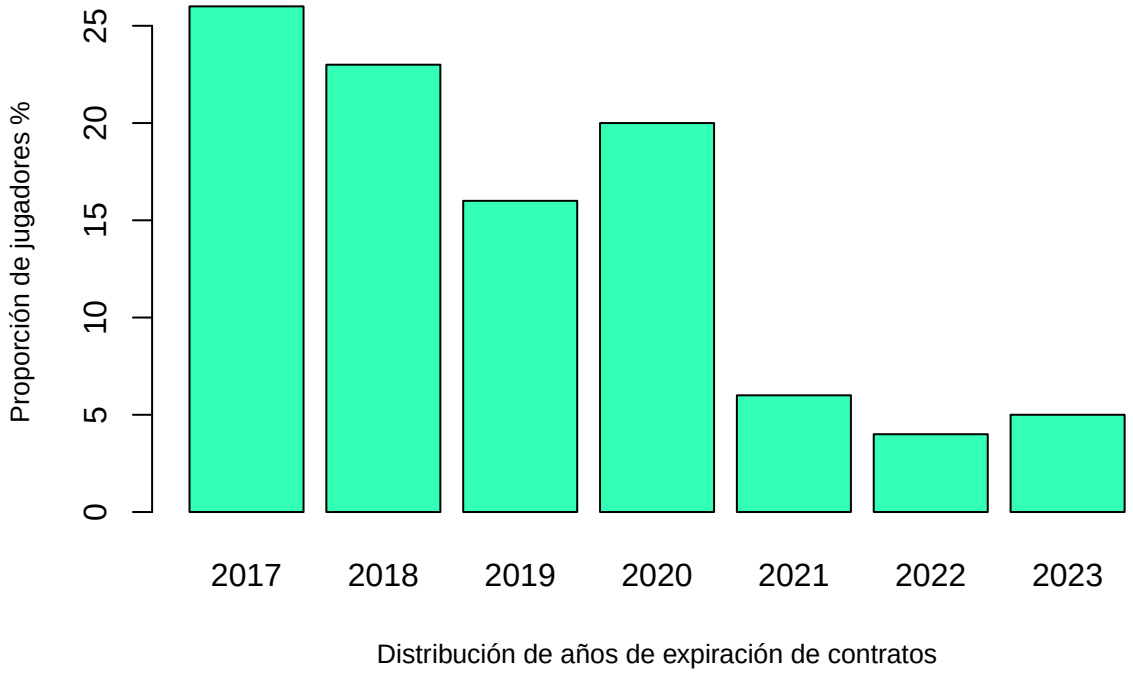
3. Dibujad un diagrama de barras que muestre el porcentaje de jugadores que finalizan el contrato en cada uno de los años.

Solución:

```
table_contract <- table(fifaNet$Contract_Expiry)
table_contract
```

```
##
## 2017 2018 2019 2020 2021 2022 2023
## 4529 4074 2844 3488  975  783  894
```

```r
pct_contract <- round(table_contract/sum(table_contract)*100)
barplot(pct_contract, ylab ="Proporción de jugadores %",
        xlab="Distribución de años de expiración de contratos",
        cex.main = 0.8, cex.lab = 0.8, col="#33FFB5")
```



Enunciado:

4. Interpretad los gráficos brevemente.

Solución:

1. En el primer gráfico podemos observar que el porcentaje de porteros frente al porcentaje de jugadores de campos es mucho menor, lo cual es razonable, ya que portero sólo hay uno en el campo y jugadores 11.

2.1 En el diagrama de cajas de la variable **Weight** con respecto a la variable **portero**, podemos observar que cambia la distribución del peso dependiendo si el jugador es un portero o no. Vemos que en el caso de los jugadores de campo, la mediana se encuentra en 75kg aproximadamente, y en el caso de los porteros, la mediana se encuentra en 85kg aproximadamente, es decir 10kg más de media. Por otro lado, existe una varianza menor en el caso de los porteros, pero esto es debido a que el número de casos es mucho menor que el de jugadores de campo.

2.2 En el diagrama de cajas de la variable **Weight** con respecto a la variable **Preffered foot**, podemos observar que la mediana se encuentra prácticamente al mismo nivel en ambos casos, 75kg. Pero sin embargo, la categoría **Right**, tiene algunos casos más de valores atípicos. Esto puede deberse a que la mayoría de los jugadores son diestros y no zurdos, por lo que hay un mayor número de casos en la categoría **Right** y esto hace que haya una mayor variedad en los diferentes jugadores.

2.3 En el diagrama de cajas de la variable **Weight** con respecto a la variable **Clasificacion**, vemos que la distribución del peso es diferente en las diferentes categorías de la variable clasificación. Siendo el caso de la mediana con **menor valor** en aquellos jugadores considerados como **"malos"** y el caso de la mediana con **mayor valor** aquellos jugadores considerados como **"excelentes"**. Esto puede deberse a la masa muscular

de los mismos, y no realmente a la materia grasa como tal, por lo que parece que a medida que la clasificación es mejor, mayor es la masa muscular de los jugadores.

2.4 En el diagrama de cajas de la variable **Weight** con respecto a la variable **Age**, podemos observar que en la mayoría de los casos, a una mayor edad mayor es el peso del jugadore, exceptuando las edades de 33 y 39 que tienen una mediana de peso más bajo con respecto a las medianas que se encuentran a sus extremos (derecho e izquierdo)

2.5 Por úlitmo, en el diagrama de barras que muestra la distribución de la frecuencia de la variable que indica el año en el que expira el contrato del jugador, podemos observar que la mayoría de los contratos terminan en el mismo año en el que se basa el dataframe, es decir, en el año 2017 y 2018. Esto es debido a que en el fútbol de manera general los contratos suelen renovarse por temporada.

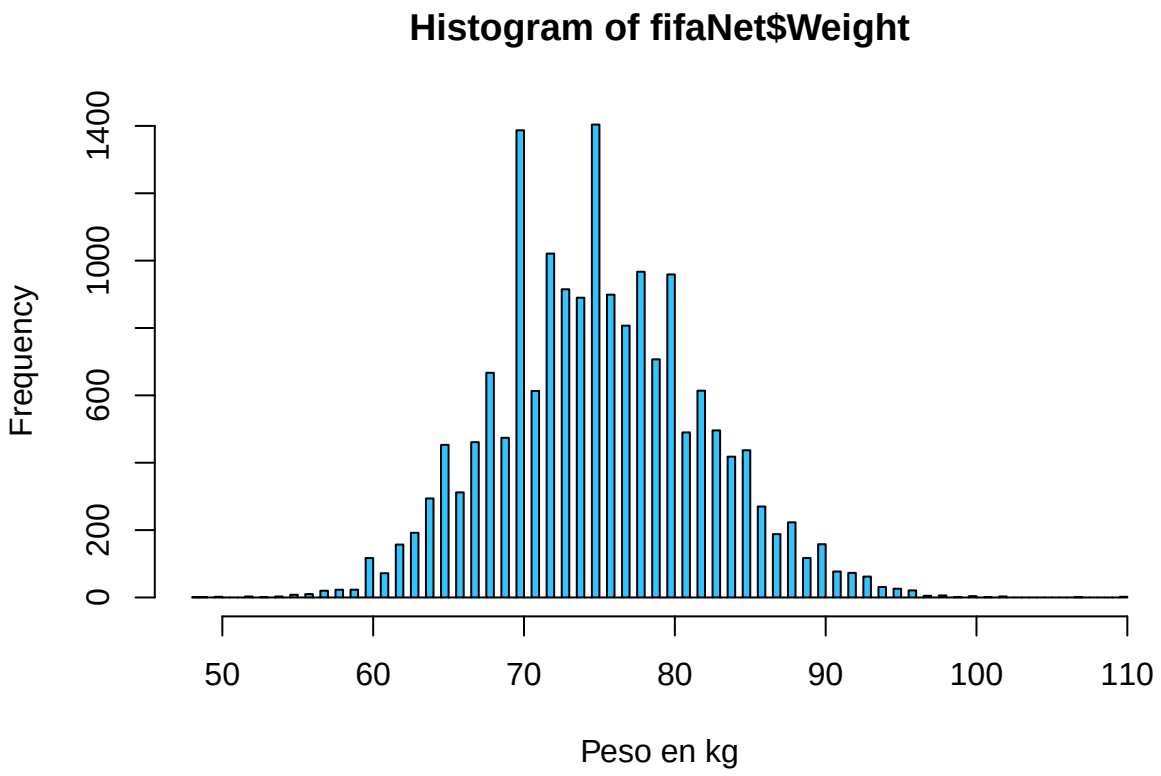
2.4 Comprobación de normalidad

Enunciado:

¿Podemos asumir que la variable Weight tiene una distribución normal? Debéis justificar la respuesta a partir de métodos visuales.

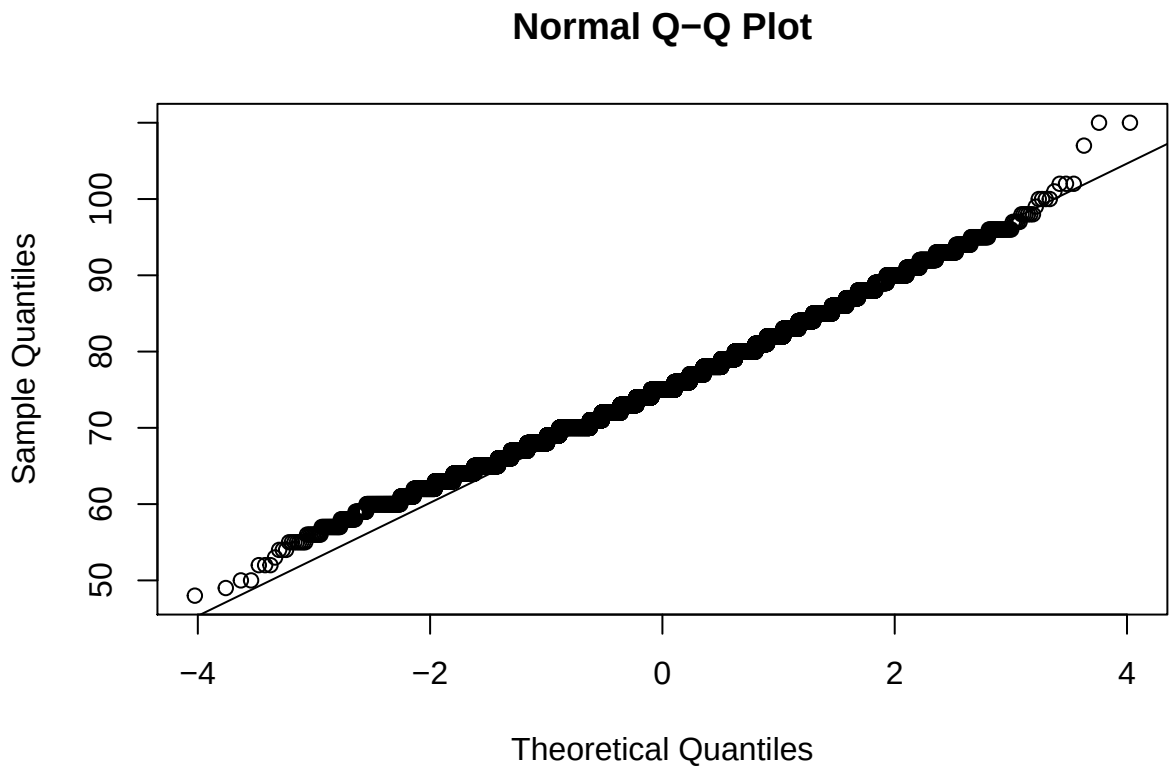
Solución: Para comprobar la normalidad de la variable **Weight**, se procede a representar la misma mediante un histograma, para ver si dicho histograma presenta la forma de una campana de Gauss, y posteriormente mediante el gráfico Q-Q, el cual representa los cuantiles de la variable y veremos si estos se ajustan a la recta que cruza en diagonal los cuadrantes de la gráfica.

```
hist(fifaNet$Weight, breaks=sqrt(dim(fifaNet)[1]),
     xlab="Peso en kg", col="#33C4FF")
```



En primer lugar, podemos observar que la distribución de la variable **Weight** en el histograma es apróximadamente de manera normal, es decir, con un intervalo donde se concentra la moda de la variable.

```r
qqnorm(fifaNet$Weight)
qqline(fifaNet$Weight)
```



Por otro lado, en la gráfica anterior podemos observar que la mayoría de los puntos se ajustan a la recta, por lo que no hay evidencias contra el supuesto de normalidad.

3 Estadística inferencial

Enunciado:

Suponemos que los jugadores del año 2017 son una muestra representativa de los jugadores de la última década (población). Utilizamos el conjunto de datos fifaNet.

3.1 Intervalo de confianza de la media poblacional de la variable Weight

a) Calculad manualmente el intervalo de confianza al 95% de la media poblacional de la variable Weight de los jugadores (No se pueden utilizar funciones como t.test o z.test para el cálculo). A partir del resultado obtenido, explicad cómo se interpreta el intervalo de confianza.

Solución:

A continuación, se procede a declarar una función que permita calcular el intervalo de confianza de cualquier variable numérica:

```r
getConfidentInterval<- function(var){
  s = sd(var)
  n = length(var)
  me = abs(qt((1-0.95)/2,n-1 )) * (s/sqrt(n))
  x = mean(var)
```

```
  confidenceInterval = c(x-me,x+me)
  return (confidenceInterval)
}
```

Una vez declarada la función, procedemos a emplear la misma para calcular el intervalo de confianza de la variable **Weight**

```
getConfidentInterval(fifaNet$Weight)
```

```
## [1] 75.15113 75.35504
```

Para demostrar que la función creada funciona correctamente, se procede a calcular el intervalo mediante la función **CI** de R.

```
CI(fifaNet$Weight, ci=0.95)
```

```
##    upper     mean    lower
## 75.35504 75.25308 75.15113
```

Como se puede observar, el resultado obtenido es el mismo por ambas funciones, solo que mostrado de distinta forma.

Interpretación: La interpretación del intervalo de confianza con un nivel de confianza de 95%, se corresponde con que el 95% de las veces que se calcule la media de la variable de la cual se está calculando el intervalo, de una muestra extraída de la misma población que esta, dicha media se encontrará entre el intervalo que ha sido calculado. En este caso, el 95% de las veces que se extraiga una muestra de la misma población que esta, la media del peso de los jugadores se encontrará entre 75.1511297 y 75.3550397

Enunciado:

b) Calculad los intervalos de confianza al 95% de la media poblacional de la variable Weight, en función de si los jugadores son de campo o porteros. ¿Qué conclusión se puede extraer de la comparación de los dos intervalos, en relación a si existe solapamiento o no en los intervalos de confianza? Justificad la respuesta.

Solución:

Para calcular el intervalo de confianza de la variable **Weight** en función de los jugadores de campo o porteros volveremos a aplicar la función **getConfidentInterval** y filtraremos por los dos posibles valores de la variable **portero**:

```
getConfidentInterval(fifaNet[fifaNet$portero=='Yes',]$Weight)
```

```
## [1] 82.53728 83.47538
```

```
getConfidentInterval(fifaNet[fifaNet$portero=='No',]$Weight)
```

```
## [1] 74.86233 75.06583
```

Como podemos observar, el intervalo de confianza de la media de peso obtenido para los porteros es diferente que el obtenido para los jugadores de campo. Por lo que, según los intervalos obtenidos, podemos asegurar con un 95% de confianza, que los porteros tienen un mayor peso que los jugadores de campo. Esto puede deberse a que un portero a lo largo de un partido de fútbol y a lo largo de un entrenamiento, no realiza el mismo ejercicio físico que un jugador de campo.

3.2 Contraste de hipótesis para la diferencia de medias

Enunciado:

¿Podemos aceptar que la altura de los porteros supera en más de 5 centímetros la altura de los jugadores de campo? Responded a la pregunta utilizando un nivel de confianza del 95%.

Nota: se deben realizar los cálculos manualmente. No se pueden usar funciones de R que calculen directamente el contraste como t.test o similar. Sí se pueden usar funciones como mean, sd, qnorm, pnorm, qt y pt.

Seguid los pasos que se detallan a continuación

3.2.1 Escribid la hipótesis nula y la alternativa

Solución:

- $H_0 : \mu_{hp} - \mu_{hj} = 5$
- $H_1 : \mu_{hp} - \mu_{hj} > 5$

3.2.2 Justificación del test a aplicar

Solución:

En primer lugar, antes de indicar el tipo de test, indicamos que podemos asumir que se trata de una muestra con distribución normal debido al tamaño de la misma (400).

Dado que se trata de una variable que se distribuye de manera normal, para evaluar si la mdeia de altura de los porteros supera en más de 5cm la de los jugadores de campo, podemos aplicar un test de hipótesis de dos muestras sobre la media.

Además, comprobaremos si la varianza de la altura de los jugadores de campo es la misma que la de los porteros, ya función de si la varianza de los dos supuestos es la misma o no, el test a aplicar será de una forma u otra. Para ello, aplicamos el test de **var.test** de R:

```
var.test(fifaNet$Height[fifaNet$portero=='Yes'], fifaNet$Height[fifaNet$portero=='No'])
```

```
##
##  F test to compare two variances
##
## data:  fifaNet$Height[fifaNet$portero == "Yes"] and fifaNet$Height[fifaNet$portero == "No"]
## F = 0.47907, num df = 631, denom df = 16954, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.4293851 0.5376901
## sample estimates:
## ratio of variances
##          0.4790692
```

Como se puede observar, el p-valor obtenido es inferior a 0.05, por lo que descartamos la igualdad de las varianzas en las dos poblaciones.

En consecuencia, aplicamos un test de dos muestras independientes sobre la media con varianza desconocida y diferente. En este caso se trata de un test unilateral por la derecha.

3.2.3 Cálculos

Solución:

En primer lugar, se declara una función que permite el cálculo de un test de hipótesis sobre dos medias:

```
mean_hip_test <- function(x1, x2, CL=0.95, equalvar=TRUE, type="bilateral",
                          value=0){

  mean1<-mean(x1)
```

```
  n1<-length(x1)
  sd1<-sd(x1)
  mean2<-mean(x2)
  n2<-length(x2)
  sd2<-sd(x2)

  if(equalvar){
    comun_std <- sqrt( ((n1-1)*sd1^2 + (n2-1)*sd2^2 )/(n1+n2-2) )
    Sb <- comun_std*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-value) / Sb

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

  solution<-data.frame(t,tcritical,pvalue,df)
  return(solution)
}
```

Una vez declarada la función, procedemos a llamar a dicha función y calcular el test:

```
mean_hip_test(x1=fifaNet$Height[fifaNet$portero=='Yes'],
              x2=fifaNet$Height[fifaNet$portero=='No'],
              equalvar=FALSE, type="greater",
              value=5)
```

```
##          t tcritical       pvalue       df
## 1 13.70655  1.646936 1.562832e-38 732.8479
```

3.2.4 Interpretación del test

Dado a que el valor obtenido por el p-valor es < 0.05 descartamos la hipótesis nula y podemos concluir que con una confianza del 95% la altura de los porteros supera en 5 centímetros a la altura de los jugadores de campo.

4 Modelo de regresión lineal

Enunciado:

Estimad 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.

Especificad el nivel base de referencia de las variables cualitativas, usando la función relevel:

– *Para la variable portero, la categoría "Portero".*

– *Para la variable Preffered_Foot, la categoría "Left".*

Solución:

En primer lugar realizaremos una selección del dataframe *fifaNet* de las columnas solicitadas en el enunciado de este apartado, para aplicar el modelo sobre dicha selección. Para ello, utilizaremos la función **select** del paquete **dplyr**.

```
features_model <- c("Age","portero","Weight","Preffered_Foot", "Vision",
                    "Ball_Control", "Rating")
fifaNet_model <- fifaNet %>% select(features_model)
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(features_model)` instead of `features_model` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
head(fifaNet_model)
```

```
##   Age portero Weight Preffered_Foot Vision Ball_Control Rating
## 1  32      No     80          Right     85           93     94
## 2  29      No     72           Left     90           95     93
## 3  25      No     68          Right     80           95     92
## 4  30      No     85          Right     84           91     92
## 5  31     Yes     92          Right     70           48     92
## 6  26     Yes     82          Right     68           31     90
```

```
str(fifaNet_model)
```

```
## 'data.frame':    17587 obs. of  7 variables:
##  $ Age           : int  32 29 25 30 31 26 28 27 35 24 ...
##  $ portero       : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 2 1 1 1 2 ...
##  $ 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 ...
##  $ Vision        : int  85 90 80 84 70 68 78 79 83 44 ...
##  $ Ball_Control  : int  93 95 95 91 48 31 87 88 90 23 ...
##  $ Rating        : int  94 93 92 92 92 90 90 90 90 89 ...
```

Una vez creada una variable que recoja las columnas que van a ser utilizadas para el modelo, procederemos a especificar el nivel base de referencia de las variables **portero y Preffered_Foot**.

```
fifaNet_model$portero <- relevel(fifaNet_model$portero,ref="Yes")
fifaNet_model$Preffered_Foot <- relevel(fifaNet_model$Preffered_Foot,ref="Left")
```

Una vez especificado el nivel base de referencia, procedemos a la aplicación del modelo de regresión lineal múltiple a través de la función **lm**:

```
lm1 = lm(Rating~.,data=fifaNet_model)
summary(lm1)
```

```
##
## Call:
## lm(formula = Rating ~ ., data = fifaNet_model)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.0537  -3.3570  -0.2433   3.0596  26.0988
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         28.818004   0.561571   51.32   <2e-16 ***
## Age                  0.446879   0.008600   51.96   <2e-16 ***
## porteroNo           -9.353242   0.231161  -40.46   <2e-16 ***
## Weight               0.244425   0.006036   40.49   <2e-16 ***
## Preffered_FootRight -0.047315   0.089212   -0.53    0.596
## Vision               0.089804   0.003946   22.76   <2e-16 ***
## Ball_Control         0.205212   0.003698   55.50   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.96 on 17580 degrees of freedom
## Multiple R-squared:  0.5096, Adjusted R-squared:  0.5094
## F-statistic:  3045 on 6 and 17580 DF,  p-value: < 2.2e-16
```

4.1 Interpretación del modelo

Enunciado:

Interpretad el modelo lineal ajustado:

– *¿Cuál es la calidad del ajuste?*

– *Explicad la contribución de las variables explicativas en el modelo*

Solución:

a) **Calidad del ajuste:** Para valorar la calidad del ajuste del modelo, nos fijaremos en la métrica **Adjusted R-squared** o R cuadrado Ajustado. Esta métrica tiene un valor de **0.5094**, lo que indica que el modelo de regresión múltiple calculado explica apróximadamente el 51% de la variabilidad de la variable **Rating**. En este caso, estamos logrando conseguir un modelo que explica apróximadamente la mitad de la variabilidad de la variable objetivo, por lo que se trata de un modelo que se podría mejorar y que no resulta muy fiable.

b) **Contribución de las variables explicativas:** Para poder evaluar la contribución de cada una de las variables explicativas nos fijaremos en la columna **Estimate** de la visualizaciñon anterior.

1. **Age**: En el caso de la variable **Age**, el resultado obtenido es de 0.44, lo que indica que por cada unidad que aumenta la edad del jugador, esto supone un aumento de 0.04 en la puntuación del mismo (Rating).
2. **portero**: En este caso, el resultado obtenido es -9.35, lo que indica que el hecho de **ser portero**, afecta de manera negativa a la puntuación obtenida por ese jugador en 9.35 puntos.
3. **Weight**: En este caso, el resultado obtenido es de 0.24, lo que indica que por cada unidad que aumente el peso del jugador, afectará a su puntuación un 0.24.
4. **Preffered_Foot**: En este caso, el resultado obtenido es de -0.047, lo que indica que el hecho de **ser zurdo**, afecta de manera negativa a la puntuación del jugadore en 0.047 unidades.

5. **Vision**: En este caso, el resultado obtenido es de 0.089, lo que indica que por cada unidad que aumente la capacidad de visión del jugador, esto afectará a la puntuación del jugador en 0.089 unidades.
6. **Ball_Control**: En este caso, el resultado obtenido es de 0.205, lo que indica que por cada unidad que aumente la capacidad de control del balón del jugador, esto afectará a la puntuación del mismo en 0.205 unidades.

Además de la examinación de la estimación del efecto de cada variable al modelo, es importante destacar los p-valores obtenidos por cada una de las variables del modelo ($Pr(>|t|)$ en la visualización anterior). Si observamos los valores obtenidos por las diferentes variables vemos que todos ellos son inferior a 0.05 menos el obtenido por la variable **Preffered_Foot**, que es igual a 0.596. Esto indica que esta variable no es explicativa para el modelo la mayoría de las veces (el 59.6% de los casos), por lo que no se debería tener en cuenta para la construcción del mismo.

4.2 Predicción

Enunciado:

Aplicad el modelo de regresión para 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.

Solución:

A continuación, procedemos a obtener la predicción solicitada a través de la función **predict**:

```
predict(lm1, newdata = data.frame(Age=24,portero="No",Weight=70,Preffered_Foot="Left",
                                  Vision=60,Ball_Control=80), type='response')
```

```
##        1
## 69.10481
```

5 Regresión logística

5.1 Modelo predictivo

Enunciado:

Ajustad un modelo predictivo basado en la regresión logística para predecir la probabilidad de jugar en la selección nacional en función de las variables: portero, Rating, Age y Work_Rate.

Para ello, cread una variable internacional que indique si el jugador es internacional, es decir, si está en la selección nacional. La variable internacional debe codificarse como una variable dicotómica, que toma el valor 0 cuando el jugador no tiene dorsal en la selección (valor ausente en National_Kit) y 1 cuando tiene dorsal (valor en National_Kit).

La variable internacional será la variable dependiente del modelo. Concretamente, se quiere evaluar la probabilidad de ser un jugador internacional en función de las variables: portero, Rating, Age y Work_Rate. Analizad la calidad del modelo y las variables que son relevantes.

Solución:

En primer lugar, crearemos una función que nos permita obtener los diferentes valores de la variable cualitativa **internacional**, en función de los valores de la variable **National_Kit** que recibirá como parámetro de entrada.

```
get_internacional <- function(x) {
  if (x == -1)
```

```
    return (0)
  else
    return (1)
}
```

En la función anterior utilizamos el valor "-1" para detectar que un jugador no juega en el equipo nacional porque es el valor que imputamos al inicio de la práctica para aquellos valores faltantes de esta columna que correspondían con los jugadores que no jugaban en el equipo nacional.

Una vez creada la función que nos permite obtener los diferentes valores de la variable **internacional**, procedemos a realizar un lapply a todas las filas de la columna **National_Kit** del dataframe **fifaNet** e imputar el resultado obtenido en la nueva variable **internacional**:

```
fifaNet$internacional <- lapply(fifaNet$National_Kit,get_internacional)
fifaNet$internacional <- unlist(fifaNet$internacional)
fifaNet$internacional <- as.factor(fifaNet$internacional)
head(fifaNet$internacional)
```

```
## [1] 1 1 1 1 1 1
## Levels: 0 1
```

```
tail(fifaNet$internacional)
```

```
## [1] 0 0 0 0 0 0
## Levels: 0 1
```

Una vez obtenida la variable **internacional**, procedemos a seleccionar las variables que formarán el modelo de regresión logística y a almacenarlas en una nueva variable:

```
features_glm <- c("portero","Rating","Age","Work_Rate","internacional")
fifaNet_glm <- fifaNet %>% select(features_glm)
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(features_glm)` instead of `features_glm` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
head(fifaNet_glm)
```

```
##   portero Rating Age       Work_Rate internacional
## 1      No     94  32      High / Low             1
## 2      No     93  29 Medium / Medium             1
## 3      No     92  25   High / Medium             1
## 4      No     92  30   High / Medium             1
## 5     Yes     92  31 Medium / Medium             1
## 6     Yes     90  26 Medium / Medium             1
```

Por último, procedemos a calcular el modelo de regresión logística solicitado:

```
glm1 <- glm(internacional~.,data=fifaNet_glm,
            family=binomial (link = logit))
summary(glm1)
```

```
##
## Call:
## glm(formula = internacional ~ ., family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -1.6060  -0.3454  -0.2231  -0.1357   3.6023
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -15.312315   0.462512 -33.107  < 2e-16 ***
## porteroYes                 0.471770   0.146771   3.214 0.001307 **
## Rating                     0.197794   0.005951  33.235  < 2e-16 ***
## Age                       -0.028663   0.008623  -3.324 0.000888 ***
## Work_RateHigh / Low       -0.437268   0.189782  -2.304 0.021220 *
## Work_RateHigh / Medium    -0.493416   0.132552  -3.722 0.000197 ***
## Work_RateLow / High       -0.537016   0.244983  -2.192 0.028375 *
## Work_RateLow / Low        -1.398818   1.068024  -1.310 0.190289
## Work_RateLow / Medium     -0.556625   0.258108  -2.157 0.031040 *
## Work_RateMedium / High    -0.435759   0.149203  -2.921 0.003494 **
## Work_RateMedium / Low     -0.857744   0.202352  -4.239 2.25e-05 ***
## Work_RateMedium / Medium  -0.623227   0.125008  -4.985 6.18e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 6447.9  on 17575  degrees of freedom
## AIC: 6471.9
##
## Number of Fisher Scoring iterations: 6
```

5.2 Matriz de confusión

Enunciado:

A continuación analizad la precisión del modelo, comparando la predicción del modelo sobre los mismos datos del conjunto de datos. Asumiremos que la predicción del modelo es 1 (internacional) si la probabilidad del modelo de regresión logística es superior o igual a 0.5 y 0 en caso contrario. Analizad la matriz de confusión y las medidas de 'sensitivity' y 'specificity'.

Nota: Tomad como variable de interés ser jugador internacional. Por tanto, internacional igual a 1 será el caso positivo en la matriz de confusión y 0 el caso negativo.

Solución:

A continuación, procedemos a obtener la matriz de confusión del modelo utilizando para ello los mismos datos que se han utilizado para entrenar al mismo.

```
y_pred <- ifelse(glm1$fitted.values < 0.5, 0, 1)
glm1.confusion_matrix <- ConfusionMatrix(y_true=fifaNet_glm$internacional,
                                         y_pred=y_pred)
glm1.confusion_matrix
```

```
##        y_pred
## y_true     0     1
##      0 16469    43
##      1   977    98
```

```r
Specificity(y_true=fifaNet_glm$internacional,
            y_pred=y_pred, positive = "1" )
```

```
## [1] 0.9973958
```

```r
Sensitivity(y_true=fifaNet_glm$internacional,
            y_pred=y_pred, positive = "1")
```

```
## [1] 0.09116279
```

Interpretación:

- **Matriz de confusión**: si evaluamos la matriz de confusión, podemos observar que el modelo calculado se encuentra sesgado hacia el caso negativo, esto quiere decir que tiende a asignar los casos a la clase "0" porque es la que más se repite. Esto puede detectarse porque de la clase 0 predicce bien la gran mayoría de los casos, fallando en menos de un 1% de los casos (lo cual ya es sospechoso que consiga predecir tan bien), y sin embargo en el caso de la clase "1" los predicce la mayoría mal, ya que tiende a estimar el valor que más se repite que es el 0. Esto puede ser debido a que no cuenta con la información suficiente para predecir correctamente los valores o que las variables no son lo suficientemente explicativas.
- **Specificity**: Esta métrica mide la proporción de negativos reales con respecto a los negativos estimados por el modelo. Su valor máximo es 1, ya que se trata de una proporción, y el valor es tan alto en este caso (0.997) por lo indicado anteriormente, porque el modelo se encuentra sesgado hacia el caso negativo.
- **Sensitivity**: Esta métrica, al contrario que la anterior, mide la proporción de positivos reales con respecto a los positivos estimados por el modelo. Su valor es muy bajo debido a que el modelo calculado no está logrando predecir correctamente los casos positivos.

5.3 Interpretación

Enunciado:

Interpretad el modelo ajustado. Concretamente, explicad la contribución de las variables explicativas para predecir si el jugador juega en la selección o no.

Solución: En primer lugar, para evaluar la contribución de las variables al modelo, se calculará un modelo que incluya únicamente la variable a analizar, para obtener la relación entre la variable objetivo (**internacional**) y la variable que se está evaluando en concreto, para realizar lo que se denomina *análisis en crudo*. Posteriormente, se analizarán los Odds Ratio (OR de ahora en adelante) en crudo para cada una de estas variables, para ver como afectan para predecir si un jugador juega en la selección o no.

Para evaluar las diferentes variables, seguiremos el mismo orden que el obtenido con el resumen del modelo.

```r
glm_portero <- glm(formula=internacional~portero,data=fifaNet_glm,
            family=binomial (link = logit))
summary(glm_portero)
```

```
##
## Call:
## glm(formula = internacional ~ portero, family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.5203  -0.3478  -0.3478  -0.3478   2.3814
##
## Coefficients:
```

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.77510    0.03267 -84.930  < 2e-16 ***
## porteroYes   0.84358    0.12401   6.802 1.03e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 8053.3  on 17585  degrees of freedom
## AIC: 8057.3
##
## Number of Fisher Scoring iterations: 5
```

En el resultado obtenido se puede observar que el signo del coeficiente es positivo, lo que significa que la posibilidad de jugar en el equipo nacional es mayor en caso de que el jugador sea un portero, lo cual es lógico porque hay una menor competencia entre porteros que entre jugadores de campo, porque son menos.

A continuación procedemos a calcular los OR de la variable portero.

```
exp(coefficients(glm_portero))
```

```
## (Intercept)  porteroYes
##  0.06234336  2.32466681
```

Se tiene un OR para la variable portero igual a 2.32, por lo que la ocurrencia de jugar en el equipo nacional en el caso de los porteros es 2.32 veces mayor que la de los jugadores de campo.

Pasamos a la variable **Rating**

```
glm_rating <- glm(formula=internacional~Rating,data=fifaNet_glm,
            family=binomial (link = logit))
summary(glm_rating)
```

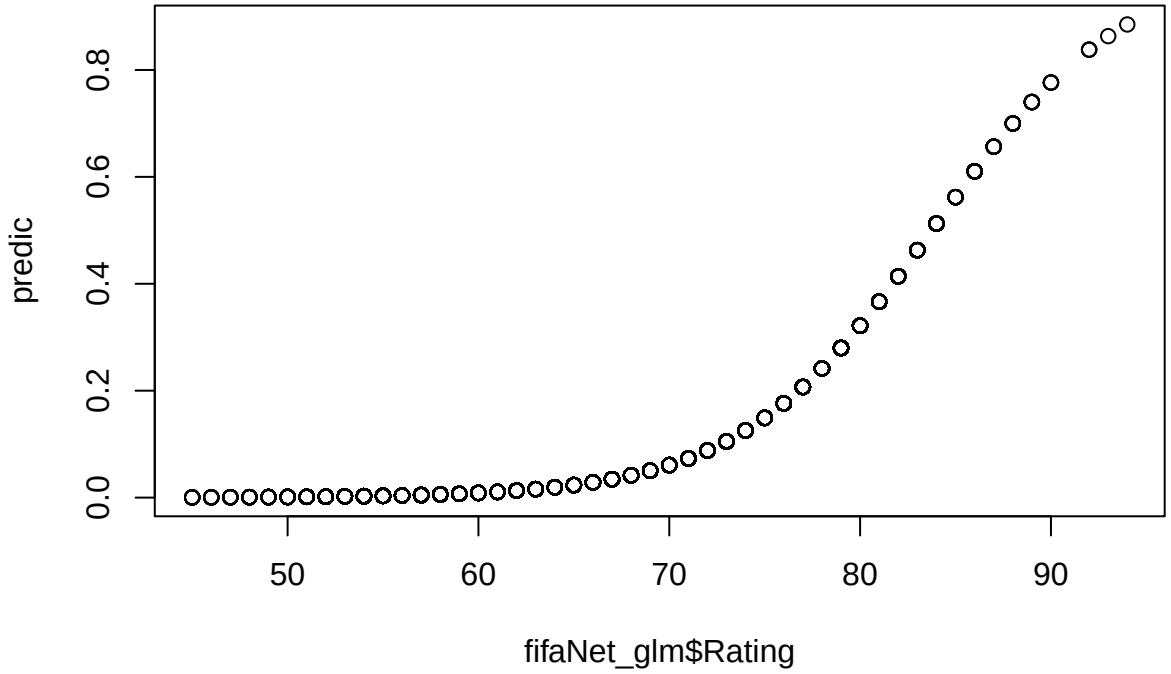
```
##
## Call:
## glm(formula = internacional ~ Rating, family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7309  -0.3542  -0.2175  -0.1327   3.5561
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -16.675200   0.415714  -40.11   <2e-16 ***
## Rating        0.199114   0.005675   35.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 6493.9  on 17585  degrees of freedom
## AIC: 6497.9
##
## Number of Fisher Scoring iterations: 6
```

```
exp(coefficients(glm_rating))
```

```
##  (Intercept)       Rating
## 5.728656e-08 1.220321e+00
```

A la vista de los resultados obtenidos, se puede concluir que, por cada unidad que aumenta la variable **Rating**, aumenta la posibilidad de jugar en el equipo nacional. Para interpretar mejor la contribución de esta variable procedemos a obtener una gráfica que represente el aumento de la misma con respecto al aumento de la posibilidad de jugar en el equipo internacional, ya que el valor obtenido por el OR es muy pequeño.

```
predic=predict(glm_rating,type="response")
plot(fifaNet_glm$Rating,predic)
```



Como podemos observar, se trata de una gráfica exponencial, donde la probabilidad va creciendo más rápido a medida que aumenta el Rating, comenzándose a notar sobre todo a partir de 80.

Pasamos a la variable **Age**

```
glm_age <- glm(formula=internacional~Age,data=fifaNet_glm,
           family=binomial (link = logit))
summary(glm_age)
```

```
##
## Call:
## glm(formula = internacional ~ Age, family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.6881  -0.3776  -0.3312  -0.3000   2.6013
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.498503   0.176733  -25.45   <2e-16 ***
## Age          0.067626   0.006488   10.42   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 7985.0  on 17585  degrees of freedom
## AIC: 7989
##
## Number of Fisher Scoring iterations: 5
```

```
exp(coefficients(glm_age))
```

```
## (Intercept)         Age
##  0.01112563  1.06996470
```

A la vista de los resultados obtenidos, podemos concluir que por cada unidad que aumenta la edad del jugador aumenta 1.069 veces la posibilidad de jugar en el equipo nacional.

Pasamos a la variable **Work_Rate**

```
glm_work_Rate <- glm(formula=internacional~Work_Rate,data=fifaNet_glm,
            family=binomial (link = logit))
summary(glm_work_Rate)
```

```
##
## Call:
## glm(formula = internacional ~ Work_Rate, family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.5945  -0.4116  -0.3058  -0.3058   2.6081
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -1.64356    0.09931 -16.550  < 2e-16 ***
## Work_RateHigh / Low       -0.92434    0.17484  -5.287 1.24e-07 ***
## Work_RateHigh / Medium    -0.78233    0.12023  -6.507 7.66e-11 ***
## Work_RateLow / High       -1.16101    0.22865  -5.078 3.82e-07 ***
## Work_RateLow / Low        -1.72373    1.02113  -1.688   0.0914 .
## Work_RateLow / Medium     -1.37104    0.24456  -5.606 2.07e-08 ***
## Work_RateMedium / High    -0.77876    0.13628  -5.715 1.10e-08 ***
## Work_RateMedium / Low     -1.33122    0.18840  -7.066 1.59e-12 ***
## Work_RateMedium / Medium -1.39600    0.11036 -12.649  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 7918.4  on 17578  degrees of freedom
## AIC: 7936.4
##
## Number of Fisher Scoring iterations: 5
```

```r
exp(coefficients(glm_work_Rate))
```

```
##              (Intercept)      Work_RateHigh / Low   Work_RateHigh / Medium
##                0.1932907                0.3967917                0.4573414
##      Work_RateLow / High       Work_RateLow / Low    Work_RateLow / Medium
##                0.3131691                0.1783986                0.2538426
##   Work_RateMedium / High    Work_RateMedium / Low Work_RateMedium / Medium
##                0.4589739                0.2641541                0.2475856
```

A la vista de los resultados obtenidos, podemos concluir que esta variable afecta de manera negativa a la posibilidad de jugar en el equipo nacional independientemente del valor que tome.

Procedemos a evaluar cada uno de los posibles valores:

- En el caso del valor "Hight/Low", la probabilidad de jugar en el equipo nacional es 0.39 veces menor.
- En el caso del valor "High/Medium", la probabilidad de jugar en el equipo nacional es 0.45 veces menor.
- En el caso del valor "Low/High", la probabilidad de jugar en el equipo nacional es 0.31 veces menor.
- En el caso del valor "Low/Low", la probabilidad de jugar en el equipo nacional es 0.17 veces menor.
- En el caso del valor "Low/Medium", la probabilidad de jugar en el equipo nacional es 0.25 veces menor.
- En el caso del valor "Medium/High", la probabilidad de jugar en el equipo nacional es 0.45 veces menor.
- En el caso del valor "Medium/Low", la probabilidad de jugar en el equipo nacional es 0.26 veces menor.
- En el caso del valor "Medium/Medium", la probabilidad de jugar en el equipo nacional es 0.24 veces menor.

5.4 Interpretación de la variable Work_Rate

Enunciado:

La variable Work_Rate es una variable categórica con 9 categorías diferentes. Volved a ajustar el modelo logístico con las variables portero, Rating, Age y Work_Rate, pero ahora considerad como categoría de referencia de la variable Work_Rate la categoría 'Medium / Medium'. Interpretad las diferencias en los resultados.

Solución:

```r
fifaNet_glm$Work_Rate <- relevel(fifaNet_glm$Work_Rate,"Medium / Medium")

glm2 <-glm(internacional~.,data=fifaNet_glm,family=binomial(link = logit))

summary(glm2)
```

```
##
## Call:
## glm(formula = internacional ~ ., family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6060  -0.3454  -0.2231  -0.1357   3.6023
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)            -15.935542   0.435135 -36.622  < 2e-16 ***
## porteroYes               0.471770   0.146771   3.214 0.001307 **
## Rating                   0.197794   0.005951  33.235  < 2e-16 ***
## Age                     -0.028663   0.008623  -3.324 0.000888 ***
## Work_RateHigh / High     0.623227   0.125008   4.985 6.18e-07 ***
## Work_RateHigh / Low      0.185959   0.164009   1.134 0.256867
## Work_RateHigh / Medium   0.129811   0.093054   1.395 0.163014
## Work_RateLow / High      0.086211   0.225283   0.383 0.701958
## Work_RateLow / Low      -0.775591   1.064006  -0.729 0.466042
## Work_RateLow / Medium    0.066602   0.239100   0.279 0.780590
## Work_RateMedium / High   0.187468   0.115433   1.624 0.104365
## Work_RateMedium / Low   -0.234517   0.178577  -1.313 0.189098
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 6447.9  on 17575  degrees of freedom
## AIC: 6471.9
##
## Number of Fisher Scoring iterations: 6
```

En este segundo modelo calculado, podemos observar que si cambiamos la categoría de referencia de la variable **Work_Rate**, cambian los valores obtenidos por el modelo, en concreto cambian los valores de los coeficientes de la variable **Work_Rate**. Antes veíamos que esta afectaba de manera negativa al modelo independientemente del valor que tomase, sin embargo ahora solo afecta de manera negativa en algunos casos.

5.5 Importancia de ser portero

Enunciado:

En el modelo anterior, interpretad los niveles de la variable portero a partir del odds ratio. ¿En qué porcentaje se ve aumentada la probabilidad de ir a la selección si eres portero? Proporcionad intervalos de confianza del 95% de los odds ratio.

Realiza el mismo análisis para la variable 'Work_Rate'.

Solución:

En el caso de la variable portero, ya se examinó anteriormente que la probabilidad de jugar en el equipo nacional se veía aumentada en 2.32 veces.

Procedemos a calular el intervalo de confianza:

```
exp(confint(glm_portero))
```

```
## Waiting for profiling to be done...
```

```
##                  2.5 %     97.5 %
## (Intercept) 0.05844016 0.06642691
## porteroYes  1.81095714 2.94647742
```

Los resultados obtenidos indican que la probabilida de jugar en el equipo nacional si el jugador es portero, aumenta en el 95% de los casos entre 1.81 y 2.94 veces.

Para la variable 'Work_Rate' se evaluaron anteriormente, pero como estos han cambiado al cambiar la categoría de referencia, procedemos a comentar los nuevos resultados y a calcular los diferentes intervalos de

confianza.

```
glm_work_Rate <- glm(formula=internacional~Work_Rate,data=fifaNet_glm,
                     family=binomial (link = logit))
summary(glm_work_Rate)
```

```
##
## Call:
## glm(formula = internacional ~ Work_Rate, family = binomial(link = logit),
##     data = fifaNet_glm)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.5945  -0.4116  -0.3058  -0.3058   2.6081
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -3.03956    0.04814 -63.136  < 2e-16 ***
## Work_RateHigh / High    1.39600    0.11036  12.649  < 2e-16 ***
## Work_RateHigh / Low     0.47166    0.15173   3.108  0.00188 **
## Work_RateHigh / Medium  0.61367    0.08313   7.382 1.56e-13 ***
## Work_RateLow / High     0.23499    0.21151   1.111  0.26658
## Work_RateLow / Low     -0.32774    1.01743  -0.322  0.74736
## Work_RateLow / Medium   0.02496    0.22861   0.109  0.91307
## Work_RateMedium / High  0.61724    0.10501   5.878 4.16e-09 ***
## Work_RateMedium / Low   0.06478    0.16718   0.387  0.69842
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8091.8  on 17586  degrees of freedom
## Residual deviance: 7918.4  on 17578  degrees of freedom
## AIC: 7936.4
##
## Number of Fisher Scoring iterations: 5
```

```
exp(coefficients(glm_work_Rate))
```

```
##            (Intercept)   Work_RateHigh / High    Work_RateHigh / Low
##             0.04785601             4.03900652             1.60264439
## Work_RateHigh / Medium    Work_RateLow / High     Work_RateLow / Low
##             1.84720480             1.26489208             0.72055329
##  Work_RateLow / Medium Work_RateMedium / High  Work_RateMedium / Low
##             1.02527189             1.85379854             1.06691994
```

```
exp(confint(glm_work_Rate))
```

```
## Waiting for profiling to be done...
```

```
##                             2.5 %     97.5 %
## (Intercept)            0.04348666 0.05252226
## Work_RateHigh / High   3.24236920 4.99918707
## Work_RateHigh / Low    1.17767478 2.13727087
## Work_RateHigh / Medium 1.56749213 2.17166989
## Work_RateLow / High    0.81531162 1.87472186
## Work_RateLow / Low     0.04039027 3.37654156
```

```
## Work_RateLow / Medium  0.63557322 1.56445500
## Work_RateMedium / High 1.50355041 2.27009206
## Work_RateMedium / Low  0.75784399 1.46198009
```

Procedemos a analizar los resultados obtenidos para cada uno de los valores posibles de la variablw **Work_Rate**:

- En el caso del valor "Hight/Hight", la probabilidad de jugar en el equipo nacional es 4.039 veces mayor de media y entre 3.24 y 4.99 según el IC obtenido.
- En el caso del valor "Hight/Low", la probabilidad de jugar en el equipo nacional es 1.60 veces mayor de media y entre 1.17 y 2.13 según el IC obtenido.
- En el caso del valor "High/Medium", la probabilidad de jugar en el equipo nacional es 1.84 veces mayor de media y entre 1.56 y 2.71 según el IC obtenido.
- En el caso del valor "Low/High", la probabilidad de jugar en el equipo nacional es 1.26 veces mayor de media y entre 0.81 y 1.84 según el IC obtenido.
- En el caso del valor "Low/Low", la probabilidad de jugar en el equipo nacional es 0.72 veces menor de media y entre 0.04 y 3.37 según el IC obtenido.
- En el caso del valor "Low/Medium", la probabilidad de jugar en el equipo nacional es 1.02 veces mayor de media y entre 0.63 y 1.56 según el IC obtenido.
- En el caso del valor "Medium/High", la probabilidad de jugar en el equipo nacional es 1.85 veces mayor de media y entre 1.50 y 2.27 según el IC obtenido.
- En el caso del valor "Medium/Low", la probabilidad de jugar en el equipo nacional es 1.06 veces mayor de media y entre 0.75 y 1.46 según el IC obtenido.

5.6 Predicción

Enunciado:

¿Con que probabilidad un portero de 25 años, con un rating de 95 puntos y una clasificación de Work_Rate como High/High irá a la selección?

Solución:

Para calcular la probabilidad aplicaremos la función **predict**:

```
predict(glm2,newdata = data.frame(portero="Yes",Age=25,Rating=95,
                                  Work_Rate="High / High"), type='response')
```

```
##         1
## 0.9620684
```

La probabilidad obtenida es de 0.96.

6 Análisis de la varianza (ANOVA) de un factor

Vamos a realizar un ANOVA para contrastar si existen diferencias en la variable Rating en función del grupode edad al que pertenecen los jugadores. Seguid los pasos que se indican.

6.1 Visualización gráfica

Enunciado:

En primer lugar, a partir de la variable Age cread una variable categórica denominada Age_Int, que clasifique al jugador en una de estas tres categorías: Junior (edad menor o igual a 20), Middle (edad entre 21 y 27), Senior (edad mayor o igual a 28).

Mostrad gráficamente la distribución de Rating según los valores de Age_Int ordenados: Junior, Middle, Senior.

Solución:

En primer lugar, para crear la variable **Age_Int** crearemos una función que devuelva los diferentes posibles valores que puede tomar la misma en función de la variable **Age** que recibirá como parámetro de entrada:

```
get_age_int<-function(x){
  if (x <= 20)
    return ("Junior")
  else if (x >= 21 & x <=27)
    return("Middle")
  else if (x>=28)
    return("Senior")
}
```

Una vez creada la función, realizaremos un lapply por cada una de las filas de la columna **Age** del dataframe fifaNet e imputaremos el resultado obtenido en la nueva variable **Age_Int**:

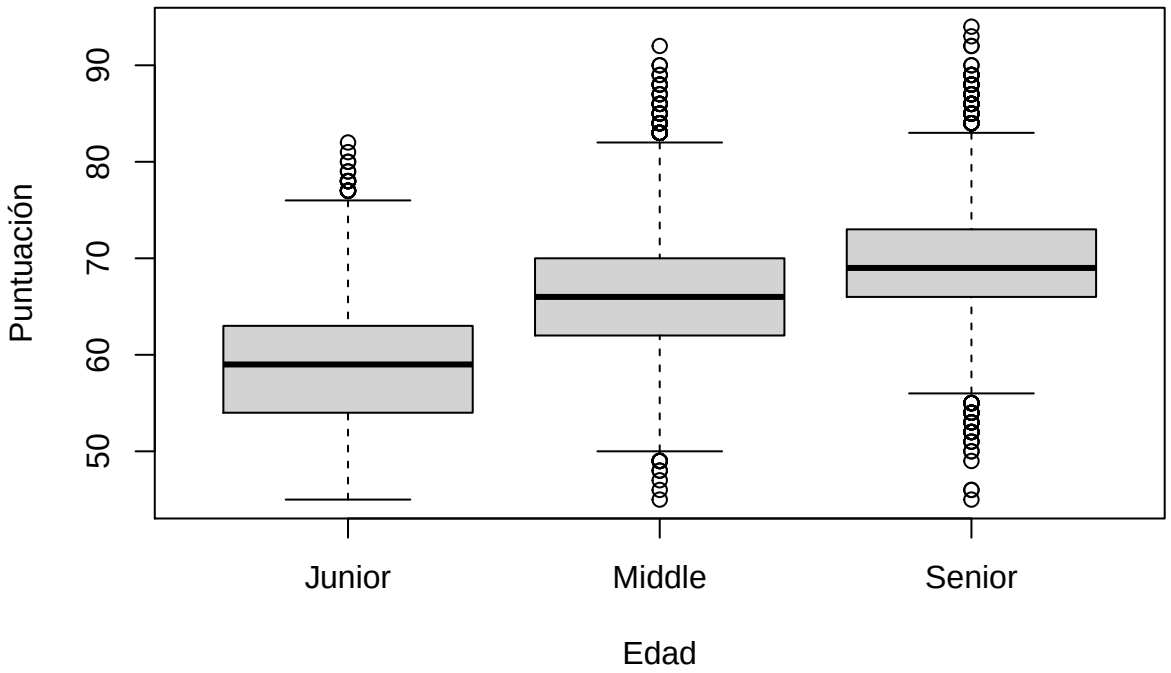
```
fifaNet$Age_Int <- lapply(fifaNet$Age,get_age_int)
fifaNet$Age_Int <- unlist(fifaNet$Age_Int)
fifaNet$Age_Int <- as.factor(fifaNet$Age_Int)

levels(fifaNet$Age_Int)
```

```
## [1] "Junior" "Middle" "Senior"
```

Una vez que ha sido incluida la nueva variable en el dataframe, procedemos a representar como se distribuye la variable **Rating** en función de los diferentes grupos de edad.

```
boxplot(Rating~Age_Int,data=fifaNet,xlab="Edad",
        ylab="Puntuación")
```



Como se puede observar, la mediana de la puntuación aumenta a medida que lo hace el grupo de edad, por lo que parece que la variable Edad si es un factor determinante.

6.2 Hipótesis nula y alternativa

Enunciado:

Escribid la hipótesis nula y la alternativa.

Solución:

- $H_0 : \alpha_1 = \alpha_2 = \alpha_3 = 0$
- $H_1 : \alpha_i != \alpha_j$ para algún i != j

6.3 Modelo

Enunciado:

Calculad el análisis de varianza, usando la función aov o lm. Interpretad el resultado del análisis, teniendo en cuenta los valores: Sum Sq, Mean SQ, F y Pr (> F).

```
lm_anova <- lm(Rating~Age_Int,data=fifaNet)
taov<-anova(lm_anova)
taov
```

```
## Analysis of Variance Table
##
## Response: Rating
##              Df Sum Sq Mean Sq F value    Pr(>F)
## Age_Int       2 212801  106401  2795.4 < 2.2e-16 ***
## Residuals 17584 669302      38
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Como se puede observar, obtenemos un p-valor muy inferior al nivel de significación del 5%. Por lo tanto, descartamos la hipótesis nula y aceptamos la hipótesis alternativa y concluimos por tanto que el factor es significativo. Es decir, la edad afecta a la puntuación obtenida en la variable **Rating**.

Por otro lado cabe destacar que la varianza del error es de 38 (Valor Mean Sq de la fila Residuals), lo que quiere decir que de media la varianza de la media de cada una de las muestras con respecto a la media total es de 38.

Y en el caso de la varianza obtenida en la variable Age_Int (Valor Mean Sq de la fila Age_Int) realmente corresponde a la media de la varianza que hay en cada uno de los grupos de la variable Age_Int, que en este caso es igual a 106401.

6.4 Efectos de los niveles del factor

Enunciado:

Proporcionad la estimación del efecto de los niveles del factor Age_Int.

Solución:

```
tapply(fifaNet$Rating, fifaNet$Age_Int, mean)
```

```
##   Junior   Middle   Senior
## 58.94176 66.41417 69.40983
```

6.5 Interpretación de los resultados

Enunciado:

Interpretad los resultados obtenidos en los apartados anteriores.

Solución:

Estos valores corresponden con la desviación de la media de cada uno de los grupos frente a la media global de la variable **Rating**

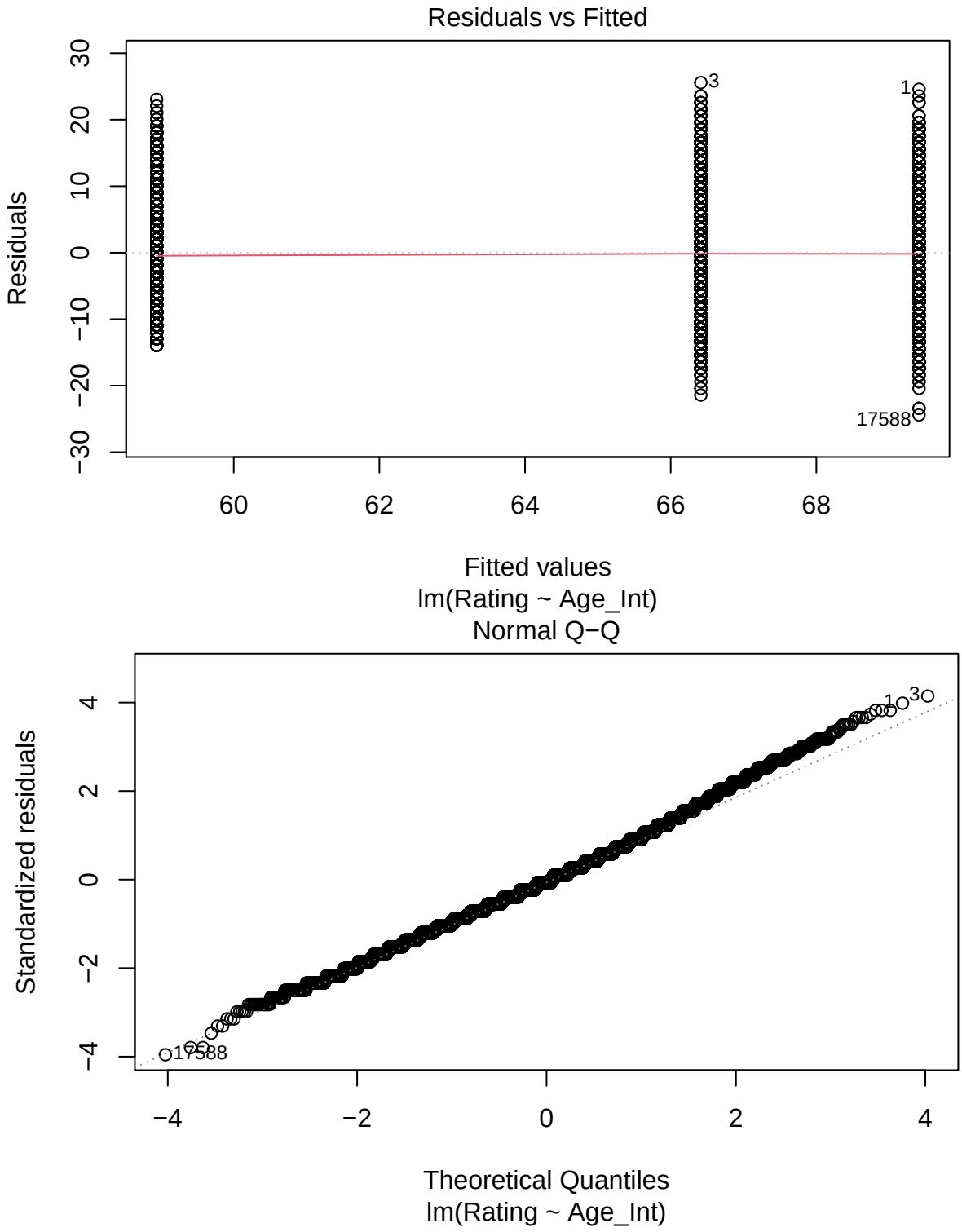
6.6 Adecuación del modelo

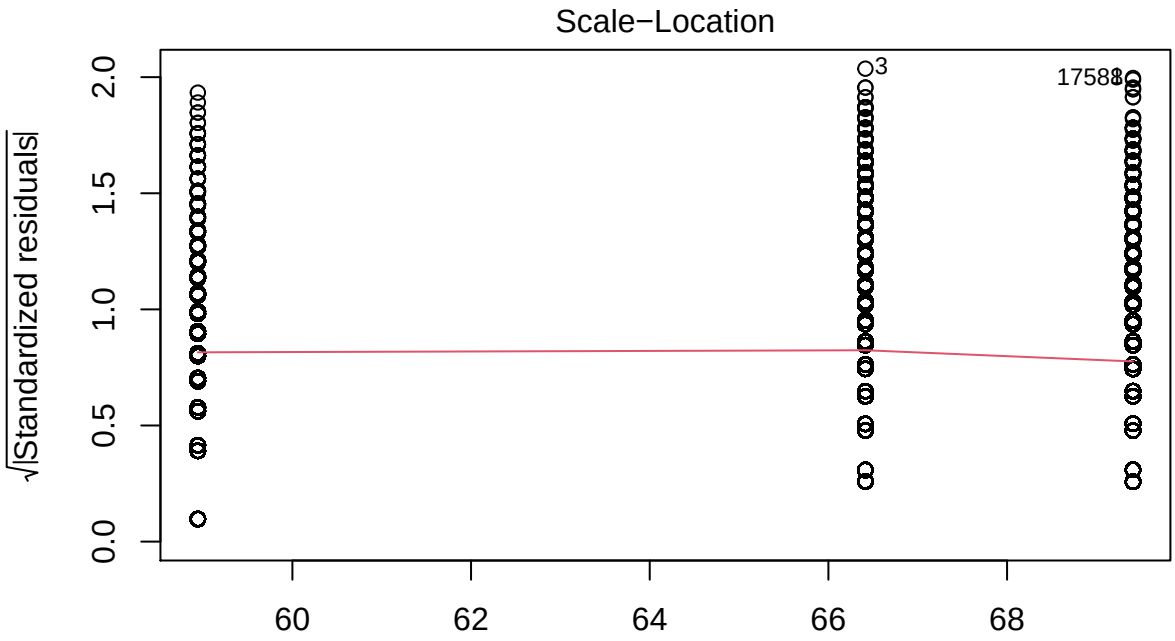
Enunciado:

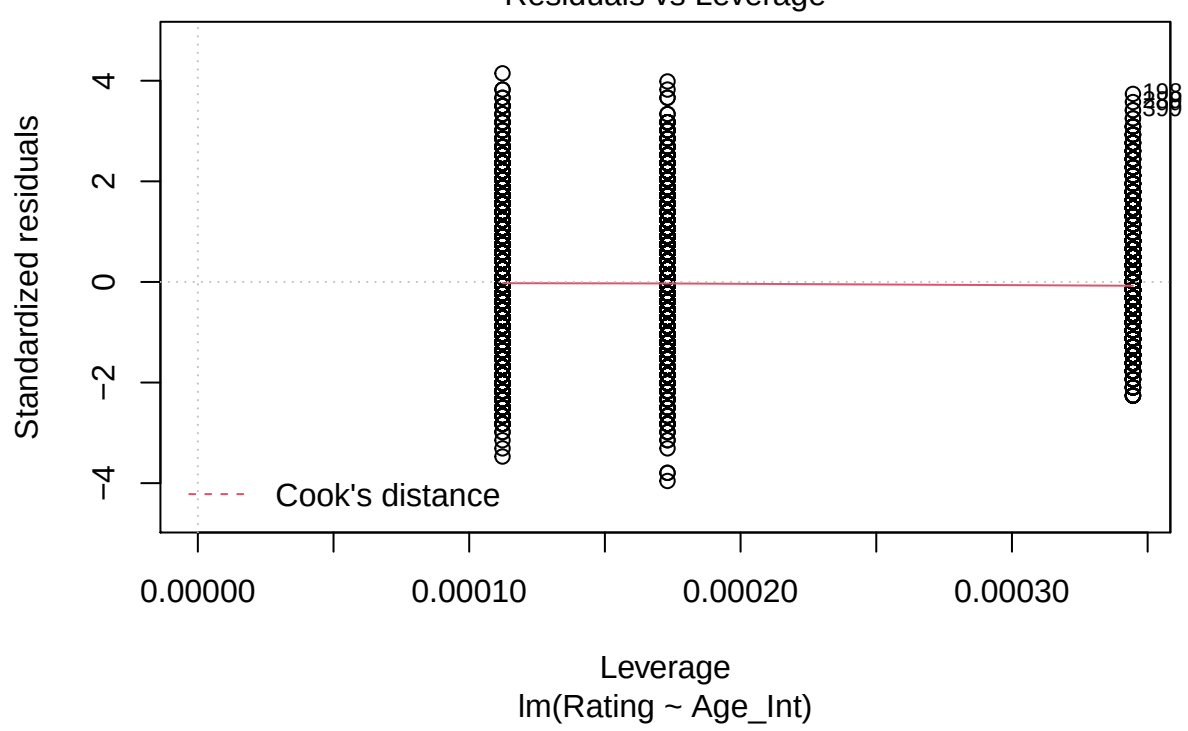
Mostrad visualmente la adecuación del modelo ANOVA. Podéis usar plot sobre el modelo ANOVA calculado.

Solución:

```
plot(lm_anova)
```


Residuals vs Fitted
Residuals
Fitted values
lm(Rating ~ Age_Int)
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Rating ~ Age_Int)





6.6.1 Normalidad de los residuos

Enunciado:

Interpretad la normalidad de los residuos a partir del gráfico Normal Q-Q que habéis mostrado en el apartado

anterior

Solución:

El gráfica Q-Q corresponde con el último mostrado en el apartado anterior. Como se puede observar, la mayoría de los puntos se ajustan a la recta que cruza los ejes de la gráfica, por lo que todo apunta a que los residuos se distribuyen de manera normal con media 0.

6.6.2 Homocedasticidad de los residuos

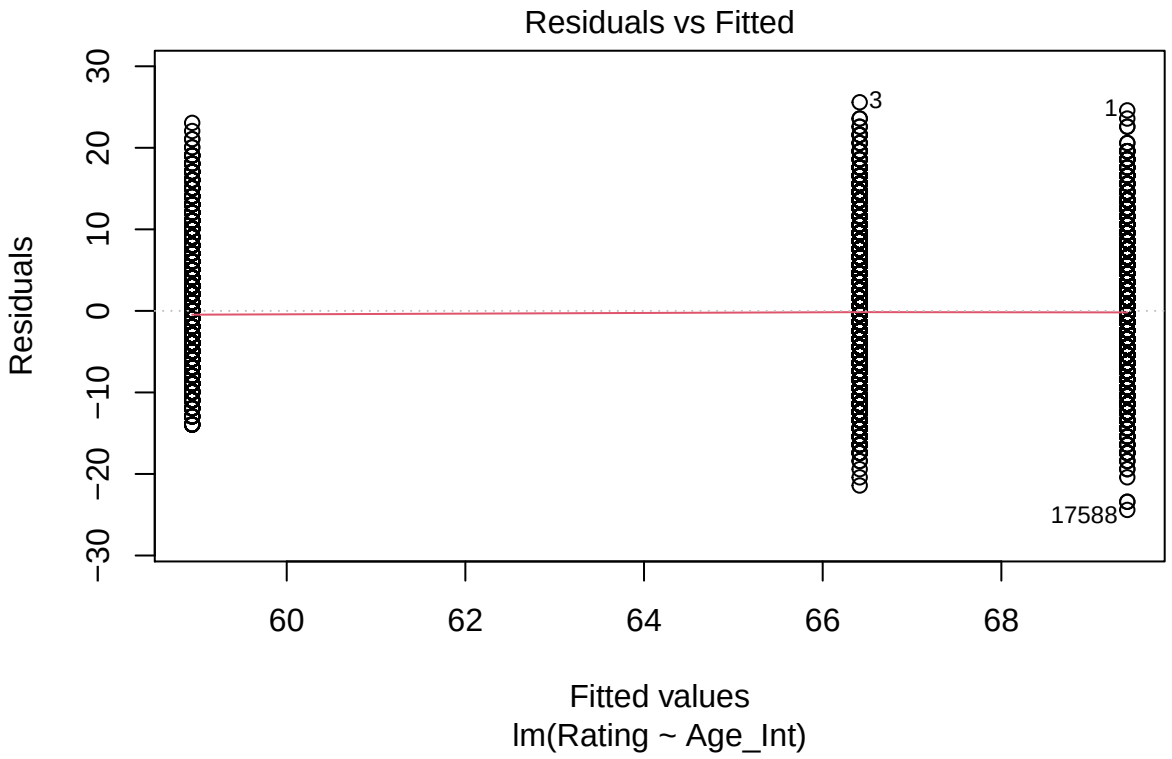
Enunciado:

El gráfico "Residuals vs Fitted" proporciona información sobre la homcedasticidad de los residuos. Mostrad e interpretad este gráfico.

Solución:

En primer lugar, procedemos a mostrar de nuevo el gráfico solicitado:

```
plot(lm_anova, which=1)
```



En este gráfico se pueden observar tres tiras verticales de puntos que están sitiadas en las medias de cada grupo. Estas corresponden a los valores ajustados de las observaciones. La disposición de los residuos muestra una dispersión parecida en cada tira, por lo que todo parece indicar que la igualdad de varianzas se cumple.

7 ANOVA multifactorial

Enunciado:

A continuación, se desea evaluar el efecto sobre Rating del grupo de edad combinado con el factor tipo de jugador (portero). Seguid los pasos que se indican a continuación.

7.1 Análisis visual de los efectos principales y posibles interacciones

Enunciado:

Dibujad en un gráfico la variable Rating en función de Age_Int y en función de portero. El gráfico debe permitir evaluar si hay interacción entre los dos factores. Por ello, se recomienda seguir estos pasos:

1. Agrupad el conjunto de datos por Age_Int y por portero. Calculad la media de rating para cada grupo. Para realizar este proceso, se puede hacer con las funciones group_by y summarise de la librería dplyr.

Solución:

```
fifaNet_mean_group <- fifaNet %>%
  dplyr::group_by(Age_Int,portero) %>%
  dplyr::summarise(mean_rating = mean(Rating))
```

```
## `summarise()` regrouping output by 'Age_Int' (override with `.groups` argument)
```

2. Mostrad el conjunto de datos en forma de tabla (data frame), donde se muestre la media de cada grupo según Age_Int y portero.

Solución:

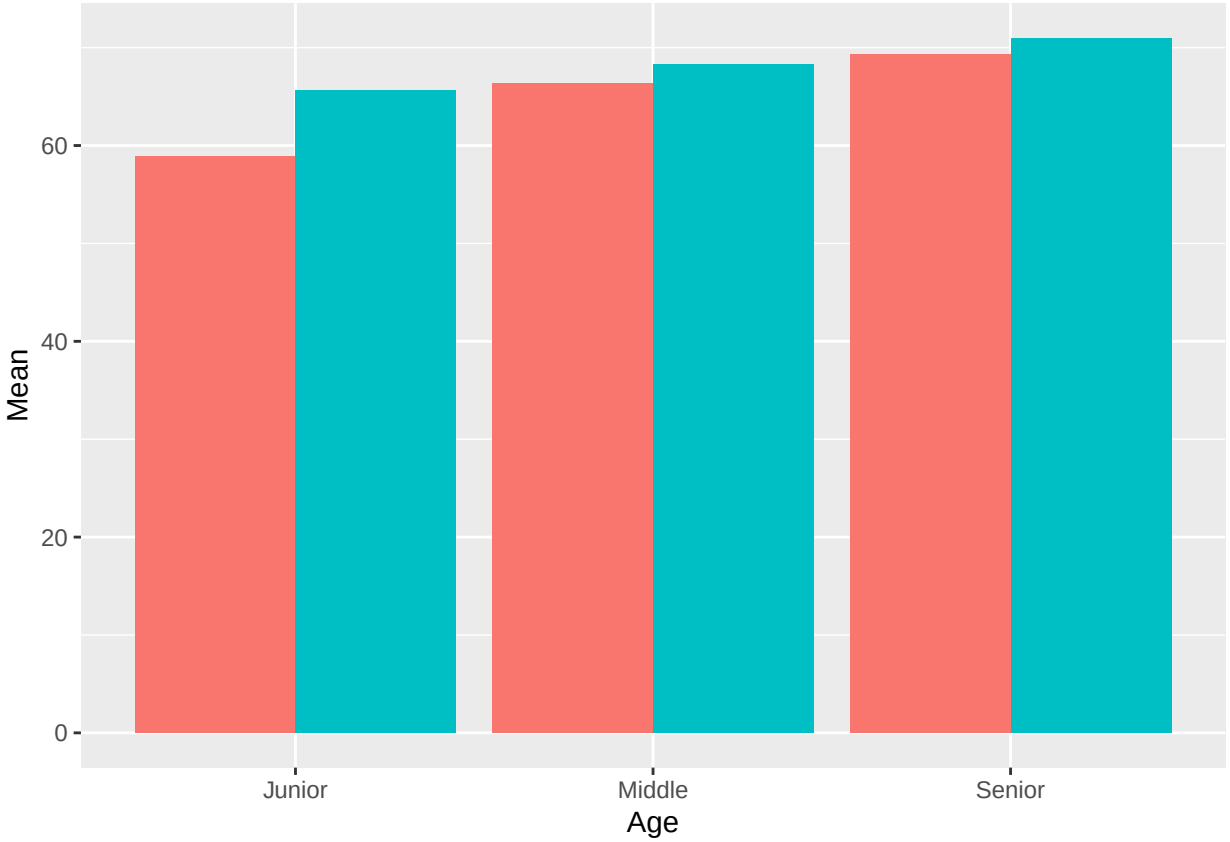
```
fifaNet_mean_group
```

```
## # A tibble: 6 x 3
## # Groups:   Age_Int [3]
##   Age_Int portero mean_rating
##   <fct>   <fct>         <dbl>
## 1 Junior  No             58.9
## 2 Junior  Yes            65.7
## 3 Middle  No             66.4
## 4 Middle  Yes            68.3
## 5 Senior  No             69.3
## 6 Senior  Yes            71.0
```

3. Mostrad en un gráfico el valor medio de la variable Rating para cada factor. Podéis inspiraros en los gráficos de López-Roldán y Fachelli (2015), p.38. Podéis realizar este tipo de gráfico usando la función ggplot de la librería ggplot2.

Solución:

```
ggplot(fifaNet_mean_group,aes(x=Age_Int,y=mean_rating,fill=factor(portero)))+
  geom_bar(stat="identity",position="dodge")+
  scale_fill_discrete(name="Portero",
                      breaks=c(1, 2),
                      labels=levels(fifaNet_mean_group))+
  xlab("Age")+ylab("Mean")
```



(El color rojo representa a aquellos que no son porteros y el azul los que sí son porteros)

4. Interpretad el resultado sobre si sólo hay efectos principales o hay interacción entre los factores. Si hay interacción, explicad cómo se observa esta interacción en el gráfico.

Solución:

Evaluando el gráfico anterior, podemos ver que la media de la variable **Rating** se va haciendo más grande conforme va aumentando la edad del jugador y que en el caso de los porteros esta media es mayor en todos los casos.

7.2 Cálculo del modelo

Enunciado:

Podéis usar la función aov.

Solución:

```
anova_two_factors <- aov(Rating ~ Age_Int*portero, data = fifaNet)
anova(anova_two_factors)
```

```
## Analysis of Variance Table
##
## Response: Rating
##                    Df Sum Sq Mean Sq  F value    Pr(>F)
## Age_Int             2 212801  106401 2805.780 < 2.2e-16 ***
## portero             1   2226    2226   58.700 1.932e-14 ***
## Age_Int:portero     2    370     185    4.877   0.00763 **
```

```
## Residuals       17581 666706      38
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

7.3 Interpretación de los resultados

Solución:

Como se puede observar, el p-valor obtenido en cada uno de los factores principales, es menor a 0.05, por lo que rechazamos la hipótesis nula y aceptamos por tanto que hay efecto de la edad y de si el jugador es un portero o no.

Por otro lado, es interesante analizar los parámetros del modelo.

En el caso de la estimación de la varianza del error, el valor obtenido es de 38.

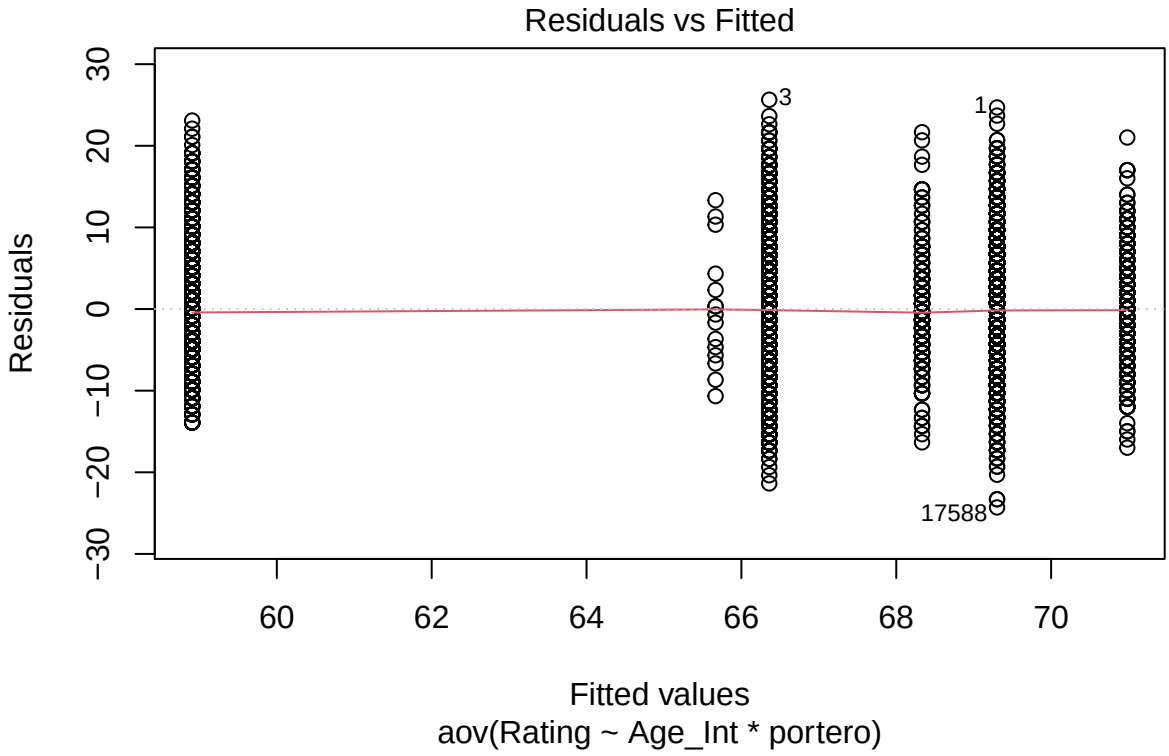
7.4 Adecuación del modelo

Enunciado:

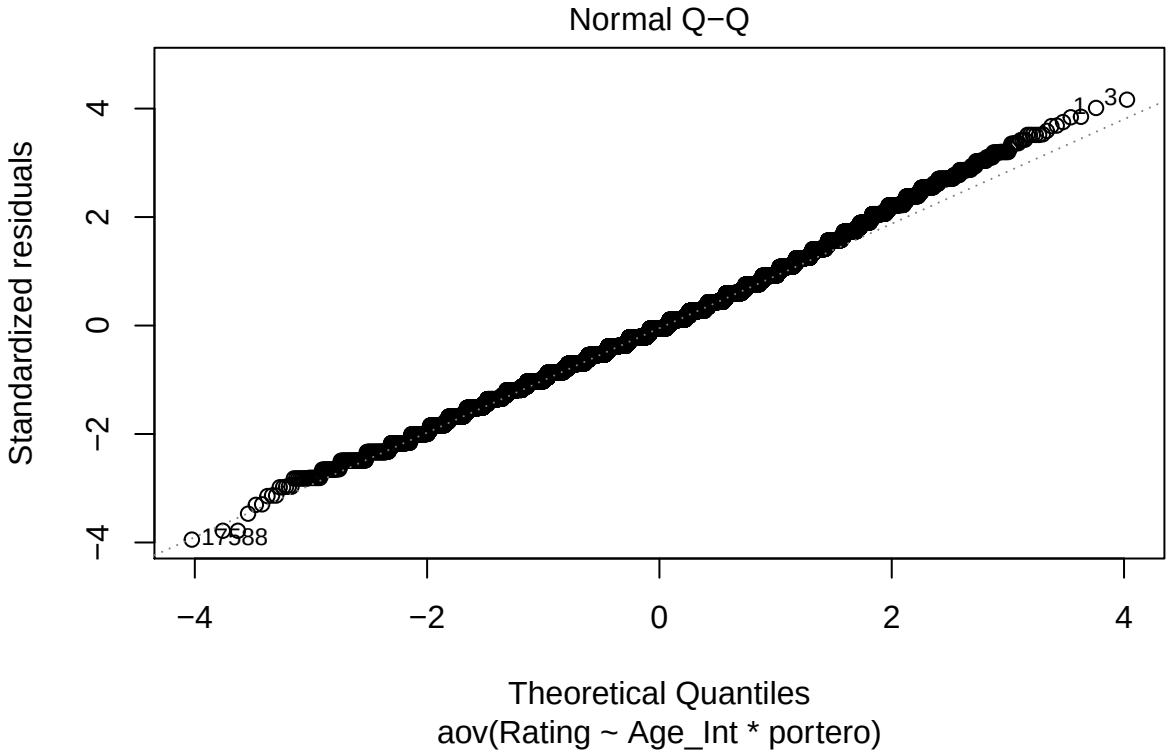
Interpretad la adecuación del modelo ANOVA obtenido usando los gráficos de los residuos.

Solución:

```
plot(anova_two_factors, which=1)
```




```
plot(anova_two_factors, which=2)
```



En el primer gráfico pordemo observar 6 tiras verticales de puntos que están situadas en las medias de cada uno de los grupos formados por las dos variables de tipo factor. La disposición de las tiras es muy similar en la mayoría de los casos menos en el caso de la segunda tira, lo que podría denotar una ligera falta de homocedasticidad, pero esto solo ocurre en uno de los grupos.

Por otro lado, en el segundo gráfico vemos que la mayoría de los puntos se adaptan a la recta que cruza los cuadrantes de la gráfica, lo que indica que los errores se distribuyen de forma normal.

8 Conclusiones

Enunciado:

Resumid las conclusiones principales del análisis. Para ello, podéis resumir las conclusiones de cada uno de los apartados.

Solución:

Procedemos a responder las diferentes cuestiones planteadas en este documento:

1. ¿Podemos asumir que la variable Weight tiene una distribución normal?

Tras representar dicha variable a través de un histograma y a través de un gráfico Q-Q, hemos concluido que la variable **Weight** tiene una distribución normal.

2. Calculad manualmente el intervalo de confianza al 95% de la media poblacional de la variable Weight de los jugadores (No se pueden utilizar funciones como t.test o z.test para el cálculo). A partir del resultado obtenido, explicad cómo se interpreta el intervalo de confianza.

Tras calcular el intervalo de confianza al 95% de la media poblacional a través de la función **getConfidentInterval**, se obtuvo el la conslusión de que el 95% de las veces que se extraiga una muestra de la misma población que esta, la media del peso de los jugadores se encontrará entre 75.15kg y 75.35kg.

3. Calculad los intervalos de confianza al 95% de la media poblacional de la variable Weight, en función de si los jugadores son de campo o porteros. ¿Qué conclusión se puede extraer de la comparación de los dos intervalos, en relación a si existe solapamiento o no en los intervalos de confianza? Justificad la respuesta.

Tras calcular los intervalos de confianza solicitados, la conclusión obtenida es que según los datos presentes en esta muestra analizada, podemos asegurar con un 95% de confianza que el peso de aquellos jugadores que son porteros es mayor que el de los jugadores de campo. En concreto, el intervalo de confianza obtenido del peso de los porteros es 82.53 kg y 83.47 kg y el de los jugadores de campo es 74.86kg y 75.06kg.

4. ¿Podemos aceptar que la altura de los porteros supera en más de 5 centímetros la altura de los jugadores de campo? Responded a la pregunta utilizando un nivel de confianza del 95%.

Tras calcular el test de hipótesis y obtener un p-valor muy por debajo del valor 0.05, podemos asegurar con un 95% de confianza que la altura de los porteros supera en más de 5 centímetros la altura de los jugadores de campo.

5. Con respecto al modelo de regresión lineal calculado en el apartado 4, ¿Cuál es la calidad del ajuste?

La calidad del ajuste del modelo ha sido medida a través de la variable R cuadrado ajustado, que en este caso tiene el valor de **0.5094**, lo que indica que el modelo calculado logra explicar aproximadamente la mitad de la varianza de los datos reales.

6. Con respecto al modelo de regresión lineal calculado en el apartado 4, ¿Cuál es la contribución de las variables explicativas?

- En el caso de la variable **Age** la contribución es de 0.44
- En el caso de la variable **portero** la contribución es de 9.35
- En el caso de la variable **Weight** la contribución es de 0.24
- En el caso de la variable **Preffered_Foot** la contribución es de -0.047

7. Aplicad el modelo de regresión para 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.

El valor predicho por el modelo es igual a 69.1

8. Con respecto al modelo de regresión logística calculado en el apartado 5, ¿Cuáles son los valores de las métricas 'sensitivity' y 'specificity'?.

El valor de la variable Sensitivity es de 0.09 y el valor de la variable Specificity es de 0.997.

9. ¿En qué porcentaje se ve aumentada la probabilidad de ir a la selección si eres portero?

La probabilidad de ir a la selección aumenta si el jugador es un portero 2.32 veces.

10. ¿Con que probabilidad un portero de 25 años, con un rating de 95 puntos y una clasificación de Work_Rate como High/High irá a la selección?

La probabilidad obtenida por el modelo es de 0.96

11. Vamos a realizar un ANOVA para contrastar si existen diferencias en la variable Rating en función del grupo de edad al que pertenecen los jugadores.

Tras calcular el ANOVA, podemos asegurar con un 95% de confianza que existen diferencias en la variable Rating en función al grupo de edad al que pertenecen los jugadores.

12. A continuación, se desea evaluar el efecto sobre Rating del grupo de edad combinado con el factor tipo de jugador (portero). Seguid los pasos que se indican a continuación.

Tras cañciñar eñ ANOVA multi factor, podemos asegurar con un 95% de confianza que existen diferencias en la variable Rating en función del grupo de edad y si el jugador es portero o jugador de campo.