

GRÁFICOS

INTRODUCCIÓN

La creación de gráficos es una cosa muy útil ya que nos permite expresar datos numéricos de una forma muy visual, lo que facilita la extracción de conclusiones y el entendimiento de estas.

A lo largo de este recurso se explican algunas de las muchas posibilidades que R tiene para de crear un gráfico y personalizarlo.

ÍNDICE

¿CÓMO CREAR UN GRÁFICO?

TIPOS DE GRÁFICAS

PERSONALIZACIÓN DEL GRÁFICO

- SIMBOLOS
- LÍNEAS
- TÍTULOS DE LOS EJES
- LIMITES DE LOS EJES
- COLORES
- ANOTACIONES

COMBINAR GRÁFICOS

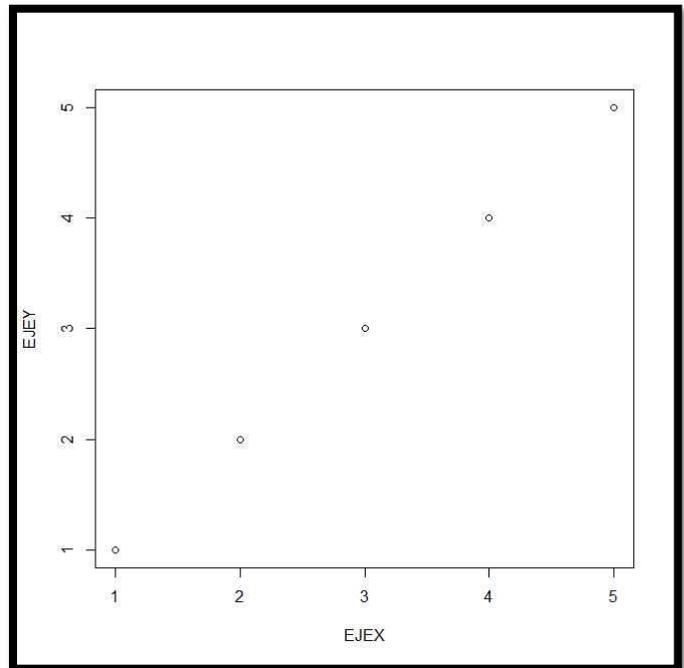
¿CÓMO CREAR UN GRÁFICO?

Para crear un gráfico en R se utiliza **la función plot** de la siguiente manera:

```
EJEX = c(1,2,3,4,5)  
EJEY = c(1,2,3,4,5)  
plot( EJEX, EJEY )
```

Lo primero que pongas tiene que ser un vector con los valores que quieres que tenga la x

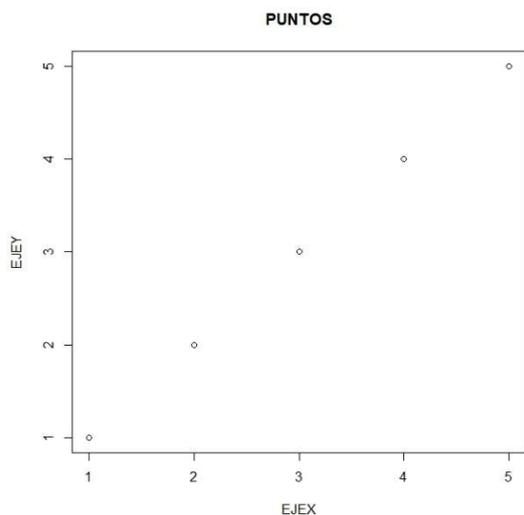
Lo segundo que pongas tiene que ser un vector con los valores que quieres que tome la y en los puntos de la x que has puesto antes.



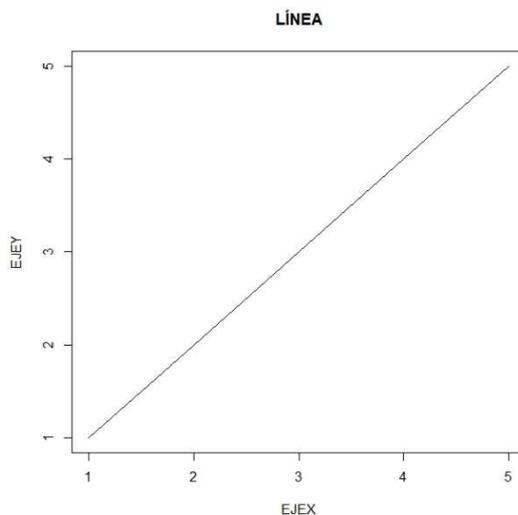
TIPOS DE GRÁFICAS

Para definir el tipo de gráfica que queremos utilizamos el parámetro **type** dentro del plot.

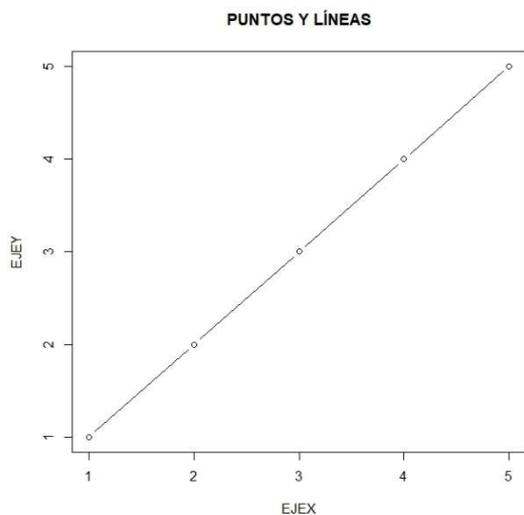
SÓLO PUNTOS → type = 'p'



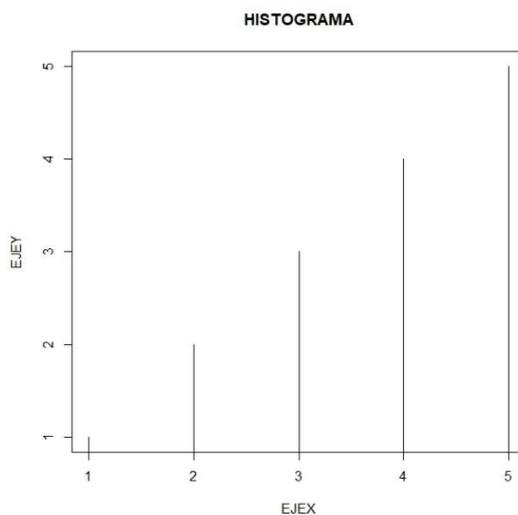
LÍNEA → type = 'l'



PUNTOS Y LÍNEA → type = 'b'



HISTOGRAMA → type = 'h'



EJEX = c(1,2,3,4,5)

EJEY = c(1,2,3,4,5)

plot(EJEX, EJEY , type='b')

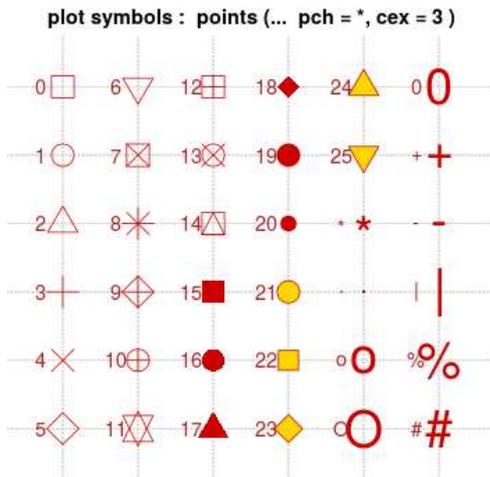
NO OLVIDAR COMILLAS "

PERSONALIZACIÓN DEL GRÁFICO

SIMBOLOS

Para cambiar el símbolo de los puntos utilizamos el parámetro **pch** dentro del plot.

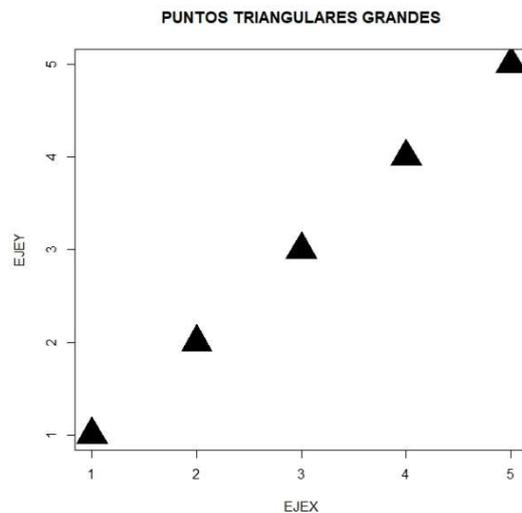
Hay muchos tipos de puntos diferentes algunos de ellos son:



Los símbolos **21 a 25** se pueden especificar con un color de fondo (distinto al del borde), haciendo uso del argumento **bg**.
Ej. `Bg= 'green'`

Además de cambiar el símbolo podemos hacer que este sea más grande o más pequeño con el parámetro **cex**.

```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
plot( EJEX, EJEY , type='p', pch= 17, cex= 4 )
```



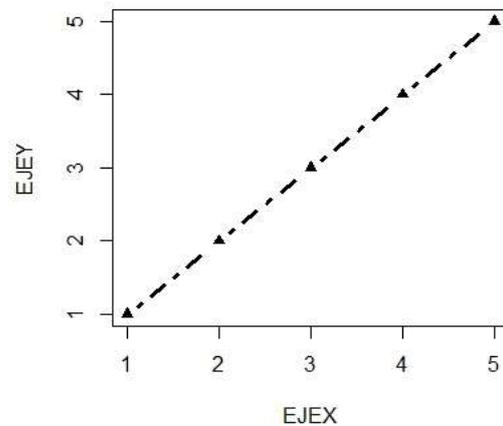
LÍNEAS

0. 'blank'	
1. 'solid'	_____
2. 'dashed'	- - - - -
3. 'dotted'
4. 'dotdash'	-. -. -. -. -. .
5. 'longdash'	- - - - -
6. 'twodash'	- - - - -

Las líneas en los gráficos también pueden personalizarse. Se usan los parámetros **lty** para determinar el tipo de línea y **lwd** para definir el ancho de la misma.

```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
plot( EJEX, EJEY , type='b', pch= 17, lty=4, lwd= 3)
```

LÍNEA GRUESA DISCONTINUA



TÍTULOS DE LOS EJES

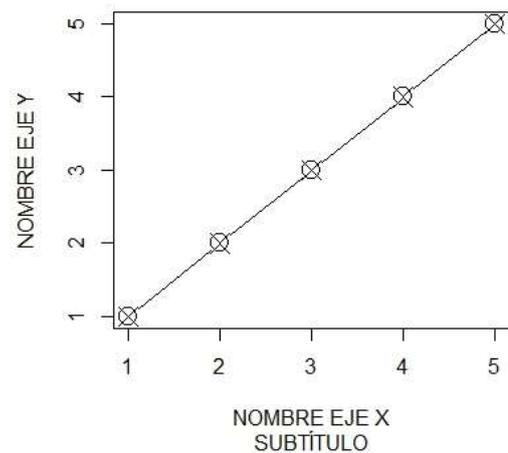
Para nombrar los ejes y ponerle el título al gráfico se usan los siguientes parámetros:

- **main** → Para poner el título al gráfico.
- **sub** → Para añadir un subtítulo.
- **ylab** y **xlab** → Para nombrar los ejes.

NO OLVIDAR COMILLAS "

```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
plot( EJEX, EJEY , type='b', pch= 13, cex= 2, main= 'TÍTULO', sub= 'SUBTÍTULO',
      xlab= 'NOMBRE EJE X', ylab= 'NOMBRE EJE Y')
```

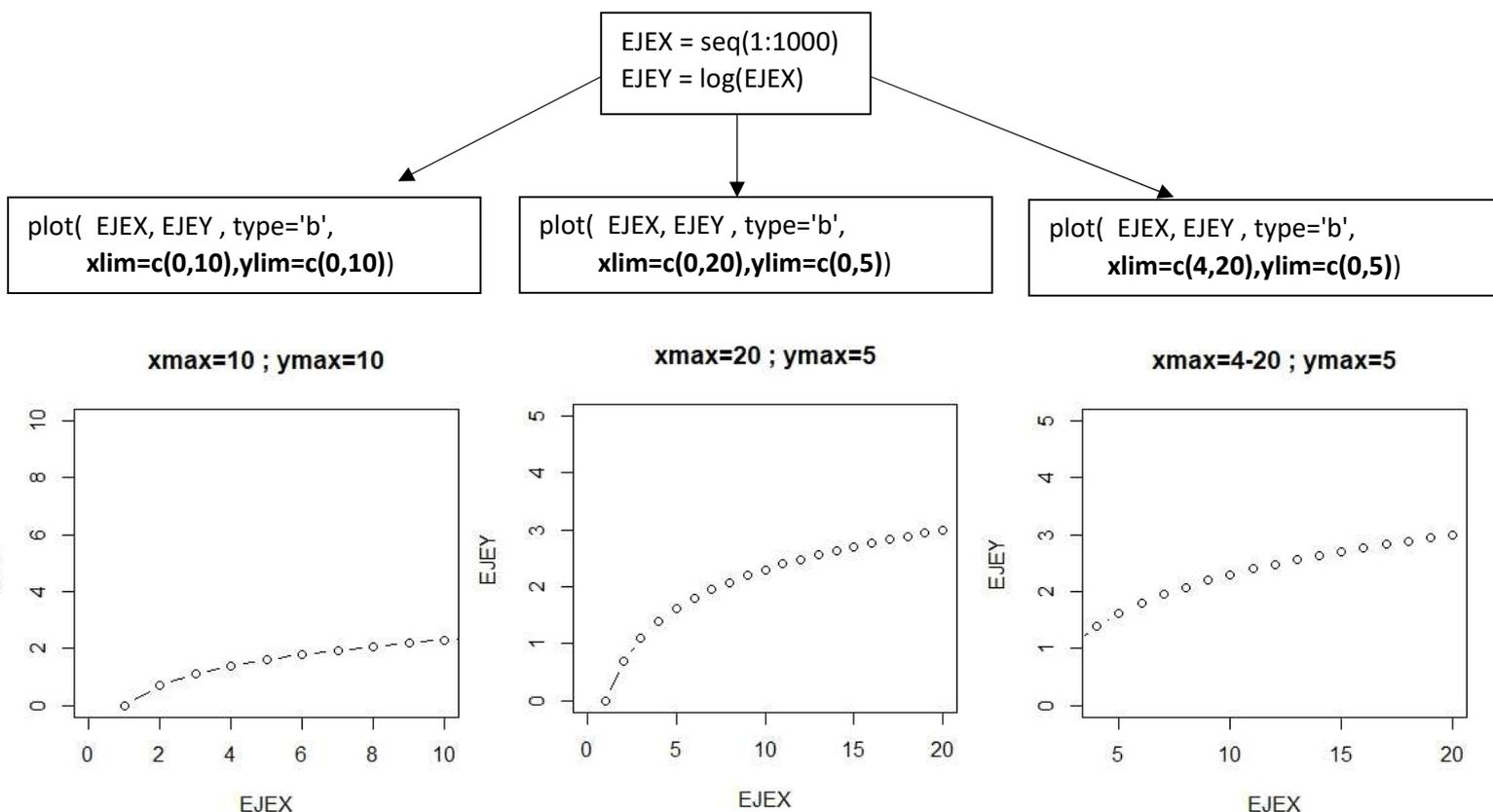
TÍTULO



Si quisiéramos que no me nombrara los ejes definiría:
xlab="" o ylab=""

LIMITES DE LOS EJES

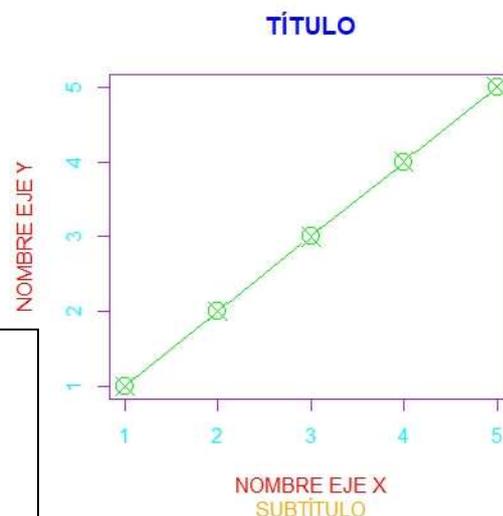
A veces nos puede interesar centrar nuestra representación en un intervalo concreto. Para ello podemos utilizar los parámetros **xlim** e **ylim** que nos permiten especificar los límites (tanto en el eje x como en el y) de lo que se va a representar. Estos parámetros suelen tomar como valor un vector de dos números (el inicio y el fin del intervalo que queremos que se muestre):



COLORES

Otra de las cosas que R nos permite a la hora de representar gráficas es cambiar los colores de los diferentes elementos de la gráfica. Para ello utilizo los siguientes parámetros especificados a continuación y pongo detrás el nombre del color que quiero **en inglés y con comillas**. (p.ej, col= 'green')

- Color gráfica → **col**
- Color etiquetas de los ejes → **col.lab**
- Color título → **col.main**
- Color subtítulo → **col.sub**
- Color ejes → **col.axis**
- Color de la caja → **fg**



```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
plot( EJEX, EJEY , type='b', pch= 13, cex= 2, main= 'TÍTULO', sub= 'SUBTÍTULO',
      xlab= 'NOMBRE EJE X', ylab= 'NOMBRE EJE Y', col='green', col.lab='red',
      col.main='blue', col.sub='orange', fg='purple', col.axis='cyan')
```

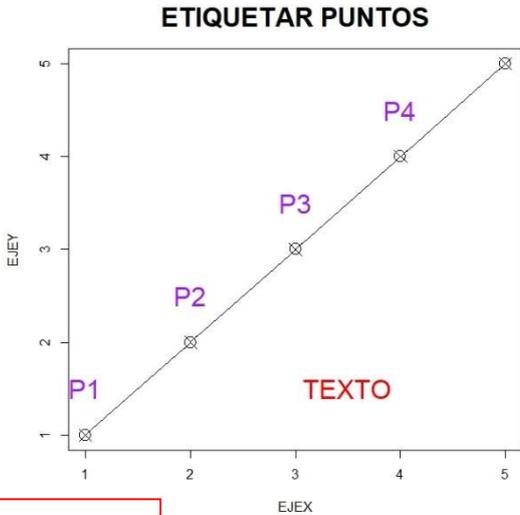
En caso de que necesites hacer algunas anotaciones en el gráfico puedes utilizar **la función text**, cuyo primer argumento es la coordenada del eje X, el segundo la del eje Y y el tercero la anotación, recuerda que **siempre con comillas**.

```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
plot( EJEX, EJEY, type='b', pch= 13, cex= 2, cex.main=2,
      main= 'ANOTACIONES PERSONALIZADAS')
text( x=3.5, y = 1.5, cex=2, col='red', "TEXTO")
```



Además también puedes poner **etiquetas a los diferentes puntos del gráfico**. Para ello previamente creas un vector con los nombres de los puntos y lo metes en una nueva función text.

```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
ETIQUETAS= c( 'P1', 'P2', 'P3', 'P4')
plot( EJEX, EJEY, type='b', pch= 13, cex= 2, cex.main=2,
      main= 'ETIQUETAR PUNTOS')
text(x=3.5, y = 1.5, cex=2, col='red', "TEXTO")
text(x=EJEX, y = EJEY + 0.5, cex=2, col='purple', ETIQUETAS)
```



Pongo el +0,5 para que la etiqueta aparezca encima del punto y no sobre él

COMBINAR GRÁFICOS

Para hacer que dos representaciones gráficas estén en el mismo gráfico primero definimos una y luego añadimos el comando **par (new = 'true')** y a continuación la otra representación que queremos introducir en el gráfico.

```
EJEX = c(1,2,3,4,5)
EJEY = c(1,2,3,4,5)
F2 = function(x){
  x^2
}
plot( EJEX, EJEY , type='b', pch= 2, col='green',xlab="", ylab="",xlim=c(-1,4),ylim=c(0,4))
par(new = 'true')
plot( F2 , type='l', col='red', xlab='x', ylab='y',xlim=c(-1,4),ylim=c(0,4))
title('COMBINAR GRÁFICOS')
```

