

Bootrap: Ejercicio 7

Iker Terán

23/10/2021

Vamos a dividir el ejercicio en dos partes: el cálculo del intervalo de confianza cuando la muestra es normal y cuando la muestra es exponencial.

Muestra N(0,1)

Definimos el coeficiente de asimetría de una muestra normal, el cual es 0.

```
library(e1071)

## Warning: package 'e1071' was built under R version 4.0.5

set.seed(100)
R <- 1000
n <- 100
m <- 100
alfa <- 0.05
coef_asim_normal<-0
```

Procedemos a calcular los intervalos con los tres métodos.

```
1. Método Híbrido

library(e1071)
set.seed(100)
R <- 1000
n <- 100
m <- 100
alfa <- 0.05
coef_asim_normal<-0
acierto <- NULL
intervalo <- NULL
for (i in 1:m){
  muestra_original <- rnorm(100,mean=0,sd=sqrt(1))
  coef_asim_original<-skewness(muestra_original)

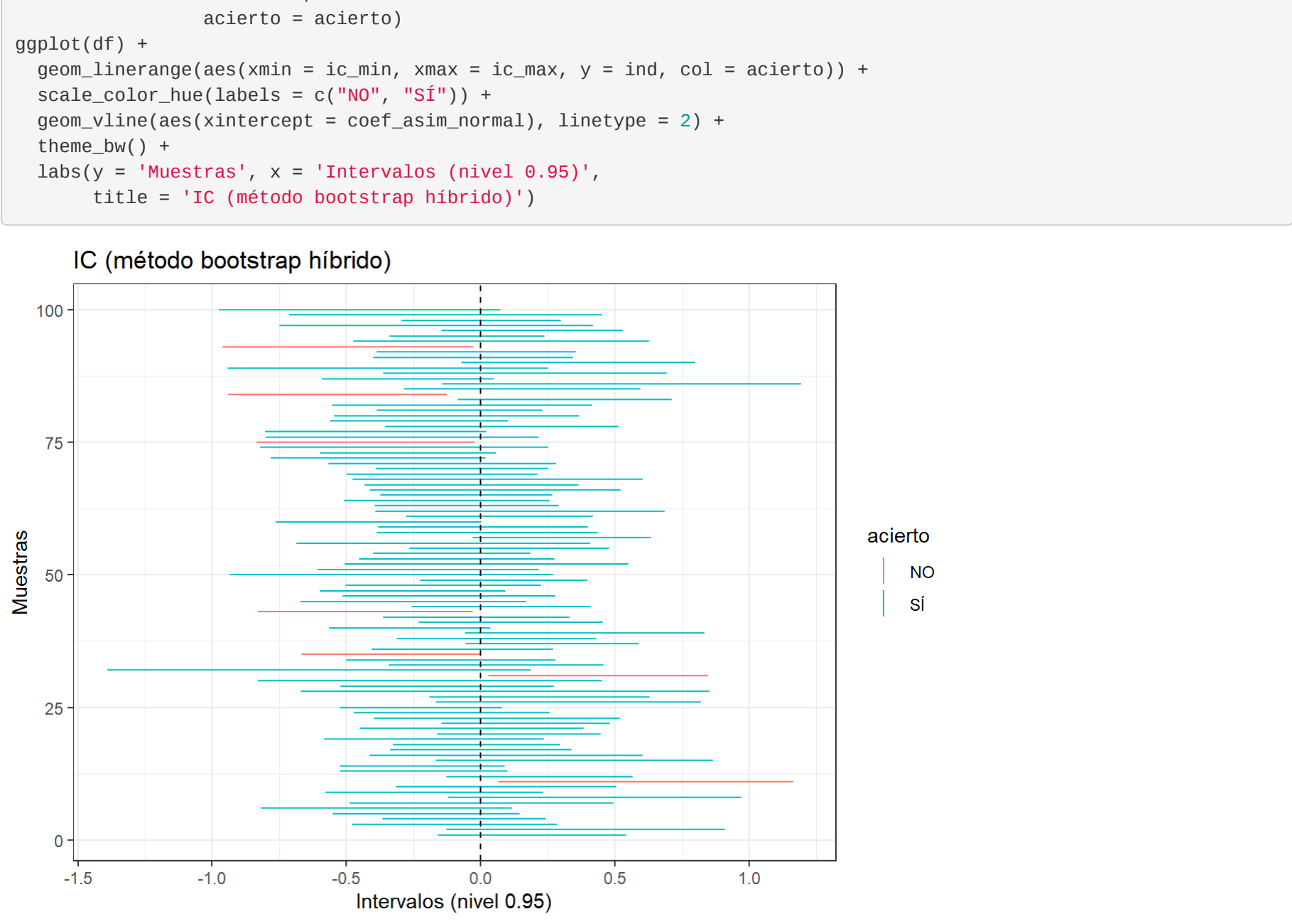
  muestras_bootstrap <- sample(muestra_original, n*R, rep = TRUE)
  muestras_bootstrap <- matrix(muestras_bootstrap, nrow = n)
  coef_asim_bootstrap <- apply(muestras_bootstrap, 2, skewness)

  T_bootstrap <- sqrt(n) * (coef_asim_bootstrap - coef_asim_original)
  ic_min <- coef_asim_original - quantile(T_bootstrap, 1-alfa/2)/sqrt(n)
  ic_max <- coef_asim_original + quantile(T_bootstrap, alfa/2)/sqrt(n)
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
  acierto <- c(acierto, ic_min < coef_asim_normal & ic_max > coef_asim_normal)
}

paste("% de acierto =",mean(acierto))

## [1] "% de acierto = 0.93"
```

Y por último, el gráfico.



```
2. Método Normal

acierto <- NULL
intervalo <- NULL
for (i in 1:m){
  muestra_original <- rnorm(100,mean=0,sd=sqrt(1))
  coef_asim_original<-skewness(muestra_original)

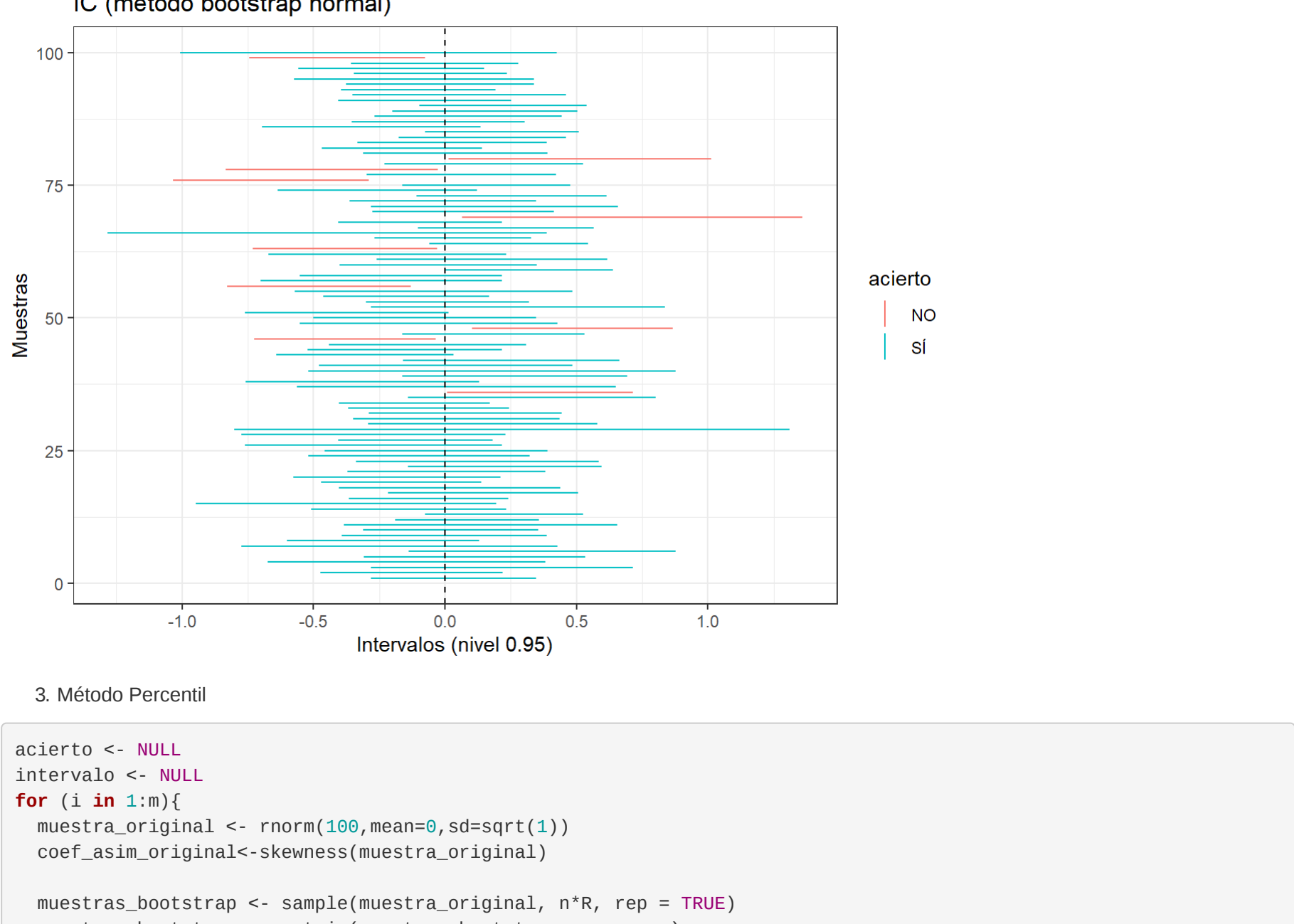
  muestras_bootstrap <- sample(muestra_original, n*R, rep = TRUE)
  muestras_bootstrap <- matrix(muestras_bootstrap, nrow = n)
  coef_asim_bootstrap <- apply(muestras_bootstrap, 2, skewness)
  et_bootstrap<-sd(coef_asim_bootstrap)
  ic_min <- coef_asim_original + qnorm(alfa/2)*et_bootstrap
  ic_max <- coef_asim_original - qnorm(alfa/2)*et_bootstrap
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
  acierto <- c(acierto, ic_min < coef_asim_normal & ic_max > coef_asim_normal)
}

paste("% de acierto =",mean(acierto))

## [1] "% de acierto = 0.9"
```

```
df <- data.frame(ic_min <- intervalo[,1],
                 ic_max <- intervalo[, 2],
                 ind = 1:m,
                 acierto = acierto)

ggplot(df) +
  geom_linerange(aes(xmin = ic_min, xmax = ic_max, y = ind, col = acierto)) +
  scale_color_hue(labels = c("NO", "SI")) +
  geom_vline(aes(xintercept = coef_asim_normal), linetype = 2) +
  theme_bw() +
  labs(y = 'Muestras', x = 'Intervalos (nivel 0.95)',
       title = 'IC (método bootstrap normal)')
```



```
3. Método Percentil

acierto <- NULL
intervalo <- NULL
for (i in 1:m){
  muestra_original <- rnorm(100,mean=0,sd=sqrt(1))
  coef_asim_original<-skewness(muestra_original)

  muestras_bootstrap <- sample(muestra_original, n*R, rep = TRUE)
  muestras_bootstrap <- matrix(muestras_bootstrap, nrow = n)
  coef_asim_bootstrap <- apply(muestras_bootstrap, 2, skewness)

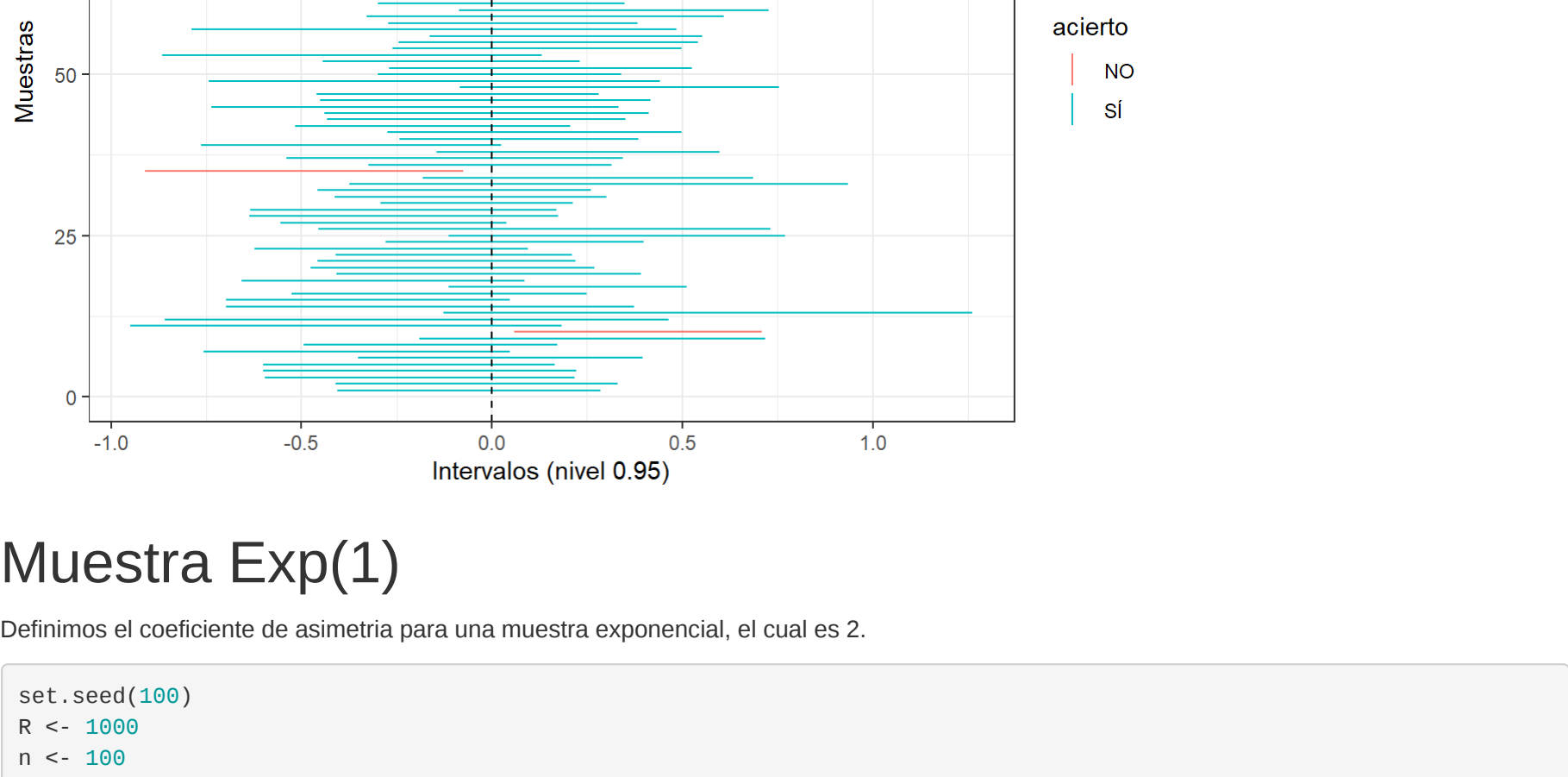
  ic_min <- quantile(coef_asim_bootstrap, alfa/2)
  ic_max <- quantile(coef_asim_bootstrap, 1-alfa/2)
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
  acierto <- c(acierto, ic_min < coef_asim_normal & ic_max > coef_asim_normal)
}

paste("% de acierto =",mean(acierto))

## [1] "% de acierto = 0.96"
```

```
df <- data.frame(ic_min <- intervalo[,1],
                 ic_max <- intervalo[, 2],
                 ind = 1:m,
                 acierto = acierto)

ggplot(df) +
  geom_linerange(aes(xmin = ic_min, xmax = ic_max, y = ind, col = acierto)) +
  scale_color_hue(labels = c("NO", "SI")) +
  geom_vline(aes(xintercept = coef_asim_normal), linetype = 2) +
  theme_bw() +
  labs(y = 'Muestras', x = 'Intervalos (nivel 0.95)',
       title = 'IC (método bootstrap percentil)')
```



Muestra Exp(1)

Definimos el coeficiente de asimetría para una muestra exponencial, el cual es 2.

```
set.seed(100)
R <- 1000
n <- 100
m <- 100
alfa <- 0.05
coef_asim_exp<-2
```

```
1. Método Híbrido

acierto <- NULL
intervalo <- NULL
for (i in 1:m){
  muestra_original <- rexp(100,1)
  coef_asim_original<-skewness(muestra_original)

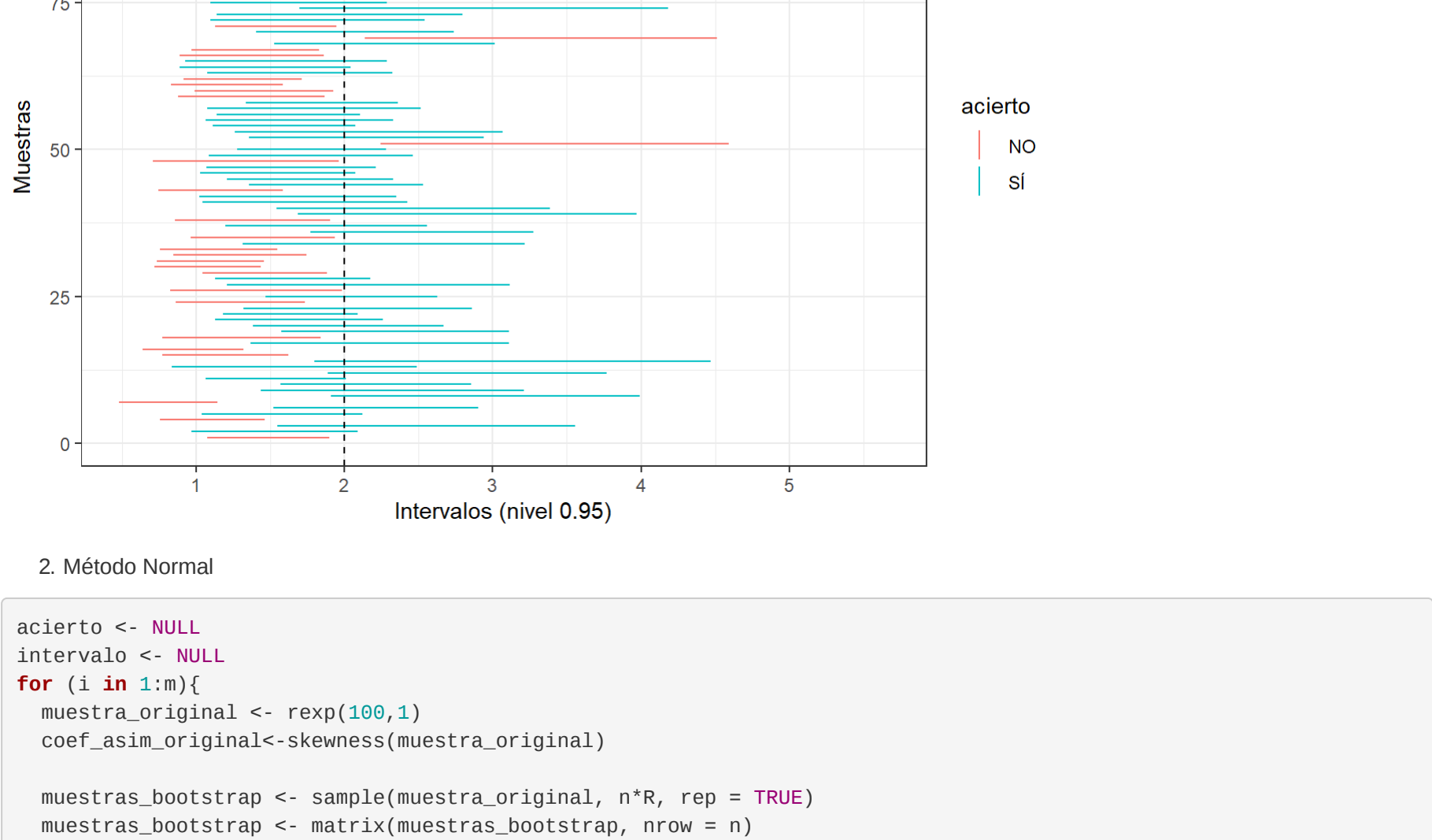
  muestras_bootstrap <- sample(muestra_original, n*R, rep = TRUE)
  muestras_bootstrap <- matrix(muestras_bootstrap, nrow = n)
  coef_asim_bootstrap <- apply(muestras_bootstrap, 2, skewness)
  T_bootstrap <- sqrt(n) * (coef_asim_bootstrap - coef_asim_original)
  ic_min <- coef_asim_original - quantile(T_bootstrap, 1-alfa/2)/sqrt(n)
  ic_max <- coef_asim_original + quantile(T_bootstrap, alfa/2)/sqrt(n)
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
  acierto <- c(acierto, ic_min < coef_asim_exp & ic_max > coef_asim_exp)
}

paste("% de acierto =",mean(acierto))

## [1] "% de acierto = 0.6"
```

```
df <- data.frame(ic_min <- intervalo[,1],
                 ic_max <- intervalo[, 2],
                 ind = 1:m,
                 acierto = acierto)

ggplot(df) +
  geom_linerange(aes(xmin = ic_min, xmax = ic_max, y = ind, col = acierto)) +
  scale_color_hue(labels = c("NO", "SI")) +
  geom_vline(aes(xintercept = coef_asim_exp), linetype = 2) +
  theme_bw() +
  labs(y = 'Muestras', x = 'Intervalos (nivel 0.95)',
       title = 'IC (método bootstrap híbrido)')
```



```
2. Método Normal

acierto <- NULL
intervalo <- NULL
for (i in 1:m){
  muestra_original <- rexp(100,1)
  coef_asim_original<-skewness(muestra_original)

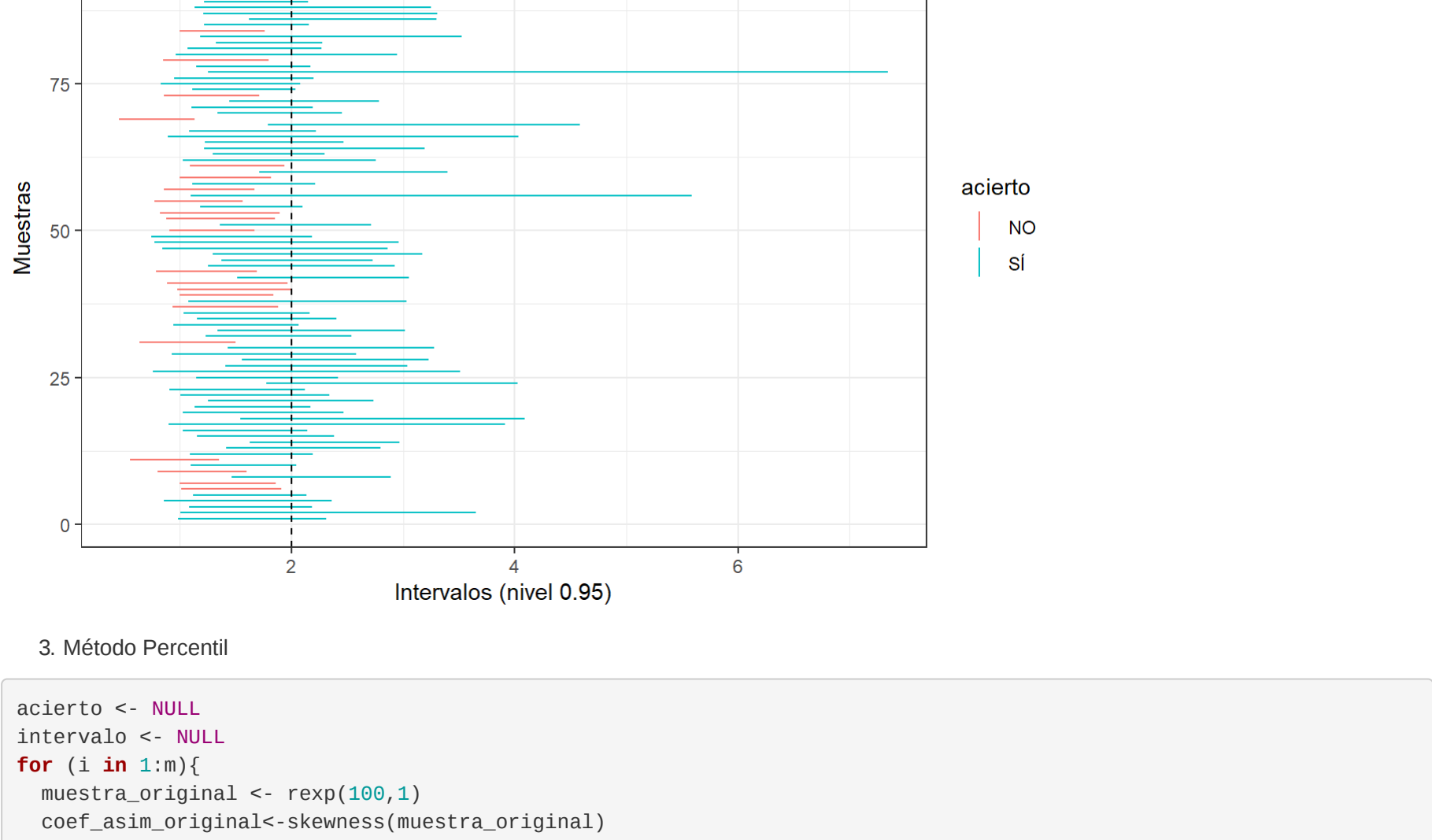
  muestras_bootstrap <- sample(muestra_original, n*R, rep = TRUE)
  muestras_bootstrap <- matrix(muestras_bootstrap, nrow = n)
  coef_asim_bootstrap <- apply(muestras_bootstrap, 2, skewness)
  et_bootstrap<-sd(coef_asim_bootstrap)
  ic_min <- coef_asim_original + qnorm(alfa/2)*et_bootstrap
  ic_max <- coef_asim_original - qnorm(alfa/2)*et_bootstrap
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
  acierto <- c(acierto, ic_min < coef_asim_exp & ic_max > coef_asim_exp)
}

paste("% de acierto =",mean(acierto))

## [1] "% de acierto = 0.75"
```

```
df <- data.frame(ic_min <- intervalo[,1],
                 ic_max <- intervalo[, 2],
                 ind = 1:m,
                 acierto = acierto)

ggplot(df) +
  geom_linerange(aes(xmin = ic_min, xmax = ic_max, y = ind, col = acierto)) +
  scale_color_hue(labels = c("NO", "SI")) +
  geom_vline(aes(xintercept = coef_asim_exp), linetype = 2) +
  theme_bw() +
  labs(y = 'Muestras', x = 'Intervalos (nivel 0.95)',
       title = 'IC (método bootstrap normal)')
```



```
3. Método Percentil

acierto <- NULL
intervalo <- NULL
for (i in 1:m){
  muestra_original <- rexp(100,1)
  coef_asim_original<-skewness(muestra_original)

  muestras_bootstrap <- sample(muestra_original, n*R, rep = TRUE)
  muestras_bootstrap <- matrix(muestras_bootstrap, nrow = n)
  coef_asim_bootstrap <- apply(muestras_bootstrap, 2, skewness)

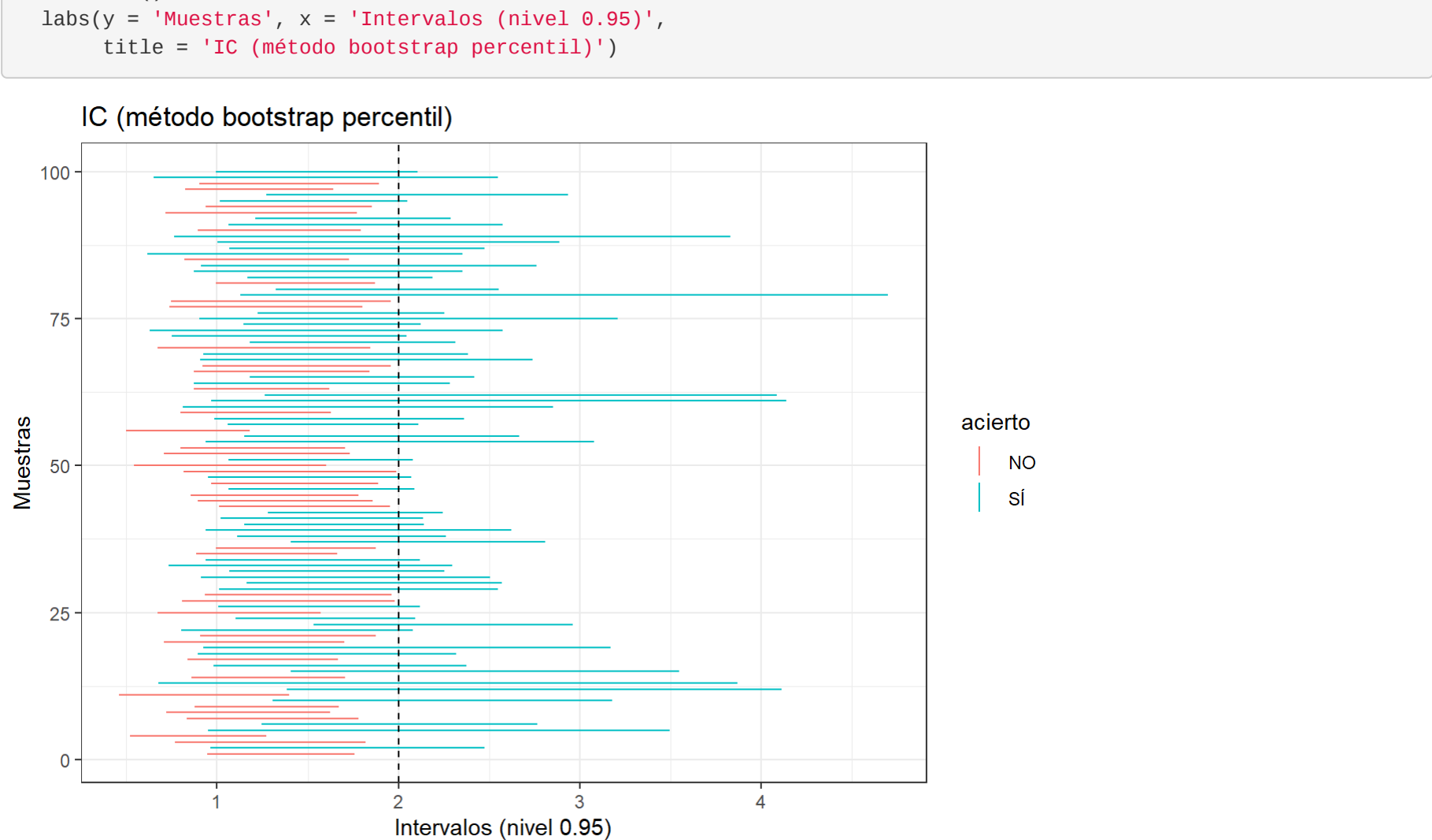
  ic_min <- quantile(coef_asim_bootstrap, alfa/2)
  ic_max <- quantile(coef_asim_bootstrap, 1-alfa/2)
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
  acierto <- c(acierto, ic_min < coef_asim_exp & ic_max > coef_asim_exp)
}

paste("% de acierto =",mean(acierto))

## [1] "% de acierto = 0.61"
```

```
df <- data.frame(ic_min <- intervalo[,1],
                 ic_max <- intervalo[, 2],
                 ind = 1:m,
                 acierto = acierto)

ggplot(df) +
  geom_linerange(aes(xmin = ic_min, xmax = ic_max, y = ind, col = acierto)) +
  scale_color_hue(labels = c("NO", "SI")) +
  geom_vline(aes(xintercept = coef_asim_exp), linetype = 2) +
  theme_bw() +
  labs(y = 'Muestras', x = 'Intervalos (nivel 0.95)',
       title = 'IC (método bootstrap percentil)')
```



Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.