

María Paula Llano

Departamento de Ciencias de la
Atmósfera y los Océanos

Laboratorio de Procesamiento de Información Meteorológica

Funciones

Subrutinas: sentencias entre llaves

En general, una subrutina es un segmento de código que se escribe sólo una vez pero puede invocarse o ejecutarse muchas veces.

Cuando agrupamos comandos o expresiones entre llaves {expre.1;expre.2; ...; expre.m}, las expresiones pueden ir:

- separadas por ; en la misma línea o
- separadas por cambio de línea

En ambos casos el valor resultante de la subrutina es el resultado de la última expresión evaluada. Puesto que una subrutina es también una expresión, podemos incluirla entre paréntesis y usarla como parte de otra expresión.

La mejor manera de poder ejecutar expresiones de varias líneas es a través de script.

Funciones y expresiones

La gran utilidad de las expresiones es que nos permiten ejecutar varios comandos de una única vez.

Pero donde gana mayor utilidad esta forma de trabajar es a la hora de crear nuevos objetos que ejecuten diversas expresiones utilizando como entrada unos objetos (argumentos) y devolviendo otros objetos.

Estos objetos (cuyo modo es **function**) constituyen las nuevas funciones de R, que se pueden utilizar a su vez en expresiones posteriores.

*Aprender a escribir funciones que nos puedan ser de futura utilidad es una de las mejores formas de conseguir que el uso de **R** nos sea cómodo y productivo.*

Función

En R las funciones constituyen una clase de objetos.

Los objetos de esta clase pueden ser asignados a variables, incluso armar una lista cuyos elementos fueran funciones.

nombre <- function(arg_1, arg_2, ..., arg_n) {expresión}

Le asignamos un **nombre**.

Una **expresión**, o un conjunto de expresiones (entre llaves {}) (es lo que hace la función)

Evalúamos en la expresión a los **argumentos** (arg_)

El valor de la expresión (la salida de la función) puede ser un número, un vector, un gráfico, una lista, un mensaje...

Para conocer los argumentos de una función podemos utilizar dos sentencias:

- **args**(nombre de la función)
- **formals**(nombre de la función)

La primera nos da la lista de argumentos que son necesarios para ejecutar la función.

La segunda nos brinda más información, nos da la lista de argumentos y si alguno de ellos tiene valores asignados por omisión, también nos da ese valor.

Funciones de control

- R nos permite utilizar funciones que controlen y frenen el funcionamiento de una función.
- Podemos hacer que compruebe “algo” y si no es un error grave que nos de un mensaje de “warning” y la función se continúe ejecutando.
- Podemos frenar mediante la función “stop”, nos devuelve un mensaje de error y frena la ejecución.
- También podemos emplear la función “missing” cuando algún argumento no fue especificado.