Trabajo-tema-01

Autor: José Pérez Yázquez

Objetivo

En este trabajo se va a trabajar con datos procedentes de la liga de béisbol de los Estados Unidos recopilados porSean Lahman desde 1871 a nuestros días. Descargue el fichero que aparece como 2014 – comma-delimited version – Updated January 24, 2015 (lahman-csv_2015-01-24.zip) en la dirección: http://seanlahman.com/baseball-archive/statistics/.

NOTA: El fichero descargado tiene algunas variaciones con respecto a lo que se pide en el texto, este tema se comentó y clase. En cualquier caso, lo comento en aquellas partes en las que me he encontrado alguna de estas variaciones

Ejercicio 1

Cargar los ficheros: Master.csv y Batting.csv en los objetos R: master (datos de jugadores) y bateos, usando funciones de paquetes distintos)

```
library(rio)

library(readr)

master = rio::import("Master.csv")

bateos = readr::read_csv("Batting.csv")
```

Apartado (a)

Extraer los nombres de las variables que contienen los dos ficheros.

```
str(master)
```

```
'data.frame': 18589 obs. of  24 variables:
 $ playerID    : chr  "aardsda01" "aaronha01" "aaronto01" "aasedo01" ...
 $ birthYear   : int  1981 1934 1939 1954 1972 1985 1854 1877 1869 1866 ...
 $ birthMonth  : int  12 2 8 9 8 12 11 4 11 10 ...
 $ birthDay    : int  27 5 5 8 25 17 4 15 11 14 ...
 $ birthCountry: chr  "USA" "USA" "USA" "USA" ...
 $ birthState  : chr  "CO" "AL" "AL" "CA" ...
 ...
```

```
str(bateos)
```

```
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':   99846 obs. of  22 variables:
 $ playerID: chr  "abercda01" "addybo01" "allisar01" "allisdo01" ...
 $ yearID  : int  1871 1871 1871 1871 1871 1871 1871 1871 1871 1871 ...
 $ stint   : int  1 1 1 1 1 1 1 1 1 1 ...
 $ teamID  : chr  "TRO" "RC1" "CL1" "WS3" ...
 $ lgID    : chr  NA NA NA NA ...
 $ G       : int  1 25 29 27 25 12 1 31 1 18 ...
 $ AB      : int  4 118 137 133 120 49 4 157 5 86 ...
 ...
```

Apartado (b)

Muestre las primeras 6 filas de los dos objetos R creados.

```
head(master)
```

```
playerID birthYear birthMonth birthDay birthCountry birthState      ...
1 aardsda01      1981         12       27          USA         CO
2 aaronha01      1934          2        5          USA         AL
3 aaronto01      1939          8        5          USA         AL
4  aasedo01      1954          9        8          USA         CA
```

```
5  abadan01      1972           8        25           USA          FL
6  abadfe01      1985          12        17          D.R.   La Romana
```

```
head(bateos)
```

```
playerID yearID stint teamID lgID  G  AB  R  H 2B 3B HR RBI SB CS BB
1 abercda01   1871     1    TRO <NA>  1   4  0  0  0  0  0   0  0  0  0
2  addybo01   1871     1    RC1 <NA> 25 118 30 32  6  0  0  13  8  1  4
3 allisar01   1871     1    CL1 <NA> 29 137 28 40  4  5  0  19  3  1  2
4 allisdo01   1871     1    WS3 <NA> 27 133 28 44 10  2  2  27  1  1  0
5 ansonca01   1871     1    RC1 <NA> 25 120 29 39 11  3  0  16  6  2  2
6 armstbo01   1871     1    FW1 <NA> 12  49  9 11  2  1  0   5  0  1  0
```

NOTAS: Alternativamente podríamos haber usado algo como master[1:6,] y bateos[1:6,]

Apartado (c)

Cree un data.frame que contenga solamente las siguientes variables del objeto master (llámelo: master2):

"lahmanID", "birthCountry", "deathYear", "nameFirst","nameNick", "bats", "finalGame", "playerID", "birthYear", "birthState", "deathCountry" "deathState", "nameLast", "weight","height", "throws", "debut", "college"

Nota: En el fichero descargado no existían las siguientes columnas

lahmanID, nameNick y college

```
master2 = data.frame(master["birthCountry"],master["deathYear"],master["nameFirst"],
master["bats"],master["finalGame"],


master["playerID"],master["birthYear"],master["birthState"],master["deathCountry"],
master["deathState"],


master["nameLast"],master["weight"],master["height"],master["throws"],master["debut"])
```

Apartado (d)

¿De cuántos países distintos hay jugadores de béisbol? Muestre el peso (weight) y la altura (height) de los jugadores de "W.Germany". Represente esos puntos con la ayuda de plot.

Cuenta el numero de paises en los que han nacido los jugadores

```
numero_paises_distintos = length(unique(master$birthCountry))
```

```
[1] 53
```

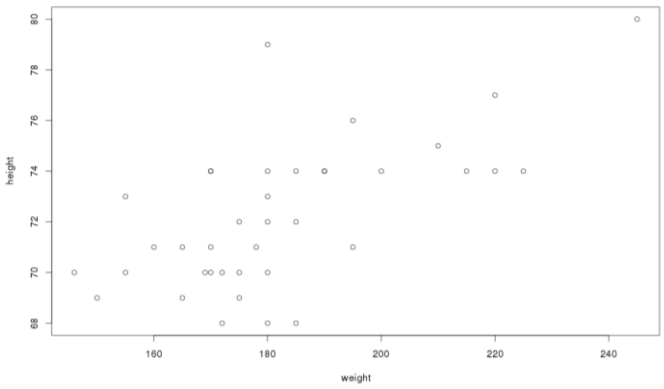
Muestra el peso y altura de los jugadores nacidos en "Germany"

```
peso_altura_germany = subset(master, master$birthCountry=="Germany", select =
c("weight","height"))
```

```
      weight height
255      220     77
629      220     74
981      200     74
1077     155     70
1423     190     74
2096     170     70
3888     185     72
```

Representa los puntos (peso, altura) correspondientes a los jugadores nacidos en "Germany"

```
plot(peso_altura_germany)
```



Apartado (e)

¿Cuántos jugadores son de “England” y tienen un peso mayor que 180 (libras)?

```
england_peso_mayor_180 = subset(master, master$birthCountry=="United Kingdom" &
master$weight>180)

nrow(england_peso_mayor_180)
```

```
[1] 11
```

Apartado (f)

Seleccione 200 jugadores al azar y calcule el índice indicado

Guarde los datos en un fichero Excel datos200.xlsx.

Seleccionamos los 200 jugadores

```
azar_200_jugadores = master[sample(nrow(master), 200), ]
```

Calculamos el índice propuesto:

Paso 1) Definimos la función que realiza dicho cálculo

```
get_index = function(w, h) (w/(h**2))*100
```

Paso 2) Usando cbind, unimos al dataframe original la columna, a la que llamaremos index, que nos devuelve la función definida en el paso anterior

```
azar_200_jugadores = cbind(azar_200_jugadores, index=mapply(get_index,
azar_200_jugadores$weight, azar_200_jugadores$height))
```

Paso 3) Guardamos los datos en un fichero (se adjunta con el nombre indicado).

```
library(openxlsx)

openxlsx::write.xlsx(azar_200_jugadores, file = "datos200.xlsx", colNames = TRUE)
```

Apartado (g)

¿De qué clase R es la variable nameFirst? Conviértela a clase character. Recodifique en la variable nameFirst (en master2) para que en lugar de Charlie aparezca Carlos. ¿Cuántos jugadores tienen como nameFirst el valor Carlos ahora? ¿Y antes?

1) Determinamos la clase dela variable nameFirst

```
class(master$nameFirst)
```

```
[1] "character"
```

2) Convertimos la clase a carácter.

```
master$nameFirst = as.character(master$nameFirst)
```

3) Numero de jugadores con el nombre Carlos antes del cambio

```
nrow(subset(master2, master2$nameFirst=="Carlos"))
```

```
[1] 65
```

Realizamos el cambio

- Con la expresion which(master2$nameFirst=="Charlie") obtenemos la posicion de todas las filas que cumplen que nameFirst=="Charlie"
- Usamos ese array para recuperar todas las filas, de entre ellas solo necesitamos la variables nameFirst por lo que añadimos ["nameFirst"]
- Finalmente asignamos el valor "Carlos" a esa variable

```
master2[which(master2$nameFirst=="Charlie"),]["nameFirst"]="Carlos"
```

Numero de jugadores con el nombre Carlos despues del cambio

```
nrow(subset(master2, master2$nameFirst=="Carlos"))
```

```
[1] 306
```

Ejercicio 2

Combine las dos data.frame en un único data.frame (llámelo todos) uniéndolos por la variable que los relaciona playerID.

```
todos = merge(master,bateos,by="playerID")
```

Apartado (a)

Guarde los 2000 primeros registros de todos en un fichero csv.

```
todos_2000_primeros = todos[1:2000,]
```

```
write.csv(todos_2000_primeros,"todos_2000_primeros.csv")
```

Ejercicio 3

Cree una función que calcule el momento de orden k de una variable, que por defecto calcule el momento de # orden 2 (nota: elimine en la función los datos faltantes o NA que puediera tener la variable). Utilícela para calcular el momento de orden 2, 3 y 4, de las variables peso y altura de todos los jugadores, y de la variable RBI pero únicamente de los jugadores nacidos en USA con yearID igual a 2008 (usa la función subset).

Para facilitar las cosas voy a usar la librería "`e1071`"

```
library(e1071)
```

Creamos la función "moment_of_order". En primer lugar omitimos los datos sin valor, para ello usamos is.na(x) el cual nos devolverá un array de booleanos que nos indica cuáles de ellos tienen el valor NA, para obtener los que no lo tienen simplemente negamos la condición !(is.na(x)). Por ultimo usamos el mapa resultante para obtener los datos concretos.

Finalmente Llamamos a la función "momen"t de la librería "e1071" pasando el valor k que hemos recibido como parámetro, el cual tiene 2 como valor por defecto

```
moment_of_order = function (x,k=2){
  x = x[!(is.na(x))]
  return (e1071::moment(x, order=k, center=TRUE))
}
```

Calculo del momento de orden 2, para peso y altura. Al ser 2 el valor por defecto, no es necesario pasarlo

```
moment_of_order(master$weight)
```

```
[1] 440.9028
```

```
moment_of_order(master$height)
```

```
[1] 6.754381
```

Calculo del momento de orden 3, para peso y altura

```
moment_of_order(master$weight,3)
```

```
[1] 6247.651
```

```
moment_of_order(master$height,3)
```

```
[1] -2.636291
```

Calculo del momento de orden 4, para peso y altura

```
moment_of_order(master$weight,4)
```

```
[1] 811302.8
```

```
moment_of_order(master$height,4)
```

```
[1] 180.9392
```

Calculo del momento de orden 2,3,4, para variable RBI con las restricciones indicadas

Usamos el dataframe “todos”, porque necesitamos filtrar por el año de nacimiento y es variable no está en el dataframe “bateos”

```
rbi_usa_year2008 = subset(todos, todos$birthCountry =="USA" & todos$yearID==2008, select = c("RBI"))
```

```
moment_of_order(rbi_usa_year2008$RBI)
```

```
[1] 691.4872
```

```
moment_of_order(rbi_usa_year2008$RBI)
```

```
[1] 691.4872
```

```
moment_of_order(rbi_usa_year2008$RBI,3)
```

```
[1] 35309.54
```

```
moment_of_order(rbi_usa_year2008$RBI,4)
```

```
[1] 2935597
```

Ejercicio 4

En este ejercicio, de libre elección, me he centrado en el estudio de los datos en lo que a la nacionalidad de los jugadores se refiere

1) Estudio de los jugadores según su país de nacimiento

Obtenemos un vector con los paises de los jugadores

```
players_by_Country = master$birthCountry
```

Normalizamos todos los paises que no son USA

```
players_by_Country[which(players_by_Country!="USA")]="RESTO DEL MUNDO"
```

Obtenemos las frecuencias relativas, en forma de %

```
FrePor = round(prop.table(table(players_by_Country))*100,1)
```

Configuramos y mostramos la gráfica

```
etiquetas = paste(rownames(FrePor)," ",FrePor,"%",sep="")

pie(FrePor,labels=etiquetas,col = rainbow(length(etiquetas)),cex=0.8, main="Distribución
jugadores USA/Resto del mundo")
```

2) Estudio de los jugadores según su país de nacimiento. Para jugadores foráneos

```
foreign_players = subset(master, master$birthCountry!="USA", select = c("birthCountry"))

FrePor  = round(prop.table(table(foreign_players))*100,1)
```

Mostraremos solo los que superen el 5%, agrupando el resto

```
over_5  = FrePor[which(FrePor>=5)]

under_5 = FrePor[which(FrePor<5)]
```

Calcularmos la suma de los porcentajes de los países con menos de un 5%

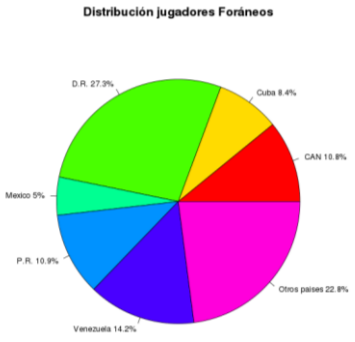
```
under_5_sum = sum(under_5)
```

Añadimos el nuevo dato a mostrar

```
l = length(over_5)+1

names = rownames(over_5)

over_5[l]=under_5_sum

names[l]="Otros paises"
```

Configuramos y mostramos la gráfica

```
etiquetas = paste(names," ",over_5,"%",sep="")

pie(over_5,labels=etiquetas,col = rainbow(length(etiquetas)),cex=0.8, main="Distribución
jugadores Foráneos")
```



3) Estudio de la evolución, en cuanto a número de jugadores debutantes. Series temporales para locales vs foráneos

Creamos una función que extrae el año de la fecha de debut, teniendo en cuenta los distintos formatos presentes.

```
get_debut_year = function(d) {
  if (grepl("-", d)) {
    # Fechas tipo 1890-04-21
    return(as.integer(substr(d,1,4)))
  } else{
    # Fechas tipo 1890-04-21
    return(as.integer(substr(d,(nchar(d)+1)-4,nchar(d))))
  }
}
```

Obtenemos todos los jugadores nacidos fuera de USA

```
foreign_players = subset(master, master$birthCountry!="USA", select = c("debut"))
```

Añadimos la columna "year" a nuestro dataframe

```
foreign_players = cbind(foreign_players, year=mapply(get_debut_year, foreign_players$debut))
```

Obtenemos la tabla de frecuencias absolutas, agrupando en 40 grupos para hacer la gráfica más manejable

```
frecAbs_foreign_players = table(cut(foreign_players$year,breaks = 40))
```

Obtenemos todos los jugadores nacidos en USA

```
local_players = subset(master, master$birthCountry=="USA", select = c("debut"))
```

Añadimos la columna "year" a nuestro dataframe

```
local_players = cbind(local_players, year=mapply(get_debut_year, local_players$debut))
```

Obtenemos la tabla de frecuencias absolutas, agrupando en 40 grupos.

```
frecAbs_local_players = table(cut(local_players$year,breaks = 40))
```

Configuramos y Pintamos la gráfica

```
yrange<-range(c(frecAbs_foreign_players,frecAbs_local_players))

plot(frecAbs_foreign_players, type="l", ylim=yrange, col=1, xlab="Año de debut", ylab="Número
de jugadores", main="Evolución de debuts de jugadores (Locales vs Foraneos)")

lines(frecAbs_local_players, type="l", col=2)
```