Sesión 6.1

Atributos

*Mikel De Juan*

* [**Objetivo**](https://rstudio-pubs-static.s3.amazonaws.com/423785_12c7b2e6fae343e08c0446e6aab3d225.html#objetivo)
* [**Introduccion**](https://rstudio-pubs-static.s3.amazonaws.com/423785_12c7b2e6fae343e08c0446e6aab3d225.html#introduccion)
* [**Nombres**](https://rstudio-pubs-static.s3.amazonaws.com/423785_12c7b2e6fae343e08c0446e6aab3d225.html#nombres)
* [**Dimensiones**](https://rstudio-pubs-static.s3.amazonaws.com/423785_12c7b2e6fae343e08c0446e6aab3d225.html#dimensiones)
* [**Clases**](https://rstudio-pubs-static.s3.amazonaws.com/423785_12c7b2e6fae343e08c0446e6aab3d225.html#clases)

**Objetivo**

El objetivo de este markdown es conocer mas en profundidad el concepto de atributos en R.

**Introduccion**

Un atributo es una parte de informacion que puedes adjuntar a un vector atomico o cualquier objeto de R. El atributo no afectara a ningun valor del objeto. Podriamos denominar a los atributos como **metadata**.

Los atributos mas comunes de un vector atomico son los nombres, dimensiones y clases.

**Nombres**

die <- c(1,2,3,4,5,6)

names(die <- c("uno","dos","tres","cuatro","cinco","seis"))

## NULL

attributes(die)

## NULL

**Dimensiones**

Podemos transformar un vector atomico en una array de n-dimensiones dandole un atributo de dimension o **dim**. El primer valor siempre hara referencia a numero de filas y el segundo valor al numero de columnas.

dim(die) <- c(2,3)

die

## [,1] [,2] [,3]

## [1,] "uno" "tres" "cinco"

## [2,] "dos" "cuatro" "seis"

Aun asi, todo lo relacionado con dimensiones en matrices y arrays las podemos trabajar con funciones especificas.

Las **matrices** almacenan valores en un array bidimensional. Podemos establecer el numero de filas y columnas a traves de **nrow** o **ncol**.

m <- matrix(die, nrow = 2)

m

## [,1] [,2] [,3]

## [1,] "uno" "tres" "cinco"

## [2,] "dos" "cuatro" "seis"

m <- matrix(die, nrow = 2, byrow = TRUE)

m

## [,1] [,2] [,3]

## [1,] "uno" "dos" "tres"

## [2,] "cuatro" "cinco" "seis"

Por otro lado, las **arrays** crean un array de n-dimensiones.

ar <- array(c(11:14, 21:24, 31:34), dim = c(2,2,3))

ar

## , , 1

##

## [,1] [,2]

## [1,] 11 13

## [2,] 12 14

##

## , , 2

##

## [,1] [,2]

## [1,] 21 23

## [2,] 22 24

##

## , , 3

##

## [,1] [,2]

## [1,] 31 33

## [2,] 32 34

**Clases**

Cambiar las dimensiones de nuestro objetivo no cambiara el tipo de objetio, pero si su clase.

dim(die) <- c(2,3)

typeof(die)

## [1] "character"

class(die)

## [1] "matrix"

**Una lase es un caso especial de un vector atomico**.Por ejemplo, la matriz “die” es un caso especial de un vector “double”. Todos los elementos en la matriz son “doubles”, pero el elemento ha derivado en una nueva estructura.

Class() se puede aplicar a objetos que no disponen de un atributo de clase. Class deolvera un valor en base al tipo de objeto atomico.

Mas alla de la informacion basica sobre los objetos en R; el programa utiliza una clase especial para representar fechas y horas. Generalmente podemos encontrar dos clases denominadas “POSIXct” o “POSIXt”.

now <- Sys.time()

now

## [1] "2018-09-27 19:57:18 CEST"

typeof(now)

## [1] "double"

class(now)

## [1] "POSIXct" "POSIXt"

POSIXct es la forma mas comun de representar fechas y horas. En este formato, cada hora se representa con el numero de segundos que han pasado entre la hora y 12:00 AM 1 de enero de 1970.

unclass(now)

## [1] 1538071039

La mayoria de las clases solo son utiles en casos especificos. sin embargo, hay una clase que predomina ante todas que son los **Factores**