

Introducción a R

Armando Ocampo

Primeros pasos

R permite realizar operaciones aritméticas

```
2+2
2+4+6+8
5-3
6-4-2+5
8*2
10*20
20*15*2
10/2
20/5
```

E incluso realizar comparaciones

```
2 > 5
10 > 2
5 < 4
8 < 20
5 == 4
5 == 2+3
10 >= 10
8 <= 8
```

Primeros operadores booleanos

```
TRUE
FALSE
```

Los operadores booleanos pueden tener un valor. TRUE equivale a 1, mientras que FALSE vale 0

```
TRUE + TRUE
1 == TRUE
FALSE + FALSE
0 == FALSE
TRUE > FALSE
```

Guardando objetos

Hasta este momento, todos nuestros resultados se han quedado en la consola. A veces, necesitamos guardar ciertos resultados para utilizarlos después. Para esto lo asignamos como un objeto que se guardará en el

ambiente. Para guardar un objeto se utiliza “<-” o “=”. Ambos funcionan de la misma forma. Nota: “<-” se puede escribir de forma rápida usando alt + -

```
respuesta <- 2 + 5  
suma = 4 + 8
```

De esta manera podemos utilizar el resultado de nuestros objetos Nota: el nombre de cada objeto es libre, teniendo incluso combinación entre numeros, letras, mayúsculas y minúsculas ej. “resultado1”, “resultado2”, “VecTor2”, “VectoR” No obstante, nunca debe llevar espacios. Los puntos y los guiones bajos te pueden servir para separar ideas ej. “SNPS_demencia_vascular”, “base.datos.demencia” Por último, nunca asignes un objeto a un número aislado ej “2 <- 2+2”, esto te generará un error.

Regresemos a la parte de trabajar con objetos

```
respuesta + 4  
suma + respuesta  
respuesta - suma
```

Y sí, también podemos guardar este nuevo resultado

```
nuevo <- suma + respuesta  
datos_nuevos <- 4+5+suma+respuesta
```

Primeras funciones

R permite realizar una serie de funciones de forma automática. Podemos obtener promedios, ds, varianza, etc

Estas funciones vienen precargadas con el lenguaje y el visor de trabajo. Comenzaremos con las funciones c() y sum()

La función c() permite concatenar o combianr una serie de elementos numericos o de tipo caracter. Nota: los elementos de tipo caracter siempre van entre comillas

```
c(2,5,6,8)  
c("a", "b", "c", "d")
```

Estos elementos tambien se pueden guardar en un objeto

```
vectorcito <- c(2,4,6,8,7)
```

¿Qué pasa si no sé usar una función? Para esto podemos usar la funcion help() o el signo de interrogación

```
help(c)  
?c
```

Ambas líneas abrirán un apartado de ayuda con la descripción de la función Nota: te recomiendo realizar esto con las funciones que utilizas, para que empieces a familiarizarte con R

R tiene funciones pre cargadas, como sum(), mean(), sd(). Las cuales nos permiten obtener la suma de un vector, la media aritmética y la desviación estándar. Antes de usarlas, veremos la seccion de ayuda con la descripcion de cada función.

```
?sum  
?mean  
?sd
```

Nota: todas las funciones comparten el argumento `na.rm = FALSE`, el cual permite trabajar o no con argumentos faltantes. Hablaremos de esto más adelante

Usaremos a `vectorcito` con cada una de estas funciones

```
sum(vectorcito)  
mean(vectorcito)  
sd(vectorcito)
```

No obstante, también podemos colocar una función, dentro de la función. Tal vez suene redundante, pero en el ejemplo siguiente vamos a utilizar la función `c()`, dentro de la función `sum()`

```
sum(c(2,5,6,7,8))
```

Uno de los argumentos de nuestras funciones es `na.rm = FALSE`, esto significa que no eliminará valores faltantes. Vamos por un ejemplo

```
vectorcito2 <- c(2,8,7,NA,7)  
sum(vectorcito2)
```

Este argumento genera un `NA` como resultado. Para obtener el resultado a pesar de tener un valor vacío, utilizaremos el argumento `na.rm` pero lo cambiaremos a `= TRUE`

```
sum(vectorcito2, na.rm = TRUE)
```

De esta forma estamos escribiendo que eliminemos los valores `NA` del vector.

También podemos hacer esto sin necesidad de crear un objeto y eliminar los `NA`'s, solo hay que tener cuidado con las comas para separar cada función.

```
sum(c(2,8,7,NA,7), na.rm = TRUE)
```

Instalando paqueterías

Cargando paqueterías