Estudiando Datos Agrupados

Santiago Pérez Moncada

4/7/2020

Frecuencias

Una primera consideración es tratar las clases obtenidas en el paso anterior como los niveles de una variable ordinal y calcular sus frecuencias.

- **La frecuencia absoluta** de una clase será el número de datos originales que pertenecen a la clase.
- **La frecuencia absoluta acumulada** de una clase será el número de datos que pertenecen a dicha clase o alguna de las anteriores.

Normalmente, las frecuencias de un conjunto de datos agrupados suele representarse de la siguiente forma.

- X_j marca de clase.
- n_j Frecuencia absoluta.
- N_j Frecuencia absoluta acumulada.
- f_j Frecuencia relativa.
- F_j Frecuencia relativa acumulada.

Intervalos	X_j	n_j	f_j	F_j
$[L_1, L_2)$	X_1	n_1	f_1	F_1
$[L_2, L_3)$	X_2	n_2	f_2	F_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$[L_k, L_{k+1})$	X_k	n_k	f_k	F_k

La Función `hist`

El cálculo de las frecuencias con `R` podemos hacerlo mediante las funciones, `table`,`prop.table` y `cumsum`.

También podemos utilizar la función `hist`, que internamente genera una list cuya componente `count` es el vector de frecuencias absolutas de las clases.Por lo consiguiente, para calcular estas frecuencias, podemos utilizar la sintaxis.

```
hist(x, breaks=..., right=FALSE, plot=FAlSE)$count
```

Conviene igualar el parámetro `breaks` al vector de los extremos del intervalo debido a que `cut` y `hist` hacen uso de diferentes métodos para agrupar los datos cuando se especifica solamente el número k de clases.

El resultado de `hist` incluye la componente `mids` que contiene el vector de puntos medios de los intervalos, es decir, nuestras marcas de clase.

Podemos automatizar el cálculo de la ya tan mencionada table de frecuencias, utilizando las dos funciones que mostramos a continuación.

La primera sirve en el caso en que vayamos a tomar las clases de la misma amplitud.Sus parámetros son: x, el vector con los datos cuantitativos; k, el numero de clases; A, su amplitud; y p la precisión de los datos($p = 1$ si la presición son unidades, $p = 0.1$ si la precisión son décimas de unidad...).

Por su parte, la segunda es para cuando conocemos los extremos de las clases.Sus parámetros son x el vector con los datos cuantitativos; L, el vector de extremos de clases y V un valor lógico, que ha de ser `True` si queremos que el último intervalo sea cerrado, y `FALSE` en caso contrario.

Primera función

```
TablaFrecs1 = function(x,k,A,p){
  L = min(x)-p/2+A*(0:k)
  x_cut = cut(x, breaks = L, right = FALSE)
  intervals = levels(x_cut)
  mc = (L[1]+L[2])/2 +A *(0:(k-1))
  Fr.abs = as.vector(table(x_cut))
  Fr.rel = round(Fr.abs/length(x),4)
  Fr.cum.abs = cumsum(Fr.abs)
  Fr.cum.rel = cumsum(Fr.rel)
  tabla = data.frame(intervals, mc, Fr.abs,Fr.cum.abs,Fr.rel,Fr.cum.rel)
  tabla
}
```

Segunda función

```
TablaFrecs2 = function(x,L,V){
  x_cut = cut(x, breaks = L, right = FALSE, iinclude.lowest = V)
  intervals = levels(x_cut)
  mc = (L[1:(length(L))-1]+L[2:(length(L))])/2
  Fr.abs = as.vector(table(x_cut))
  Fr.rel = round(Fr.abs/length(x),4)
  Fr.cum.abs = cumsum(Fr.abs)
  Fr.cum.rel = cumsum(Fr.rel)
  tabla = data.frame(intervals, mc, Fr.abs,Fr.cum.abs,Fr.rel,Fr.cum.rel)
  tabla
}
```

La tabla de frecuencias de la longitud de los petalos de iris.

```
petals = iris$Petal.Length

TablaFrecs1(petals, k = 6, A = 1, p = 0.1)
```

```
##     intervals   mc Fr.abs Fr.cum.abs Fr.rel Fr.cum.rel
## 1 [0.95,1.95) 1.45     50         50 0.3333     0.3333
```

```
## 2 [1.95,2.95) 2.45      0         50 0.0000     0.3333
## 3 [2.95,3.95) 3.45     11         61 0.0733     0.4066
## 4 [3.95,4.95) 4.45     43        104 0.2867     0.6933
## 5 [4.95,5.95) 5.45     35        139 0.2333     0.9266
## 6 [5.95,6.95) 6.45     11        150 0.0733     0.9999
```

```
TablaFrecs2(petals, L = c(1:7), V = FALSE )
```

```
##   intervals  mc Fr.abs Fr.cum.abs Fr.rel Fr.cum.rel
## 1     [1,2) 1.5     50         50 0.3333     0.3333
## 2     [2,3) 2.5      0         50 0.0000     0.3333
## 3     [3,4) 3.5     11         61 0.0733     0.4066
## 4     [4,5) 4.5     43        104 0.2867     0.6933
## 5     [5,6) 5.5     35        139 0.2333     0.9266
## 6     [6,7) 6.5     11        150 0.0733     0.9999
```

Ejemplo

Siguiendo con el ejemplo de las anchuras de los cangrejos, vamos a calcular sus tablas de frecuecnias haciendo uso de todo lo aprendido anteriormente.

```
data = read.table("../data/datacrab.txt",header = TRUE)
CW = data$width

k = nclass.Sturges(CW)
A = (max(CW)-min(CW))/k
A
```

```
## [1] 1.388889
```

```
A = 1.5
p = 0.1
TablaFrecs1(CW,k,A,p)
```

```
##     intervals   mc Fr.abs Fr.cum.abs Fr.rel Fr.cum.rel
## 1 [20.9,22.4) 21.7      2          2 0.0116     0.0116
## 2 [22.4,23.9) 23.2     21         23 0.1214     0.1330
## 3 [23.9,25.4) 24.7     37         60 0.2139     0.3469
## 4 [25.4,26.9) 26.2     48        108 0.2775     0.6244
## 5 [26.9,28.4) 27.7     38        146 0.2197     0.8441
## 6 [28.4,29.9) 29.2     18        164 0.1040     0.9481
## 7 [29.9,31.4) 30.7      6        170 0.0347     0.9828
## 8   [31.4,33) 32.2      2        172 0.0116     0.9944
## 9   [33,34.5) 33.7      1        173 0.0058     1.0002
```