

vectores, matrices, data frames & listas

Armando Ocampo

Vectores

Los vectores son representaciones lineales de la información. Es decir, solo tienen una dimensión. Los vectores pueden tener elementos de tipo carácter, numéricos continuos, numéricos enteros, fechas, o una mezcla de varios elementos. Para unir varios elementos utilizamos la función `c()`.

Nota: si no sabes utilizar una función, recuerda que puedes buscarla en internet o con los siguientes comandos `?c` `help(c)`

```
# vector tipo caracter
c("a", "b", "c", "d")
# vector numerico continuo <-
c(1.2,1.3,1.4,1.5)
# vector numerico entero
c(1,2,3,4,5,6,7)
# vector mixto
c("a", "b",1, "c",1.3,5)
```

Nota 1: los elementos de tipo carácter van entre comillas, los vectores de tipo numérico se escriben sin comillas. Nota 2: puedes o no colocar un espacio entre cada elemento, el agregar un espacio no afecta al vector

Hasta este momento nuestros vectores solo se han proyectado en la consola. Para guardarlos en el ambiente utilizamos “<-” o “=”. Recuerda que el atajo para asignar es “alt” + “-”. Vamos a crear un vector con números del 1 al 5. Llamaremos a este vector *numeros*

```
numeros <- c(1,2,3,4,5)
```

Este vector con una secuencia numérica corta es fácil de realizar de forma manual, pero ¿qué sucede si nos piden realizar un vector con números del 1 al 100? Para esto tenemos un atajo, los dos puntos “:” permiten hacer secuencias solo necesitamos el valor mínimo y el valor máximo

```
c(1:100)
```

De hecho, podemos hacer secuencias con diferente rango, separando cada secuencia con una coma.

```
c(1:5,15:40,50:100)
```

Nota: las comas sirven para separar argumentos, solo debes tener cuidado que no omitas o pongas una coma extra cuando haces tu código.

Los vectores numéricos también pueden utilizarse en expresiones matemáticas. Si sumamos un número a un vector, el número se sumará a cada uno de los elementos del vector, generando un nuevo vector con el producto de la expresión matemática.

```
# sumando 5 a cada elemnto del vector numeros
numeros + 5
# restando 15 a cada elemento del vector numeros
numeros - 15
# multiplicando por 10
numeros * 10
```

Este resultado solo se observa en la consola, podemos guardarlo asignandolo a un objeto

```
vectorcito <- numeros + 5
```

Tambien podemos seleccionar un elemento especifico de cada vector a partir del uso de []. Para seleccionar el tercer elemento del vector numeros se realiza lo siguiente

```
numeros[3]
# seleccion del cuarto elemento del vector vectorcito
vectorcito[4]
# si queremos conocer dos elementos utilizamos la funcion concatenar c() dentro de los
# corchetes. Por ejemplo, seleccionaremos los elementos tres, cuatro y cinco del vector
#vectorcito
vectorcito[c(3,4,5)]
# e incluso podemos realizar comparaciones y saber que elemento cumple cada condicon

# en el siguiente ejemplo conoceremos los elementos que son mayor que 8 en el vector
#vectorcito
vectorcito > 8
# de esta forma podemos seleccionar los elementos que cumplen la condicion
vectorcito[c(4,5)]
```

tarea: realiza un vector de tipo caracter con 5 nombres y guardalo con el nombre personas, realiza un vector con numeros del 5 al 35, realiza un vector de tipo caracter con 5 animales.

Matrices

Las matrices son elementos rectangulares en dos dimensiones que almacenan informacion de un solo tipo. Para crear una matriz se necesita la funcion matrix(). Si no sabes utilizar esta funcion utiliza el comando help(matrix) o ?matrix

```
matrix(data = (1:9), nrow = 3, byrow = TRUE)

matrix(data = (1:9), nrow = 3, byrow = FALSE)
```

Esta informacion se puede almacenar como un objeto

```
mi_primer_matriz <- matrix(data = (1:9), nrow = 3, byrow = TRUE)
```

Las matrices tambien pueden tener una evaluacion a partir de funciones aritméticas. Cuando sumamos una matriz con un numero, este se sumará a cada elemento de la matriz

```
mi_primer_matriz + 5
# tambien podemos hacer restas
mi_primer_matriz - 20
# multiplicaciones
mi_primer_matriz * 50
```

Al igual que los vectores, podemos seleccionar un elemento en específico dentro de la matriz a partir del uso de corchetes []. Recordando que las matrices son elementos en dos dimensiones, necesitamos colocar dos números dentro de los corchetes. El primero corresponde a las filas y el segundo a las columnas. A continuación seleccionaremos el elemento de la fila 2 de la columna 3 del elemento `mi_primer_matriz`

```
mi_primer_matriz[2,3]
```

De la misma forma, podemos hacer comparaciones. Evaluaremos que elementos de `mi_primer_matriz` son mayores a 6

```
mi_primer_matriz > 6
# en este caso vemos que toda la fila 3 cumple con esta condicion, para seleccionar una
# fila completa colocamos la fila en el corchete y el segundo elemnto lo dejamos en blanco
mi_primer_matriz[3,]
# con las columnas se hace algo similar, solo que el primer elemento se deja en blanco y el
# segundo numero es la columna que seleccionaremos. A continuacion seleccionaremos los
# elementos de la columna 2
mi_primer_matriz[,2]
# No obstante, tambien podemos colocar la condicion dentro de los corchetes y saber
# los elementos que se cumple la condicion
mi_primer_matriz[mi_primer_matriz > 6]
# podemos hacer la condicion como queramos
mi_primer_matriz[mi_primer_matriz == 8]
mi_primer_matriz[mi_primer_matriz > 4]
# queremos obtener los elementos mayores a 4 y menores a 8
mi_primer_matriz[mi_primer_matriz > 4 & mi_primer_matriz < 8]
```