Entrega de ejercicios Tema 3

Blanca Cano Camarero

9 de noviembre de 2022

Indice de contenidos


```
library(purrr)
library(ggplot2)
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v tibble  3.1.8      v dplyr   1.0.10
v tidyr   1.2.0      v stringr 1.4.1
v readr   2.1.2      v forcats 0.5.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
set.seed(5)
```

Ejercicio 7

```
muestra <- c(1, 2, 3.5, 4, 7, 7.3, 8.6, 12.4, 13.8, 18.1)
varianza <- var(muestra)
```

Apartado 1. Usa bootstrap para determinar el error típico de este estimador de σ^2.

Solución Generaremos nuevas muestras a partir de las que ya tenemos, calcularemos sus varianzas y finalemente el error típico de éstas.

```r
size <- length(muestra)
number_of_samples <- 1000

# Paso 1: Remuestro de los datos
remuestreo <- matrix(
  sample(muestra, size*number_of_samples, replace=TRUE),
  nrow = number_of_samples
)
# Paso 2: Cálculo de la varianza de cada remuestreo
varianzas_remuestreo <- apply(remuestreo, 1, var)

# Paso 3: Cálculo del error típico
et <- sd(varianzas_remuestreo)

cat('El error típico de remuestreo es ', et)
```

```
El error típico de remuestreo es  10.34001
```

Apartado 2 Compara el resultado con el error típico que darías si, por ejemplo, supieras que los datos proceden de una distribución normal.

Solución

Bajo hipótesis de normalidad podría aplicarse el lema de Fisher, que dice así:

Sean $X_1, X_2, \dots, X_n$ variables aleatorias independientes e identicamente distribuidas de una normal $\mathcal{N}(\mu, \sigma^2)$. Entonces:

1. $\bar{X}$ y S^2 son independientes.
2. $$\frac{(n-1)}{\sigma^2} S^2 \sim \chi^2_{n-1}$$
3. $\bar{X} \sim \mathcal{N}(\mu, \frac{\sigma^2}{n})$.

Puesto que nuestro objetivo es encontrar un estimador de la $Var(S^2)$ tomando la varianza de ambos miembros (2) resulta:

$$Var\left(\frac{n-1}{\sigma^2} S^2\right) = Var(\chi^2_{n-1})$$

que por las propiedades de la variaza y que $var(\chi^2_k) = 2k$ se tiene

$$\frac{(n-1)^2}{\sigma^4} Var\left(S^2\right) = 2(n-1),$$

Por lo que concluímos que

$$Var\left(S^2\right) = \frac{2\sigma^4}{n-1}.$$

Por el apartado anterior, σ puede ser estimada con S, por lo que finalmente podemos concluir que

$$\widehat{Var(S^2)} = \frac{2S^4}{n-1} \tag{1}$$

```
# El estimador de la varianza de una normal
et_2 <- 2*(varianza ^2)/(size - 1)
cat('Var(S^2) aprox ',et_2)
```

```
Var(S^2) aprox  211.3888
```

```
cat('\nEl erro típico es ', sqrt(et_2))
```

```
El erro típico es  14.53922
```

Notemos que esto solo se puede utilizar para una normal.

Además si comparamos los resultados con los del apartado primero, vemos que conocida la normal, la estimación del error típico es mayor; algo esperable ya que analizando la Ecuación 1, a mayor tamaño de muesta menor varianza y en nuestro caso tenemos una muestra relativamente pequeña.

Como moraleja, a pequeños tamaños de muestra no deberíamos de fiarnos en exceso del resultado boostrap (y ojo, tengamos presente que nuestro error típico procede de una estimación).

Apartado 3 Calcula un intervalo de confianza para σ^2 usando el método bootstrap híbrido. Fija $1 - \alpha = 0.95$.

Solución

Para explicar la idea que subyace en el diseño del algoritmo de *boostrap híbrido*, comenzaremos con las siguientes

Se define la proporción como

$$\tilde{H}_n(x) = \frac{1}{B}\sum_{b}^{B} I_{T^{*(b)} \leq x}.$$

Sea

$$H_n(x) = P_F\left(\sqrt{n}\left(\bar{X} - \mu\right) \leq x\right)$$

que por no ser conocido aproximaremos como

$$\hat{H}_n(x) = P_F\left(\sqrt{n}\left(\bar{X}^* - \bar{X}\right) \leq x\right)$$

$$1 - \alpha = P\left\{H_n^{-1}\left(\frac{\alpha}{2}\right) \leq \sqrt{n}\left(\hat{\theta} - \theta\right) \leq H_n^{-1}\left(1 - \frac{\alpha}{2}\right)\right\}$$

dando lugar al intervalo de confianza

$$\left[\hat{\theta} - \sqrt{n}H_n^{-1}\left(1 - \frac{\alpha}{2}\right), \hat{\theta} - \sqrt{n}H_n^{-1}\left(\frac{\alpha}{2}\right)\right]$$

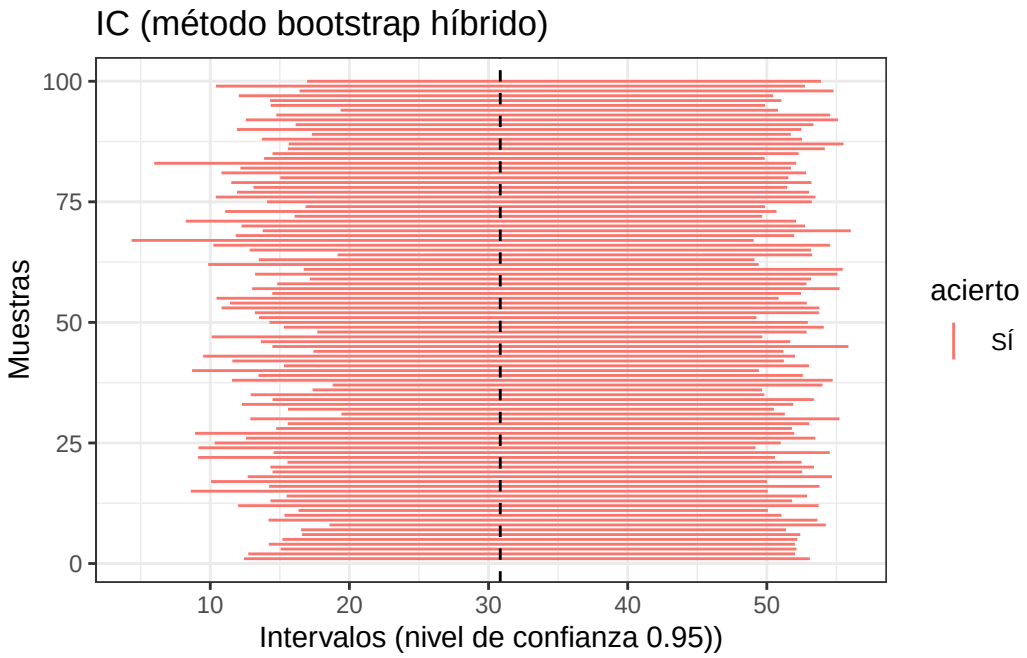
Puesto que H_n no es conocido los sustituiremos por el estimador de *bootstrap* $\hat{H}_n$ (que será la función `quantile` definida en R) y es el llamado *método híbrido*.

De esta manera resulta:

```
# --- Funciones auxiliares ---
# Construción de la inversa de H(H, muestra_ordenada, B^{-1})
H_inv <- function (alpha, muestra_ordenada, B_inv, acumulado = 0, index = 0) {
  if(acumulado < alpha){
    return (H_inv(alpha, muestra_ordenada, B_inv, acumulado + B_inv, index+1 ))
  }
  else{
    return(muestra_ordenada[index])
  }
}
# En lugar de emplear esta función utilizaremos la función `quantile`

# \hat \theta: Estimador de la varianza
```

```
# Parámetros
a = 0.05 # alpha
B = length(muestra) # tamaño del reemuestro
numero_remuestreos = 100
repeticiones_experimento = 100
```

```r
## variable auxiliares
B_inv = 1/B
acierto <- NULL
intervalo <- NULL

for(i in 1:repeticiones_experimento){

  muestras_boostrap <- matrix(
    sample(muestra, B*numero_remuestreos, rep=TRUE),
    nrow = numero_remuestreos
  )

  varianzas_bootstrap = apply(muestras_boostrap, 1, var)
  muestras_normalizadas <- sqrt(B)*(varianzas_bootstrap - varianza)
  ic_min <-varianza - quantile(muestras_normalizadas, 1-a/2)/sqrt(B)
  ic_max <-varianza - quantile(muestras_normalizadas, a/2)/sqrt(B)
  intervalo <- rbind(intervalo, c(ic_min, ic_max))
   acierto <- c(acierto, ic_min < varianza & ic_max > varianza)
}

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


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


IC (método bootstrap híbrido)
Muestras
100
75
50
25
0
10
20
30
40
50
Intervalos (nivel de confianza 0.95))
acierto
sí