

Vectores y tipos de datos

Téllez Gerardo Rubén

13/3/2021

Un vector es una secuencia ordenada de datos. R contiene por ejemplo:

logical: lógicos (**TRUE** o **FALSE**)

integer: números enteros, \mathbb{Z}

numeric: números reales, \mathbb{R}

complex: números complejos, \mathbb{C}

character: palabras

Todos los datos tienen que ser del mismo tipo en el vector. Para mezclarlos habría que hacer listas

Una mezcla de datos definirá el vector según la jerarquía (inferior más relevante)

Vector básico

- `c()`: para definir un vector
- `scan()`: para definir un vector a partir de un documento, o leer manualmente
- `fix(x)`: para modificar visualmente el vector `x`

```
c(1,2,3)
```

```
## [1] 1 2 3
```

```
rep(c("Azúl"), 7) #la función rep() repite otra función, n veces
```

```
## [1] "Azúl" "Azúl" "Azúl" "Azúl" "Azúl" "Azúl" "Azúl"
```

```
a = c(2, TRUE, 3.25, 4i, "Rubén")
```

```
a
```

```
## [1] "2"      "TRUE"   "3.25"   "0+4i"   "Rubén"
```

```
fix(a) #Para modificarlo manualmente
```

```
a
```

```
## [1] "2"      "FALSE"  "3.25"   "0+4i"   "Rubén"
```

```
class(a) #Para identificar qué clase de vector es
```

```
## [1] "character"
```

Para definir un vector a partir de una URL con datos, sólo hace falta la función `scan()` y la url o txt

Función scan

Para crear un vector que contenga 3 copias de 1 9 9 8 0 7 2 6 con la función `scan()`:

```
# En consola  
#w <- scan()  
#1: 1 9 9 8 0 7 2 6  
#9: 1 9 9 8 0 7 2 6  
#17: 1 9 9 8 0 7 2 6  
#Read 24 items  
#[1] 1 9 9 8 0 7 2 6 1 9 9 8 0 7 2 6 1 9 9 8 0 7 2 6
```

Ejercicio

- Repetir mi año de nacimiento 10 veces

```
#> a = scan()  
#1: 2001  
#2: 2001  
#3: 2001  
#4: 2001  
#5: 2001  
#6: 2001  
#7: 2001  
#8: 2001  
#9: 2001  
#10: 2001  
#11:  
#Read 10 items
```

- Crear el vector que contenga entradas 16, 0, 1, 20, 1, 7, 88, 5, 1, 9, llamarlo `vec` y modificar la cuarta entrada con la función `fix()`, para cambiar 20 por 12

```
vec <- c(16, 0, 1, 20, 1, 7, 88, 5, 1, 9)  
vec
```

```
## [1] 16 0 1 20 1 7 88 5 1 9
```

```
fix(vec)  
vec
```

```
## [1] 16 0 1 12 1 7 88 5 1 9
```

- Si hubiera que cambiar el separador decimal, en la función `scan()` indicar el (`dec=`) Ejemplo:
`scan(dec=",")`
- Para especificar de antemano el tipo de vector, usar el parámetro (`what=`) Ejemplo: `scan(what="character")`

Para hacer una repetición de varios valores con `rep`:

```
s <- rep(c(78, 79, 80), 10)
s
```

```
## [1] 78 79 80 78 79 80 78 79 80 78 79 80 78 79 80 78 79 80 78 79 80 78
## [26] 79 80 78 79 80
```