

Introducción a la Representación Gráfica

Ramon Ceballos

19/1/2021

Añadir elementos a un gráfico

1. Añadir puntos y rectas al gráfico

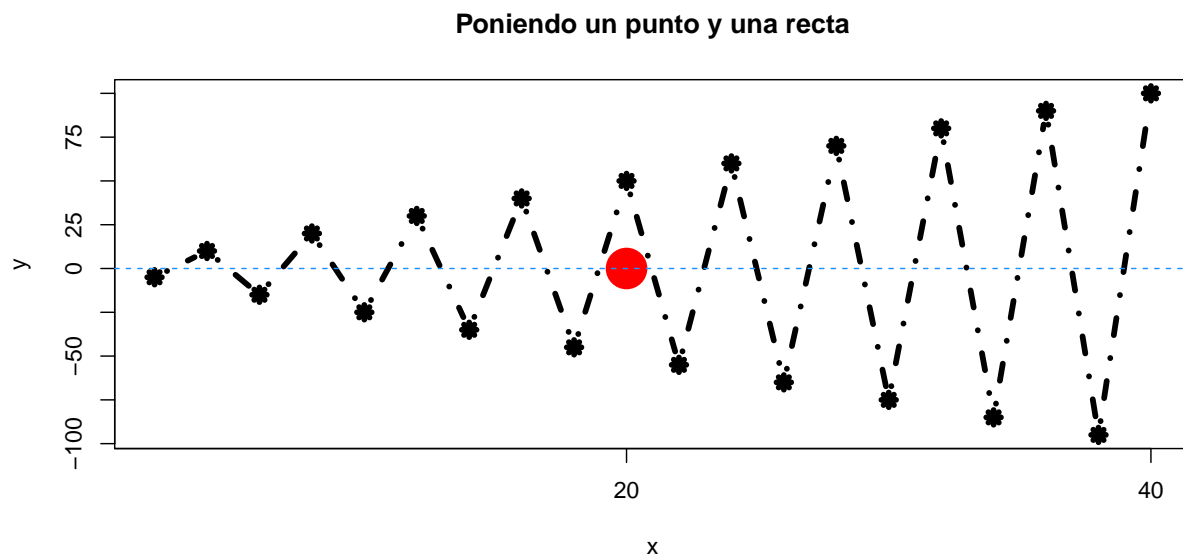
Existen dos instrucciones que se deben de conocer para añadir elementos a un gráfico:

- `points(x,y)`: añade un punto de coordenadas (x,y) a un gráfico ya existente. También se pueden añadir vectores de puntos.
- `abline`: para añadir una recta a un gráfico ya existente
 - `abline(a,b)`: añade la recta $y = bx + a$. Es de regresión lineal
 - `abline(v = x0)`: añade la recta vertical $x = x_0$. v puede estar asignado a un vector
 - `abline(h = y0)`: añade la recta horizontal $y = y_0$. h puede estar asignado a un vector

(**Importante**) Una vez se añaden elementos a un gráfico, no se podrá modificar la estructura del gráfico.

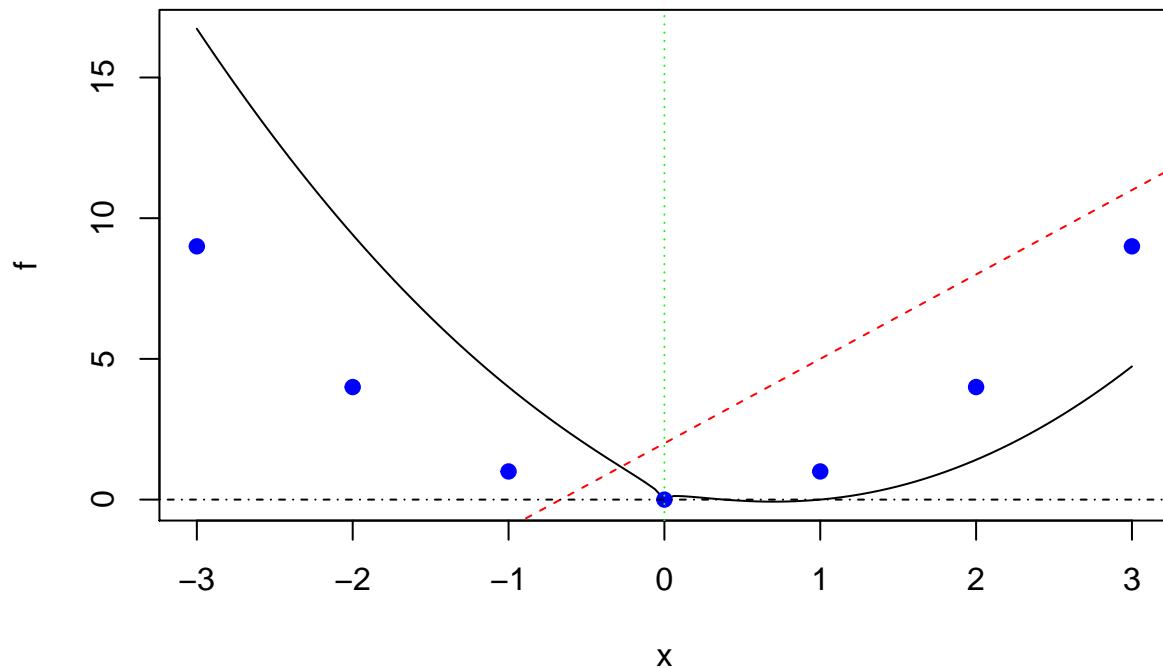
Ejemplos

1. Añadir un punto y una recta horizontal



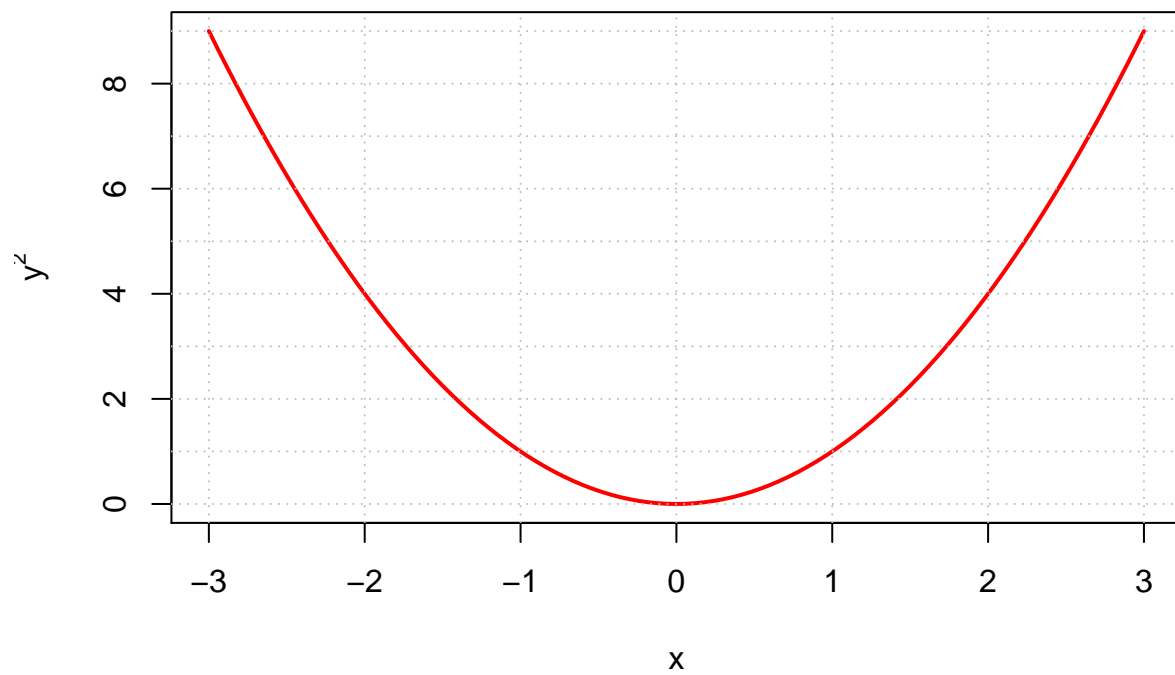
2. Añadir puntos y rectas

```
f <- function(x){  
  x^2 - 2*x + sqrt(abs(x))  
}  
plot(f, xlim = c(-3,3))  
#poner punto en (0,0)  
points(0,0, pch = 19, cex=0.7)  
#poner puntos como si fuera una parabola  
points(-3:3, (-3:3)^2, col = "blue", pch=19)  
#se dibuja recta de pendiente 3 que pasa por 2  
abline(2,3, lty = "dashed", col = "red")  
  
abline(v = 0, lty = "dotted", col = "green")  
  
abline(h = 0, lty = "dotdash", col = "black")
```



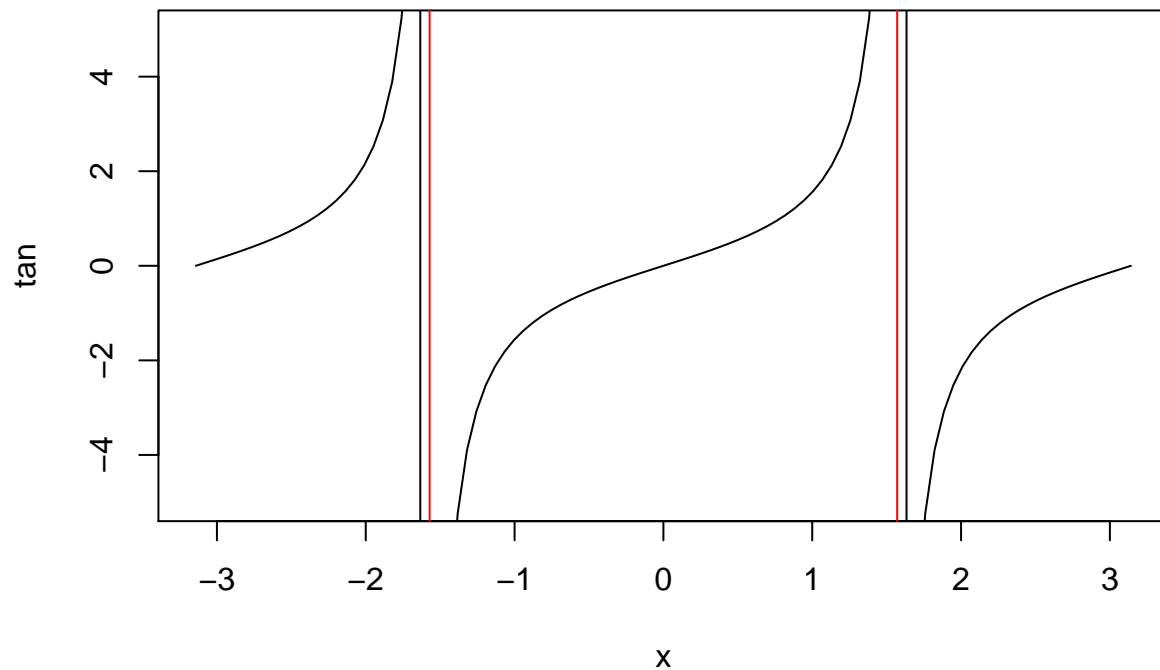
3. Genera lineas para ver donde se situa la linea

```
f <- function(x){x^2}  
plot(f, xlim = c(-3,3), col = "red",  
     lwd = 2,  
     ylab = expression(y^2),  
     xlab = "x")  
#Establecer un grid  
abline(h=0:9, v = -3:3, lty="dotted", col = "grey")
```



4. Función tangente. Colocar asíntotas verticales

```
plot(tan, xlim = c(-pi, pi),
      ylim = c(-5,5))
#Coloca la asíntota vertical de la función
abline(v = c(-pi/2, pi/2), col = "red")
```



2. Añadir texto y curvas a un gráfico

1. Añadir texto

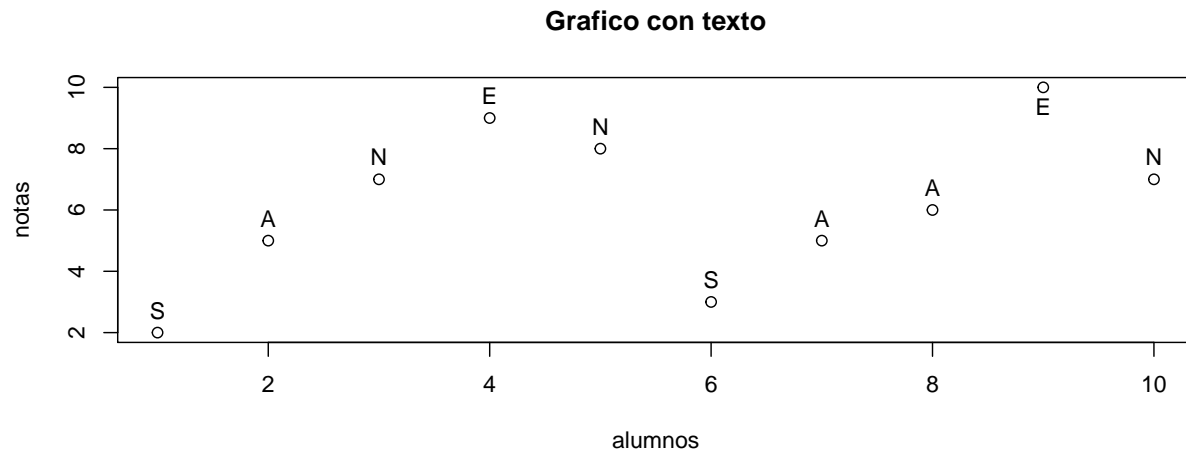
- `text(x,y,labels = "...")`: añade en el punto de coordenadas (x,y) el texto especificado como argumento de labels
 - `pos`: permite indicar la posición del texto alrededor de las coordenadas (x,y) . Admite los siguientes valores:
 - * 1: abajo
 - * 2: izquierda
 - * 3: arriba
 - * 4: derecha
 - * 5: sin especificar: el texto se sitúa centrado en el punto (x,y)

Ejemplo función `text()`

Añadir etiquetas

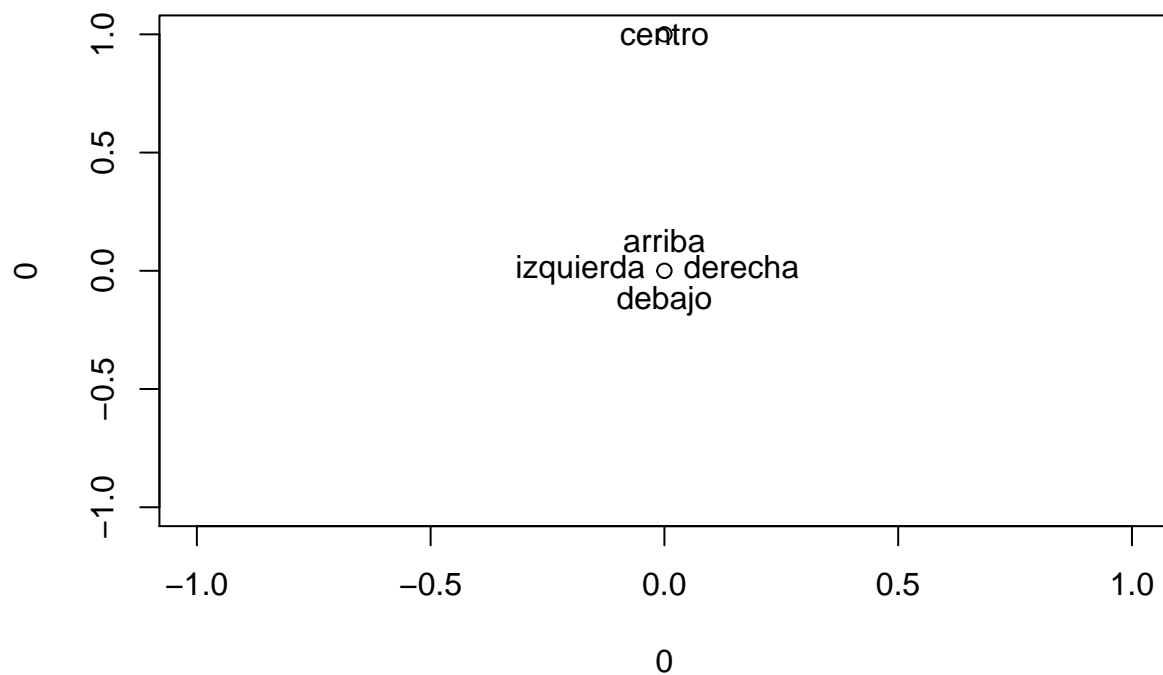
```
alumnos = c(1:10)
notas = c(2,5,7,9,8,3,5,6,10,7)
plot(alumnos,notas, main = "Grafico con texto")
#Se repite 3 ocho veces con rep()
text(alumnos,notas,
```

```
labels = c("S","A","N","E","N","S","A","A","E","N"),
pos = c(rep(3,times = 8),1,3))
```



Diferentes posiciones en 0,0

```
plot(0,0)
text(0,0, labels = "debajo", pos = 1)
text(0,0, labels = "izquierda", pos = 2)
text(0,0, labels = "arriba", pos = 3)
text(0,0, labels = "derecha", pos = 4)
points(0,1)
text(0,1, labels = "centro")
```



2. Añadir líneas y curvas

- `lines(x, y)`: añade a un gráfico existente una línea poligonal que une los puntos (x_i, y_i) sucesivos. x, y son vectores numéricos
- `curve(curva)`: permite añadir la gráfica de una curva a un gráfico existente
 - `add=TRUE`: si no, la curva no se añade
 - La curva se puede especificar mediante una expresión algebraica con variable x , o mediante su nombre si la hemos definido antes

Ejemplos

Uso de la función `lines()` y `curve()`

```
f <- function(x){x^2}

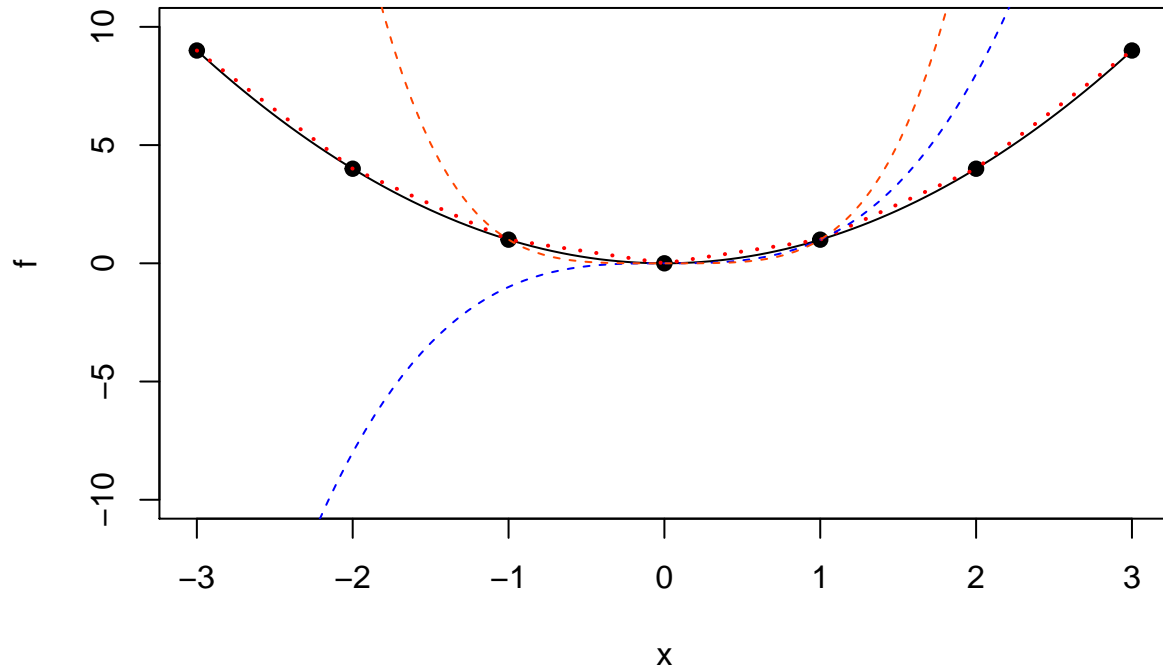
plot(f, xlim = c(-3,3),
     ylim = c(-10,10))

points(-3:3, f(-3:3), pch = 19)

lines(-3:3, f(-3:3), lwd = 2,
      lty = "dotted", col = "red")

curve(x^3, lty = "dashed", col = "blue", add = TRUE)
```

```
curve(x^4, lty = "dashed", col = "orangered", add=TRUE)
```



Ejemplo con lines() y curve()

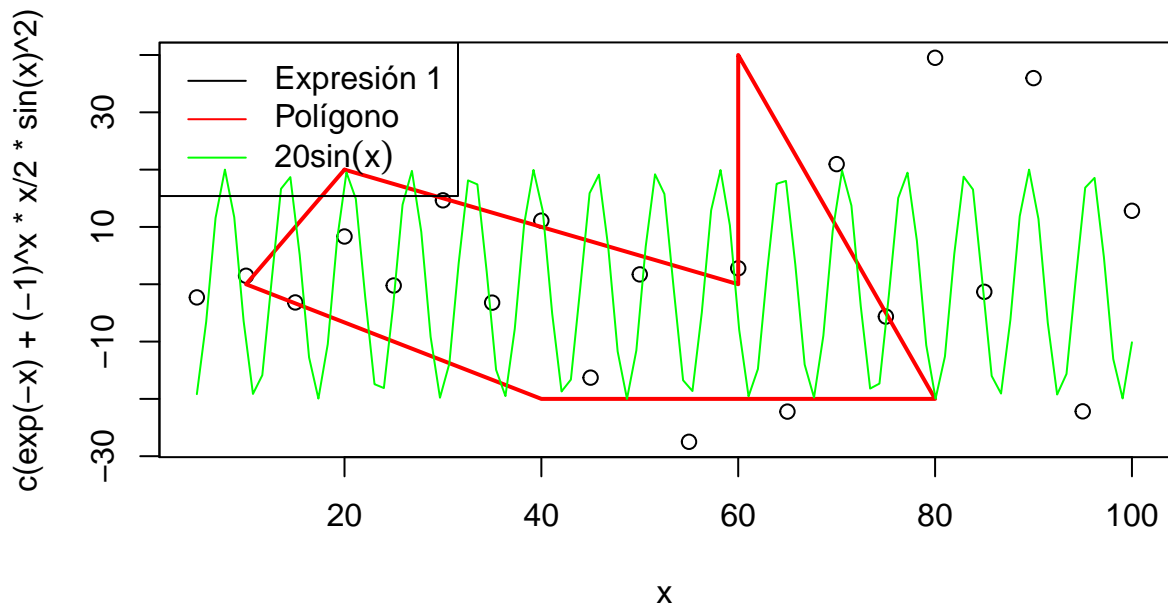
```
x = c(5*(1:20))

plot(x, c(exp(-x) + (-1)^x * x / 2 * sin(x)^2), col = "black")

lines(c(20, 10, 40, 80, 60, 60, 20),
      c(20, 0, -20, -20, 40, 0, 20),
      lwd = 2, col = "red")

curve(20 * sin(x), add = TRUE,
      col = "green")

legend("topleft",
      legend = c("Expresión 1", "Polígono", expression(20 * sin(x))),
      col = c("black", "red", "green"),
      lty = 1)
```



3. Añadir leyendas a los gráficos

- `legend(posición, legend = ...)`: para añadir una leyenda
 - La posición indica donde queremos situar la leyenda. Puede ser o bien las coordenadas de la esquina superior izquierda de nuestra leyenda, o bien una de las palabras siguientes:
 - * “bottom” / “bottomright” / “bottomleft”
 - * “top” / “topright” / “topleft”
 - * “center” / “right” / “left”
 - `legend`: contiene el vector de nombres entre comillas con los que queremos identificar a las curvas en la leyenda

Ejemplos

Ejemplo 1 de leyenda

```
x = seq(0,2*pi,0.1)

plot(x,sin(x),type="l",col="blue",
     lwd=3, xlab="", ylab="")

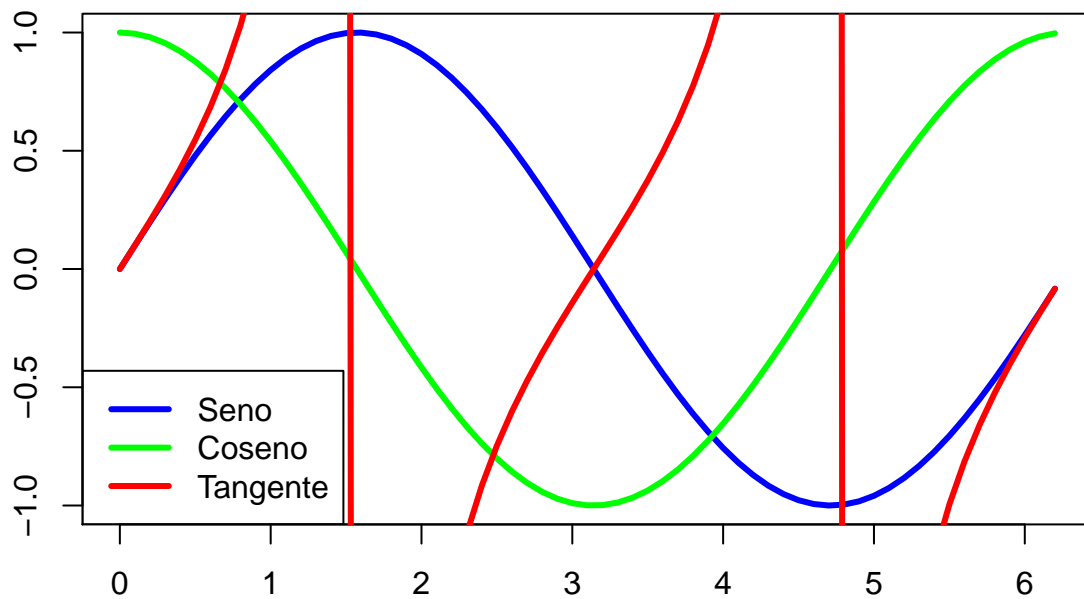
lines(x,cos(x),col="green",lwd=3)

lines(x, tan(x), col="red",lwd=3)

legend("bottomleft",
     col=c("blue","green","red"),
```



```
legend=c("Seno","Coseno", "Tangente"),
lwd=3, bty="l")
```



Ejemplo 2 de leyenda

```
f <- function(x){x^2}

plot(f, xlim = c(-3,3),
     ylim = c(-10,10))

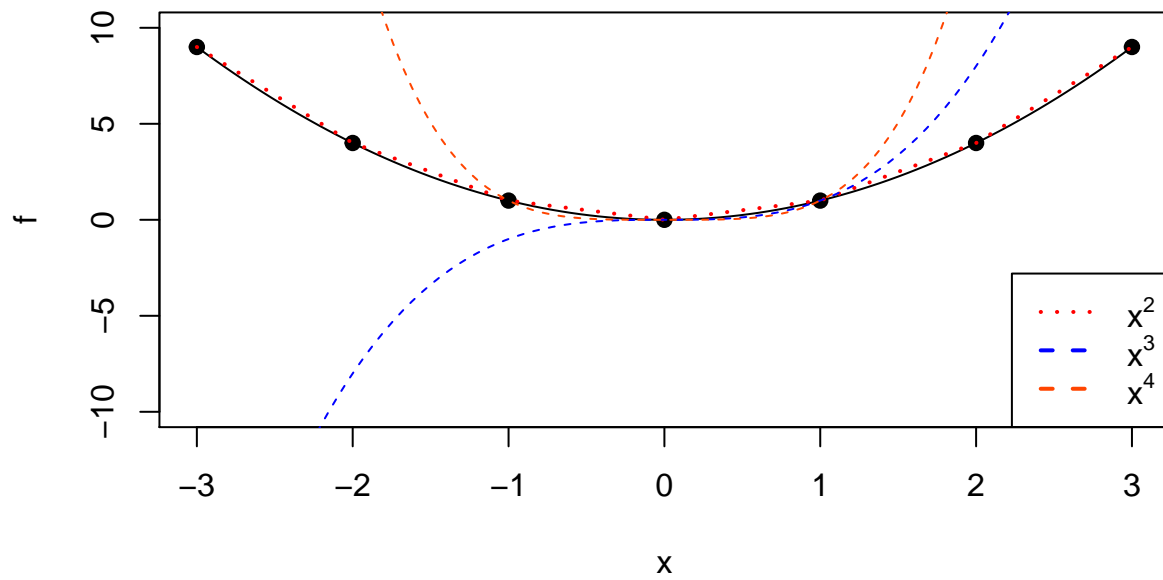
points(-3:3, f(-3:3), pch = 19)

lines(-3:3, f(-3:3), lwd = 2,
      lty = "dotted", col = "red")

curve(x^3, lty = "dashed", col = "blue", add = TRUE)

curve(x^4, lty = "dashed", col = "orangered", add=TRUE)

legend("bottomright",
      legend = c(expression(x^2), expression(x^3), expression(x^4)),
      lwd = 2,
      col = c("red", "blue", "orangered"),
      lty = c("dotted", "dashed", "dashed")
    )
```



4. Añadir segmentos, flechas, símbolos y polígonos

Añadir elementos al gráfico

- **segments:** para añadir segmentos a un gráfico existente. Trozo de línea
- **arrows:** para añadir flechas a un gráfico existente
- **symbols:** para añadir símbolos a un gráfico existente
- **polygon:** para añadir polígonos cerrados especificando sus vértices a un gráfico existente

Ejemplos

Grafico con varios elementos

