

Estructura de datos en R - Listas

Ramon Ceballos

16/1/2021

Hasta ahora hemos trabajado con vectores, que son sucesiones de elementos de un mismo tipo.

Este problema se resuelve con las listas generalizadas o función `list()`

LISTAS EN R / FUNCIÓN LIST()

List. Lista formada por diferentes objetos, no necesariamente del mismo tipo, cada cual con un nombre interno

- `list(...)`: función que crea una list
 - Para obtener una componente concreta usamos la instrucción `list$componente`
 - También podemos indicar el objeto por su posición usando dobles corchetes: `list[[i]]`. Lo que obtendremos es una list formada por esa única componente, no el objeto que forma la componente

```
#Esto sería un vector como hemos trabajado hasta ahora
```

```
x = c(1,3,-4,6,8,13,-3)
```

```
x
```

```
## [1]  1  3 -4  6  8 13 -3
```

```
#Crear una lista emplando además el vector anterior para algunos datos
```

```
L = list(nombre = "Temperaturas", datos = x, media = mean(x), sumas = cumsum(x))
```

```
L
```

```
## $nombre
```

```
## [1] "Temperaturas"
```

```
##
```

```
## $datos
```

```
## [1]  1  3 -4  6  8 13 -3
```

```
##
```

```
## $media
```

```
## [1] 3.428571
```

```
##
```

```
## $sumas
```

```
## [1]  1  4  0  6 14 27 24
```

```
#Para obtener una determinada componente de la lista se emplea L$objeto
```

```
L$datos
```

```
## [1] 1 3 -4 6 8 13 -3
```

```
#También se puede acceder por L[[posición_objeto]]  
L[[2]]
```

```
## [1] 1 3 -4 6 8 13 -3
```

Funciones para una Lista

Hay dos funciones principales.

- `str(list)`: para conocer la estructura interna de una list. La naturaleza de los objetos que la componen.
- `names(list)`: para saber los nombres de la list.

```
str(L)
```

```
## List of 4  
## $ nombre: chr "Temperaturas"  
## $ datos : num [1:7] 1 3 -4 6 8 13 -3  
## $ media : num 3.43  
## $ sumas : num [1:7] 1 4 0 6 14 27 24
```

```
names(L)
```

```
## [1] "nombre" "datos" "media" "sumas"
```