

Introducción a los gráficos en R

El R es una herramienta muy potente para hacer gráficos. Hay dos tipos de funciones gráficas: las de alto nivel que crean una nueva gráfica y las de bajo nivel que agregan elementos a un gráfico existente. Además, se pueden modificar los parámetros gráficos.

Parámetros gráficos (función *par*)

Con la función *par()* modificamos los parámetros gráficos. Puede ser invocada previamente a cualquier gráfico. En general, sus argumentos pueden ser especificados directamente cuando se crea el gráfico pero depende del tipo de gráfico.

Algunos de los muchos parámetros que se pueden modificar son:

col: Especifica el color de los símbolos usados en el gráfico. Se escriben los nombres de los colores en inglés como cadena de caracteres (entre comillas). Una lista de los colores disponibles se puede obtener con la función *colors()*

las: Especifica el estilo de las etiquetas de las anotaciones en los ejes:

- 0: paralelo al eje (valor por defecto)
- 1: horizontal
- 2: perpendicular al eje
- 3: vertical

pch: Especifica el carácter a utilizar al graficar puntos. Puede ser un número (que refiere a un determinado carácter para el R, por ejemplo *pch=23* es un diamante relleno) o cualquier símbolo entre comillas.

mfrow: Parte el dispositivo gráfico en distintas áreas. El argumento es un vector con el número de filas y de columnas en el que se desea partir: *c(nrow,ncol)*

lty: Es el tipo de línea (punteada, etc)

Ejemplo:

```
par(col="red",mfrow=c(1,3), las=1, pch="*")
```

La función *plot*

Es una función genérica para hacer gráficos en R

plot(x,y,...)

Argumentos

x: es el vector de coordenadas que se grafican en el eje de las abscisas (horizontal)

y: es el vector de coordenadas que se grafican en el eje de las ordenadas (vertical)

x e y tienen que ser del mismo largo

En vez de esto se le puede dar como argumento una matriz de dos columnas o una lista con dos elementos. Para algunas clases de objetos como series temporales (*ts*), hojas de datos

(*data.frame*) o factores (*factor*) la función `plot` devuelve un gráfico ajustado a las características de esa clase de objetos (por ejemplo, un gráfico de barras para el factor).

Opcionales:

type: es el tipo de gráfico. Los posibles son: "p" (puntos), "l" (líneas), "b"(puntos y líneas),"h"(histograma), etc. Por defecto es "p".

main: es el título del gráfico

xlab: es el nombre del eje de las abscisas

ylab: es el nombre del eje de las ordenadas

xlim: es un vector con el mínimo y el máximo del eje de las abscisas

ylim: es un vector con el mínimo y el máximo del eje de las ordenadas

axes: argumento lógico. Si es FALSE no dibuja los ejes.

Ejemplo:

```
x=seq(1,20,0.5)
```

```
y=x^2
```

```
plot(x,y,type="l",main="Gráfico de una parábola", xlab="eje de las x",ylab="f(x)=x^2", axes=T)
```

Nota: existen muchas otras funciones gráficas de primer nivel en R como: `curve`, `hist`, `truhist{MASS}`, `histogram{lattice}`, `boxplot`, `pie`, `stripchart`, `qqplot`, `coplot`, `barplot`, etc, tanto univariados como multivariados (ver manuales)

Algunas funciones gráficas de bajo nivel

Se añaden a los gráficos existentes (antes de cerrar los dispositivos gráficos -*graphic devices*)

points(x,y): Añade una secuencia de puntos al gráfico en esas coordenadas.

x e y son dos vector de igual largo.

x contiene las coordenadas de las abscisas e y de las ordenadas.

lines(x,y): Añade líneas al gráfico actual

x e y son dos vector de igual largo y son las coordenadas de los puntos que se van a unir con líneas.

abline(h=y): Añade una recta horizontal de altura y

abline(v=x): Añade una recta vertical de altura x

abline(a,b): Añade una recta con ordenada en el origen a y pendiente b.

axis(side, ...): Dibuja el eje indicado con el argumento side. Si es 1

text(x,y,labels): Escribe los nombres o etiquetas al lado de cada punto en el par de coordenadas dado por x e y

polygon(x,y): Dibuja un polígono en las coordenadas dadas.

Ejemplo:

```
x=seq(0,2*pi, len=51) #en vez de darle como argumento by que establece cuál va a ser la separación entre puntos, le doy como argumento len que determina la longitud del vector a crear.
```

```
plot(x, sin(x),col="red",type="l")
```

```
abline(h=0,col="blue")
```

```
a=seq(-6,6,2)
```

```
b=seq(-1,1,(1/3))
```

```
points(a,b,pch=23,col="violet")
```

```
lines(a,b,col="green")
```