

Representación de datos: tablas y gráficas

En este apartado veremos cómo podemos representar los datos almacenados en R, centrándonos principalmente en Tablas data.frame y en gráficas. En este apartado vamos a emplear repetidamente los **vectores** por lo que recomendamos consultar el apartado en el que se explican si no se saben manejar correctamente.

Dada su utilidad, ambas funciones son muy empleadas en el programa, siendo especialmente frecuente la aparición de gráficas en los ejercicios de R.

<https://www.youtube.com/watch?v=yDuZlBT6ofE&t=1s>

Tablas Data.frame

Si lo que queremos es ordenar la información en forma de tablas, usaremos el comando data.frame. En él, los vectores que queramos representar (pueden contener tanto números como letras), se ordenarán en forma de **columnas**, y obtendremos tantas filas como componentes tenga el vector. Es por esto que es importante que todos los vectores tengan la misma longitud. El título del vector dará nombre a las columnas de la tabla final.

```
> Nombre<-c('Olivia', 'Irene', 'Marco', 'Blanca')
> Edad<-c(19,18,15,22)
> Altura<-c(1.64,1.59,1.76,1.70)
> TablaDatos<-data.frame(Nombre, Edad, Altura)
> TablaDatos
  Nombre Edad Altura
1 Olivia   19   1.64
2 Irene   18   1.59
3 Marco   15   1.76
4 Blanca  22   1.70
> |
```

Gráficas:

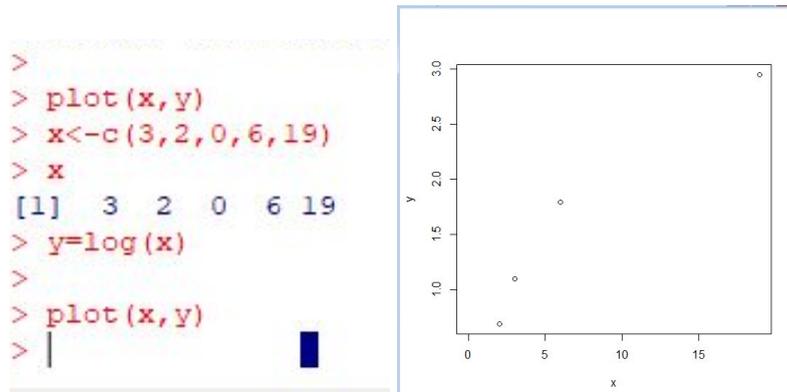
Pasamos ahora a la representación gráfica de datos. En primer lugar, el comando principal utilizado para la representación de funciones es **plot(x,y)**. En él, se deberán establecer previamente las variables x, y (en forma de vectores).

Sin embargo, al tratarse de una función en la que y es la variable dependiente de x, este parámetro podrá indicarse en función de la x.

Nota: es importante que ambos vectores tengan el mismo número de componentes, si no, el programa dará error (ver **errores frecuentes**).

Por ejemplo, para representar la función “ $f(x)=\log(x)$ ” se seguirán los siguientes pasos:

- Definir la variable x, como un vector tal que: $x<-c(3,2,0,6,19)$
- Definir la variable y, en función de x: $y=\log(x)$
- Finalmente, se introduce el comando plot: $\text{plot}(x,y)$



Sin embargo, a la hora de conseguir una representación “completa” de una función este método no resulta el más efectivo.

Para conseguir la representación total, se empleará el comando “seq”, el cual genera un número determinado de valores aleatorios dentro de un intervalo. A la hora de emplearlo, se deberá incluir la siguiente información seq(a, b, length=c), siendo a el valor inicial del intervalo, b el valor final y c el número de valores que queremos obtener.

A partir de este vector, se puede obtener una función mucho más definida.

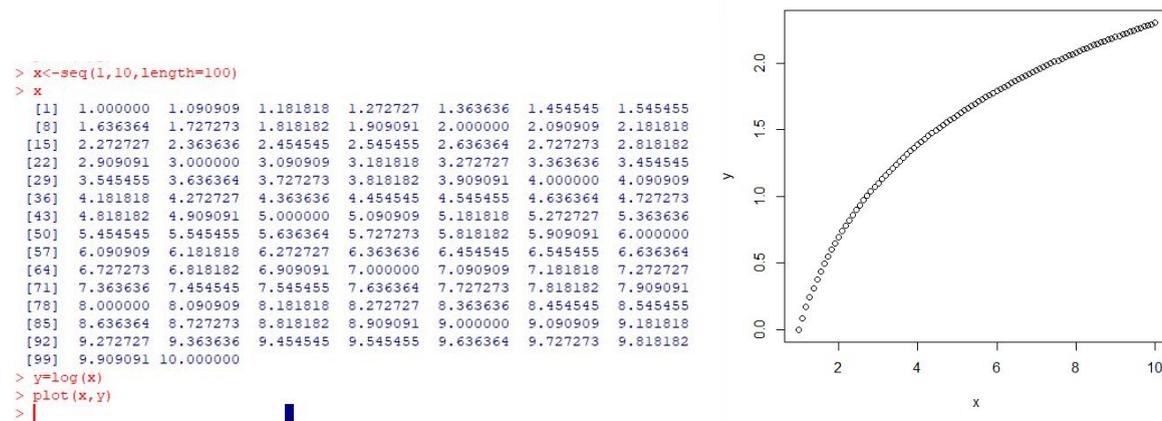
Como ejemplo:

Se obtiene un vector de 100 componentes, variando del 1 al 10, que actuará de variable x:

`x<-seq(1,10,length=100)`

De nuevo, la variable y se establece a partir de x: `y=log(x)`.

Finalmente, se crea de nuevo la función: `plot(x,y)`



Hay diversos comandos que pueden usarse para modificar la función, para ver una explicación en detalle consultar [“comandos y herramientas útiles”](#).

Incluimos aquí un breve resumen de las opciones:

