

R básico

10-2022

- 1 Analisis de datos ordinales
- 2 Descripción de datos ordinales con R
- 3 Descripción de datos cuantitativos
- 4 Medidas de tendencia central
- 5 Medidas de posición

Sección 1

Analisis de datos ordinales

Descripción de datos ordinales

Los **datos ordinales** son parecidos a los cualitativos, en el sentido de que son cualidades de los individuos u objetos.

La diferencia existente entre los datos cualitativos y los ordinales reside en las características que expresan. En el caso de los ordinales, éstas tienen un orden natural que permite “acumular” observaciones.

Frecuencias para datos ordinales

Al trabajar con datos ordinales, el orden de los niveles de los datos nos permite calcular no solo frecuencias absolutas y relativas, sino también **frecuencias acumuladas**.

Es decir, podemos contar cuantas veces hemos observado un dato menor o igual a este.

Ejemplo 1

Ejemplo 1

Suponed que tenemos una muestra de 15 estudiantes de los cuales sabemos su nota en el examen de Estadística. Clasificamos todos estos resultados en Suspenso (S), Aprobado (A), Notable (N) y Excelente (Ex) y consideramos su orden natural $S < A < N < Ex$.

Las notas obtenidas han sido las siguientes

$S, A, N, Ex, S, S, Ex, Ex, N, A, A, A, A, N, S$

Como recordaréis, para saber cuantas hay de cada una (su frecuencia absoluta), utilizamos la función `table()`

Ejemplo 1

```
notas = ordered(c("S", "A", "N", "Ex", "S", "S",  
                  "Ex", "Ex", "N", "A", "A", "A",  
                  "A", "N", "S"),  
               levels = c("S", "A", "N", "Ex"))  
table(notas)
```

```
## notas  
##  S  A  N Ex  
##  4  5  3  3
```

Como podréis observar, hay 4 *S*, 5 *A*, 3 *N* y 3 *Ex*.

Ejemplo 1

En lo referente a **frecuencias absolutas acumuladas**, hay

- 4 estudiantes con S o menos. Ello implica que la frecuencia acumulada de S es 4
- 9 estudiantes que han obtenido A o menos. Entonces, la frecuencia acumulada de A es 9
- 12 estudiantes los cuales han obtenido N o menos. Así, la frecuencia acumulada de N es 12
- 15 estudiantes (todos) que han obtenido Ex o menos. De este modo, la frecuencia acumulada de Ex es 15, o sea, el total.

Ejemplo 1

Frecuencia relativa acumulada. Es la fracción del total de las observaciones en tanto por 1 que representa su frecuencia absoluta acumulada

Así, las frecuencias relativas acumuladas respectivas son

- $S: \frac{4}{15} \approx 0.27$
- $A: \frac{9}{15} \approx 0.6$
- $N: \frac{12}{15} \approx 0.8$
- $Ex: \frac{15}{15} = 1$

Frecuencia relativa acumulada

En general, supongamos que realizamos n observaciones

$$x_1, \dots, x_n$$

de un cierto tipo de datos ordinales, cuyos posibles niveles ordenados son

$$l_1 < l_2 < \dots < l_k$$

Por tanto, cada una de las observaciones x_j es igual a algún l_i . Diremos que todas estas observaciones forman una **variable ordinal**. En nuestro ejemplo anterior, los 4 niveles eran

$$S < A < N < Ex$$

Frecuencia relativa acumulada

Además, nuestro $n = 15$ y nuestros x_1, \dots, x_{15} son las calificaciones obtenidas por los alumnos.

De este modo, con estas notaciones

- Las definiciones de frecuencias absolutas n_j y las relativas f_j , para cada nivel l_j son las mismas que en una variable cualitativa.
- La frecuencia absoluta acumulada del nivel l_j en esta variable ordinal es el número N_j de observaciones x_i tales que $x_i \leq l_j$. Es decir,

$$N_j = \sum_{i=1}^j n_i$$

Frecuencia relativa acumulada

- La frecuencia relativa acumulada del nivel l_j en esta variable ordinal es la fracción en tanto por 1 F_j de observaciones x_i tales que $x_i \leq l_j$. Es decir,

$$F_j = \frac{N_j}{n} = \sum_{i=1}^j f_i$$

Ejemplo 2

Ejemplo 2

En un estudio, a un grupo de clientes de un restaurante se les hizo la siguiente pregunta:

“¿Estás contento con el trato ofrecido por los trabajadores del establecimiento?”

Las posibles respuestas forman una escala ordinal con $1 < 2 < 3 < 4 < 5$.

Supongamos que se recogieron las siguientes respuestas de 50 técnicos:

```
set.seed(2018)
clientes = sample(1:5, 50, replace = TRUE)
clientes

## [1] 3 4 5 2 5 1 3 4 2 4 3 3 1 1 5 3 1 3 3 5 1 4 2 5 3 4 5 1 2 2 1 5 5 2 1 2 5 5
## [39] 2 1 2 1 3 2 1 2 3 3 1 2
```

```
set.seed(NULL)
```

Ejemplo 2

En este caso tenemos 5 niveles ($k = 5$) y 50 observaciones ($n = 50$) que forman una variable ordinal a la que hemos llamado `clientes`.

Hemos calculado todas sus frecuencias (absoluta, relativa, acumulada y relativa acumulada) y las hemos representado en la siguiente tabla.

##	Absoluta	Relativa	Acumulada	Rel. Acumulada
## 1	12	0.24	12	0.24
## 2	12	0.24	24	0.48
## 3	11	0.22	35	0.70
## 4	5	0.10	40	0.80
## 5	10	0.20	50	1.00

Ejercicio. Calculad todas las frecuencias y comprobad que son exactamente estas.

Frecuencia relativa acumulada

Los gráficos para frecuencias absolutas y relativas absolutas de variables ordinales son exactamente los mismos que para las variables cualitativas.

También podemos utilizar diagramas de barras para describir frecuencias acumuladas: en este caso, la altura de cada barra debe ser igual a la frecuencia acumulada del nivel respectivo. Además, estos niveles deben de aparecer ordenados de manera ascendente, de forma que las alturas de las barras también tengan un orden ascendente.

No obstante, se recomienda no hacer uso de diagramas circulares a la hora de representar frecuencias acumuladas, debido a que éstos no representan la información sobre la acumulación de datos de forma fácil de entender a simple vista.

Sección 2

Descripción de datos ordinales con R

Función cumsum()

¿Recordáis la función `cumsum()`? Pues esta puede ser utilizada a la hora de calcular frecuencias acumuladas.

Retomemos el ejemplo anterior de las notas de los estudiantes y calculemos y representemos en un diagrama de barras las frecuencias acumuladas de la muestra de notas.

notas

```
## [1] S A N Ex S S Ex Ex N A A A A N S
## Levels: S < A < N < Ex
```

```
fAbs = table(notas) #Frec. abs.
cumsum(fAbs) #Frec. abs. acumuladas
```

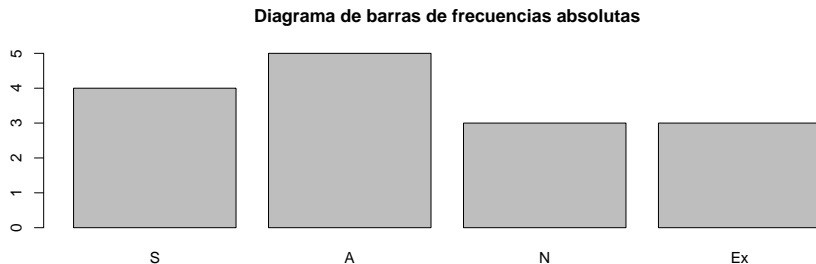
```
## S A N Ex
## 4 9 12 15
```

Función cumsum()

```
cumsum(prop.table(fAbs)) #Frec. relativas acumuladas
```

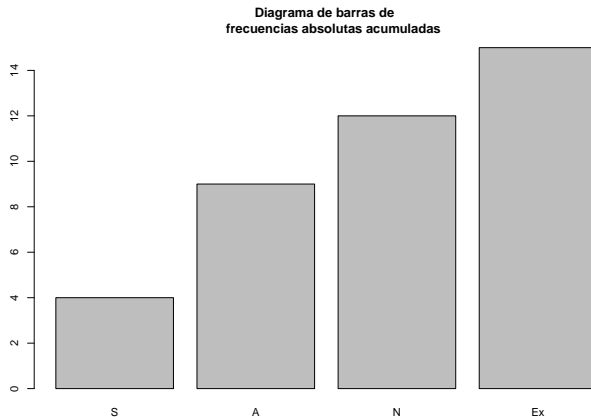
```
##           S           A           N           Ex
## 0.2666667 0.6000000 0.8000000 1.0000000
```

```
barplot(fAbs, main = "Diagrama de barras de frecuencias absolutas")
```



Función cumsum()

```
barplot(cumsum(fAbs), main = "Diagrama de barras de  
frecuencias absolutas acumuladas")
```



Función cumsum()

Podríamos haber calculado las frecuencias relativas acumuladas de la forma

```
cumsum(table(notas))/length(notas)
```

```
##           S           A           N           Ex
## 0.2666667 0.6000000 0.8000000 1.0000000
```

```
cumsum(table(notas)/length(notas))
```

```
##           S           A           N           Ex
## 0.2666667 0.6000000 0.8000000 1.0000000
```

Función `cumsum()`

Pero no podemos hacer `prop.table(cumsum(table(notas)))`.

Ejercicio. Pensad qué ha entendido R que queríamos hacer con esta última instrucción.

Ejemplo 3

Ejemplo 3

Se ha evaluado el tamaño de los cuellos de 100 jirafas. Los niveles que se han utilizado se los considera ordenados de la siguiente manera:

$$\text{Muy.corto} < \text{Corto} < \text{Normal} < \text{Largo} < \text{Muy.largo}$$

Los valores obtenidos en dicho estudio han sido los siguientes

Ejemplo 3

longitud

```
## [1] Normal    Largo    Muy.largo Corto    Muy.largo Muy.corto Normal
## [8] Largo     Corto    Largo     Normal    Normal    Muy.corto Muy.corto
## [15] Muy.largo Normal    Muy.corto Normal    Normal    Muy.largo Muy.corto
## [22] Largo     Corto    Muy.largo Normal    Largo     Muy.largo Muy.corto
## [29] Corto     Corto    Muy.corto Muy.largo Muy.largo Corto    Muy.corto
## [36] Corto     Muy.largo Muy.largo Corto    Muy.corto Corto    Muy.corto
## [43] Normal    Corto    Muy.corto Corto    Normal    Normal    Muy.corto
## [50] Corto     Normal    Muy.corto Largo    Largo     Corto    Muy.corto
## [57] Corto     Normal    Normal    Normal    Normal    Muy.corto Normal
## [64] Muy.corto Corto    Largo    Muy.corto Corto    Muy.corto Muy.largo
## [71] Muy.corto Corto    Muy.largo Largo    Muy.largo Normal    Corto
## [78] Corto     Normal    Largo    Largo    Corto    Corto    Muy.largo
## [85] Largo     Largo    Normal    Normal    Muy.corto Normal    Corto
## [92] Normal    Muy.corto Corto    Muy.corto Normal    Corto    Corto
## [99] Muy.corto Corto
## Levels: Muy.corto < Corto < Normal < Largo < Muy.largo
```

Ejemplo 3

Estudiamos sus frecuencias

```
Fr.Abs = table(longitud)
```

```
Fr.Abs
```

```
## longitud
## Muy.corto Corto Normal Largo Muy.largo
##      23      26      24      13      14
```

```
Fr.Rel = prop.table(Fr.Abs)
```

```
Fr.Rel
```

```
## longitud
## Muy.corto Corto Normal Largo Muy.largo
##      0.23      0.26      0.24      0.13      0.14
```


Ejemplo 3

```
Fr.Acum = cumsum(Fr.Abs)
```

```
Fr.Acum
```

```
## Muy.corto    Corto    Normal    Largo Muy.largo
##           23      49      73      86      100
```

```
Fr.RAcum = cumsum(Fr.Rel)
```

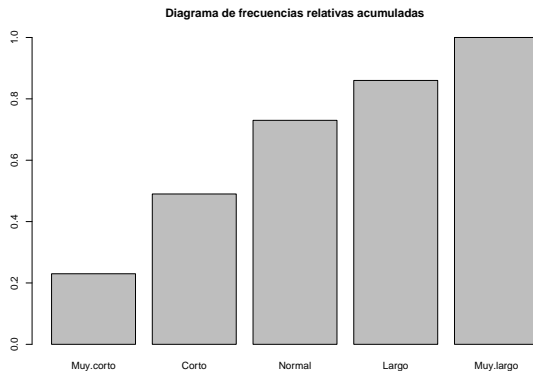
```
Fr.RAcum
```

```
## Muy.corto    Corto    Normal    Largo Muy.largo
##           0.23    0.49    0.73    0.86    1.00
```

Ejemplo 3

La instrucción `barplot` produce el siguiente diagrama de barras de frecuencias relativas acumuladas

```
barplot(Fr.RAcum, main = "Diagrama de frecuencias relativas acumuladas")
```



Función `cumsum()`

Para calcular frecuencias acumuladas en una tabla multidimensional, hay que aplicar a la tabla la función `cumsum` mediante la función `apply` que ya explicábamos para matrices. En este caso en concreto, la sintaxis de la instrucción sería

```
apply(tabla, MARGIN=..., FUN=cumsum)
```

donde el valor `MARGIN` ha de ser el de la dimensión en la que queremos acumular las frecuencias: 1 si queremos hacerlo por filas, 2 para hacerlo por columnas, etc. Lo veremos todo más claro con un ejemplo

Ejemplo 4

Ejemplo 4

Supongamos que en el ejemplo anterior, el de las jirafas, estas provienen de 4 zonas diferentes, A,B,C y D, de manera que las 30 primeras son de la zona A, las 25 siguientes de la B, las 35 siguientes de la C y las 10 últimas de la D. Nos interesa estudiar la distribución de las longitudes según la zona.

Vamos a organizar todos estos datos en un data frame llamado `jirafas`. Para que nos sea más fácil visualizar la información, es conveniente que las filas de las tablas de frecuencias correspondan a las zonas. Por lo tanto, al definir el data frame, entraremos como primera variable la de la muestra las zonas. Así, conseguiremos que éstas aparezcan en las filas al aplicarle la función `table`.

Ejemplo 4

```
zonas = rep(c("A","B","C","D"), c(30,25,35,10))
jirafas = data.frame(zonas,longitud)
str(jirafas)
```

```
## 'data.frame':    100 obs. of  2 variables:
## $ zonas      : chr  "A" "A" "A" "A" ...
## $ longitud: Ord.factor w/ 5 levels "Muy.corto"<"Corto"<...: 3 4 5 2 5 1 3 4 2 4 ...
```

```
head(jirafas)
```

```
##   zonas longitud
## 1    A   Normal
## 2    A    Largo
## 3    A Muy.largo
## 4    A    Corto
## 5    A Muy.largo
## 6    A Muy.corto
```

Ejemplo 4

Para calcular la tabla de frecuencias absolutas acumuladas de las longitudes por zonas y como las zonas definen las filas de la tabla anterior, debemos utilizar la función `apply` con `MARGIN = 1`.

```
apply(table(jirafas), MARGIN = 1, FUN = cumsum)
```

```
##           zonas
## longitud   A  B  C  D
## Muy.corto  6  7  7  3
## Corto     11 15 15  8
## Normal     19 19 25 10
## Largo      24 21 31 10
## Muy.largo  30 25 35 10
```

Ejemplo 4

Fijaos que la tabla se ha traspuesto. Resulta que cuando se aplica `apply` a una table bidimensional, R intercambia, en caso de ser necesario, filas por columnas en el resultado para que la dimensión de la tabla resultante en la que se haya aplicado la función sea la de las columnas.

Con lo cual, para volver a tener las zonas en las filas, hay que trasponer el resultado de la función `apply`.

```
t(apply(table(jirafas), MARGIN = 1, FUN = cumsum))
```

```
##      longitud
## zonas Muy.corto Corto Normal Largo Muy.largo
##   A         6    11    19    24     30
##   B         7    15    19    21     25
##   C         7    15    25    31     35
##   D         3     8    10    10     10
```

Ejemplo 4

Vamos ahora a calcular la tabla de frecuencias relativas acumuladas de las longitudes de cuello por zonas. Para conseguirlo, y en una única instrucción, primero calculamos la tabla de frecuencias relativas por filas, a continuación, con las funciones `apply` y `cumsum` las acumulamos y, finalmente, trasponemos el resultado.

```
t(apply(prop.table(table(jirafas),
                        margin = 1),
        MARGIN = 1, FUN = cumsum))
```

```
##      longitud
## zonas Muy.corto   Corto   Normal   Largo Muy.largo
## A      0.20 0.3666667 0.6333333 0.8000000      1
## B      0.28 0.6000000 0.7600000 0.8400000      1
## C      0.20 0.4285714 0.7142857 0.8857143      1
## D      0.30 0.8000000 1.0000000 1.0000000      1
```


Ejemplo 4

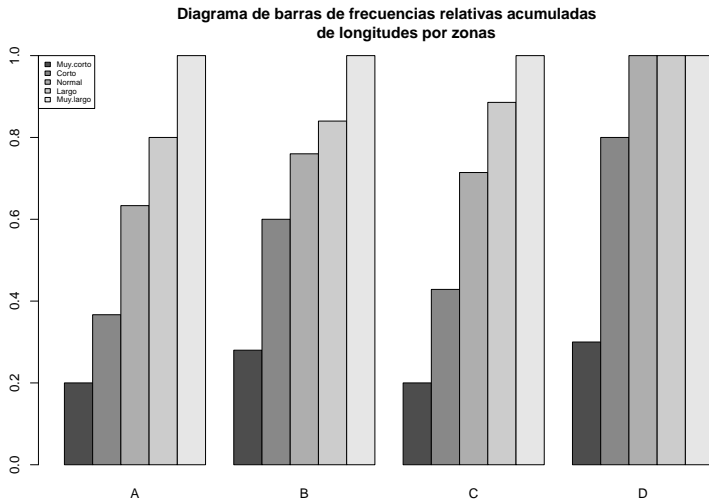
Vamos ahora a dibujar el diagrama de barras por bloques de esta tabla. Nos interesa que las barras de este diagrama se agrupen por zonas. Entonces, tendremos que aplicar `barplot` a la tabla sin trasponer.

Además, vamos a colocar la leyenda en la esquina superior izquierda para que no se superponga a ninguna barra. También reduciremos el tamaño del texto de la leyenda para que quepa completamente.

Ejemplo 4

```
Diagrama = apply(prop.table(table(jirafas), margin = 1),  
                  MARGIN = 1, FUN = cumsum)  
barplot(Diagrama, beside = TRUE, legend = TRUE,  
main = "Diagrama de barras de frecuencias relativas acumuladas  
       de longitudes por zonas",  
args.legend=list(x="topleft", cex=0.55))
```

Ejemplo 4



Ejemplo 5

Ejemplo 5

Consideremos el data frame `datacrab` y arreglemos los datos.

```
crabs = read.table("../data/datacrab.txt", header = TRUE)
crabs = crabs[,-1] #Omitimos la primera columna
str(crabs)
```

```
## 'data.frame':    173 obs. of  5 variables:
## $ color : int  3 4 2 4 4 3 2 4 3 4 ...
## $ spine : int  3 3 1 3 3 3 1 2 1 3 ...
## $ width : num  28.3 22.5 26 24.8 26 23.8 26.5 24.7 23.7 25.6 ...
## $ satell: int  8 0 9 0 4 0 0 0 0 0 ...
## $ weight: int  3050 1550 2300 2100 2600 2100 2350 1900 1950 2150 ...
```

La variable numérica `width` contiene la anchura de cada cangrejo

Ejemplo 5

```
table(crabs$width)
```

```
##
##  21  22 22.5 22.9  23 23.1 23.2 23.4 23.5 23.7 23.8 23.9  24 24.1 24.2 24.3
##    1    1    3    3    2    3    1    1    1    3    3    1    2    1    2    2
## 24.5 24.7 24.8 24.9  25 25.1 25.2 25.3 25.4 25.5 25.6 25.7 25.8 25.9  26 26.1
##    7    5    1    3    6    2    2    1    3    3    2    6    7    1    6    2
## 26.2 26.3 26.5 26.7 26.8  27 27.1 27.2 27.3 27.4 27.5 27.6 27.7 27.8 27.9  28
##    8    1    6    3    3    5    2    2    1    3    6    1    2    2    2    3
## 28.2 28.3 28.4 28.5 28.7 28.9  29 29.3 29.5 29.7 29.8  30 30.2 30.3 30.5 31.7
##    4    3    2    4    2    1    6    2    1    1    1    3    1    1    1    1
## 31.9 33.5
##    1    1
```

Ejemplo 5

Vamos a convertir a la variable `width` en una variable ordinal que agrupe las entradas de la variable original en niveles.

La manera más sencilla de llevarlo a cabo es utilizando la función `cut`, que estudiaremos en detalle en lecciones posteriores. Por ahora, basta con saber que la instrucción dividirá el vector numérico `crabs$width` en intervalos de extremos los puntos especificados en el argumento `breaks`. El parámetro `right = FALSE` sirve para indicar que los puntos de corte pertenecen la intervalo de su derecha, e `Inf` indica ∞ .

Por lo tanto, nosotros llevaremos a cabo la siguiente instrucción

```
intervalos = cut(crabs$width, breaks = c(21,25,29,33,Inf), right = FALSE,  
                labels = c("21-25", "25-29", "29-33", "33-..."))
```

Ejemplo 5

El resultado de la instrucción es un factor que tiene como niveles estos intervalos, identificados con las etiquetas especificadas en el parámetro `labels`. Como nosotros vamos a usar estos intervalos como niveles de una variable ordinal, además convertiremos este factor en ordenado.

```
crabs$width.rank = ordered(intervalos)
str(crabs)
```

```
## 'data.frame':    173 obs. of  6 variables:
## $ color      : int  3 4 2 4 4 3 2 4 3 4 ...
## $ spine      : int  3 3 1 3 3 3 1 2 1 3 ...
## $ width      : num  28.3 22.5 26 24.8 26 23.8 26.5 24.7 23.7 25.6 ...
## $ satell     : int  8 0 9 0 4 0 0 0 0 0 ...
## $ weight     : int  3050 1550 2300 2100 2600 2100 2350 1900 1950 2150 ...
## $ width.rank: Ord.factor w/ 4 levels "21-25"<"25-29"<...: 2 1 2 1 2 1 2 1 1 2 ...
```

Ejemplo 5

Nos interesa estudiar la distribución de las anchuras de los cangrejos según el número de colores. Por lo tanto, vamos a calcular las tablas bidimensionales de frecuencias relativas y relativas acumuladas de los intervalos de las anchuras en cada nivel de color y las representaremos por medio de diagramas de barras.

La tabla de frecuencias absolutas de los pares se puede obtener aplicando `table` al data frame formado por la primera y última columnas.

```
Tabla = table(crabs[,c(1,6)])
```

Tabla

```
##      width.rank
## color 21-25 25-29 29-33 33-...
##    2      1      9      2      0
##    3     19     62     13     1
##    4     17     24      3     0
##    5      9     12      1     0
```


Ejemplo 5

```
Fr.rel = round(prop.table(Tabla,margin = 1),3)
```

```
Fr.rel
```

```
##      width.rank
## color 21-25 25-29 29-33 33-...
##      2 0.083 0.750 0.167 0.000
##      3 0.200 0.653 0.137 0.011
##      4 0.386 0.545 0.068 0.000
##      5 0.409 0.545 0.045 0.000
```

Ejemplo 5

```
Fr.rel.acu = round(apply(prop.table(Tabla, margin = 1),
                          MARGIN = 1, FUN = cumsum), 3)
t(Fr.rel.acu)
```

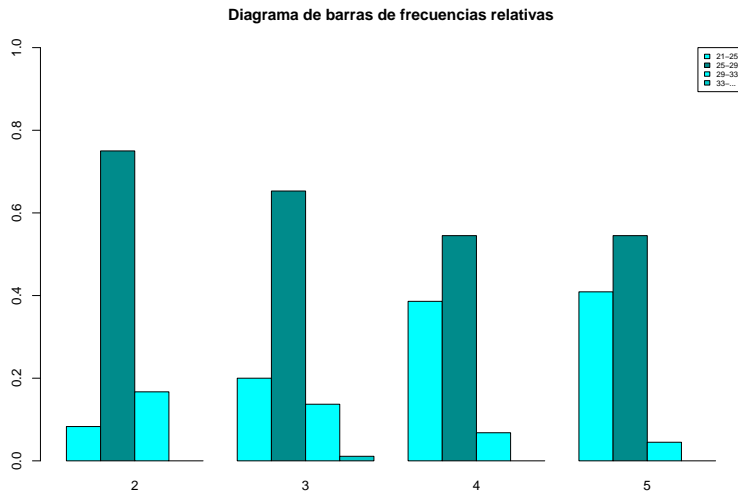
```
##      width.rank
## color 21-25 25-29 29-33 33-...
##      2 0.083 0.833 1.000      1
##      3 0.200 0.853 0.989      1
##      4 0.386 0.932 1.000      1
##      5 0.409 0.955 1.000      1
```

Ejemplo 5

```
azul = c("cyan", "cyan4", "cyan1", "cyan3")

barplot(t(Fr.rel), beside = TRUE, legend = TRUE, ylim = c(0,1), col = azul,
        main = "Diagrama de barras de frecuencias relativas",
        args.legend=list(x = "topright", cex=0.55))
```

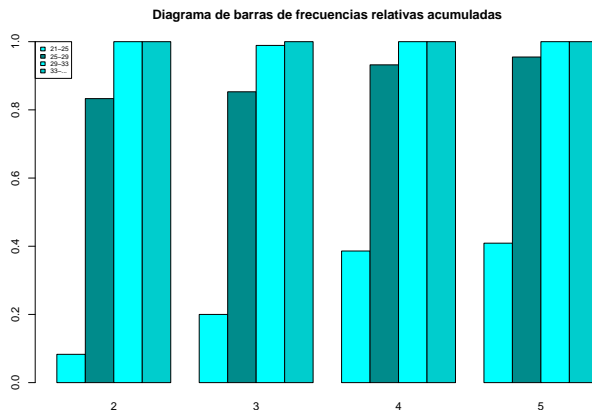
Ejemplo 5



Ejemplo 5

```
barplot(Fr.rel.acu, beside = TRUE, legend = TRUE, col = azul,  
        main = "Diagrama de barras de frecuencias relativas acumuladas",  
        args.legend=list(x = "topleft", cex=0.55))
```

Ejemplo 5



Los **datos ordinales** son parecidos a los cualitativos, en el sentido de que son cualidades de los individuos u objetos.

Sección 3

Descripción de datos cuantitativos

Datos cuantitativos

Los **datos cuantitativos** son los que expresan cantidades que se representan mediante números. Éstos se suelen clasificar en continuos y discretos.

- Los **datos continuos** son los que, si existiese la posibilidad de medirlos con precisión infinita, en principio podrían tomar todos los valores de un intervalo de la recta real. A modo de ejemplo, el peso, la altura, el tiempo... son datos de este tipo.
- Por su parte, los **datos discretos** son los que pueden tomar un solo conjunto contable de valores. El número de colores de un gato, el número de individuos que conforman una población son algunos ejemplos de este tipo de datos.

Conviene tener en cuenta que esta división es solo teórica. Es decir, en la práctica, todos estos datos son discretos puesto que la precisión infinita no existe. Sin embargo, es necesario de vez en cuando suponer los datos de tipo continuo para así poder utilizar técnicas específicas en su análisis.

Datos cuantitativos

A la hora de estudiar **variables cuantitativas**, podemos utilizar las frecuencias que hemos visto hasta el momento: absoluta, relativa, acumulada y relativa acumulada. Esto se debe a que podemos ordenar los datos cuantitativos en el orden natural de los números reales.

En este caso, disponemos de muchas otras técnicas descriptivas aparte de las frecuencias, puesto que estamos trabajando con números reales y podemos operar con ellos.

Los datos cuantitativos admiten dos tipos de tratamiento según trabajemos con los **raw data** (datos brutos u originales) o bien los agrupemos en clases o intervalos.

En esta lección trabajaremos sobre la primera situación. En la siguiente, estudiaremos la descripción de datos cuantitativos agrupados.

Frecuencias de datos cuantitativos

El tratamiento de las frecuencias de datos cuantitativos es similar al de los datos ordinales. La cosa cambia ligeramente debido a que no se tienen en cuenta todos los niveles posibles, sino únicamente los observados.

Ejemplo 1

Ejemplo 1

Se han pedido las edades a 20 clientes de un museo. Las respuestas obtenidas han sido las siguientes:

```
edad = c(15, 18, 25, 40, 30, 29, 56, 40, 13, 27, 42, 23, 11, 26, 25, 32, 30, 40, 33, 29)
```

Recordemos que solamente nos interesan las frecuencias de las edades observadas. Es decir, solamente nos interesan

```
table(edad)
```

```
## edad
## 11 13 15 18 23 25 26 27 29 30 32 33 40 42 56
##  1  1  1  1  1  2  1  1  2  2  1  1  3  1  1
```

Ejemplo 1

Calculemos el resto de frecuencias como ya sabemos

```
round(prop.table(table(edad)),3)
```

```
## edad
##  11  13  15  18  23  25  26  27  29  30  32  33  40  42  56
## 0.05 0.05 0.05 0.05 0.05 0.10 0.05 0.05 0.10 0.10 0.05 0.05 0.15 0.05 0.05
```

```
cumsum(table(edad))
```

```
## 11 13 15 18 23 25 26 27 29 30 32 33 40 42 56
##  1  2  3  4  5  7  8  9 11 13 14 15 18 19 20
```

Ejemplo 1

```
round(cumsum(prop.table(table(edad))),3)
```

```
##    11    13    15    18    23    25    26    27    29    30    32    33    40    42    56  
## 0.05 0.10 0.15 0.20 0.25 0.35 0.40 0.45 0.55 0.65 0.70 0.75 0.90 0.95 1.00
```

Frecuencias de datos cuantitativos

En general, supongamos que tenemos n observaciones de una propiedad que se mide con un número real y obtenemos la variable cuantitativa formada por los datos

$$x_1, \dots, x_n$$

Sean ahora X_1, \dots, X_k los valores distintos que aparecen en esta lista de datos y considerémoslos ordenados

$$X_1 < X_2 < \dots < X_k$$

Frecuencias de datos cuantitativos

Entonces, en esta variable cuantitativa

- La frecuencia absoluta de X_i es el número n_i de elementos que son iguales a X_i
- La frecuencia relativa de X_i es $f_i = \frac{n_i}{n}$
- La frecuencia absoluta acumulada de X_i es $N_i = \sum_{j=1}^i n_j$
- La frecuencia relativa acumulada de X_i es $F_i = \frac{N_i}{n}$

Ejemplo 2

Ejemplo 2

Lanzamos 25 veces un dado de 6 caras y anotamos las puntuaciones obtenidas en cada tirada.

En este caso, $n = 25$ y, los distintos valores observados son

$$X_1 = 1, X_2 = 2, X_3 = 3, X_4 = 4, X_5 = 5, X_6 = 6$$

Nos interesa ahora calcular las frecuencias de este experimento. Además, las organizaremos en un data frame para observarlas de forma más clara y sencilla en una tabla.

```
set.seed(162017)
datos = sample(1:6,25,replace = TRUE)
datos
```

```
## [1] 1 1 5 5 5 5 1 6 5 4 1 3 1 3 2 2 1 1 1 4 2 1 6 3 1
```

```
set.seed(NULL)
```


Ejemplo 2

```
table(dados)
```

```
## dados  
##  1  2  3  4  5  6  
## 10  3  3  2  5  2
```

```
round(prop.table(table(dados)),2)
```

```
## dados  
##    1    2    3    4    5    6  
## 0.40 0.12 0.12 0.08 0.20 0.08
```

```
cumsum(table(dados))
```

```
##  1  2  3  4  5  6  
## 10 13 16 18 23 25
```

Ejemplo 2

```
round(cumsum(prop.table(table(dados))),2)
```

```
##      1      2      3      4      5      6  
## 0.40 0.52 0.64 0.72 0.92 1.00
```

```
dados.df = data.frame(  
  Puntuacion = 1:6,  
  Fr.abs = as.vector(table(dados)),  
  Fr.rel = as.vector(round(prop.table(table(dados)),2)),  
  Fr.acu = as.vector(cumsum(table(dados))),  
  Fr.racu = as.vector(round(cumsum(  
    prop.table(table(dados))),2)))
```

Ejemplo 2

```
datos.df
```

```
##   Puntuacion Fr.abs Fr.rel Fr.acu Fr.racu
## 1          1     10  0.40     10   0.40
## 2          2      3  0.12     13   0.52
## 3          3      3  0.12     16   0.64
## 4          4      2  0.08     18   0.72
## 5          5      5  0.20     23   0.92
## 6          6      2  0.08     25   1.00
```

¡OJO! Para entrar una tabla unidimensional como una variable en un data frame, es conveniente transformarla en vector con `as.vector`. Si no, cada `table` y cada `prop.table` añadirían una columna extra con los nombres de los niveles.

Sección 4

Medidas de tendencia central

Medidas de tendencia central

Las **medidas de tendencia central** son las que dan un valor representativo a todas las observaciones. Algunas de las más importantes son:

- La **media aritmética** o **valor medio**

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} = \frac{\sum_{j=1}^k n_j X_j}{n} = \sum_{j=1}^k f_j X_j$$

- La **mediana**, que representa el valor central en la lista ordenada de observaciones.
- La **moda** es el valor (o valores) de máxima frecuencia (absoluta o relativa, el resultado será el mismo).

La mediana

La definición formal de la mediana es la siguiente. Denotando por

$$x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$$

los datos de la variable cuantitativa ordenados de menor a mayor, la mediana es

- Si n par, la medio de los dos datos centrales

$$\frac{x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)}}{2}$$

- Si n impar, el dato central $x_{(\frac{n+1}{2})}$

Ejemplo 1

Recordemos el ejemplo de las edades.

```
sort(edad) #Ordenamos los datos por su orden natural
```

```
## [1] 11 13 15 18 23 25 25 26 27 29 29 30 30 32 33 40 40 40 42 56
```

```
table(edad)
```

```
## edad
## 11 13 15 18 23 25 26 27 29 30 32 33 40 42 56
##  1  1  1  1  1  2  1  1  2  2  1  1  3  1  1
```

En este caso, la moda es 40, la mediana es $\frac{29+29}{2} = 29$ y la media aritmética es

$$\frac{11 + 13 + 15 + 18 + 23 + 25 + 25 + 26 + 27 + 29 + 29 + 30 + 30 + 32 + 33 + 40 + 40 + 40 + 42 + 56}{20} = 29.2$$

Ejemplo 2

Recordemos el ejemplo de los dados.

`datos.df`

```
##   Puntuacion Fr.abs Fr.rel Fr.acu Fr.racu
## 1          1     10  0.40     10   0.40
## 2          2      3  0.12     13   0.52
## 3          3      3  0.12     16   0.64
## 4          4      2  0.08     18   0.72
## 5          5      5  0.20     23   0.92
## 6          6      2  0.08     25   1.00
```

En este caso, la moda son dos valores: el 2 y el 3. La mediana es $x_{(13)} = 3$ y la media aritmética es 2.8

Medidas de tendencia central en R

Vamos a calcular la media aritmética, mediana y moda de los dos ejemplos anteriores con instrucciones de R.

```
mean(edad) #La media aritmética
```

```
## [1] 29.2
```

```
mean(dados)
```

```
## [1] 2.8
```

```
median(edad) #La mediana
```

```
## [1] 29
```

Medidas de tendencia central en R

```
median(dados)
```

```
## [1] 2
```

```
as.numeric(names(which(
  table(edad)==max(table(edad))))) #La moda
```

```
## [1] 40
```

```
as.numeric(names(which(
  table(dados)==max(table(dados)))))
```

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

Cuando trabajamos con datos cuantitativos, es conveniente que el resultado lo demos como un número. De ahí que hayamos aplicado la función `as.numeric`.

Sección 5

Medidas de posición

Medidas de posición

Las **medidas de posición** estiman qué valores dividen las observaciones en unas determinadas proporciones.

Los valores que determinan estas posiciones son conocidos como los **cuantiles**.

Pensándolo de este modo, la mediana puede interpretarse como una medida de posición, debido a que divide la variable cuantitativa en dos mitades.

Medidas de posición

Dada una proporción $p \in (0, 1)$, el **cuantil de orden p** de una variable cuantitativa, Q_p , es el valor más pequeño tal que su frecuencia relativa acumulada es mayor o igual a p .

Dicho de otro modo, si tenemos un conjunto de observaciones x_1, \dots, x_n y los ordenamos de menor a mayor, entonces Q_p será el número más pequeño que deja a su izquierda (incluyéndose a sí mismo) como mínimo a la fracción p de los datos. Es decir, $p \cdot n$.

Así, ahora es más claro ver que la mediana vendría a ser $Q_{0.5}$, el cuantil de orden 0.5.

Ejemplo 3

Ejemplo 3

Consideremos un experimento en el que lanzamos 50 veces un dado de 4 caras y obtenemos los siguientes resultados

```
set.seed(260798)
dado = sample(1:4, 50, replace = TRUE)
set.seed(NULL)
length(dado)

## [1] 50

dado = sort(dado) #Los ordenamos de menor a mayor
dado

## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 4 4
## [39] 4 4 4 4 4 4 4 4 4 4 4 4
```

Ejemplo 3

```
df.dado = data.frame(  
  Puntuacion = 1:4,  
  Fr.abs = as.vector(table(dado)),  
  Fr.rel = as.vector(round(prop.table(table(dado)),2)),  
  Fr.acu = as.vector(cumsum(table(dado))),  
  Fr.racu = as.vector(round(cumsum(  
    prop.table(table(dado))),2))  
)  
df.dado
```

##	Puntuacion	Fr.abs	Fr.rel	Fr.acu	Fr.racu
## 1	1	16	0.32	16	0.32
## 2	2	15	0.30	31	0.62
## 3	3	5	0.10	36	0.72
## 4	4	14	0.28	50	1.00

Ejemplo 3

Si nos piden el cuantil $Q_{0.3}$, sabemos que este es el primer elemento de la lista cuya frecuencia relativa acumulada es mayor o igual a 0.3. Este se corresponde con la puntuación 1.

Ejemplo 3

También podríamos hallarlo de otro modo: fijándonos en la lista ordenada de puntuaciones, el cuantil $Q_{0.3}$ sería el primer elemento de dicha lista tal que fuera mayor o igual que, como mínimo, el 30% de los datos. Si calculamos el 30% de 50, obtenemos que es 15. Esto lo que nos dice es que el cuantil que buscamos es el número que se encuentra en la quinceava posición de la lista ordenada.

```
dato[15]
```

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

Cuantiles

Algunos cuantiles tienen nombre propio:

- Los **cuartiles** son los cuantiles $Q_{0.25}$, $Q_{0.5}$ y $Q_{0.75}$. Respectivamente, son llamados primer, segundo y tercer cuartil. El primer cuartil, $Q_{0.25}$, será el menor valor que es mayor o igual a una cuarta parte de las observaciones y $Q_{0.75}$, el menor valor que es mayor o igual a tres cuartas partes de los datos observados.
- El cuantil $Q_{0.5}$ es la mediana
- Los **deciles** son los cuantiles Q_p con p un múltiplo de 0.1.
- Los **percentiles** son los cuantiles Q_p con p un múltiplo de 0.01.

Cuantiles

La definición de cuantil anteriormente dada es orientativa. La realidad es que, exceptuando el caso de la mediana, no hay consenso sobre cómo deben calcularse los cuantiles. En verdad, existen diferentes métodos que pueden dar lugar a soluciones distintas.

Al fin y al cabo, nuestro objetivo no es el de encontrar el primer valor de una muestra cuya frecuencia relativa acumulada en la variable sea mayor o igual a p , sino estimar el valor de esta cantidad para el total de la población.

Para calcular los cuantiles de orden p de una variable cualitativa x con R, se utiliza la instrucción `quantile(x,p)`, la cual dispone de 9 métodos diferentes que se especifican con el parámetro `type`. El valor por defecto es `type = 7` y no hace falta especificarlo, como veremos en el siguiente ejemplo. Para más información sobre todos los valores posibles de este parámetro, haced click en el enlace a [Wikipedia](#)

Ejemplo 4

```
set.seed(0)
datos2 = sample(1:6,15, replace = TRUE)
datos2
```

```
## [1] 6 1 4 1 2 5 3 6 2 3 3 1 5 5 2
```

```
set.seed(NULL)
quantile(datos2,0.25) #Primer cuartil
```

```
## 25%
## 2
```

```
quantile(datos2,0.8)
```

```
## 80%
## 5
```