Bootstrap

Contents

1 Introducción

El bootstrap es un método para estimar varianzas de estadísticos e intervalos de confianza utilizando simulaciones.

2 Estimación de la varianza con bootstrap

Sea $\{X_1, X_2, \cdots, X_n\}$ una muestra aleatoria simple (luego los datos son independientes y con igual distribución). Y sea $T = f(X_1, X_2, \cdots, X_n)$ un estadístico, es decir, T es cualquier función de los datos. Para calcular la varianza del estimador, $Var(T)$, el método bootstrap consiste en:

- Paso 1: generar, mediante simulación, una muestra con reemplazamiento a partir de $\{X_1, X_2, \cdots, X_n\}$ que llamaremos $\{X_1^*, X_2^*, \cdots, X_n^*\}$.
- Paso 2: Calcular la estimación de T a partir de la muestra bootstrap: $T^* = f(X_1^*, X_2^*, \cdots, X_n^*)$.
- Paso 3: Repetir los pasos 1 y 2 un total de B veces, obteniendo $T_1^*, T_2^*, \ldots, T_B^*$.
- Paso 4: estimar la varianza de T mediante la varianza de $T_1^*, T_2^*, \ldots, T_B^*$.

Por ejemplo, sea el número de viajeros diarios de una determinada línea de autobuses interurbana durante 12 días seleccionados aleatoriamente:

```
# muestra
x = c(47,66,55,53,49,65,48,44,50,61,60,55)
```

Supongamos que el número de viajeros de un día determinado tiene distribución normal: $X_i \sim N(\mu, \sigma)$. El estadístico que vamos a considerar en este ejemplo es la media muestral

$$T = \bar{X} = \frac{X_1 + X_2 + \cdots + X_{12}}{12}$$

La varianza de este estadístico tiene solución teórica:

$$Var(\bar{X}) = \frac{\sigma^2}{12}$$

Como σ^2 no es conocido esta varianza no se puede calcular. sin embargo, se puede aproximar utilizando el estimador de σ^2, la varianza de la muestra s_x^2:

$$Var(\bar{X}) \approx \frac{s_x^2}{12}$$

```
var(x)/length(x)
```

```
## [1] 4.370581
```

Vamos a obtener otra aproximación a este valor utilizando el método bootstrap:

```
set.seed(123)
B = 1000
medias = rep(0,B)
for (b in 1:B){
  replica = sample(x, replace = T)
  medias[b] = mean(replica)
}
var(medias)
```

```
## [1] 4.196051
```

Otro ejemplo puede ser si consideramos como estadístico la **mediana muestral**. En este caso no hay una fórmula teórica como en el caso de la media muestral. Así que aplicamos diréctamente bootstrap:

```
set.seed(123)
B = 1000
medianas = rep(0,B)
for (b in 1:B){
  replica = sample(x, replace = T)
  medianas[b] = median(replica)
}
var(medianas)
```

```
## [1] 10.22714
```

3 Estimación de intervalos de confianza utilizando bootstrap

Hay varios métodos para calcular el intervalo de confianza de un parámetro θ con bootstrap. Nosotros vamos a utilizar el método de los percentiles:

- Paso 1: generar, mediante simulación, una muestra con reemplazamiento a partir de $\{X_1, X_2, \cdots, X_n\}$ que llamaremos $\{X_1^*, X_2^*, \cdots, X_n^*\}$.
- Paso 2: Calcular la estimación de θ a partir de la muestra bootstrap: $\theta^* = T(X_1^*, X_2^*, \cdots, X_n^*)$.
- Paso 3: Repetir los pasos 1 y 2 un total de B veces, obteniendo $\theta_1^*, \theta_2^*, \ldots, \theta_B^*$.
- Paso 4: estimar el intervalo de θ mediante $(\theta_{\alpha/2}^*, \theta_{1-\alpha/2}^*)$.

Supongamos que el número de viajeros de un día determinado tiene distribución normal: $X_i \sim N(\mu, \sigma)$. Vamos a calcular el intervalo de confianza de μ. El intervalo teórico se calcula

$$\bar{x} \pm t_{\alpha/2;n-1} * \sqrt{\frac{s_x^2}{n}}$$

```
alfa = 0.05
n = length(x)
c(mean(x) + qt(alfa/2, n-1)*sqrt(var(x)/length(x)), mean(x) - qt(alfa/2, n-1)*sqrt(var(x)/length(x)) )
```

```
## [1] 49.81530 59.01803
```

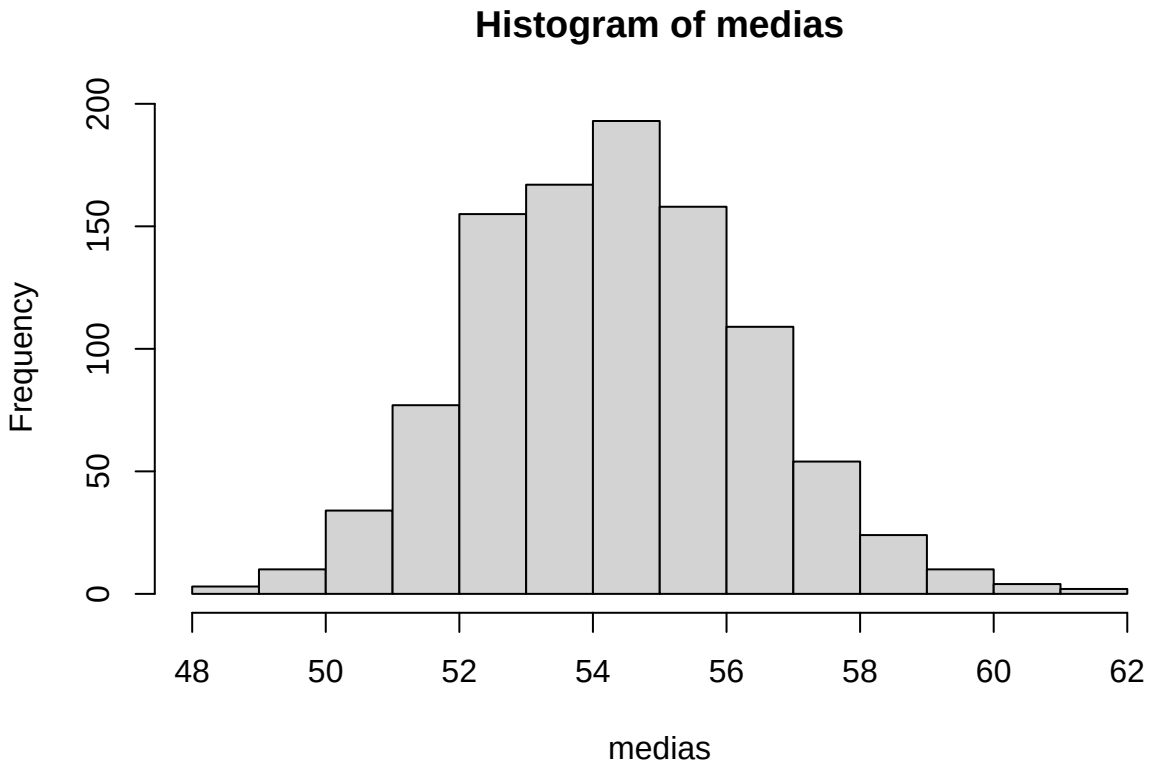
Vamos a calcularlo con bootstrap:

```
set.seed(123)
B = 1000
medias = rep(0,B)
for (b in 1:B){
  replica = sample(x, replace = T)
  medias[b] = mean(replica)
}
quantile(medias, c(alfa/2, 1-alfa/2))
```

```
##     2.5%    97.5%
## 50.58333 58.33333
```

En realidad tenemos la distribución del estimador de la media:

```
hist(medias)
```



También podemos calcular el intervalo de confianza de la mediana:

```
quantile(medianas, c(alfa/2, 1-alfa/2))
```

```
##  2.5% 97.5%
##  48.5  60.5
```

4 Bootstrap en el modelo de regresión

La hipótesis fundamental para poder aplicar bootstrap es que los datos tienen que ser independientes. En el caso del modelo de regresión, la opción es muestrear los residuos ya que son independientes. El método consiste en:

- Paso 1: estimar el modelo de regresión $y = X\hat{\beta} + e$ y calcular los residuos: $\{e_1, e_2, \cdots, e_n\}$.

- Paso 2: generar, mediante simulación, una muestra con reemplazamiento a partir de $\{e_1, e_2, \cdots, e_n\}$ que llamaremos $\{e_1^*, e_2^*, \cdots, e_n^*\}$.
- Paso 3: generar, a partir de los residuos bootstrap, una replica bootstrap de los datos: $y^* = X\hat{\beta} + e^*$.
- Paso 4: estimar los parámetros del modelo a partir de la muestra bootstrap $y^* = X\hat{\beta}^* + e$.
- Paso 5: Repetir los pasos 2, 3 y 4 un total de B veces, obteniendo $\beta_1^*, \beta_2^*, \ldots, \beta_B^*$.
- Paso 6: calcular la varianza de los estimadores o los intervalos de confianza de los parámetros a partir de los valores calculados en el paso 5.

Por ejemplo, vamos a calcular la varianza de los estimadores y los intervalos de confianza para el modelo:

```
d = read.csv("datos/kidiq.csv")
d$mom_hs = factor(d$mom_hs, labels = c("no", "si"))
```

```
m1 = lm(kid_score ~ mom_iq + mom_hs, data = d)
beta_e = coef(m1)
```

```
# BOOTSTRAP
# residuos
e = resid(m1)
# matrix de las X
X = model.matrix(m1)
# muestreamos los residuos CON REPOSICION
B = 1000
beta_e_b = matrix(0, nrow = B, ncol = 3)
for (i in 1:B){
  eb = sample(e, rep = T)
  yb = X %*% beta_e + eb
  mb = lm(yb ~ mom_iq + mom_hs, data = d)
  beta_e_b[i,] = coef(mb)
}
```

- Varianza de los parémetros estimados

```
# aplicando la teoría
diag(vcov(m1))
```

```
##  (Intercept)       mom_iq     mom_hssi
## 34.518069224  0.003669219  4.892113136
```

```
# aplicando bootstrap
apply(beta_e_b, 2, var)
```

```
## [1] 34.388633690  0.003732659  4.846163232
```

- intervalos de confianza

```
# aplicando la teoría
confint(m1)
```

```
##                 2.5 %     97.5 %
## (Intercept) 14.1839148 37.2791615
## mom_iq       0.4448487  0.6829634
## mom_hssi     1.6028370 10.2973969
```

```
# aplicando bootstrap
t(apply(beta_e_b, 2, quantile, probs = c(0.025,0.975)))
```

```
##            2.5%      97.5%
## [1,] 14.0208709 37.3910135
## [2,]  0.4442435  0.6818668
```

```
## [3,]  1.5552243 10.4017840
```