Ejercicio de simulación

Cedric Prieels

21/11/2016

Este ejercicio consiste en desarrollar una simulación que tiene por objetivo principal la estimación de diferentes parámetros estadísticos y el cálculo del intervalo de confianza sobre los parámetros obtenidos por de un ajuste lineal de datos experimentales (pendiente, termino independiente), usando un método general de bootstrap.

Primero, vamos a crear a mano una medidas experimentales suponiendo que siguen una ley de probabilidad definida.

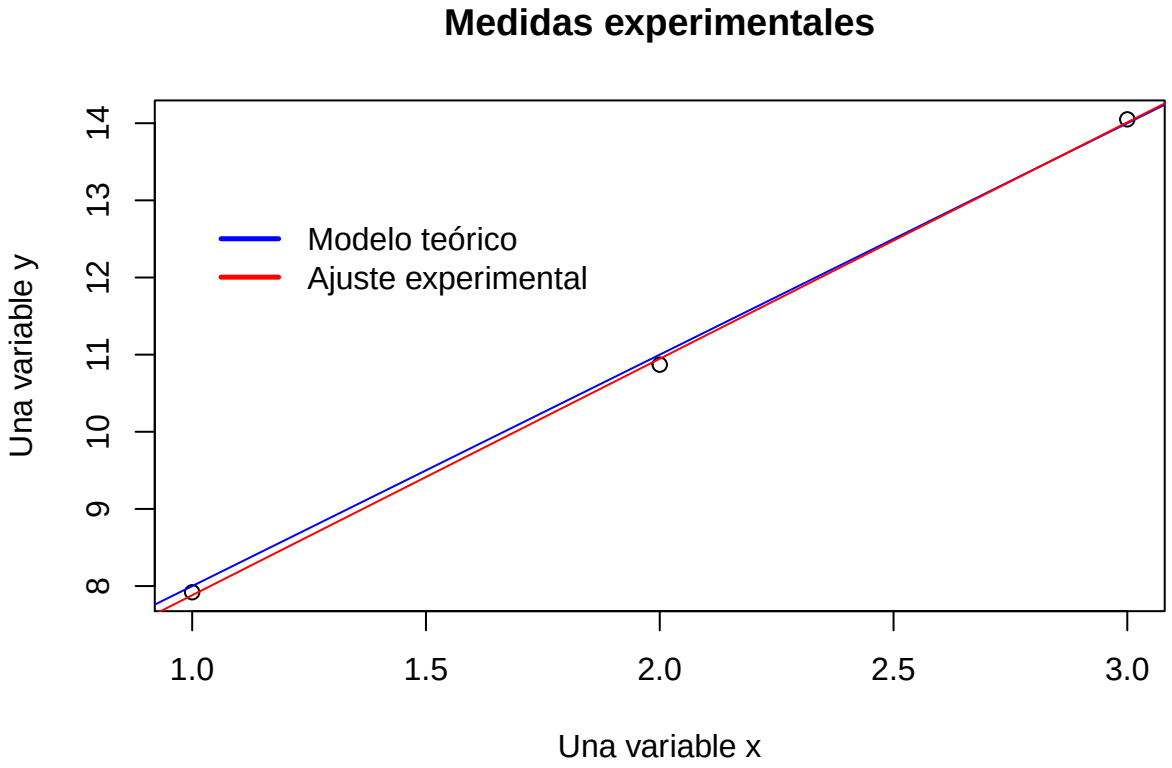
```
rm(list=ls())
par(mfrow=c(1,1))

#Medidas experimentales que siguen el modelo lineal 3x+5
x <- c(1, 2, 3)
y <- c(7.92, 10.87, 14.05)
plot(x, y, main="Medidas experimentales", xlab = "Una variable x", ylab = "Una variable y")

#Calculamos un ajuste lineal de nestos datos experimentales
primerAjuste <- lm(y ~ x)

terminoIndependiente <- primerAjuste$coefficients[1]
terminoIndependiente
```

```
## (Intercept)
##    4.816667
```

```
pendiente <- primerAjuste$coefficients[2]
pendiente
```

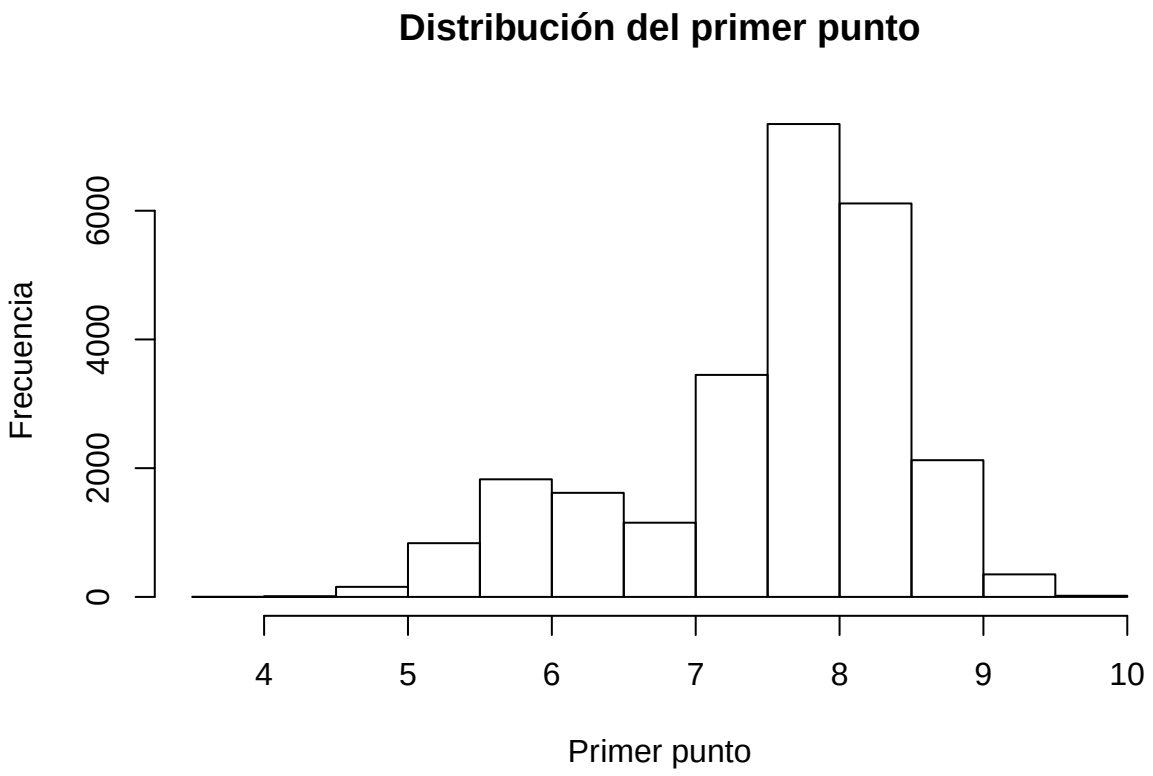
```
##     x
## 3.065
```

```
abline(a=5, b=3, col="blue")
abline(a=terminoIndependiente, b=pendiente, col="red")
legend(1, 13 , legend = c("Modelo teórico", "Ajuste experimental"),
       lty=c(1, 1), lwd=c(2.5, 2.5), col=c("blue","red"), bty = "n")
```

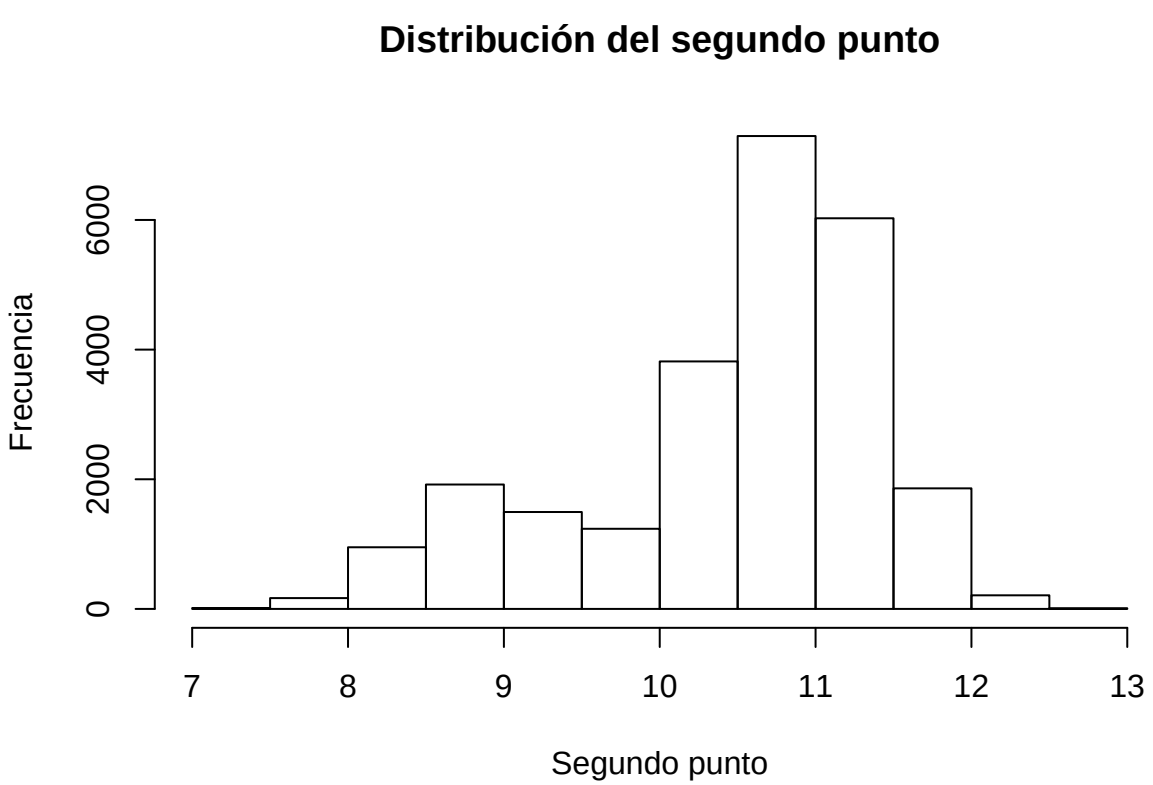


Vemos que el ajuste sobre estos datos experimentales nos devuelve un valor de pendiente de 3.065 y un valor de termino independiente de 4.817. Lo que nos interesa ahora es determinar el valor del error asociado a estos dos parámetros, de manera más precisa que con un simple cálculo propagación de errores. Para esto vamos a generar pseudoexperimentos, para después estudiar la distribución del valor de la pendiente y del termino independiente.

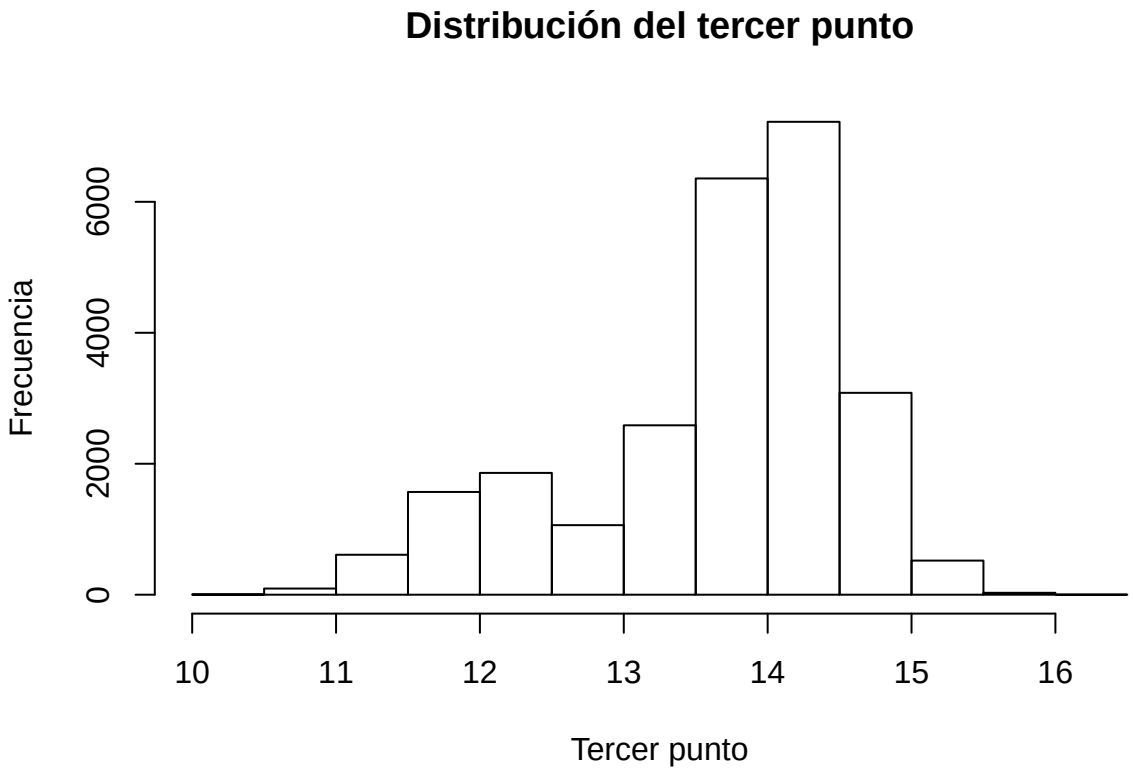
```
N <- 20000

#Simulamos pseudoexperimentos según una suma de dos gausianas de varianza 0.5
y1 <- c()
y1 <- append(rnorm(N, y[1], 0.5), y1)
y1 <- append(rnorm(N/4, y[1] - 2, 0.5), y1)

y2 <- c()
y2 <- append(rnorm(N, y[2], 0.5), y2)
y2 <- append(rnorm(N/4, y[2] - 2, 0.5), y2)

y3 <- c()
y3 <- append(rnorm(N, y[3], 0.5), y3)
y3 <- append(rnorm(N/4, y[3] - 2, 0.5), y3)

#Mezclamos de manera aleatoria los vectores creados para tener puntos que pueden partenecen
#a las dos gausianes en cualquiera posición del vector
y1 <- sample(y1)
y2 <- sample(y2)
y3 <- sample(y3)

hist(y1, main="Distribución del primer punto", xlab="Primer punto", ylab="Frecuencia")
```



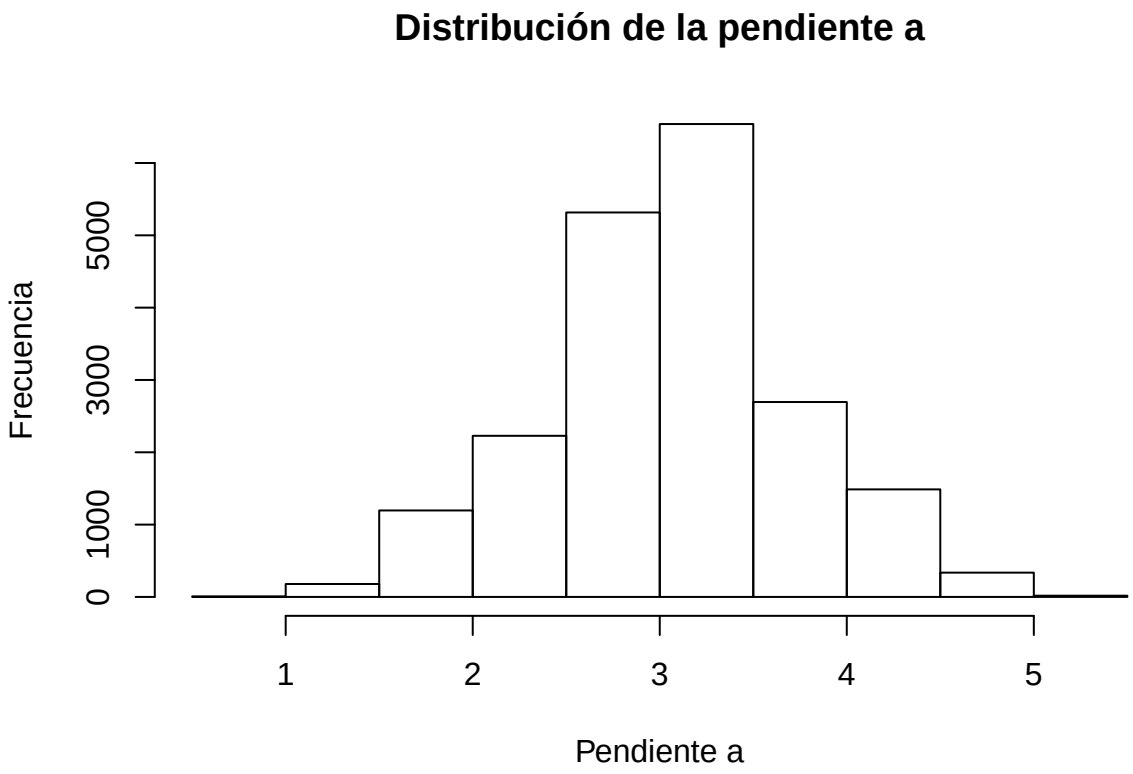

```r
hist(y2, main="Distribución del segundo punto", xlab="Segundo punto", ylab="Frecuencia")
```



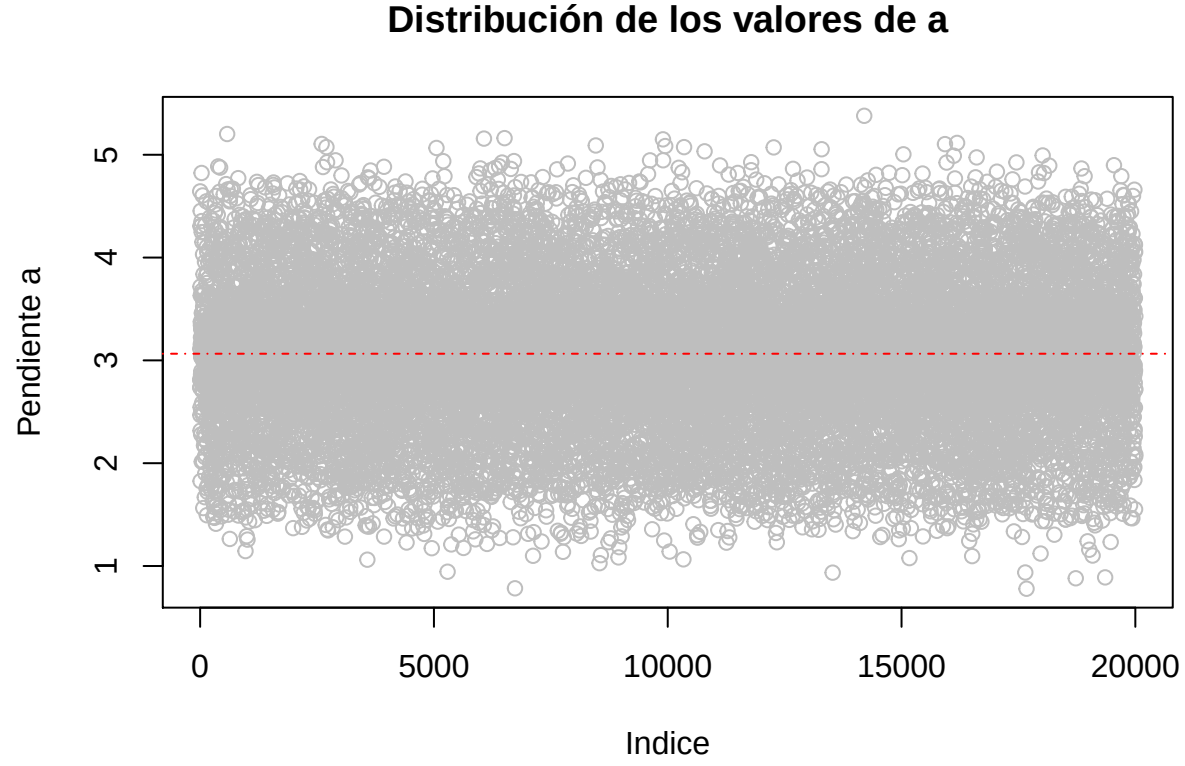
```r
hist(y3, main="Distribución del tercer punto", xlab="Tercer punto", ylab="Frecuencia")
```



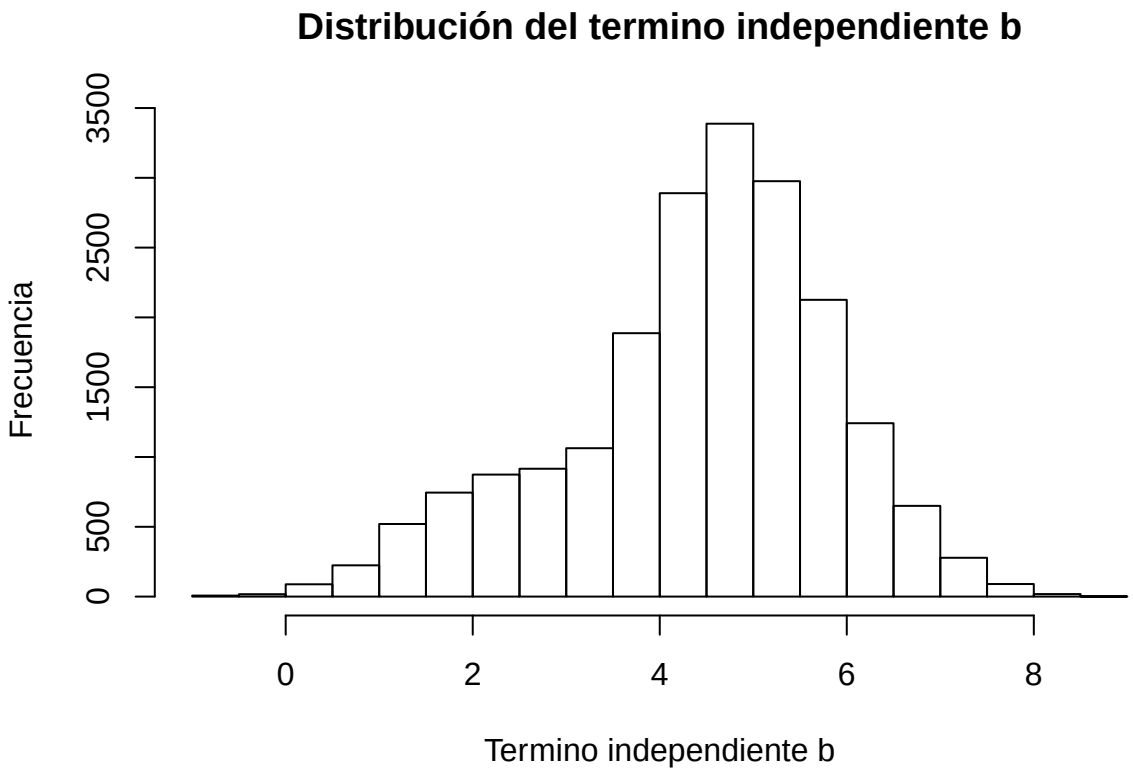

```r
#Los parámetros a y b son definidos de la manera siguiente : y=a*x+b
a <- matrix(rep(NA,N))
b <- matrix(rep(NA,N))

for(i in 1:N) {
  experimento <- c(y1[i], y2[i], y3[i])
  ajuste <- lm(experimento ~ x)
  a[i] <- ajuste$coefficients[2]
  b[i] <- ajuste$coefficients[1]
}

hist(a, main="Distribución de la pendiente a", xlab="Pendiente a", ylab="Frecuencia")
```



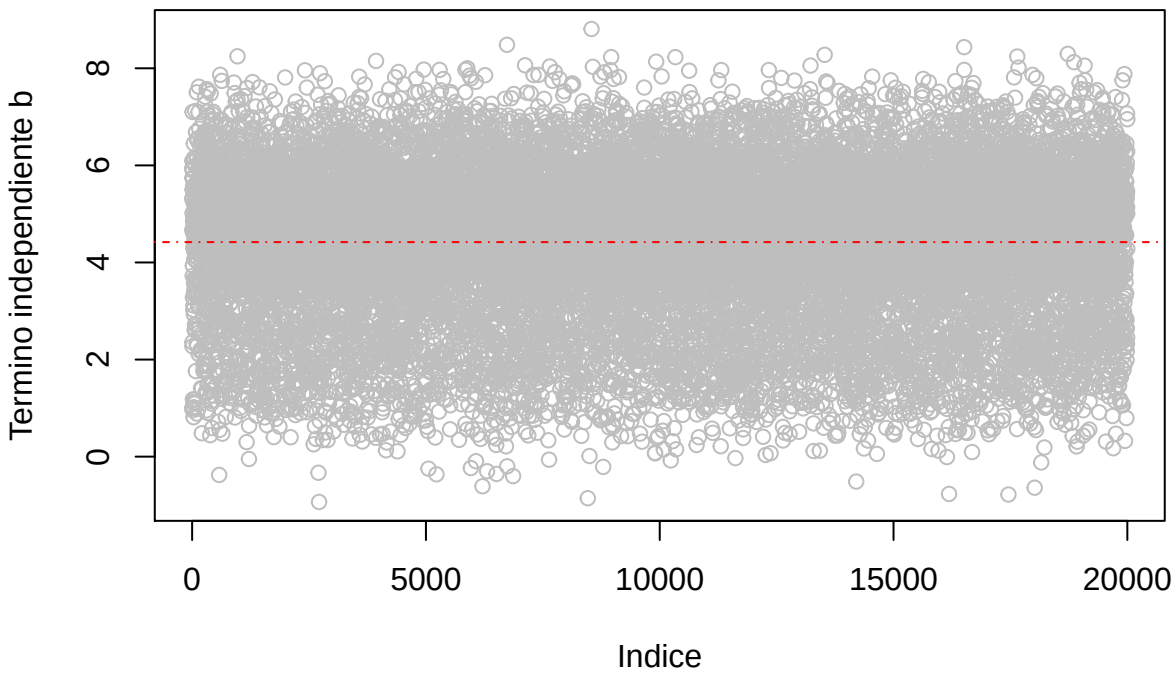

```
plot(a, ty = "p" , col="grey", xlab="Indice",
     ylab="Pendiente a", main="Distribución de los valores de a")
abline(h=mean(a), col="red", lty=10)
```



```r
hist(b, main="Distribución del termino independiente b",
     xlab="Termino independiente b", ylab="Frecuencia")
```




```r
plot(b, ty = "p" , col="grey", xlab="Indice",
     ylab="Termino independiente b", main="Distribución de los valores de b")
abline(h=mean(b), col="red", lty=10)
```



Vemos que estos dos histogramas tienen buena pinta : tienen su maximo donde lo esperamos (en más o menos 3 para la pendiente y 5 para el termino independiente, como nuestro modelo inicial). Ahora estudiamos estas distribuciones para calcular de forma más detallada el intervalo de confianza del ajuste lineal (su pendiente, su ordenada en el origen, y la correlación de ambos parámetros). Primero, calculamos los valores más altos, más bajos y la media de cada distribución.

```
maxa <- max(a)
mina <- min(a)
meana <- mean(a)
medianaa <- median(a)
meana
```

```
## [1] 3.064787
```

```
maxb <- max(b)
minb <- min(b)
meanb <- mean(b)
medianab <- median(b)
meanb
```

```
## [1] 4.419196
```

Vemos que en mi caso (aunque cambiará un poco, dependiendo de las simulaciones hechas por R), obtengo como valor de pendiente medio 3.05406 y como valor de termino independiente medio 4.45111. Ahora, podemos calcular facilmente el valor de la desviación estándar de estas dos distribuciones, para estimar el error sobre estos parámetros. Por supuesto, la desviación estándar depende del la ley que hemos elegido para calcular nuestros pseudoexperimentos.

```r
sda <- sd(a)
sda
```

```
## [1] 0.6677975
```

```r
sdb <- sd(b)
sdb
```

```
## [1] 1.441944
```

Lo correlación (lineal) entre estos dos parámetros se puede calcular también, usando por ejemplo el método de Pearson para calcular primero la probabilidad de tener correlación o no. Como en muchos casos en estadística, solo podremos calcular el valor de la probabilidad nula H0 que consiste en decir que "no tenemos correlación". Si obtenemos un valor P(H0) muy pequeño, podremos concluir que no existe correlación entre las variables que estamos estudiando. Vamos también a calcular el coeficiente de correlación lineal r para saber cuanto de fuerte es la eventual correlación. Este coeficiente de correlación lineal puede tomar valores entre -1 (anticorrelación lineal perfecta) y 1 (correlación lineal perfecta).

```r
pearsonResults <- cor.test(a, b, method="pearson")
pearsonPValue <- pearsonResults$p.value
pearsonPValue
```

```
## [1] 0
```

```r
pearsonR <- pearsonResults$estimate
pearsonR
```

```
##        cor
## -0.9261052
```

En nuestro caso, vemos que el valor de probabilidad de la hipotésis nula vale 0, lo que significa que no tenemos ninguna correlación. Por lo tanto, no tiene mucho sentido mirar al valor del coeficiente de correlación lineal r.

En conclusión, se han generado en este ejercicio unas medidas que siguen una ley suma de dos gausianas de varianza 0.5. El objetivo de esta simulación era la determinación de los parámetros de pendiente y de termino independiente de una recta que modeliza nuestros puntos experimentales, para determinar el error sobre estos parámetros (sin usar la tipica formula de propagación de errores) y la eventual correlación que existe entre ellos (en este caso, vimos que no hay ninguna correlación entre los dos parámetros a y b).