

Estadística Descriptiva con Datos Cualitativos

Ramon Ceballos

24/1/2021

TABLAS DE FRECUENCIAS BIDIMENSIONALES

1. Función `table()`. Frecuencias absolutas

La función `table()` también permite construir tablas de frecuencias conjuntas de dos o más variables.

Supongamos que el vector `Respuestas` anterior contiene las respuestas a una pregunta dadas por unos individuos cuyos sexos tenemos almacenados en un vector `Sexo`, en el mismo orden que sus respuestas. En este caso, podemos construir una tabla que nos diga cuántas personas de cada sexo han dado cada respuesta.

```
Respuestas=factor(sample(c("Si", "No"), size = 12, replace = TRUE))
Sexo= sample(c("H", "M"), size = length(Respuestas), replace = T) #H = hombre, M = mujer

#La tabla de contingencia obtenida es bidimensional
#El primer valor determina las filas
#El segundo valor determina las columnas
table(Respuestas ,Sexo)
```

```
##           Sexo
## Respuestas H M
##           No 5 2
##           Si 2 3
```

Ejercicio

- Comprobad qué ocurre si cambiamos el orden de las columnas en la función `table()`.

```
table(Sexo, Respuestas)
```

```
##           Respuestas
## Sexo No Si
##   H   5 2
##   M   2 3
```

- Usad la función `t()` para transponer ambas tablas y comprobad el resultado

```
#El girar dos columnas en la tabla de contingencia
#es equivale a hacer la traspuesta de una matriz
t(table(Respuestas,Sexo))
```

```
##      Respuestas
## Sexo No Si
##    H  5  2
##    M  2  3
```

```
t(table(Sexo,Respuestas))
```

```
##              Sexo
## Respuestas H M
##           No 5 2
##           Si 2 3
```

2. Entradas en una Tabla Bidimensional

Para referirnos a una entrada de una tabla bidimensional podemos usar el sufijo `[,]` como si estuviéramos en una matriz o un data frame. Dentro de los corchetes, tanto podemos usar los índices como los nombres (entre comillas) de los niveles.

```
table(Respuestas ,Sexo)[1,2]
```

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

```
table(Respuestas ,Sexo)["No","M"]
```

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

3. Función `prop.table()`. Frecuencias relativas

Como en el caso unidimensional, la función `prop.table()` sirve para calcular tablas bidimensionales de frecuencias relativas conjuntas de pares de variables. Pero en el caso bidimensional tenemos *dos tipos de frecuencias relativas*:

Frecuencias relativas globales: para cada par de niveles, uno de cada variable, la fracción de individuos que pertenecen a ambos niveles respecto del total de la muestra.

Frecuencias relativas marginales: dentro de cada nivel de una variable y para cada nivel de la otra, la fracción de individuos que pertenecen al segundo nivel respecto del total de la subpoblación definida por el primer nivel.

Dadas dos variables, se pueden calcular dos familias de **frecuencias relativas marginales**, según cuál sea la variable que defina las subpoblaciones en las que calculemos las frecuencias relativas de los niveles de la otra variable; no es lo mismo la fracción de mujeres que han contestado que sí respecto del total de mujeres, que la fracción de mujeres que han contestado que sí respecto del total de personas que han dado esta misma respuesta.

3.1. Frecuencias Relativas Globales

La tabla de frecuencias relativas globales se calcula aplicando sin más la función `prop.table()` a la `table`.

```
prop.table(table(Sexo,Respuestas)) #Global
```

```
##      Respuestas
## Sexo      No      Si
##   H 0.4166667 0.1666667
##   M 0.1666667 0.2500000
```

De este modo, la tabla `prop.table(table(Sexo,Respuestas))` nos da la fracción del total que representa cada pareja (sexo, respuesta).

3.2. Frecuencias Relativas Marginales

Para obtener las marginales, debemos usar el parámetro `margin` al aplicar la función `prop.table()` a la `table`. Con `margin=1` obtenemos las frecuencias relativas de las filas y con `margin=2`, de las columnas.

```
prop.table(table(Sexo,Respuestas), margin=1) #Por sexo (filas)
```

```
##      Respuestas
## Sexo      No      Si
##   H 0.7142857 0.2857143
##   M 0.4000000 0.6000000
```

```
prop.table(table(Sexo,Respuestas), margin=2) #Por respuesta (columnas)
```

```
##      Respuestas
## Sexo      No      Si
##   H 0.7142857 0.4000000
##   M 0.2857143 0.6000000
```

La tabla `prop.table(table(Sexo,Respuestas), margin=1)` nos da la fracción que representa cada respuesta dentro de cada sexo. Por ejemplo, un X % de las mujeres han contestado que no.

La tabla `prop.table(table(Sexo,Respuestas), margin=2)` nos da la fracción que representa cada sexo dentro de cada respuesta. Por ejemplo, las mujeres representan el X % del total de las personas que han contestado que no.

4. Función `CrossTable()` del paquete `gmodels`

4.1 Definición y modo de uso

La función `CrossTable()` del paquete `gmodels` permite producir (especificando el parámetro `prop.chisq=FALSE`) un resumen de la tabla de frecuencias absolutas y las tres tablas de frecuencias relativas de dos variables en un formato adecuado para su visualización.

La leyenda *Cell Contents* explica los contenidos de cada celda de la tabla: la frecuencia absoluta, la frecuencia relativa por filas, la frecuencia relativa por columnas, y la frecuencia relativa global. Esta función dispone de muchos parámetros que permiten modificar el contenido de las celdas, y que podéis consultar en `help(CrossTable)`.

```
#Instalamos y cargamos el paquete gmodels
#install.packages("gmodels")
library(gmodels)
CrossTable(Sexo,Respuestas,prop.chisq=FALSE)
```

```
##
##
##      Cell Contents
## |-----|
## |                      N |
## |      N / Row Total |
## |      N / Col Total |
## |      N / Table Total |
## |-----|
##
##
## Total Observations in Table:  12
##
##
##      | Respuestas
##      Sexo |      No |      Si | Row Total |
## -----|-----|-----|-----|
##      H |      5 |      2 |      7 |
##      |      0.714 |      0.286 |      0.583 |
##      |      0.714 |      0.400 |      |
##      |      0.417 |      0.167 |      |
## -----|-----|-----|-----|
##      M |      2 |      3 |      5 |
##      |      0.400 |      0.600 |      0.417 |
##      |      0.286 |      0.600 |      |
##      |      0.167 |      0.250 |      |
## -----|-----|-----|-----|
## Column Total |      7 |      5 |      12 |
##      |      0.583 |      0.417 |      |
## -----|-----|-----|-----|
##
##
```

3. Funciones aplicables a tablas bidimensionales cualesquiera

Una **tabla de contingencia bidimensional** es, básicamente, una matriz con algunos atributos extra. En particular, podemos usar sobre estas tablas la mayoría de las funciones para matrices que tengan sentido para tablas:

- `rowSums()` y `colSums()` se pueden aplicar a una tabla y suman sus filas y sus columnas, respectivamente.
- También podemos usar sobre una tabla bidimensional (o, en general, multidimensional) la función `apply()` con la misma sintaxis que para matrices.

```
table(Sexo,Respuestas)
```

```
##      Respuestas
## Sexo No Si
##   H  5  2
##   M  2  3
```

```
#Suma por columnas
colSums(table(Sexo,Respuestas))
```

```
## No Si
##  7  5
```

```
#Suma por filas
rowSums(table(Sexo,Respuestas))
```

```
## H M
## 7 5
```

```
#Se obtiene la suma en proporción de las columnas (la proporción)
colSums(prop.table(table(Sexo,Respuestas)))
```

```
##           No           Si
## 0.5833333 0.4166667
```

```
#Se obtiene la suma en proporción de las filas
rowSums(prop.table(table(Sexo,Respuestas)))
```

```
##           H           M
## 0.5833333 0.4166667
```

Ejemplo función CroosTable()

```
#Cargo librería gmodels
library(gmodels)
sex = factor(c("H", "M", "M", "M", "H", "H", "M", "M"))
ans = factor(c("S", "N", "S", "S", "S", "N", "N", "S"))
#Función CrossTable()
CrossTable(sex, ans, prop.chisq = FALSE)
```

```
##
##
##      Cell Contents
## |-----|
## |                      N |
## |          N / Row Total |
## |          N / Col Total |
## |          N / Table Total |
## |-----|
##
##
```

```
## Total Observations in Table:  8
##
##
##      | ans
##      sex |      N |      S | Row Total |
## -----|-----|-----|-----|
##      H |      1 |      2 |      3 |
##      | 0.333 | 0.667 | 0.375 |
##      | 0.333 | 0.400 |      |
##      | 0.125 | 0.250 |      |
## -----|-----|-----|-----|
##      M |      2 |      3 |      5 |
##      | 0.400 | 0.600 | 0.625 |
##      | 0.667 | 0.600 |      |
##      | 0.250 | 0.375 |      |
## -----|-----|-----|-----|
## Column Total |      3 |      5 |      8 |
##      | 0.375 | 0.625 |      |
## -----|-----|-----|-----|
##
##
```

La leyenda **Cell Contents** explica los contenidos de cada una de las celdas de la tabla: la frecuencia absoluta (N), la frecuencia relativa por filas (N / row total), la frecuencia relativa por columnas (N / col total), y la frecuencia relativa global (N / table total). Video 89.

Ejemplo Sumas por filas y columnas

```
tt <- table(sex, ans)
tt# Frec. absolutas
```

```
##      ans
## sex N S
##  H 1 2
##  M 2 3
```

```
prop.table(tt)#Frec. Rel. Global
```

```
##      ans
## sex      N      S
##  H 0.125 0.250
##  M 0.250 0.375
```

```
prop.table(tt, margin = 1)#Frec. Rel. Por sexo
```

```
##      ans
## sex      N      S
##  H 0.3333333 0.6666667
##  M 0.4000000 0.6000000
```

```
prop.table(tt, margin = 2)#Frec. Rel. Por respuesta
```

```
##      ans
## sex      N      S
##  H 0.3333333 0.4000000
##  M 0.6666667 0.6000000
```

```
colSums(tt) #suma por columns Frec. absolutas
```

```
## N S
## 3 5
```

```
rowSums(tt) #suma por rows Frec. absolutas
```

```
## H M
## 3 5
```

```
colSums(prop.table(tt))
```

```
##      N      S
## 0.375 0.625
```

```
rowSums(prop.table(tt))
```

```
##      H      M
## 0.375 0.625
```

```
#función que proviene de las matrices vistas  
#es lo mismo que rowSums(tt)  
apply(tt, FUN = sum, MARGIN = 1)
```

```
## H M
## 3 5
```

```
#MARGIN = c(1,2) se aplica la función por elemento de la matriz  
apply(tt, FUN = sqrt, MARGIN = c(1,2))
```

```
##      ans
## sex      N      S
##  H 1.000000 1.414214
##  M 1.414214 1.732051
```