Métodos de Remuestreo

Tema 6. Intervalos de confianza basados en remuestreos

basado en

B. Efron, R. Tibshirani (1993). An Introduction to the bootstrap.

O. Kirchkamp (2017). Resampling methods.

Curso 2018/19

Introducción

- Los errores estándar se usan a menudo para calcular intervalos de confianza ***aproximados*** para los parámetros que se estudian.

- Dado un estimador $\hat{\theta}$ y un error estándar $\widehat{se}$, el intervalo de confianza al 95 % habitual es

$$\hat{\theta} \pm 1{,}96 \cdot \widehat{se}$$

  donde $z_{0,025} = 1{,}96$ procede de una distribución normal estándar.

- En el campo de los intervalos de confianza existen diferentes técnicas bootstrap y es un área de desarrollo teórico en evolución constante.

Introducción

- Supongamos que los datos $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ proceden de una distribución desconocida F.

- Si el tamaño muestral n es grande la distribución de $\hat{\theta}$ converge a una normal de media θ y varianza $\widehat{se}^2$, es decir

$$\frac{\hat{\theta} - \theta}{\widehat{se}} \sim N(0,1)$$

- De este modo, se obtiene un intervalo de confianza estándar con probabilidad de recubrimiento igual a $1 - \alpha$

$$\left[\hat{\theta} + z_{1-\frac{\alpha}{2}} \widehat{se};\ \hat{\theta} + z_{\frac{\alpha}{2}} \widehat{se}\right]$$

Introducción

- La propiedad de recubrimiento implica que aproximadamente en el $(1-\alpha)\cdot 100\,\%$ de las ocasiones este intervalo contiene el verdadero valor de θ.

- El bootstrap se puede usar para mejorar los intervalos de confianza; de hecho, cuando n es muy grande los intervalos bootstrap y los aproximados convergen a los mismos valores.

- Se puede calcular el estimador *plug-in* $\hat{\theta} = t(\hat{F})$ del estadístico de interés $\theta = t(F)$ también, a su vez, un error estándar $\widehat{se}$ basado en un método bootstrap o en un *jackknife*.

Ejemplo

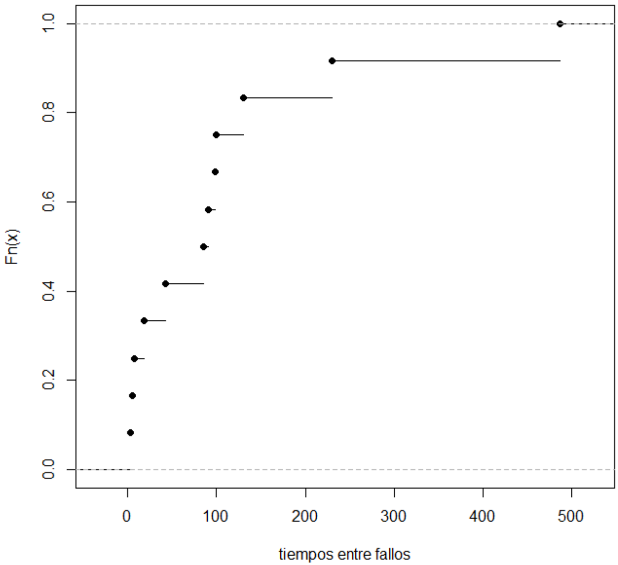
- **Ejemplo**: Datos sobre la fiabilidad de aparatos de aire acondicionado (tiempos de fallo).

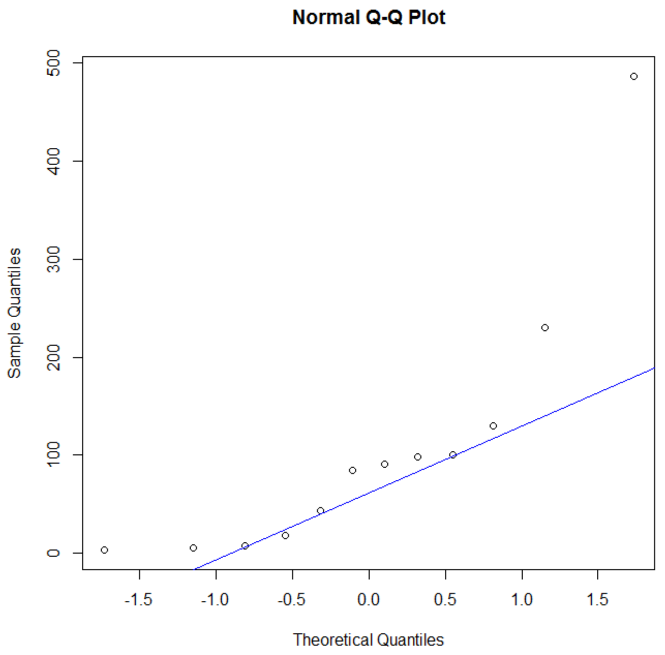
- Los datos no parecen distribuirse como una normal.

```
data(aircondit, package="boot")

plot(ecdf(aircondit$hours),main="",
xlab="tiempos entre fallos")

qqnorm(aircondit$hours)
qqline(aircondit$hours, col="blue")
```


Fn(x)
0.0
0.2
0.4
0.6
0.8
1.0
0
100
200
300
400
500
tiempos entre fallos



Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
0
100
200
300
400
500
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5

Ejemplo

- Se podría usar un enfoque *exacto* para los intervalos de confianza:
  1. Funciona incluso con muestras pequeñas.
  2. Pero requiere el conocimiento de la *verdadera* distribución.
  3. Para aplicarse a veces hay que hacer transformaciones de los datos.
- En el caso del ejemplo de aires acondicionados, si el número de fallos sigue una distribución de Poisson, entonces los tiempos entre fallos siguen una distribución exponencial.

Ejemplo

- Supongamos que la función de densidad es

$$f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

- Para estimar λ se puede usar el estimador de máxima verosimilitud (MLE):

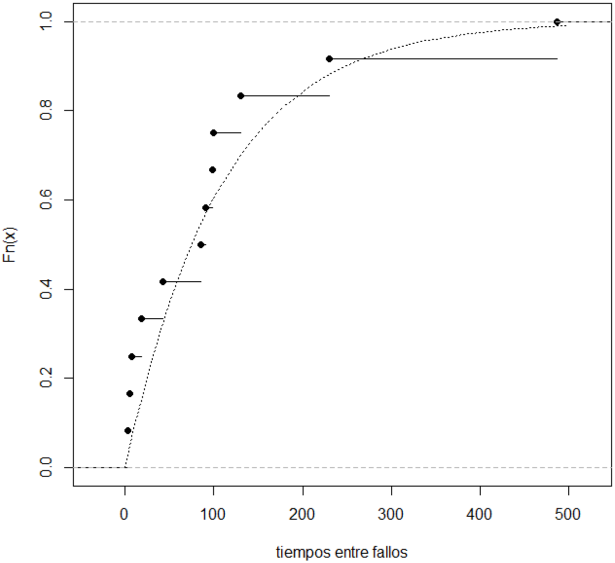
$$\hat{\lambda} = \frac{n}{\sum_i x_i} = \frac{1}{\overline{x}}$$

Ejemplo

- En este caso,

```
n = length(aircondit$hours)
lambda = length(aircondit$hours)/sum(aircondit$hours)

plot(ecdf(aircondit$hours),main="",
xlab="tiempos entre fallos")
with(list(x=seq(0,500,10)),
lines(1-exp(-lambda*x) ~ x,lty="dotted"))
```



1.0
0.8
0.6
0.4
0.2
0.0
Fn(x)
0
100
200
300
400
500
tiempos entre fallos

Ejemplo

- Al asumir una distribución exponencial, se puede calcular el intervalo de confianza exacto para el parámetro (o para la media). Ver:

```
en.wikipedia.org/wiki/Exponential_distribution
```

$$IC_{\frac{1}{\lambda}} = \left[\frac{\bar{x} \cdot 2n}{\chi^2_{2n,1-\frac{\alpha}{2}}}; \; \frac{\bar{x} \cdot 2n}{\chi^2_{2n,\frac{\alpha}{2}}} \right]$$

- De este modo

```
n = length(aircondit$hours)
media = sum(aircondit$hours)/length(aircondit$hours)
alfa = 0.05

sapply(c(1-alfa/2, alfa/2),
function(alfa) (2*n*media)/qchisq(alfa,2*n))
```

```
[1]  65.89765 209.17415
```

Aproximación a la normal

- La aproximación a la normal se puede hacer cuando la muestra es grande.
- Así, la media muestral se distribuye asintóticamente como una normal aplicando el *TCL* (teorema Central del Límite).
- Aproximación normal basada en estimadores paramétricos de σ:

```
confint(lm(hours ~ 1, data=aircondit))
```

```
                2.5 %   97.5 %
(Intercept) 21.52561 194.6411
```

Aproximación a la normal

- Aproximación normal basada en estimadores bootstrap de μ y σ.

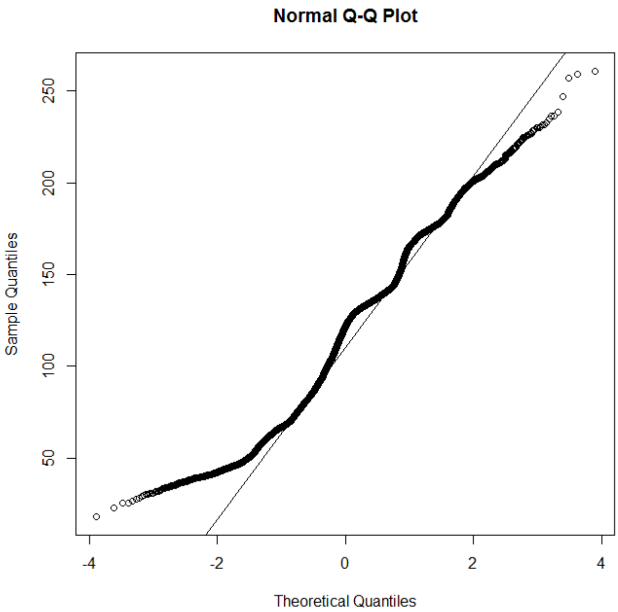
```
mediasdboot = with(aircondit, replicate(5000,{
guy = sample(hours, replace=TRUE);
c(media=mean(guy), sdev=sd(guy))}))

mean(mediasdboot["media",]) +
c(1,-1)*qnorm(alfa/2)*sd(mediasdboot["media",])
```

```
[1]  34.83928 182.09636
```

- Comprobamos el ajuste de la distribución de la media a la normal. Tampoco funciona bien.

```
qqnorm(mediasdboot)
qqline(mediasdboot)
```


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
50
100
150
200
250
-4
-2
0
2
4

Intervalos t de Student

- Una manera de mejorar los intervalos asintóticos, cuando las muestras son pequeñas, es usar intervalos t de Student.
- En 1908 Gosset (the *Student*) calculó la distribución de:

$$t = \frac{\hat{\theta} - \theta}{\widehat{se}} \sim t_{n-1}$$

  donde t_{n-1} es una distribución t de Student.
- Mediante esta aproximación el intervalo queda como

$$\left[\hat{\theta} - t_{n-1,1-\frac{\alpha}{2}}\widehat{se};\ \hat{\theta} + t_{n-1,\frac{\alpha}{2}}\widehat{se}\right]$$

- Pero no tiene en cuenta el ajuste necesario de los intervalos de confianza cuando aparece *asimetría*.
- Se puede mejorar esta aproximación mediante los intervalos bootstrap t.

Intervalos bootstrap t

- Usando el bootstrap se pueden obtener intervalos ajustados sin necesidad de asumir normalidad en los datos.
- Los intervalos que se construyen reciben el nombre de aproximación *bootstrap-t*.
- En este método se calcula una tabla de cuantiles de t directamente a partir de los datos que se tienen.
- Esta tabla se usa entonces para construir intervalos de confianza de la misma manera que en el caso de los intervalos clásicos.
- La tabla de valores remuestreados se construye generando B muestras bootstrap y luego calculando la versión bootstrap del estadístico t para cada uno de ellos.

Intervalos bootstrap t

- El método consiste en generar B muestras bootstrap $\mathbf{x}^{*1}, \mathbf{x}^{*2}, \ldots, \mathbf{x}^{*B}$. A continuación, para cada una de ellas se calcula

$$Z^*(b) = \frac{\hat{\theta}^*(b) - \hat{\theta}}{\widehat{se}^*(b)}$$

- donde $\hat{\theta}^*(b) = s(\mathbf{x}^{*b})$ es el valor de $\hat{\theta}$ para la muestra bootstrap $\mathbf{x}^{*b}$ y $\widehat{se}^*(b)$ es el valor estimado del error estándar de $\hat{\theta}^*$ para la muestra bootstrap $\mathbf{x}^{*b}$.

- El α-esimo percentil de $Z^*(b)$ se estima mediante el valor $\hat{t}_\alpha$ tal que

$$\frac{\#\left\{Z^*(b) \leq \hat{t}_\alpha\right\}}{B} = \alpha$$

Intervalos bootstrap t

- Por ejemplo, si $B = 1000$ el estimador del punto 5 % es el 50-esimo valor mayor de los $Z^*(b)$ ordenados y el estimador del punto 95 % es el 950-esimo valor mayor de los $Z^*(b)$ ordenados.
- De este modo, el estimador bootstrap-t que queda es

$$\left[\hat{\theta} - \hat{t}_{1-\alpha} \cdot \widehat{se};\ \hat{\theta} - \hat{t}_{\alpha} \cdot \widehat{se}\right]$$

- Es necesario considerar un número de réplicas B mayor que en el caso del cálculo de los errores estándar bootstrap.
- Por otro lado, los percentiles bootstrap-t pueden diferir bastante de los percentiles t de Student habituales.

Ejemplo

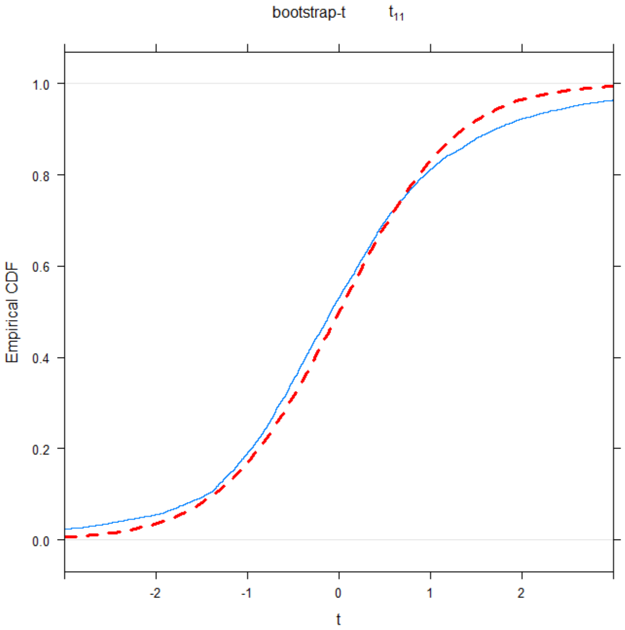
- En el ejemplo del aire acondicionado:

```
N = dim(aircondit)[1]

thetaHat = mean(aircondit[["hours"]])
bT = (mediasdboot["mean",]-thetaHat)/
     (mediasdboot["sd",]/sqrt(N))

library(latticeExtra)

ecdfplot(bT,xlim=c(-3,3),xlab="t",
key=simpleKey(c("bootstrap-t",expression(t[11])),
points=FALSE,lines=FALSE,columns=2))+
layer(panel.curve(pt(x,N-1),col='red',lty=2,lwd=3))
```


bootstrap-t
t_{11}
Empirical CDF
0.0
0.2
0.4
0.6
0.8
1.0
-2
-1
0
1
2
t

Ejemplo

- Se puede observar que se necesitan extraer muchas muestras bootstrap para obtener unos resultados ajustados.

- Si comparamos ahora los cuantiles de la distribución original t de Student con los cuantiles bootstrap se obtienen resultados muy diferentes.

```
alfa = 0.05
(extremos = quantile(bT, c(1-alfa/2, alfa/2)))
```

```
     2.5%      97.5%
-2.869611   3.610294
```

```
(t.standard = qt(c(1-alfa/2, alfa/2),N-1))
```

```
[1] 2.200985 -2.200985
```

Ejemplo

```
# Intervalo bootstrap-t
mean(mediasdboot["media",]) -
extremos*sd(mediasdboot["media",])
```

```
   97.5%      2.5%
 50.44279 285.69572
```

```
# Intervalo t Student original
mean(mediasdboot["media",]) -
t.standard*sd(mediasdboot["media",])
```

```
[1]  25.58384 190.23446
```

Ejemplo

- Si se usa la librería bootstrap:

```
# Con la libreria bootstrap
library(bootstrap)

lamedia = function(x){mean(x)}
resulta = boott(aircondit[["hours"]], lamedia,
VS=TRUE, perc=c(0.025, 0.975))

resulta$confpoints
```

```
    0.025     0.975
62.41809  258.1767
```

Características de los intervalos bootstrap-t

- Los valores de la t de Student o de la normal son simétricos alrededor del cero. De este modo se obtienen intervalos simétricos alrededor del estimador del parámetro $\hat{\theta}$.

- Por el contrario, los percentiles bootstrap-t pueden ser *asimétricos* con respeto al estimador, lo que lleva a obtener intervalos que son más largos o cortos a la derecha o la izquierda.

- Esta asimetría representa una mejora en el recubrimiento que tienen estos intervalos.

- El procedimiento bootstrap-t es una buena opción particularmente en estadísticos de localización como la media muestral, la mediana o los percentiles muestrales.

Desventajas de los intervalos bootstrap-t

- Pero hay problemas computacionales y de interpretación en los intervalos bootstrap-t.
- En el denominador del estadístico $Z^*(b)$ se necesita conocer $\widehat{se}^*(b)$, es decir, la desviación estándar de $\hat{\theta}^*$ para cada muestra bootstrap $\mathbf{x}^{*b}$.
- En el caso de la media, existe una expresión explícita para calcularlo, pero para muchos estadísticos no se tiene una expresión formal para ello.
- La única opción sería calcular un estimador bootstrap del error estándar para **cada** muestra bootstrap lo cual no es muy operativo.

Desventajas de los intervalos bootstrap-t

- Otro problema que se presenta en los intervalos bootstrap-t es que se puede comportar de manera errática con muestras pequeñas.
- También los intervalos bootstrap-t **no** son invariantes cuando se aplican transformaciones al parámetro original que se estima.
- A veces, interesa transformar los datos para encontrar mejores estimadores de θ. Por ejemplo, cambiando la escala de las observaciones.
- Pero a menudo no se sabe cuál es la mejor transformación para aplicar a los datos.

Intervalos *básicos*

- Otra opción para intervalos bootstrap, son los *intervalos básicos*

- Si se asume que $\hat{\theta} - \theta$ tiene la misma distribución que $\hat{\theta}^* - \hat{\theta}$, entonces, el intervalo básico se define como

$$IC_{bas} = \left[2\hat{\theta} - \hat{\theta}^*_{1-\frac{\alpha}{2}};\ 2\hat{\theta} - \hat{\theta}^*_{\frac{\alpha}{2}}\right]$$

- Propiedades de los intervalos básicos:

  1. No son invariantes mediante transformaciones.

  2. Corrigen el sesgo siempre que $\hat{\theta} - \theta$ tenga la misma distribución que $\hat{\theta}^* - \hat{\theta}$.

Intervalos tipo percentil

- Se pueden considerar otro tipo de intervalos basados en los *percentiles*.
- Supongamos que se obtiene una muestra $\mathbf{x}^*$ a partir de $\hat{P} \rightarrow \mathbf{x}^*$ y se calculan luego los estadísticos $\hat{\theta}^* = s(\mathbf{x}^*)$.
- Si se denomina $\hat{G}$ la función de distribución acumulada de $\hat{\theta}^*$, entonces, el *intervalo percentil* de nivel $1-\alpha$ se define como los percentiles $\frac{\alpha}{2}$ y $1-\frac{\alpha}{2}$ de $\hat{G}$:

$$\left[\hat{\theta}_{\%low};\ \hat{\theta}_{\%up}\right] = \left[\hat{G}^{-1}\left(\frac{\alpha}{2}\right);\ \hat{G}^{-1}\left(1-\frac{\alpha}{2}\right)\right]$$

Intervalos tipo percentil

- Como por definición $\hat{G}^{-1}(\alpha) = \hat{\theta}^*_{(\alpha)}$ se puede reescribir también el intervalo como

$$\left[\hat{\theta}_{\%low};\ \hat{\theta}_{\%up}\right] = \left[\hat{\theta}^*_{(\frac{\alpha}{2})};\ \hat{\theta}^*_{(1-\frac{\alpha}{2})}\right]$$

- La expresión anterior se refiere a una situación ideal donde hay infinitas réplicas bootstrap. En la práctica se toma un número finito B de ellas.

- Se generan B conjuntos de datos bootstrap independientes $\mathbf{x}^{*1}, \mathbf{x}^{*2}, \ldots, \mathbf{x}^{*B}$ y se calculan los respectivos estimadores $\hat{\theta}^*(b) = s(\mathbf{x}^{*b})$, donde $b = 1, \ldots, B$.

Intervalos tipo percentil

- Se denomina como $\hat{\theta}^*_{B(\alpha)}$ el precentil α-esimo de los valores $\hat{\theta}^*(b)$, es decir, el $B \cdot \alpha$-esimo valor en la lista ordenada de las B réplicas de $\hat{\theta}^*$.
- Por ejemplo, si $B = 2000$ y $\alpha = 0{,}05$ entonces $\hat{\theta}^*_{B(\alpha)}$ es el valor 100-esimo en la lista ordenada de réplicas.
- Del mismo modo, se razonaría con $\hat{\theta}^*_{B(1-\alpha)}$, el precentil $1-\alpha$-esimo de los valores.

Intervalos tipo percentil

- De este modo, el intervalo percentil $1-\alpha$ se define como

$$\left[\hat{\theta}_{\%low};\ \hat{\theta}_{\%up}\right] \approx \left[\hat{\theta}^{*}_{B(\frac{\alpha}{2})};\ \hat{\theta}^{*}_{B(1-\frac{\alpha}{2})}\right]$$

- Si la distribución de $\hat{\theta}^{*}$ es aproximadamente normal, entonces los intervalos precentil y normal estándar coinciden.

```
data(aircondit, package="boot")

alfa = 0.05
meansdboot = with(aircondit, replicate(5000,
c(mean=mean(sample(hours, replace=TRUE)),
sd=sd(sample(hours, replace=TRUE)))))

quantile(meansdboot["mean",], c(alfa/2, 1-alfa/2))
```

```
   2.5%    97.5%
48.6625 190.7583
```

Intervalos tipo percentil

- Para mejorar los intervalos de confianza es necesario realizar a veces transformaciones distintas a la clásica del logaritmo.
- El método del percentil permite incorporar de manera automática dicho tipo de transformaciones.
- El siguiente lema formaliza el hecho de que el método del percentil siempre *descubre* cuál es la transformación correcta.

Intervalos tipo percentil

- **Lema del intervalo del percentil**:
  Supongamos que existe una función o transformación m del parámetro tal que convierte en normal a la distribución de $\hat{\theta}$, es decir $\hat{\phi} = m(\hat{\theta})$

  $$\hat{\phi} \sim N(\phi, c^2)$$

  con una desviación estándar dada c.

- Entonces el intervalo del percentil basado en $\hat{\theta}$ se obtiene como la transformación inversa

  $$\left[m^{-1}\left(\hat{\phi} - z_{1-\alpha} \cdot c\right);\ m^{-1}\left(\hat{\phi} - z_{\alpha} \cdot c\right)\right]$$

Intervalos tipo percentil

- Supongamos un ejemplo donde se genera una muestra de tamaño 10 de una distribución normal estándar.
- El parámetro de interés que se trata de estimar es $\theta = e^{\mu}$ donde μ es la media poblacional.
- El verdadero valor de θ es $e^0 = 1$ y el estimador muestral es $\hat{\theta} = e^{\overline{x}}$.
- Si se generan muestras bootstrap $\hat{\theta}^*$ se obtiene una distribución bastante asimétrica y los intervalos percentil al 95 % resultan ser

$$\left[\hat{\theta}_{\%low};\ \hat{\theta}_{\%up}\right] = [0{,}75;\ 2{,}07]$$

- Si se calculan los intervalos estándar normales basados en un error estándar $\widehat{se} = 0{,}34$, se obtiene $1{,}25 \pm 1{,}96 \cdot 0{,}34 = [0{,}59;\ 1{,}92]$ que son muy diferentes a los anteriores.

Intervalos tipo percentil

- Sin embargo esta aproximación no es muy buena dado que la distribución del estadístico no es normal.
- Si se toma la transformación logaritmo sobre los datos y se considera el parámetro $\mu = \log(\theta)$ entonces se obtiene un intervalo estándar normal igual a $[-0{,}28;\ 0{,}73]$
- La distribución de los datos transformados es más simétrica y el intervalo estándar normal es más ajustado.
- Si se toma el antilogaritmo de los extremos de este intervalo, se obtiene $[0{,}76;\ 2{,}08]$ que es casi idéntico al obtenido mediante el percentil bootstrap.
- Es decir, el intervalo percentil encuentra de manera automática la transformación adecuada para los datos.

Intervalo de sesgo-corregido y acelerado *BCa*

- Los intervalos bootstrap-t y el percentil pueden conducir a estimaciones del intervalo de confianza algo erráticas, especialmente el intervalo percentil cuando el estimador es sesgado respecto al parámetro cuyo intervalo de confianza queremos aproximar.
- Se puede considerar una version mejorada del intervalo percentil llamada *BCa*, abreviatura que procede de sesgo-corregido (*bias-corrected*) y acelerado (*accelerated*).
- En la determination del intervalo *BCa* utilizaremos dos cantidades, $\hat{z}_0$ y $\hat{a}$.
- La primera, $\hat{z}_0$, se introduce para corregir el sesgo del estimador $\hat{\theta}$.

Intervalo de sesgo-corregido y acelerado *BCa*

- $\hat{z}_0$ se define como

$$\boxed{\hat{z}_0 = \Phi^{-1}\left(\frac{\#\left\{\hat{\theta}^*(b) < \hat{\theta}\right\}}{B}\right)}$$

- Representa el sesgo mediano de la estimation dada por las B réplicas bootstrap del estimador.

- Es decir, la discrepancia entre la mediana de dicha distribution de frecuencias y $\hat{\theta}$, en unidades normales.

- Obtenemos el valor $\hat{z}_0 = 0$, cuando exactamente la mitad de las observaciones son menores que $\hat{\theta}$.

Intervalo de sesgo-corregido y acelerado *BCa*

- La segunda cantidad, $\hat{a}$, denominada *aceleración*, corrige para el caso en que el error estándar $se(\hat{\theta})$ no sea constante, y se define en términos de los valores *jackknife*, como

$$\hat{a} = \frac{\sum_{i=1}^{n}\left(\hat{\theta}_{(\cdot)} - \hat{\theta}_{(i)}\right)^{3}}{6\left[\sum_{i=1}^{n}\left(\hat{\theta}_{(\cdot)} - \hat{\theta}_{(i)}\right)^{2}\right]^{3/2}}$$

- donde $\hat{\theta}_{(i)}$ es la *i*-esima réplica *jackknife* del estimador tal que

$$\begin{aligned} \hat{\theta}_{(i)} &= s(\mathbf{x}_{(i)}) \\ \hat{\theta}_{(\cdot)} &= \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_{(i)} \end{aligned}$$

Intervalo de sesgo-corregido y acelerado BCa

- El intervalo de sesgo-corregido y acelerado (intervalo BCa) de coeficiente de confianza $1-\alpha$ se define igual que el intervalo percentil con unos valores particulares α_1 y α_2

$$\left[\hat{\theta}^*_{\alpha_1};\ \hat{\theta}^*_{\alpha_2}\right]$$

- tal que

$$\begin{aligned}
\alpha_1 &= \Phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z_{1-\frac{\alpha}{2}}}{1-\hat{a}\left(\hat{z}_0 + z_{1-\frac{\alpha}{2}}\right)}\right)\\
\alpha_2 &= \Phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z_{\frac{\alpha}{2}}}{1-\hat{a}\left(\hat{z}_0 + z_{\frac{\alpha}{2}}\right)}\right)
\end{aligned}$$

- siendo $z_{\frac{\alpha}{2}}$ y $z_{1-\frac{\alpha}{2}}$ los cuantiles habituales de la normal estándar y Φ la función de distribución de una normal estándar.

Intervalo de sesgo-corregido y acelerado *BCa*

- Se puede observar que si $\hat{z}_0 = \hat{a} = 0$ queda

$$\begin{aligned} \alpha_1 &= \Phi\left(z_{1-\frac{\alpha}{2}}\right) = \frac{\alpha}{2} \\ \alpha_2 &= \Phi\left(z_{\frac{\alpha}{2}}\right) = 1 - \frac{\alpha}{2} \end{aligned}$$

  de forma que el intervalo *BCa* coincide con el intervalo percentil.

- El valor de $\hat{z}_0$ translada el intervalo a la derecha o la izquierda y $\hat{a}$ hace que el intervalo sea más ancho o más estrecho.

- En las aplicaciones de este intervalo se recomienda utilizar, al menos $B = 1000$ muestras bootstrap.

- Al igual que el intervalo percentil, el *BCa* también *adivina* las transformaciones del parámetro.

Intervalo *BCa* con R

```
boot.BCa =
  function(x, th0, th, stat, conf=0.95) {
      # bootstrap con intervalos de confianza BCa
      # th0 es el estadistico observado
      # th es el vector de replicas bootstrap
      # stat es la funcion que calcula el estadistico

      x = as.matrix(x)
      n = nrow(x)      # observaciones en filas
      N = 1:n
      alfa = (1 + c(-conf, conf))/2
      zalfa = qnorm(alfa)

      # Factor de correccion del sesgo
      z0 = qnorm(sum(th < th0) / length(th))
```

Intervalo *BCa* con R

```
# factor de aceleracion (est. jackknife)
th.jack = numeric(n)
for (i in 1:n) {
   J = N[1:(n-1)]
   th.jack[i] = stat(x[-i, ], J)
}
L = mean(th.jack) - th.jack
a = sum(L^3)/(6 * sum(L^2)^1.5)

# limites confianza BCa
adj.alfa = pnorm(z0 + (z0+zalfa)/(1-a*(z0+zalfa)))
limits = quantile(th, adj.alfa, type=6)
return(list("est"=th0, "BCa"=limits))
}
```

Intervalo *BCa* con R

```
data(patch, package="bootstrap")
n = nrow(patch)
B = 2000
x = cbind(patch$y, patch$z)
theta.b = numeric(B)
theta.hat = mean(patch$y) / mean(patch$z)

for (b in 1:B) {
   i = sample(1:n, size=n, replace=TRUE)
   y = patch$y[i]
   z = patch$z[i]
   theta.b[b] = mean(y)/mean(z)
}

estadis = function(dat, index) {
      mean(dat[index, 1])/mean(dat[index, 2])}
```

Intervalo *BCa* con R

- Se obtiene como resultado

```
boot.BCa(x, th0=theta.hat, th=theta.b, stat=estadis)
```

```
$est
[1] -0.0713061

$BCa
 3.671113%  98.40853%
-0.2183023  0.1910999
```

Ejemplos de intervalo *BCa*

- Con la librería `bootstrap`:

```
library(bootstrap)

xdata = matrix(rnorm(30),ncol=2)
n = 15

theta = function(ind, xdata){
cor(xdata[ind, 1],xdata[ind, 2])}

bcanon(1:n, 100, theta, xdata,
alpha=c(0.025, 0.975))$confpoints
```

```
     alpha   bca point
[1,] 0.025 -0.5110059
[2,] 0.975  0.5230146
```

Ejemplos de intervalo *BCa*

- Con la librería boot:

```
x = c(10, 27, 30, 40, 46, 51, 52, 104, 146)

library(boot)

mean.boot = function(x,ind){
return(c(mean(x[ind]), var(x[ind])/length(ind)))
}

eso = boot(x, mean.boot, 1000)
```

Ejemplos de intervalo *BCa*

- Se obtienen como resultados

```
boot.ci(eso, index=1)
```

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates

CALL :
boot.ci(boot.out = eso, index = 1)

Intervals :
Level      Normal              Basic
95%   (31.70, 81.81 )   (28.67, 78.66 )

Level     Percentile            BCa
95%   (33.78, 83.78 )   (37.67, 92.03 )
```

Ejemplos de intervalo *BCa*

```
boot.ci(eso, index=2)
```

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates

CALL :
boot.ci(boot.out = eso, index = 2)

Intervals :
Level      Normal              Basic
95%   ( 46.0, 405.5 )   ( 43.9, 384.1 )

Level     Percentile            BCa
95%   ( 16.8, 357.1 )   ( 71.5, 434.2 )
```

Ejemplos de intervalo *BCa*

- Con la librería boot y regresión:

```
d = data.frame(x=1:20, y=runif(20))
m1 = lm(y~x, data=d)

lmcoef = function(data, i){
      d = data[i, ]
      d.reg = lm(y~x, d)
      c(coef(d.reg)) }

m1
```

```
Call:
lm(formula = y ~ x, data = d)

Coefficients:
(Intercept)            x
   0.566358    -0.009771
```

Ejemplos de intervalo *BCa*

- Se obtiene como resultado

```
lmboot = boot(d, lmcoef, R=1000)
boot.ci(lmboot, index=2)
```

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates

CALL :
boot.ci(boot.out = lmboot, index = 2)

Intervals :
Level      Normal              Basic
95%   (-0.0267, 0.0053)   (-0.0270, 0.0048)

Level     Percentile            BCa
95%   (-0.0243, 0.0075)   (-0.0259, 0.0046)
```