



Programación y estadística con R

| Equipo 16

Ana Elizabeth Guzmán Jiménez

Carlos Paz

Fred Jordi Miramontes Arias

Luis Arturo Rosas León

Ludim Sánchez

Norma Arrazola Herrera

Contenido

PostWork 1: Introducción a R y Software	4
Ejercicio 1: Importa los datos de soccer de la temporada 2019/2020 de la primera división de la liga española a R, los datos los puedes encontrar en el siguiente enlace: https://www.football-data.co.uk/spainm.php	4
Ejercicio 2: Del data frame que resulta de importar los datos a R, extrae las columnas que contienen los números de goles anotados por los equipos que jugaron en casa (FTHG) y los goles anotados por los equipos que jugaron como visitante (FTAG)	4
Ejercicio 3: Consulta cómo funciona la función table en R al ejecutar en la consola ?table	5
Ejercicio 4: Posteriormente elabora tablas de frecuencias relativas para estimar las siguientes probabilidades:	6
Parte 4.1: La probabilidad (marginal) de que el equipo que juega en casa anote x goles ($x = 0, 1, 2, \dots$)	6
Parte 4.2: La probabilidad (marginal) de que el equipo que juega como visitante anote y goles ($y = 0, 1, 2, \dots$)	9
Parte 4.3: La probabilidad (conjunta) de que el equipo que juega en casa anote x goles y el equipo que juega como visitante anote y goles ($x = 0, 1, 2, \dots, y = 0, 1, 2, \dots$)	9
PostWork 2: Programación y manipulación de datos en R	10
Ejercicio 1: Importa los datos de soccer de las temporadas 2017/2018, 2018/2019 y 2019/2020 de la primera división de la liga española a R, los datos los puedes encontrar en el siguiente enlace: https://www.football-data.co.uk/spainm.php	10
Ejercicio 2: Revisa la estructura de de los data frames al usar las funciones: str, head, View y summary	11
Ejercicio 3: Con la función select del paquete dplyr selecciona únicamente las columnas Date, HomeTeam, AwayTeam, FTHG, FTAG y FTR; esto para cada uno de los data frames. (Hint: también puedes usar lapply)	11
Ejercicio 4: Asegúrate de que los elementos de las columnas correspondientes de los nuevos data frames sean del mismo tipo (Hint 1: usa as.Date y mutate para arreglar las fechas). Con ayuda de la función rbind forma un único data frame que contenga las seis columnas mencionadas en el punto 3 (Hint 2: la función do.call podría ser utilizada)	12
PostWork 3: Análisis Exploratorio de Datos (AED o EDA) con R	13
Ejercicio 1: Con el último data frame obtenido en el postwork de la sesión 2, elabora tablas de frecuencias relativas para estimar las siguientes probabilidades:	13
Parte 1.1: La probabilidad (marginal) de que el equipo que juega en casa anote x goles ($x=0,1,2,$)	14
Parte 1.2: La probabilidad (marginal) de que el equipo que juega como visitante anote y goles ($y=0,1,2,$)	14

Parte 1.3: La probabilidad (conjunta) de que el equipo que juega en casa anote x goles y el equipo que juega como visitante anote y goles ($x=0,1,2,, y=0,1,2,$)	15
Ejercicio 2: Realiza lo siguiente:	15
Parte 2.1: Un gráfico de barras para las probabilidades marginales estimadas del número de goles que anota el equipo de casa.....	15
Parte 2.2: Un gráfico de barras para las probabilidades marginales estimadas del número de goles que anota el equipo visitante.	17
Parte 2.3: Un HeatMap para las probabilidades conjuntas estimadas de los números de goles que anotan el equipo de casa y el equipo visitante en un partido.....	18
PostWork 4: Algunas distribuciones, teorema central del límite y contraste de hipótesis ..	19
Ejercicio 1: Ya hemos estimado las probabilidades conjuntas de que el equipo de casa anote $X=x$ goles ($x=0,1,...,8$), y el equipo visitante anote $Y=y$ goles ($y=0,1,...,6$), en un partido. Obtén una tabla de cocientes al dividir estas probabilidades conjuntas por el producto de las probabilidades marginales correspondientes.	19
Ejercicio 2: Mediante un procedimiento de bootstrap, obtén más cocientes similares a los obtenidos en la tabla del punto anterior. Esto para tener una idea de las distribuciones de la cual vienen los cocientes en la tabla anterior. Menciona en cuáles casos le parece razonable suponer que los cocientes de la tabla en el punto 1, son iguales a 1 (en tal caso tendríamos independencia de las variables aleatorias X y Y).	21

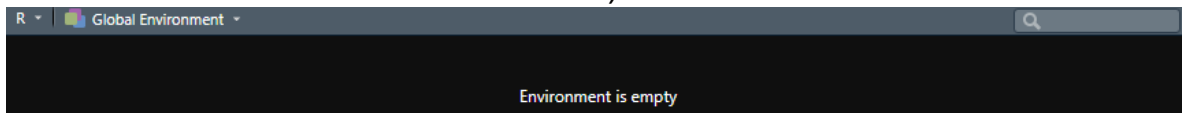
PostWork 1: Introducción a R y Software

Ejercicio 1: Importa los datos de soccer de la temporada 2019/2020 de la primera división de la liga española a R, los datos los puedes encontrar en el siguiente enlace: <https://www.football-data.co.uk/spainm.php>

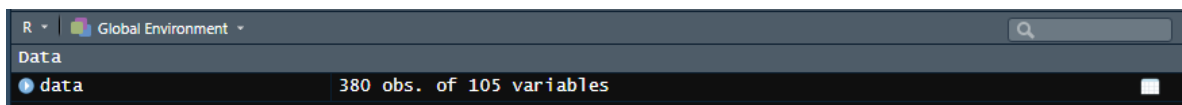
```
#<variable> <- read.csv("enlace o ubicación donde se encuentra el  
archivo csv")
```

```
#La variable almacenara los datos que contenga el archivo el cual se  
#obtendrá mediante la lectura del método read.csv que recibe como  
#parámetro tanto como el enlace del csv como la dirección local donde  
#se encuentre el archivo csv.
```

```
data <- read.csv("https://www.football-  
data.co.uk/mmz4281/1920/SP1.csv")
```



```
data <- read.csv("https://www.football-data.co.uk/mmz4281/1920/SP1.csv")  
> data <- read.csv("https://www.football-data.co.uk/mmz4281/1920/SP1.csv")  
> |
```



Ejercicio 2: Del data frame que resulta de importar los datos a R, extrae las columnas que contienen los números de goles anotados por los equipos que jugaron en casa (FTHG) y los goles anotados por los equipos que jugaron como visitante (FTAG)

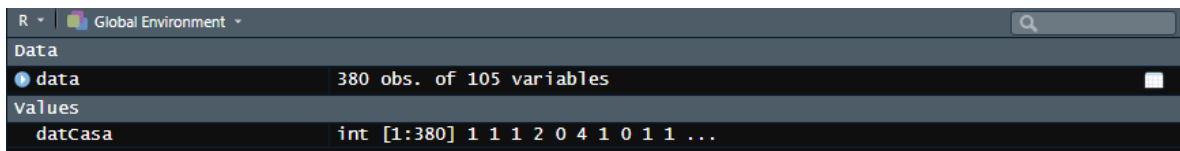
```
#<variable> <- <DataFrame>$<columna>
```

#La variable almacenara los datos los cuales se extraen de la
#<columna> del <DataFrame> que se obtienen mediante el símbolo \$.

```
datCasa <- data$FTHG
```

```
datCasa<-data$FTHG
```

```
> datCasa<-data$FTHG  
> |
```



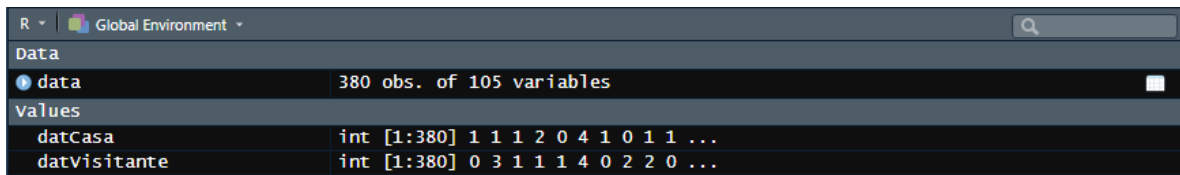
R Global Environment

Data	
data	380 obs. of 105 variables
Values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...

```
datVisitante <- data$FTAG
```

```
datVisitante<-data$FTAG
```

```
> datVisitante<-data$FTAG  
> |
```



R Global Environment

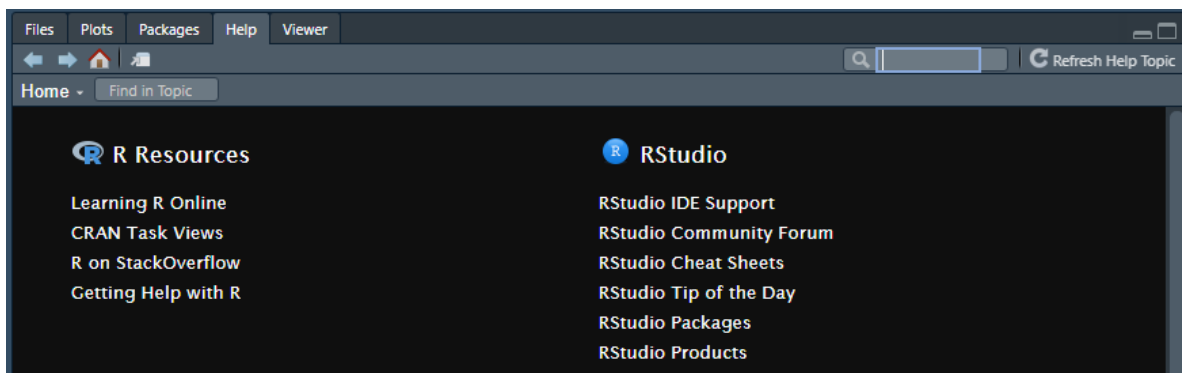
Data	
data	380 obs. of 105 variables
Values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...
datVisitante	int [1:380] 0 3 1 1 1 4 0 2 2 0 ...

Ejercicio 3: Consulta cómo funciona la función table en R al ejecutar en la consola ?table

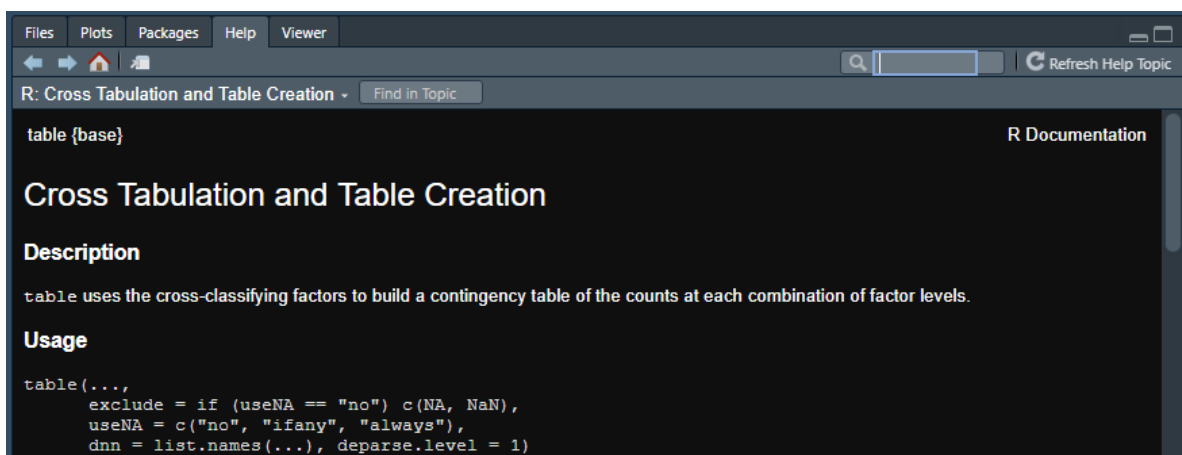
#?<método>

#Esta es una de las formas de poder acceder a la ayuda que se ofrece
#para saber el funcionamiento del <método> tanto como lo que realiza
#como los parámetros que se requieran para utilizarlo.

?table



```
> ?table
> |
```



Ejercicio 4: Posteriormente elabora tablas de frecuencias relativas para estimar las siguientes probabilidades:

Parte 4.1: La probabilidad (marginal) de que el equipo que juega en casa anote x goles ($x = 0, 1, 2, \dots$)

```
#(<variable> <- table(<vector>))
```

```
#El método table va a convertir los datos del vector en una matriz la se
#termina almacenando en la variable para su posterior manipulación
#/y/o consulta, al estar encerrados entre paréntesis al final de
#almacenar los datos en la variable se terminarán consultando.
```

```
(tablaCasa<-table(datCasa))
```

```
(tablaCasa<-table(datCasa))
```

```
> (tablaCasa<-table(datCasa))
datCasa
  0    1    2    3    4    5    6
88 132  99  38  14    8    1
> |
```

Environment	History	Connections	Tutorial
Import Dataset 112 MiB			
R Global Environment			
Data			
data 380 obs. of 105 variables			
Values			
datCasa	int [1:380]	1 1 1 2 0 4 1 0 1 1 ...	
datVisitante	int [1:380]	0 3 1 1 1 4 0 2 2 0 ...	
tablaCasa	'table' int [1:7(1d)]	88 132 99 38 14 8 1	

```
(totalGoles<-table(datVisitante,datCasa))
```

```
(totalGoles<-table(datvisitante,datCasa))
```

```
> (totalGoles<-table(datvisitante,datCasa))
      datCasa
datvisitante  0  1  2  3  4  5  6
  0  33 43 39 14  4  2  1
  1  28 49 35 14  5  3  0
  2  15 32 20  7  4  3  0
  3   8  5  3  2  0  0  0
  4   2  3  2  1  1  0  0
  5   2  0  0  0  0  0  0
> |
```

Environment	History	Connections	Tutorial
Import Dataset 112 MiB			
R Global Environment			
Data			
data 380 obs. of 105 variables			
Values			
datCasa	int [1:380]	1 1 1 2 0 4 1 0 1 1 ...	
datvisitante	int [1:380]	0 3 1 1 1 4 0 2 2 0 ...	
tablaCasa	'table' int [1:7(1d)]	88 132 99 38 14 8 1	
totalGoles	'table' int [1:6, 1:7]	33 28 15 8 2 2 43 49 32 5 ...	

```
#(<variable> <- sum(<table>))
```

#El método sum va retornar la suma de todos los elementos
#contenidos en la <table>.

```
(totalFrecAbsoluta <- sum(totalGoles))
```

```
(totalFrecAbsoluta <- sum(totalGoles))
```

```
> (totalFrecAbsoluta <- sum(totalGoles))  
[1] 380  
>
```

R Global Environment	
Data	
data	380 obs. of 105 variables
values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...
datVisitante	int [1:380] 0 3 1 1 1 4 0 2 2 0 ...
tablaCasa	'table' int [1:7(1d)] 88 132 99 38 14 8 1
totalFrecAbsoluta	380L
totalGoles	'table' int [1:6, 1:7] 33 28 15 8 2 2 43 49 32 5 ...

```
#(<variable> <- round(<table>/<int>,<int>))
```

#El método round recibe 2 parámetros el primero es una tabla que
#contiene la cantidad de goles por columna el cual se divide en el
#número total de goles y el segundo parámetro es para limitar el
#número de decimales a imprimir por lo que round retornara una tabla
#con los resultados por columna limitados al numero de decimales
#colocados en el segundo parámetro.

```
(FrecRelCasa <- round (tablaCasa/totalFrecAbsoluta,4))
```

```
(FrecRelCasa<-round (tablaCasa/totalFrecAbsoluta,4))
```

```
> (FrecRelCasa<-round (tablaCasa/totalFrecAbsoluta,4))  
datCasa  
      0      1      2      3      4      5      6  
0.2316 0.3474 0.2605 0.1000 0.0368 0.0211 0.0026  
>
```

R Global Environment	
Data	
data	380 obs. of 105 variables
values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...
datVisitante	int [1:380] 0 3 1 1 1 4 0 2 2 0 ...
FrecRelCasa	'table' num [1:7(1d)] 0.2316 0.3474 0.2605 0.1 0.0368 ...
tablaCasa	'table' int [1:7(1d)] 88 132 99 38 14 8 1
totalFrecAbsoluta	380L
totalGoles	'table' int [1:6, 1:7] 33 28 15 8 2 2 43 49 32 5 ...

Parte 4.2: La probabilidad (marginal) de que el equipo que juega como visitante anote y goles ($y = 0, 1, 2, \dots$)

```
(tablaCasa <- table(datVisitante))
```

```
(tablaCasa<-table(datvisitante))  
datvisitante  
  0   1   2   3   4   5  
136 134  81  18   9   2  
> |
```

R - Global Environment	
Data	
data	380 obs. of 105 variables
Values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...
datVisitante	int [1:380] 0 3 1 1 1 4 0 2 2 0 ...
FrecRelCasa	'table' num [1:7(1d)] 0.2316 0.3474 0.2605 0.1 0.0368 ...
tablaCasa	'table' int [1:6(1d)] 136 134 81 18 9 2
totalFrecAbsoluta	380L
totalGoles	'table' int [1:6, 1:7] 33 28 15 8 2 2 43 49 32 5 ...

```
(FrecRelVisitante <- round(tablaCasa/totalFrecAbsoluta,4))
```

```
(FrecRelVisitante<-round (tablaCasa/totalFrecAbsoluta,4))  
> (FrecRelVisitante<-round (tablaCasa/totalFrecAbsoluta,4))  
datvisitante  
  0   1   2   3   4   5  
0.3579 0.3526 0.2132 0.0474 0.0237 0.0053  
> |
```

R - Global Environment	
Data	
data	380 obs. of 105 variables
Values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...
datVisitante	int [1:380] 0 3 1 1 1 4 0 2 2 0 ...
FrecRelCasa	'table' num [1:7(1d)] 0.2316 0.3474 0.2605 0.1 0.0368 ...
FrecRelVisitante	'table' num [1:6(1d)] 0.3579 0.3526 0.2132 0.0474 0.0237 ...
tablaCasa	'table' int [1:6(1d)] 136 134 81 18 9 2
totalFrecAbsoluta	380L
totalGoles	'table' int [1:6, 1:7] 33 28 15 8 2 2 43 49 32 5 ...

Parte 4.3: La probabilidad (conjunta) de que el equipo que juega en casa anote x goles y el equipo que juega como visitante anote y goles ($x = 0, 1, 2, \dots, y = 0, 1, 2, \dots$)

```
(FrecRelCon <- round(totalGoles/totalFrecAbsoluta,4))
```

```
(FrecRelCon<-round(totalGoles/totalFrecAbsoluta,4))
```

```
> (FrecRelCon<-round(totalGoles/totalFrecAbsoluta,4))
      datCasa
datvisitante  0    1    2    3    4    5    6
      0 0.0868 0.1132 0.1026 0.0368 0.0105 0.0053 0.0026
      1 0.0737 0.1289 0.0921 0.0368 0.0132 0.0079 0.0000
      2 0.0395 0.0842 0.0526 0.0184 0.0105 0.0079 0.0000
      3 0.0211 0.0132 0.0079 0.0053 0.0000 0.0000 0.0000
      4 0.0053 0.0079 0.0053 0.0026 0.0026 0.0000 0.0000
      5 0.0053 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
> |
```

R Global Environment	
Data	
data	380 obs. of 105 variables
Values	
datCasa	int [1:380] 1 1 1 2 0 4 1 0 1 1 ...
datVisitante	int [1:380] 0 3 1 1 1 4 0 2 2 0 ...
FrecRelCasa	'table' num [1:7(1d)] 0.2316 0.3474 0.2605 0.1 0.0368 ...
FrecRelCon	'table' num [1:6, 1:7] 0.0868 0.0737 0.0395 0.0211 0.0053 ...
FrecRelVisitante	'table' num [1:6(1d)] 0.3579 0.3526 0.2132 0.0474 0.0237 ...
tablacasa	'table' int [1:6(1d)] 136 134 81 18 9 2
totalFrecAbsoluta	380L
totalGoles	'table' int [1:6, 1:7] 33 28 15 8 2 2 43 49 32 5 ...

PostWork 2: Programación y manipulación de datos en R

Ejercicio 1: Importa los datos de soccer de las temporadas 2017/2018, 2018/2019 y 2019/2020 de la primera división de la liga española a R, los datos los puedes encontrar en el siguiente enlace: <https://www.football-data.co.uk/spainm.php>

```
#Importamos los datos de soccer de las temporadas 2017/2018, 2018/2019 y
2019/2020 de la primera división de la liga española a R, desde
https://www.football-data.co.uk/spainm.php"
setwd ("C:/Users/arraz/Documents/Bedu_statisticsWithR/spainleague")
#Cambia el directorio al tuyo
```

```
e11920 <- "https://www.football-data.co.uk/mmz4281/1920/SP1.csv"
e11819 <- "https://www.football-data.co.uk/mmz4281/1819/SP1.csv"
e11718 <- "https://www.football-data.co.uk/mmz4281/1718/SP1.csv"
```

```
download.file(url = e11920, destfile = "e1-1920.csv", mode = "wb")
download.file(url = e11819, destfile = "e1-1829.csv", mode = "wb")
download.file(url = e11718, destfile = "e1-1719.csv", mode = "wb")
```

```
dir()
```

```
ligaesp <- lapply(dir(), read.csv) # leemos los archivos descargados
usando la funcion lapply y guardandolos en un dataframe
```

Ejercicio 2: Revisa la estructura de de los data frames al usar las funciones: str, head, View y summary

#Obtenemos una mejor idea de las características de los data frames al usar las funciones: str, head, View y summary

```
str(ligaesp[[1]]); str(ligaesp[[2]]); str(ligaesp[[3]])
head(ligaesp[[1]]); head(ligaesp[[2]]); head(ligaesp[[3]])
View(ligaesp[[1]]); View(ligaesp[[2]]); View(ligaesp[[3]])
summary(ligaesp[[1]]); summary(ligaesp[[2]]); summary(ligaesp[[3]])
```

Showing 1 to 9 of 380 entries, 64 total columns

Div	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	HST	AST	HF	AF	HC	AC	HY	AY	HR	AR
1	SP1	18/08/17	Leganes	Alaves	1	0	H	1	0	H	16	6	9	3	14	18	4	2	0	1	0
2	SP1	18/08/17	Valencia	Las Palmas	1	0	H	1	0	H	22	5	6	4	25	13	5	2	3	3	0
3	SP1	19/08/17	Celta	Sociedad	2	3	A	1	1	D	16	13	5	6	12	11	5	4	3	1	0
4	SP1	19/08/17	Girona	Ath Madrid	2	2	D	2	0	H	13	9	6	3	15	15	6	0	2	4	0
5	SP1	19/08/17	Sevilla	Espanol	1	1	D	1	1	D	9	9	4	6	14	12	7	3	2	4	1
6	SP1	20/08/17	Ath Bilbao	Getafe	0	0	D	0	0	D	12	8	2	2	16	15	7	6	1	3	0
7	SP1	20/08/17	Barcelona	Betis	2	0	H	2	0	H	15	3	2	0	16	15	8	0	2	1	0
8	SP1	20/08/17	La Coruna	Real Madrid	0	3	A	0	2	A	12	16	6	8	16	12	4	4	5	1	0
9	SP1	21/08/17	Levante	Villarreal	1	0	H	0	0	D	14	9	3	1	18	14	11	6	1	3	0

Console output:

```
R 4.1.0 - ~/Bedu_statisticsWithR/spainleague/
> View(ligaesp[[1]]); View(ligaesp[[2]]); View(ligaesp[[3]])
> summary(ligaesp[[1]]); summary(ligaesp[[2]]); summary(ligaesp[[3]])
```

Summary statistics for the first three data frames:

Div	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	HST	AST	HF	AF	HC	AC	HY	AY	HR	AR
Length:380	Length:380	Length:380	Length:380	Min.:0.000	Min.:0.000	Length:380	Min.:0.0000	Min.:0.0000	Length:380	Min.:2.00	Min.:1.00	Min.:0.000	Min.:0.000	Min.:4.00	Min.:0.00	Min.:0.000	Min.:0.000	Min.:0.000	Min.:0.000	Min.:0.000	Min.:1.050
Class:character	Class:character	Class:character	Class:character	1st Qu.:0.750	1st Qu.:0.000	Class:character	1st Qu.:0.0000	1st Qu.:0.0000	Class:character	1st Qu.:10.00	1st Qu.:8.00	1st Qu.:3.000	1st Qu.:2.000	1st Qu.:11.00	1st Qu.:11.00	1st Qu.:4.000	1st Qu.:2.000	1st Qu.:1.000	1st Qu.:2.000	1st Qu.:0.0000	1st Qu.:1.617
Mode:character	Mode:character	Mode:character	Mode:character	Median:1.000	Median:1.000	Mode:character	Median:0.0000	Median:0.0000	Mode:character	Median:13.00	Median:10.00	Median:4.500	Median:3.000	Median:13.00	Median:13.00	Median:5.000	Median:3.000	Median:1.000	Median:3.000	Median:0.0000	Median:2.075
				Mean:1.547	Mean:1.147		Mean:0.6605	Mean:0.4868		Mean:13.53	Mean:10.47	Mean:4.758	Mean:3.805	Mean:13.73	Mean:13.73	Mean:5.613	Mean:4.192	Mean:2.339	Mean:2.676	Mean:0.1105	Mean:2.777
				3rd Qu.:2.000	3rd Qu.:2.000		3rd Qu.:1.0000	3rd Qu.:1.0000		3rd Qu.:16.00	3rd Qu.:13.00	3rd Qu.:6.000	3rd Qu.:5.000	3rd Qu.:17.00	3rd Qu.:17.00	3rd Qu.:5.000	3rd Qu.:3.000	3rd Qu.:1.000	3rd Qu.:3.000	3rd Qu.:0.0000	3rd Qu.:2.075
				Max.:7.000	Max.:6.000		Max.:5.0000	Max.:3.0000		Max.:30.00	Max.:24.00	Max.:14.000	Max.:13.000	Max.:29.00	Max.:29.00	Max.:5.613	Max.:4.192	Max.:2.339	Max.:2.676	Max.:0.1105	Max.:2.777

Ejercicio 3: Con la función select del paquete dplyr selecciona únicamente las columnas Date, HomeTeam, AwayTeam, FTHG, FTAG y FTR; esto para cada uno de los data frames. (Hint: también puedes usar lapply).

#Con la función select del paquete dplyr seleccionamos las columnas Date, HomeTeam, AwayTeam, FTHG, FTAG y FTR para cada data frame.

```
ligaesp <- lapply(ligaesp, select, c("Date", "HomeTeam", "AwayTeam",
"FTHG", "FTAG", "FTR"))
```

	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR
1	18/08/17	Leganes	Alaves	1	0	H
2	18/08/17	Valencia	Las Palmas	1	0	H
3	19/08/17	Celta	Sociedad	2	3	A
4	19/08/17	Girona	Ath Madrid	2	2	D
5	19/08/17	Sevilla	Espanol	1	1	D
6	20/08/17	Ath Bilbao	Getafe	0	0	D
7	20/08/17	Barcelona	Betis	2	0	H
8	20/08/17	La Coruna	Real Madrid	0	3	A
9	21/08/17	Levante	Villarreal	1	0	H
10	21/08/17	Malaga	Eibar	0	1	A
11	25/08/17	Betis	Celta	2	1	H
12	25/08/17	Sociedad	Villarreal	3	0	H
13	26/08/17	Alaves	Barcelona	0	2	A

Ejercicio 4: Asegúrate de que los elementos de las columnas correspondientes de los nuevos data frames sean del mismo tipo (Hint 1: usa `as.Date` y `mutate` para arreglar las fechas). Con ayuda de la función `rbind` forma un único data frame que contenga las seis columnas mencionadas en el punto 3 (Hint 2: la función `do.call` podría ser utilizada).

"Aseguramos de que los elementos de las columnas correspondientes de los nuevos data frames sean del mismo tipo
usamos `as.Date` y `mutate` para arreglar las fechas"

```
ligaesp[[1]] <- mutate(ligaesp[[1]], Date = as.Date(Date,
format="%d/%m/%y"))
ligaesp[[2]] <- mutate(ligaesp[[2]], Date = as.Date(Date, "%d/%m/%Y"))
ligaesp[[3]] <- mutate(ligaesp[[3]], Date = as.Date(Date, "%d/%m/%Y"))
```

"Con la función `rbind` y `do.call` formamos un único data frame que contenga las seis columnas mencionadas en el punto 3"

```
data <- do.call(rbind, ligaesp)
head(data)
dim(data)
```

```
> head(data)
      Date HomeTeam AwayTeam FTHG FTAG FTR
1 2017-08-18   Leganes    Alaves     1     0   H
2 2017-08-18 Valencia Las Palmas     1     0   H
3 2017-08-19    Celta  Sociedad     2     3   A
4 2017-08-19   Girona Ath Madrid     2     2   D
5 2017-08-19   Sevilla Espanol     1     1   D
6 2017-08-20 Ath Bilbao   Getafe     0     0   D
> dim(data)
[1] 1140    6
```

PostWork 3: Análisis Exploratorio de Datos (AED o EDA) con R

Ejercicio 1: Con el último data frame obtenido en el postwork de la sesión 2, elabora tablas de frecuencias relativas para estimar las siguientes probabilidades:

```
df <- read.csv("https://github.com/sh4rkd/Equipo-16-R/raw/master/PostWork-2/csv/total.csv")
```

```
datCasa <- df$FTHG
```

```
datVis <- df$FTAG
```

```
FrecAbs <- table(datVis, datCasa)
```

```
sumaFrecAbs <- sum(FrecAbs)
```

```
df <- read.csv("https://github.com/sh4rkd/Equipo-16-R/raw/master/PostWork-2/csv/total.csv")
datCasa<-df$FTHG
datVis<-df$FTAG
FrecAbs<-table(datVis,datCasa)
sumaFrecAbs<-sum(FrecAbs)
```

```
> df <- read.csv("https://github.com/sh4rkd/Equipo-16-R/raw/master/Postwork-2/csv/total.csv")
> datCasa<-df$FTHG
> datVis<-df$FTAG
> FrecAbs<-table(datVis,datCasa)
> sumaFrecAbs<-sum(FrecAbs)
> |
```

R Global Environment	
Data	
df	1140 obs. of 6 variables
Values	
datCasa	int [1:1140] 1 1 2 2 1 0 2 0 1 0 ...
datVis	int [1:1140] 0 0 3 2 1 0 0 3 0 1 ...
FrecAbs	'table' int [1:7, 1:9] 89 92 52 21 6 5 0 132 131 78 ...
sumaFrecAbs	1140L

Parte 1.1: La probabilidad (marginal) de que el equipo que juega en casa anote x goles (x=0,1,2,)

(ProbCasa <- round(table(datCasa)/sumaFrecAbs,4))

```
(ProbCasa<-round(table(datCasa)/sumaFrecAbs,4))
```

```
> (ProbCasa<-round(table(datCasa)/sumaFrecAbs,4))
datCasa
  0      1      2      3      4      5      6      7      8
0.2325 0.3272 0.2667 0.1123 0.0351 0.0193 0.0053 0.0009 0.0009
> |
```

R Global Environment	
Data	
df	1140 obs. of 6 variables
Values	
datCasa	int [1:1140] 1 1 2 2 1 0 2 0 1 0 ...
datVis	int [1:1140] 0 0 3 2 1 0 0 3 0 1 ...
FrecAbs	'table' int [1:7, 1:9] 89 92 52 21 6 5 0 132 131 78 ...
ProbCasa	'table' num [1:9(1d)] 0.2325 0.3272 0.2667 0.1123 0.0351 ...
sumaFrecAbs	1140L

Parte 1.2: La probabilidad (marginal) de que el equipo que juega como visitante anote y goles (y=0,1,2,)

(ProbVis<-round(table(datVis)/sumaFrecAbs,4))

```
(ProbVis<-round(table(datVis)/sumaFrecAbs,4))
```

```
> (ProbVis<-round(table(datVis)/sumaFrecAbs,4))
datVis
  0      1      2      3      4      5      6
0.3518 0.3404 0.2123 0.0544 0.0289 0.0096 0.0026
> |
```

R Global Environment	
Data	
df	1140 obs. of 6 variables
Values	
datCasa	int [1:1140] 1 1 2 2 1 0 2 0 1 0 ...
datVis	int [1:1140] 0 0 3 2 1 0 0 3 0 1 ...
FrecAbs	'table' int [1:7, 1:9] 89 92 52 21 6 5 0 132 131 78 ...
ProbCasa	'table' num [1:9(1d)] 0.2325 0.3272 0.2667 0.1123 0.0351 ...
ProbVis	'table' num [1:7(1d)] 0.3518 0.3404 0.2123 0.0544 0.0289 ...
sumaFrecAbs	1140L

Parte 1.3: La probabilidad (conjunta) de que el equipo que juega en casa anote x goles y el equipo que juega como visitante anote y goles (x=0,1,2,, y=0,1,2,)

(FrecRel<-round(FrecAbs/sumaFrecAbs,4))

```
(FrecRel<-round(FrecAbs/sumaFrecAbs,4))
```

```
> (FrecRel<-round(FrecAbs/sumaFrecAbs,4))
      datCasa
datvis  0      1      2      3      4      5      6      7      8
0 0.0781 0.1158 0.0877 0.0447 0.0140 0.0088 0.0026 0.0000 0.0000
1 0.0807 0.1149 0.0939 0.0325 0.0105 0.0053 0.0018 0.0009 0.0000
2 0.0456 0.0684 0.0614 0.0246 0.0070 0.0044 0.0000 0.0000 0.0009
3 0.0184 0.0175 0.0114 0.0061 0.0000 0.0000 0.0009 0.0000 0.0000
4 0.0053 0.0088 0.0088 0.0018 0.0035 0.0009 0.0000 0.0000 0.0000
5 0.0044 0.0018 0.0018 0.0018 0.0000 0.0000 0.0000 0.0000 0.0000
6 0.0000 0.0000 0.0018 0.0009 0.0000 0.0000 0.0000 0.0000 0.0000
>
```

R Global Environment	
Data	
df	1140 obs. of 6 variables
Values	
datCasa	int [1:1140] 1 1 2 2 1 0 2 0 1 0 ...
datVis	int [1:1140] 0 0 3 2 1 0 0 3 0 1 ...
FrecAbs	'table' int [1:7, 1:9] 89 92 52 21 6 5 0 132 131 78 ...
FrecRel	'table' num [1:7, 1:9] 0.0781 0.0807 0.0456 0.0184 0.0053 ...
ProbCasa	'table' num [1:9(1d)] 0.2325 0.3272 0.2667 0.1123 0.0351 ...
ProbVis	'table' num [1:7(1d)] 0.3518 0.3404 0.2123 0.0544 0.0289 ...
sumaFrecAbs	1140L

Ejercicio 2: Realiza lo siguiente:

library(ggplot2)

```
library(ggplot2)
```

```
> library(ggplot2)
>
```

	Name	Description	Version	
<input checked="" type="checkbox"/>	ggplot2	Create Elegant Data Visualisations Using the Grammar of Graphics	3.3.4	

Parte 2.1: Un gráfico de barras para las probabilidades marginales estimadas del número de goles que anota el equipo de casa.

#barplot(<table>,<mensaje en eje x>,<mensaje en eje y>,<mensaje en #la cabecera>,<vector con valor inicial en la primera posición y valor

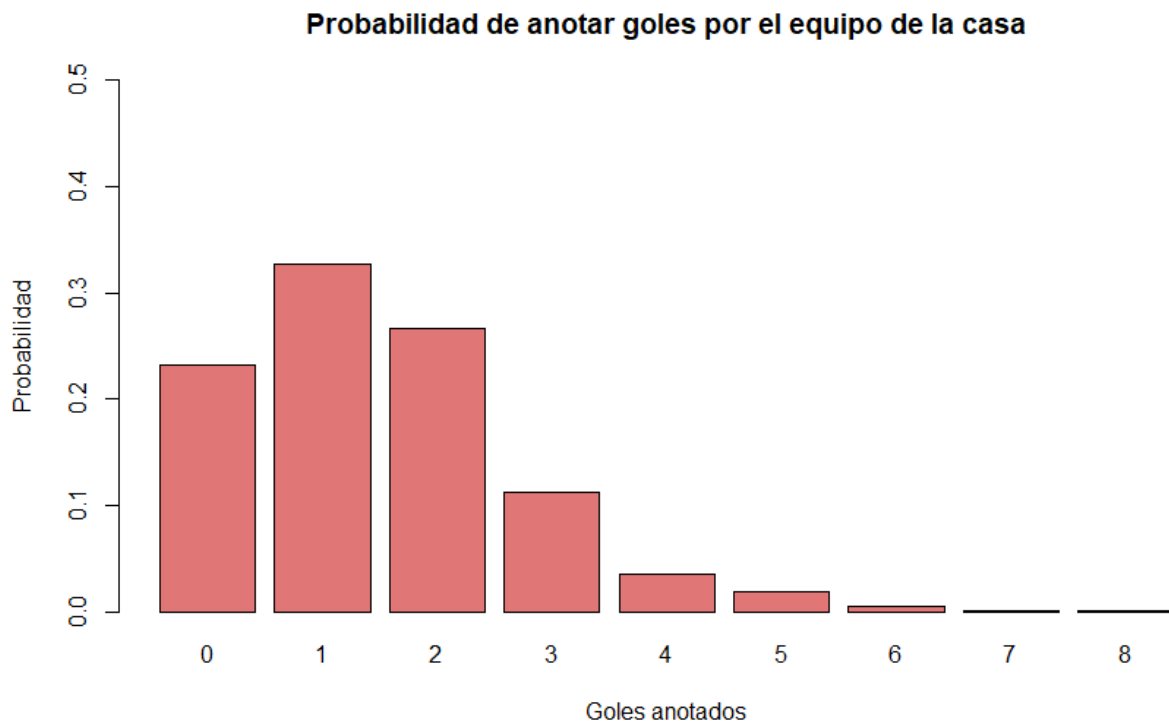
#final en la segunda posición, para determinar el tamaño de altura
#máximo para las barras>, <color en valor rgba>)

#el método barplot recibirá los parámetros anteriormente mencionados
#para graficar en forma de barra del color los elementos y medidas
#recibidas por parámetro.

```
barplot(ProbCasa,xlab="Goles anotados", ylab="Probabilidad",  
        main = "Probabilidad de anotar goles por el equipo de la casa",  
        ylim =c(0,0.5),  
        col = rgb(0.8,0.1,0.1,0.6))
```

```
barplot(ProbCasa,xlab="Goles anotados", ylab="Probabilidad",  
        main = "Probabilidad de anotar goles por el equipo de la casa",  
        ylim =c(0,0.5),  
        col = rgb(0.8,0.1,0.1,0.6))
```

```
> barplot(ProbCasa,xlab="Goles anotados", ylab="Probabilidad",  
+         main = "Probabilidad de anotar goles por el equipo de la casa",  
+         ylim =c(0,0.5),  
+         col = rgb(0.8,0.1,0.1,0.6))  
>
```

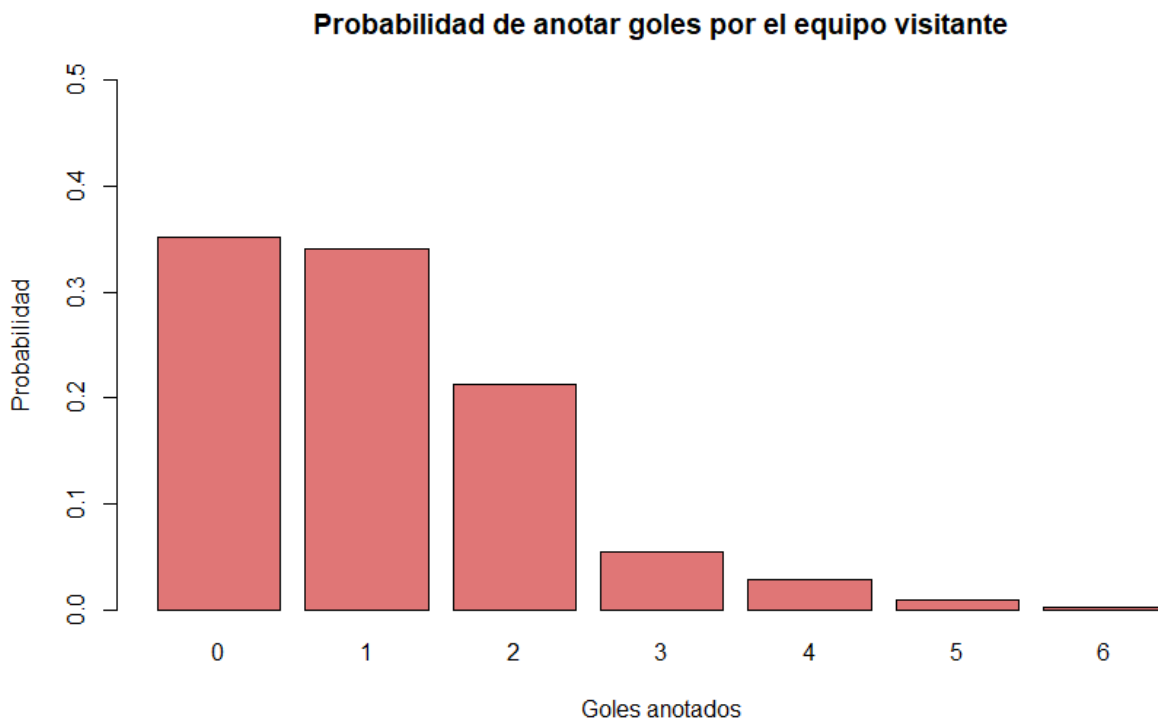


Parte 2.2: Un gráfico de barras para las probabilidades marginales estimadas del número de goles que anota el equipo visitante.

```
barplot(ProbVis,xlab="Goles anotados", ylab="Probabilidad",  
        main = "Probabilidad de anotar goles por el equipo visitante",  
        ylim =c(0,0.5),  
        col = rgb(0.8,0.1,0.1,0.6))
```

```
barplot(ProbVis,xlab="Goles anotados", ylab="Probabilidad",  
        main = "Probabilidad de anotar goles por el equipo visitante",  
        ylim =c(0,0.5),  
        col = rgb(0.8,0.1,0.1,0.6))
```

```
> barplot(ProbVis,xlab="Goles anotados", ylab="Probabilidad",  
+         main = "Probabilidad de anotar goles por el equipo visitante",  
+         ylim =c(0,0.5),  
+         col = rgb(0.8,0.1,0.1,0.6))  
>
```

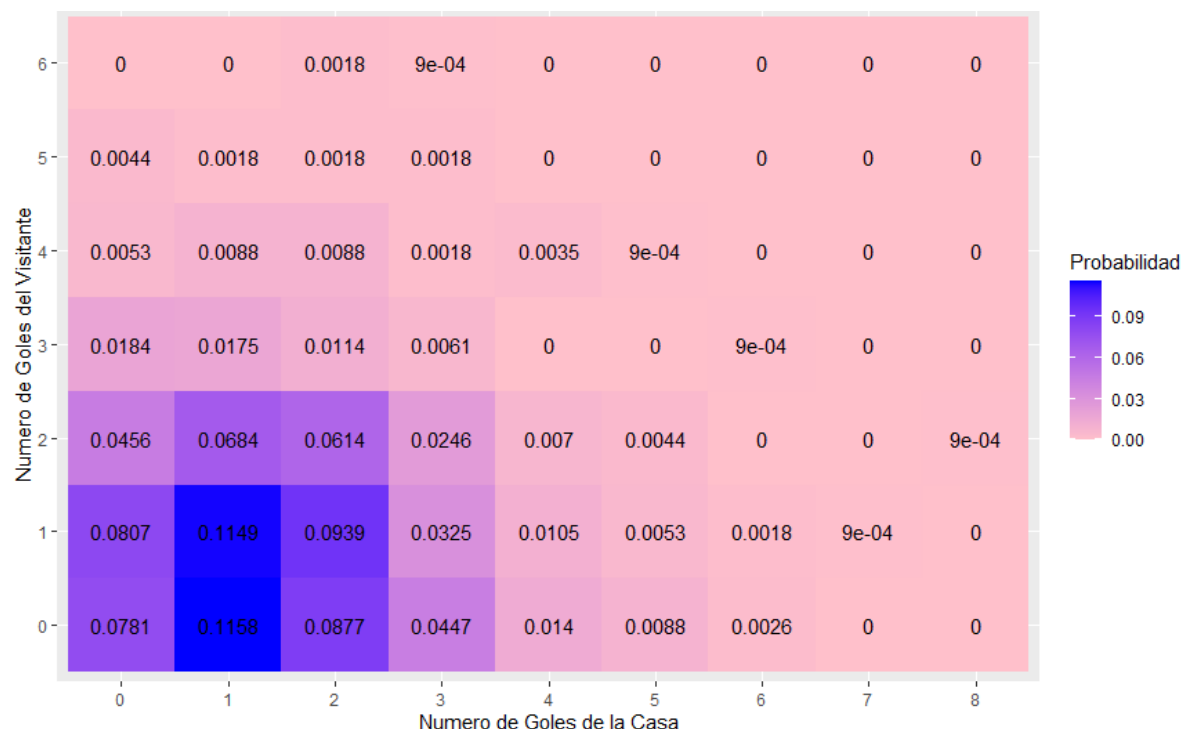


Parte 2.3: Un HeatMap para las probabilidades conjuntas estimadas de los números de goles que anotan el equipo de casa y el equipo visitante en un partido.

```
ggplot(as.data.frame(FrecRel), aes(x=datCasa, y=datVis, fill = Freq)) +
  geom_tile()+
  geom_text(aes(label=round(Freq,4))) +
  scale_fill_gradient(low="pink", high="blue") + labs(x="Numero de Goles
de la Casa") + labs(y="Numero de Goles del Visitante")+
  labs(fill="Probabilidad")
```

```
ggplot(as.data.frame(FrecRel), aes(x=datCasa, y=datVis, fill = Freq))+
  geom_tile()+
  geom_text(aes(label=round(Freq,4)))+
  scale_fill_gradient(low="pink", high="blue") +
  labs(x="Numero de Goles de la Casa")+
  labs(y="Numero de Goles del Visitante")+
  labs(fill="Probabilidad")
```

```
> ggplot(as.data.frame(FrecRel), aes(x=datCasa, y=datVis, fill = Freq))+
+   geom_tile()+
+   geom_text(aes(label=round(Freq,4)))+
+   scale_fill_gradient(low="pink", high="blue") +
+   labs(x="Numero de Goles de la Casa")+
+   labs(y="Numero de Goles del visitante")+
+   labs(fill="Probabilidad")
>
```



PostWork 4: Algunas distribuciones, teorema central del límite y contraste de hipótesis

```
library(dplyr)
library(ggplot2)
df <- read.csv("https://github.com/sh4rkd/Equipo-16-R/raw/master/Postwork-2/csv/total.csv")
datCasa<-df$FTHG
datVis<-df$FTAG
FrecAbs<-table(datVis,datCasa)
sumaFrecAbs<-sum(FrecAbs)
FrecRel<-round(FrecAbs/sumaFrecAbs,4)
ProbCasa<-round(table(datCasa)/sumaFrecAbs,4)
ProbVis<-round(table(datVis)/sumaFrecAbs,4)
listaCasa <- as.vector(ProbCasa)
listavis <- as.vector(ProbVis)
```

```
> library(dplyr)
> library(ggplot2)
> df <- read.csv("https://github.com/sh4rkd/Equipo-16-R/raw/master/Postwork-2/csv/total.csv")
> datCasa<-df$FTHG
> datVis<-df$FTAG
> FrecAbs<-table(datVis,datCasa)
> sumaFrecAbs<-sum(FrecAbs)
> FrecRel<-round(FrecAbs/sumaFrecAbs,4)
> ProbCasa<-round(table(datCasa)/sumaFrecAbs,4)
> ProbVis<-round(table(datVis)/sumaFrecAbs,4)
> listaCasa <- as.vector(ProbCasa)
> listavis <- as.vector(ProbVis)
> |
```

R Global Environment	
Data	
df	1140 obs. of 6 variables
Values	
datCasa	int [1:1140] 1 1 2 2 1 0 2 0 1 0 ...
datVis	int [1:1140] 0 0 3 2 1 0 0 3 0 1 ...
FrecAbs	'table' int [1:7, 1:9] 89 92 52 21 6 5 0 132 131 78 ...
FrecRel	'table' num [1:7, 1:9] 0.0781 0.0807 0.0456 0.0184 0.0053 ...
listaCasa	num [1:9] 0.2325 0.3272 0.2667 0.1123 0.0351 ...
listavis	num [1:7] 0.3518 0.3404 0.2123 0.0544 0.0289 ...
ProbCasa	'table' num [1:9(1d)] 0.2325 0.3272 0.2667 0.1123 0.0351 ...
ProbVis	'table' num [1:7(1d)] 0.3518 0.3404 0.2123 0.0544 0.0289 ...
sumaFrecAbs	1140L

Ejercicio 1: Ya hemos estimado las probabilidades conjuntas de que el equipo de casa anote $X=x$ goles ($x=0,1,\dots,8$), y el equipo visitante anote $Y=y$ goles ($y=0,1,\dots,6$), en un partido. Obtén una tabla de cocientes al dividir estas probabilidades conjuntas por el producto de las probabilidades marginales correspondientes.

```
#<variable> <- matrix(<con que va llenar la matrix>,<número de  
#filas>,<número de columnas>)
```

```
#se crea una matriz de un tamaño en especifico que se rellena con 0
```

```
#Matriz[i,j] = round(<vector en la posición i> * <vector en la posición  
#j>,<número de decimales>)
```

```
#Se recorre la matriz para irle almacenando el resultado de la operación  
#que es el recorrido de dos vectores, limitándole a 4 decimales
```

```
M <- matrix(0,nrow = length(listaCasa), ncol = length(listaVis))
```

```
for (i in 1:9) {  
  for (j in 1:7) {  
    M[i,j] = round(listaCasa[i]*listaVis[j],4)  
  }  
}
```

```
(ProbCon<-as.matrix(FrecRel))
```

```
M <- matrix(0,nrow = length(listaCasa), ncol = length(listaVis))  
for (i in 1:9) {  
  for (j in 1:7) {  
    M[i,j] = round(listaCasa[i]*listaVis[j],4)  
  }  
}  
(ProbCon<-as.matrix(FrecRel))
```

```
> (ProbCon<-as.matrix(FrecRel))  
      datCasa  
datVis 0      1      2      3      4      5      6      7      8  
0 0.0781 0.1158 0.0877 0.0447 0.0140 0.0088 0.0026 0.0000 0.0000  
1 0.0807 0.1149 0.0939 0.0325 0.0105 0.0053 0.0018 0.0009 0.0000  
2 0.0456 0.0684 0.0614 0.0246 0.0070 0.0044 0.0000 0.0000 0.0009  
3 0.0184 0.0175 0.0114 0.0061 0.0000 0.0000 0.0009 0.0000 0.0000  
4 0.0053 0.0088 0.0088 0.0018 0.0035 0.0009 0.0000 0.0000 0.0000  
5 0.0044 0.0018 0.0018 0.0018 0.0000 0.0000 0.0000 0.0000 0.0000  
6 0.0000 0.0000 0.0018 0.0009 0.0000 0.0000 0.0000 0.0000 0.0000  
> |
```

```
(ProbCon<-t(ProbCon))
```

```
(ProbCon<-t(ProbCon))
```

```
> (ProbCon<-t(ProbCon))
      datVis
datCasa 0      1      2      3      4      5      6
0 0.0781 0.0807 0.0456 0.0184 0.0053 0.0044 0.0000
1 0.1158 0.1149 0.0684 0.0175 0.0088 0.0018 0.0000
2 0.0877 0.0939 0.0614 0.0114 0.0088 0.0018 0.0018
3 0.0447 0.0325 0.0246 0.0061 0.0018 0.0018 0.0009
4 0.0140 0.0105 0.0070 0.0000 0.0035 0.0000 0.0000
5 0.0088 0.0053 0.0044 0.0000 0.0009 0.0000 0.0000
6 0.0026 0.0018 0.0000 0.0009 0.0000 0.0000 0.0000
7 0.0000 0.0009 0.0000 0.0000 0.0000 0.0000 0.0000
8 0.0000 0.0000 0.0009 0.0000 0.0000 0.0000 0.0000
```

Ejercicio 2: Mediante un procedimiento de bootstrap, obtén más cocientes similares a los obtenidos en la tabla del punto anterior. Esto para tener una idea de las distribuciones de la cual vienen los cocientes en la tabla anterior. Menciona en cuáles casos le parece razonable suponer que los cocientes de la tabla en el punto 1, son iguales a 1 (en tal caso tendríamos independencia de las variables aleatorias X y Y).

#Limpieza de Datos para obtener la media, cambiamos infly y NAN por la media

```
IndepEst<-(round(M/ProbCon,3))
```

```
(IndepEst<-as.data.frame(IndepEst))
```

```
IndepEst<-(round(M/ProbCon,3))
(IndepEst<-as.data.frame(IndepEst))
```

```
> IndepEst<-(round(M/ProbCon,3))
> (IndepEst<-as.data.frame(IndepEst))
```

```
      datCasa datVis Freq
1          0      0 1.047
2          1      0 0.994
3          2      0 1.070
4          3      0 0.884
5          4      0 0.879
6          5      0 0.773
7          6      0 0.731
8          7      0  Inf
9          8      0  Inf
10         0      1 0.980
```

#El metodo mutate_if se utiliza para hacer transformaciones a varias columnas de una vez.

```
IndepEst <- IndepEst %>% mutate_if(is.numeric, function(x)
ifelse(is.infinite(x), 0, x))
```

#El metodo which retornara la posición.

```
IndepEst$Freq[which(is.na(IndepEst$Freq))]<-
mean(IndepEst$Freq,na.rm = TRUE)
```

```
IndepEst$Freq[IndepEst$Freq == 0]<-mean(IndepEst$Freq)
```

```
media = mean(IndepEst$Freq)
```

```
varianza = var(IndepEst$Freq)
```

```
IndepEst <- IndepEst %>% mutate_if(is.numeric, function(x) ifelse(is.infinite(x), 0, x))
IndepEst$Freq[which(is.na(IndepEst$Freq))]<-mean(IndepEst$Freq,na.rm = TRUE)
IndepEst$Freq[IndepEst$Freq == 0]<-mean(IndepEst$Freq)
media = mean(IndepEst$Freq)
varianza = var(IndepEst$Freq)
```

```
> IndepEst <- IndepEst %>% mutate_if(is.numeric, function(x) ifelse(is.infinite(x), 0, x))
> IndepEst$Freq[which(is.na(IndepEst$Freq))]<-mean(IndepEst$Freq,na.rm = TRUE)
> IndepEst$Freq[IndepEst$Freq == 0]<-mean(IndepEst$Freq)
> media = mean(IndepEst$Freq)
> varianza = var(IndepEst$Freq)
> |
```

#Bootstrap muestreo con reemplazo Suponiendo que la muestra de

#La frecuencia viene de una distribución normal

#set.seed(n) hace un setter a una semilla del parámetro n

```
set.seed(1200)
```

```
muestra<- rnorm(n=200,mean=media, sd=sqrt(varianza))
```

```
xbarra = mean(muestra)
```

```
bootstrap <- replicate(n=1000, sample(muestra,replace=TRUE))
```

```
set.seed(1200)
muestra<- rnorm(n=200,mean=media, sd=sqrt(varianza))
xbarra = mean(muestra)
bootstrap <- replicate(n=1000, sample(muestra,replace=TRUE))
```

```
> set.seed(1200)
> muestra<- rnorm(n=200,mean=media, sd=sqrt(varianza))
> xbarra = mean(muestra)
> bootstrap <- replicate(n=1000, sample(muestra,replace=TRUE))
```

(medias<-apply(bootstrap, MARGIN = 2, FUN = mean))

```
(medias<-apply(bootstrap, MARGIN = 2, FUN = mean))
[1] 0.8239870 0.8491317 0.8179948 0.8170085 0.8565071 0.8459122 0.8289991 0.8047890 0.8273591 0.8516189 0.8221616
[12] 0.8358944 0.8400116 0.7995695 0.8680041 0.8178539 0.8108257 0.8065931 0.8469422 0.8182778 0.8548632 0.8610127
[23] 0.7913689 0.7854748 0.8705484 0.8206771 0.8318013 0.8206019 0.8799975 0.8449874 0.8140751 0.8426678 0.8180335
[34] 0.8135795 0.7868157 0.7927643 0.8169684 0.8557421 0.8273480 0.8145357 0.7996100 0.8437083 0.7875638 0.8809075
[45] 0.8404129 0.8363718 0.7971722 0.8317357 0.8590528 0.8313809 0.7985908 0.7939713 0.8231289 0.8339677 0.8707326
[56] 0.8490283 0.8114916 0.8340093 0.8240180 0.7880141 0.8268408 0.8425345 0.8042721 0.8597115 0.8345809 0.8320059
[67] 0.8545982 0.7543932 0.8339795 0.8510087 0.7770766 0.8230852 0.7858326 0.8007203 0.8579565 0.8646555 0.8245345
[78] 0.8508792 0.8531833 0.8187505 0.7935874 0.8346081 0.8347130 0.8255660 0.8521533 0.8123087 0.8282939 0.7943142
[89] 0.8566851 0.8164182 0.7911842 0.8060182 0.7973936 0.8555633 0.8190714 0.8160339 0.8633471 0.7961734 0.8148563
[100] 0.7969375 0.8044236 0.8320057 0.8030922 0.8073002 0.7822468 0.8076351 0.8265842 0.8202542 0.8021087 0.8696692

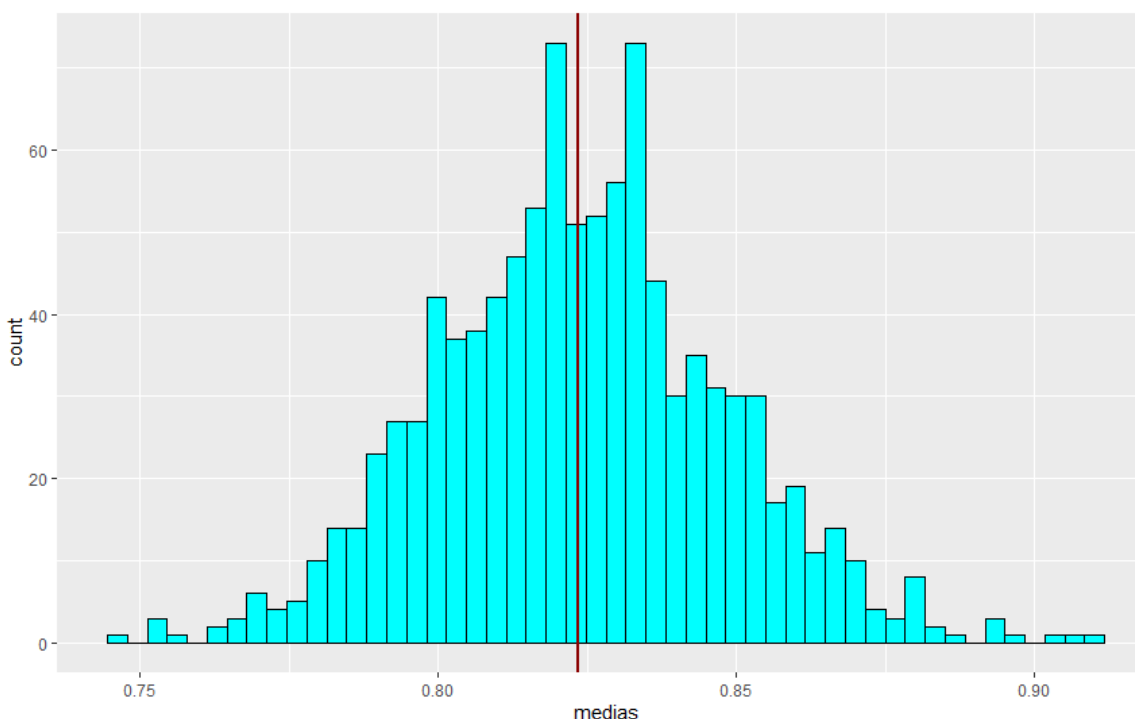
[947] 0.8318385 0.8480680 0.8281335 0.8098290 0.8190818 0.8188554 0.8096497 0.8288187 0.8506105 0.8050777 0.8183061
[958] 0.8306207 0.8410849 0.8250607 0.8522610 0.8654068 0.8523087 0.8282420 0.7994577 0.7953669 0.8315454 0.7855416
[969] 0.8286798 0.8215860 0.8122233 0.8414897 0.8215862 0.8594873 0.8532613 0.8079408 0.8012412 0.8373048 0.8365344
[980] 0.8452870 0.8037825 0.8281180 0.8451673 0.7926196 0.8219382 0.8147347 0.8752732 0.8414567 0.8271528 0.7943896
[991] 0.8451241 0.8172361 0.8221437 0.8409569 0.8060172 0.7989757 0.7682803 0.7885321 0.8151387 0.7930628
```

ggplot()+

geom_histogram(aes(x=medias), bins = 50, color="black", fill = "cyan")+
"cyan")+

geom_vline(xintercept = xbarra, size = 1, color="darkred")

```
ggplot()+
  geom_histogram(aes(x=medias), bins = 50, color="black", fill = "cyan")+
  geom_vline(xintercept = xbarra, size = 1, color="darkred")
```



```
(var_estimada = sum((medias-xbarra)^2)/ncol(bootstrap))
```

```
(var_estimada = sum((medias-xbarra)^2)/ncol(bootstrap))
```

```
> (var_estimada = sum((medias-xbarra)^2)/ncol(bootstrap))  
[1] 0.000588671
```

Media

```
media
```

```
> media  
[1] 0.8173721
```

Xbarra

```
xbarra
```

```
> xbarra  
[1] 0.8235147  
> |
```

#Conclusion PostWork 4:

#De las muestras generadas aleatoriamente se puede concluir que al
#valor que tiende la media es a 0.8235 por lo tanto, los cocientes que
#están cercanos a esta relación Se les puede considerar
#independientes estadísticamente hablando. Es decir, si son
#independientes quiere decir que la probabilidad de que el equipo
#visitante anote 6 goles no depende de la probabilidad de que la casa
#anote 1 gol; Sin embargo, en un sentido estricto, aunque la varianza
#muestral haya sido pequeña (0.00058) y la media 0.82. Esto no
#asegura que sean completamente independientes ya que
#estrictamente se requiere que el cociente sea 1.