Funciones lineales

J. Abellán

05/11/2015

Función de variable aleatoria

El caso más sencillo pero más importante: $Y = Y(X; a, b) = a + bX$

- Sea la variable aleatoria $X \sim N(\mu_X, \sigma_X)$
- Sea $Y = Y(X; a, b) = a + bX$

¿Cómo será la función de distribución de Y, $f_Y(y)$?

De acuerdo con el teorema:

$$f_Y(y) = f_X(x(y))|\frac{dx}{dy}(y)|$$

que en este caso tan sencillo toma la forma:

$$f_Y(y) = f_X(\frac{y-a}{b})\frac{1}{b} = \frac{1}{\sqrt{2\pi\sigma_Y^2}}e^{-\frac{1}{2}\frac{(y-\mu_Y)^2}{\sigma_Y^2}}$$

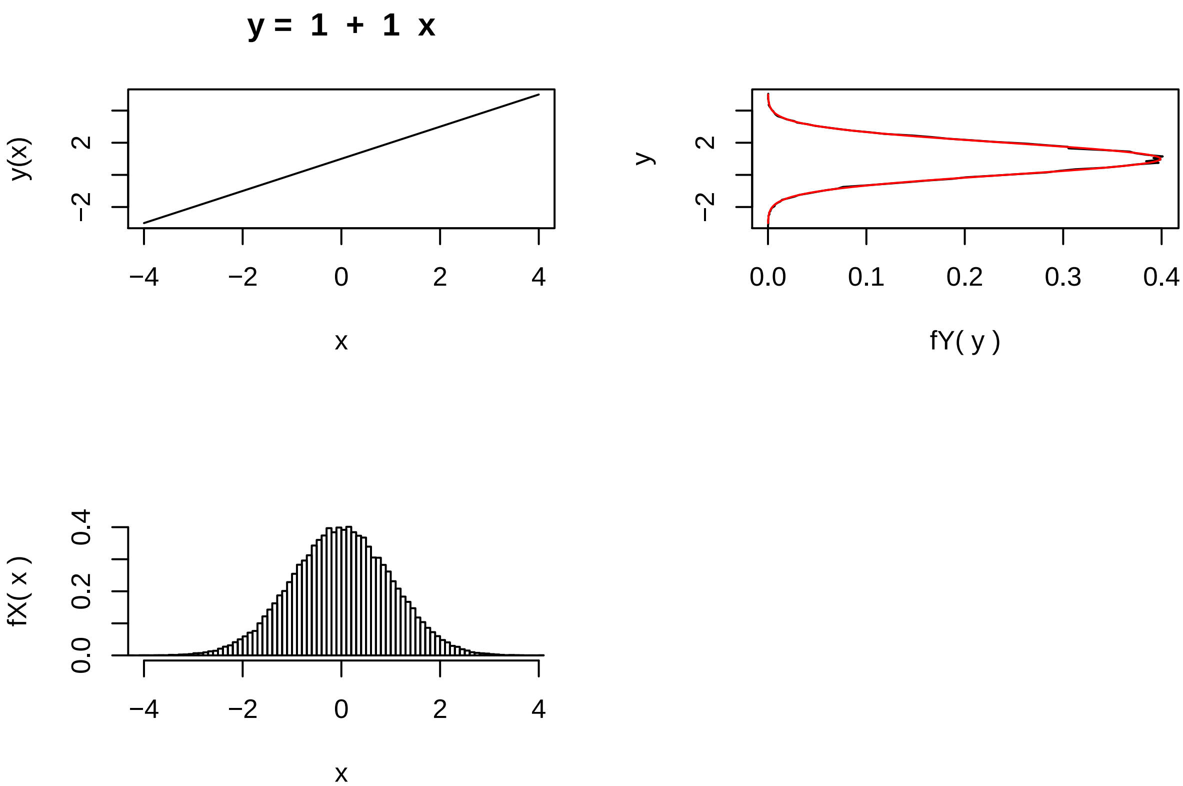
donde hemos definido $\sigma_Y^2 \equiv b^2\sigma_X^2$ y $\mu_Y \equiv a + b\mu_X$

Comprobaremos el teorema de la forma habitual: generando al azar un número grande de valores de la variable normal X y transformándolos de acuerdo con la función $Y = Y(X)$. A continuación haremos el histograma de los valores de Y y superpondremos al histograma la función de distribución teórica.

```r
#Parámetros de la variable normal X
muX <- 0 ; deX <- 1

#Valores que toma la variable X
xmin <- muX - 4 * deX

xmax <- muX + 4 * deX

x <- seq( xmin, xmax, len = 1000 )

#Parámetros de la función Y
a <- 1 ; b <- 1

#Por tanto, los valores de y serán
y <- function(x) a + b * x

#Límites de dibujo
ymin <- y( xmin )

ymax <- y( xmax )

#Para cuatro gráficas
```

```r
matriz <- matrix( 1 : 4, 2, 2 )

layout( matriz )

#Primera gráfica fila=1, columna=1
plot( x, y(x),

      type = "l",

      main = paste( " y = ", a, " + ", b, " x " )

    )

#Generamos los valores X y los transformamos
N <- 100000

X <- rnorm( N, muX, deX )

# La función de X
Y <- y(X)

#Segunda gráfica: fila=2, columna=1
hist( X, 100,

      xlab = " x ",

      ylab = " fX( x ) ",

      xlim = c( xmin, xmax ),

      prob = T,

      main = ""

    )

#Tercera gráfica: fila=1, columna=2
#Calculamos el histograma pero no lo dibujamos
hY <- hist( Y, 100, plot = FALSE )

#Extraemos la información del objeto hY
fYe <- hY$density

ye <- hY$mids

#Dibujamos pero girando la gráfica
plot( fYe, ye,

      xlab = " fY( y ) ",

      ylab = " y ",

      ylim = c( ymin, ymax ),
```

```r
      type = "l"
    )

#Y superponemos la curva teórica
muY <- y(muX)

deY <- b * deX

fY <- dnorm( y(x), muY, deY )

lines( fY, y(x), col = 2 )
```



Así se puede ver cómo la función $Y = Y(X)$, en este caso, una recta, *refleja* sin distorsión el histograma de X.

¿Qué ocurrirá si la dependencia $Y = Y(X)$ no es lineal?

La respuesta es sencilla si σ_X es pequeña. En este caso los valores que toma X no serán muy distintos de μ_X. Por tanto, se puede hacer un desarrollo Taylor de la función $Y = Y(X)$ entorno al punto $x_o = \mu_X$, que en primera aproximación da:

$$y = y(x) \simeq y(\mu_X) + \frac{dy}{dx}(\mu_X)(x - \mu_X) = a + bx$$

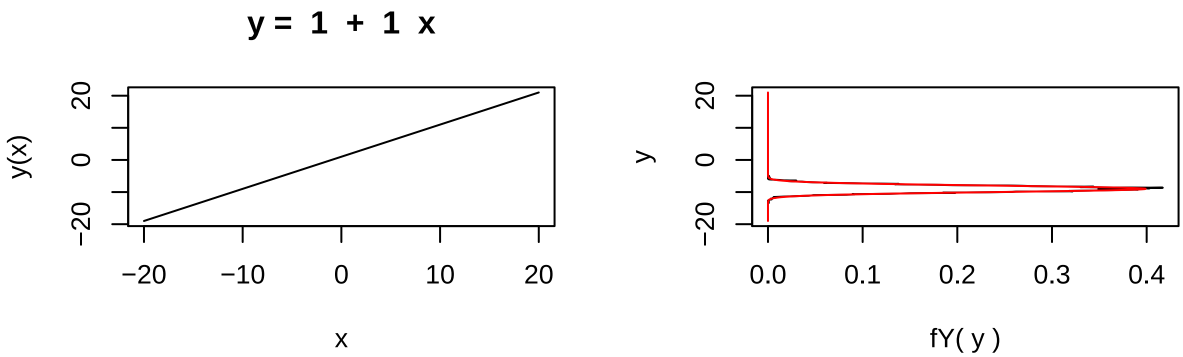
con las definiciones $b \equiv \frac{dy}{dx}(\mu_X)$ y $a \equiv y(\mu_X) - b\mu_X$.

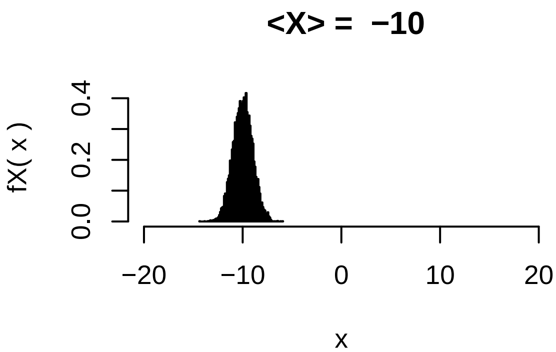
Dicho de otro modo, todas las funciones son rectas si no nos apartamos mucho del punto de interés, $x = x_o$ y el paso de la función de distribución de X a Y es bien sencilla.

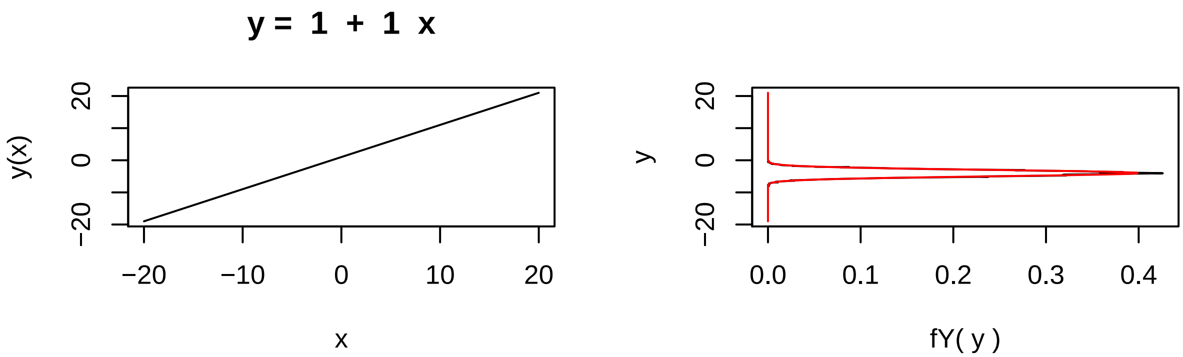
Repetimos variando μ_X

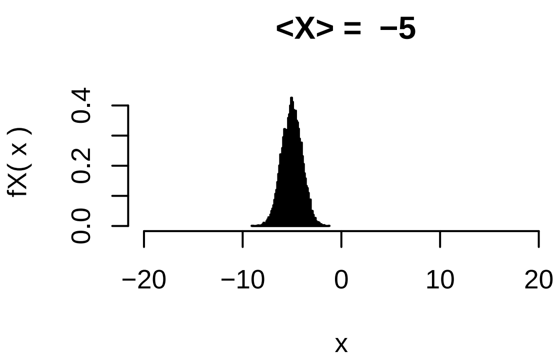
```r
#Parámetros de la variable normal X
deX <- 1

MUX <- c( - 10, - 5, 0, 2, 5, 10 )

n <- length( MUX )

#Parámetros de la función Y
a <- 1 ; b <- 1

#Valores que toma la variable X
xmin <- - 20 ; xmax = 20

x <- seq( xmin, xmax, len = 1000 )

#Por tanto, los valores de y serán
y <- function( x) a + b * x

#Límites de dibujo
ymin <- y( xmin ) ; ymax <- y( xmax )

for (i in 1 : n ) {

    muX <- MUX[ i ]

    #Para cuatro gráficas
    matriz <- matrix( 1 : 4, 2, 2 )

    layout( matriz )

    #Primera gráfica fila=1, columna=1
    plot( x, y(x),

          type = "l",

          main = paste(" y = ", a, " + " , b, " x " )

        )

    #Generamos los valores X y los transformamos
    N <- 10000

    X <- rnorm( N, muX, deX )

    Y <- y( X )

    #Segunda gráfica: fila=2, columna=1
    hist( X, 100,

          xlab = " x ",

          ylab = " fX( x ) ",
```

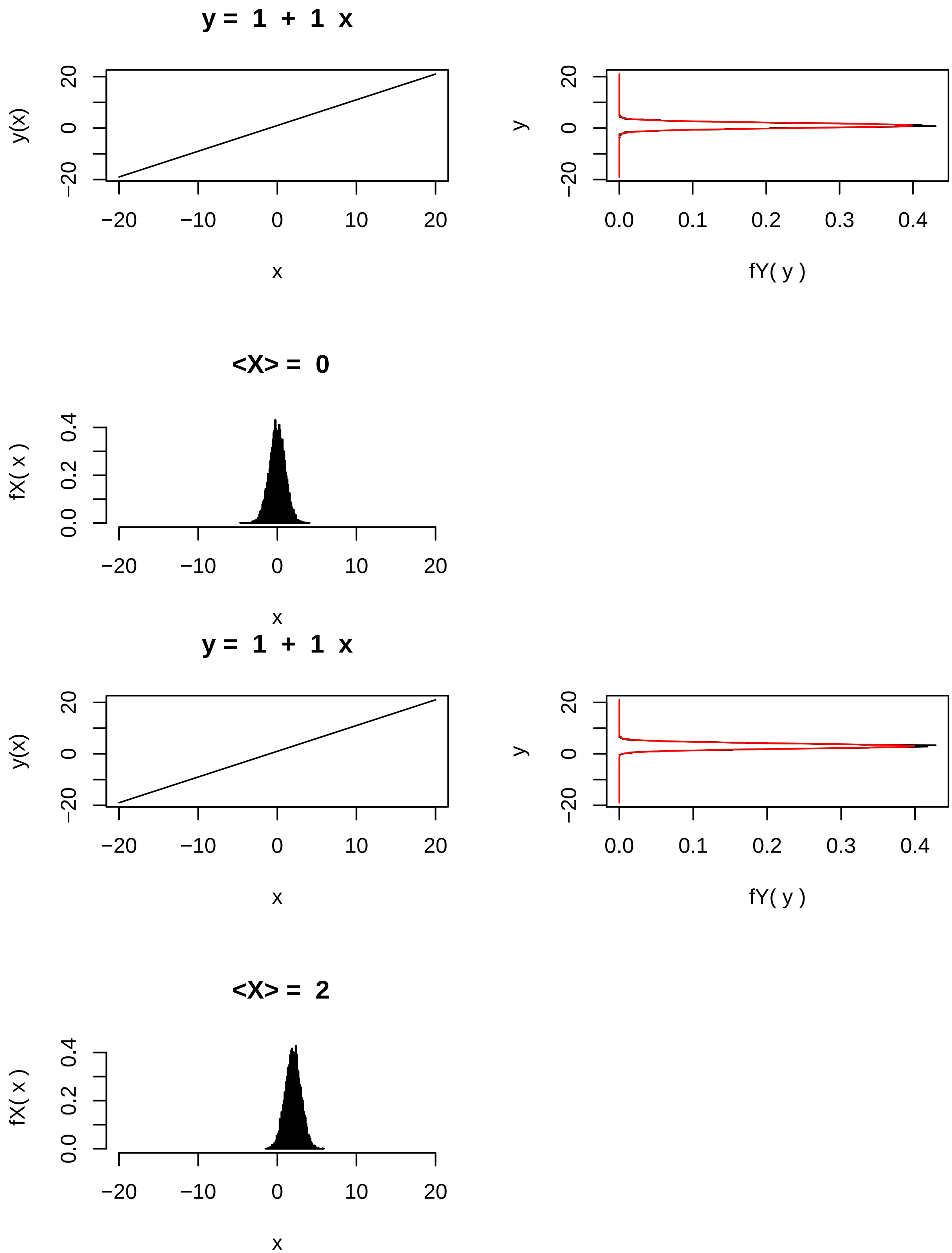
```
          xlim = c( xmin, xmax ),

          prob = T,

          main = paste(" <X> = ", muX )

        )

    #Tercera gráfica: fila=1, columna=2
    #Calculamos el histograma pero no lo dibujamos
    hY <- hist( Y, 100, plot = FALSE )

    #Extraemos la información del objeto hY
    fYe <- hY$density

    ye <- hY$mids

    #Dibujamos pero girando la gráfica
    plot( fYe, ye,

          xlab = " fY( y ) ",

          ylab = " y ",

          ylim = c( ymin, ymax),

          type = "l"

        )

    #Y superponemos la curva teórica
    muY <- y(muX)

    deY <- b * deX

    fY <- dnorm( y( x ), muY, deY )

    lines( fY, y( x ), col = 2 )

}
```

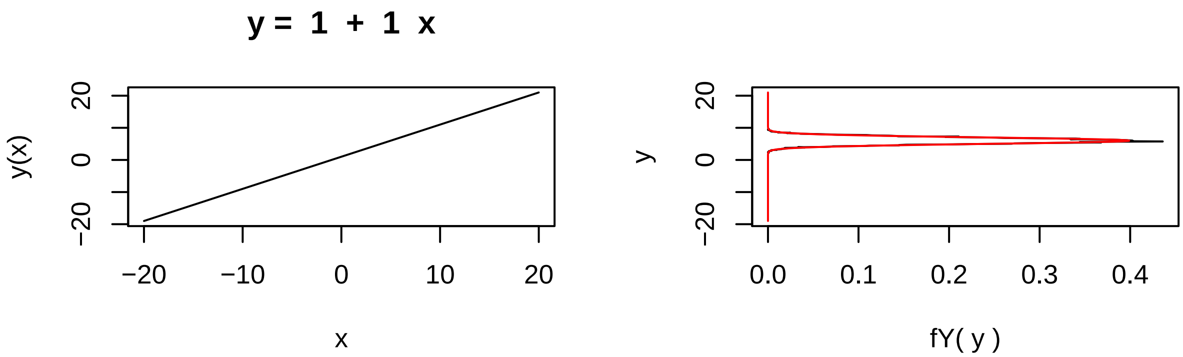

y = 1 + 1 x
y(x)
−20 0 20
−20 −10 0 10 20
x
y
−20 0 20
0.0 0.1 0.2 0.3 0.4
fY(y)

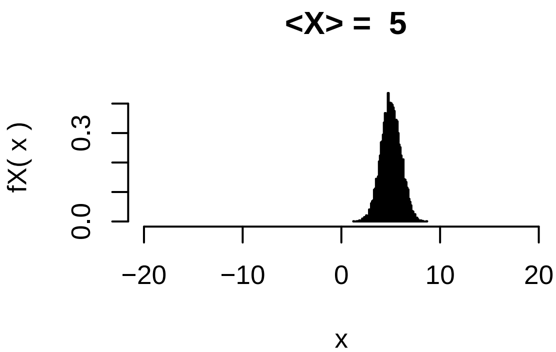

<X> = −10
fX(x)
0.0 0.2 0.4
−20 −10 0 10 20
x

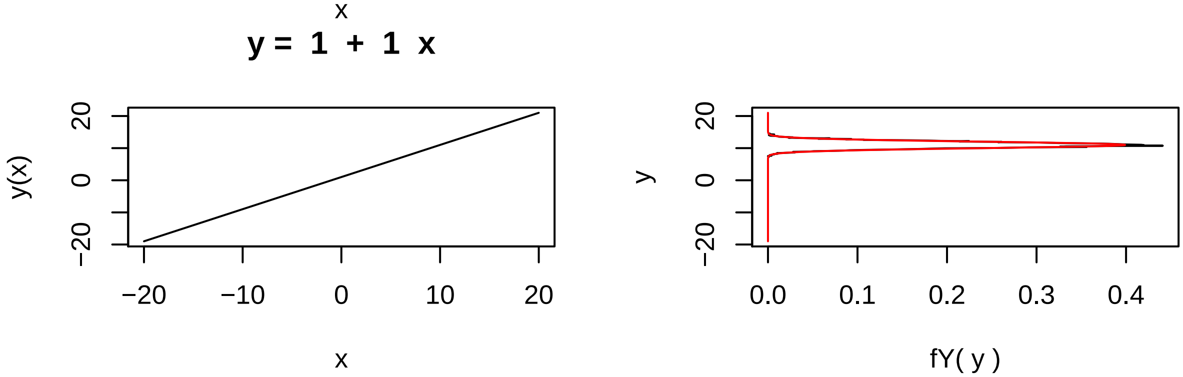

y = 1 + 1 x
y(x)
−20 0 20
−20 −10 0 10 20
x
y
−20 0 20
0.0 0.1 0.2 0.3 0.4
fY(y)

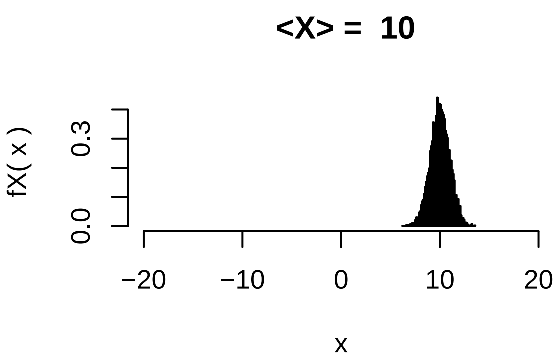

<X> = −5
fX(x)
0.0 0.2 0.4
−20 −10 0 10 20
x


y = 1 + 1 x
y(x)
x
y
fY(y)
<X> = 0
fX(x)
x
y = 1 + 1 x
y(x)
x
y
fY(y)
<X> = 2
fX(x)
x


y = 1 + 1 x
y(x)
−20
0
20
−20
−10
0
10
20
x
y
0.0
0.1
0.2
0.3
0.4
fY(y)


<X> = 5
fX(x)
0.0
0.3
−20
−10
0
10
20
x


y = 1 + 1 x
y(x)
−20
0
20
−20
−10
0
10
20
x
y
0.0
0.1
0.2
0.3
0.4
fY(y)


<X> = 10
fX(x)
0.0
0.3
−20
−10
0
10
20
x