# Intervalo de confianza Bootstrap

*Francisco Paz*

*30/8/2019*

```
library(pander) #Paquetería para sacar tablas más bonitas
library(tidyverse)
```

### Intervalo de confianza Bootstrap

Hasta ahora vimos lo que es una estimación plug-in y la idea de bootstrap, pero vale la pena recalcar un par de cosas.

- Para el caso en el que el estadistico $\theta = \bar{x}$ el principio de plug-in para el calculo del error estandar es inmediato. En general no siempre es fácil e inmediata.
- La idea del método de bootstrap es utilizar la simulación para aproximar el error estandar

Para cambiar un poco las cosas el siguiente ejemplo será con la asimetría o skewness. Sabemos que para la distribución normal esta debe valer 0

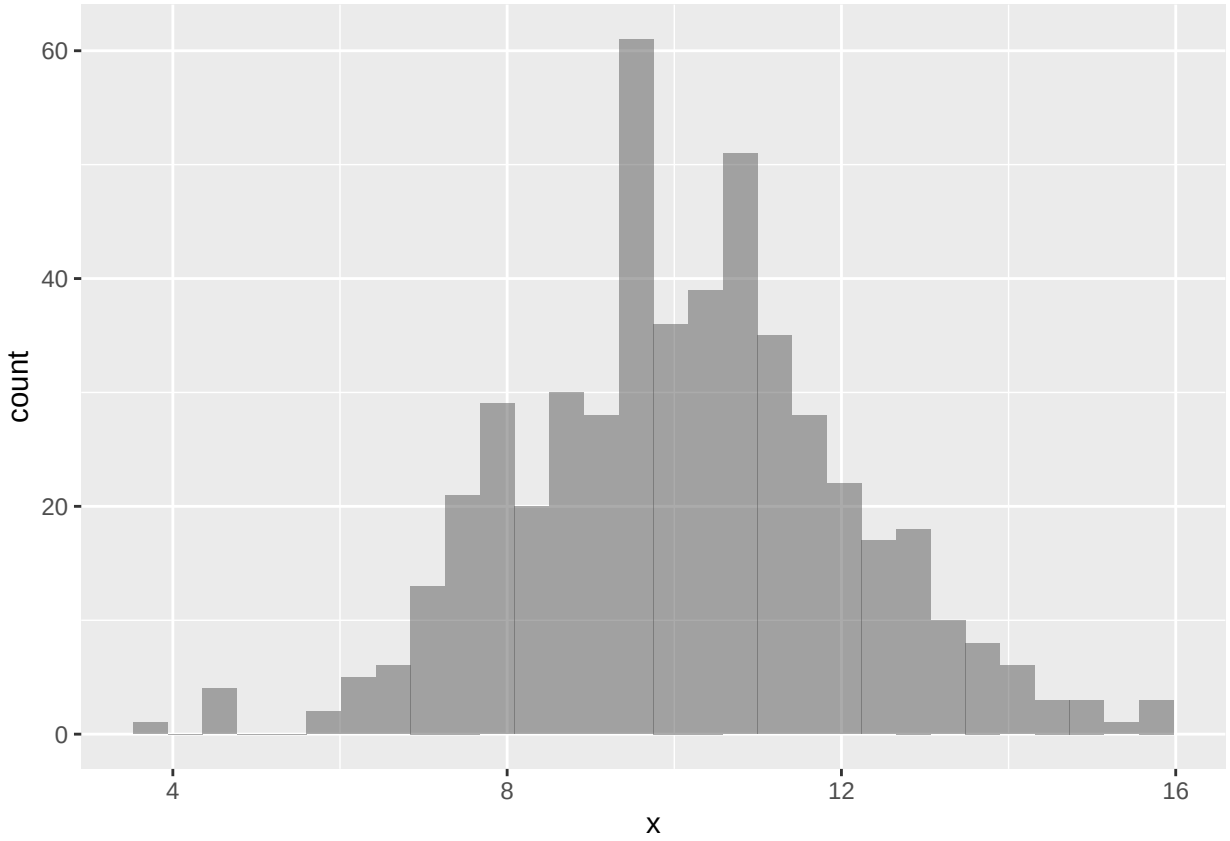
```
set.seed(7926)
s <-  function(x){
  n <- length(x)
  1/n * sum((x - mean(x))^3) / (sd(x)^3)
}
```

```
x <- rnorm(500,10,2)
s_value <- s(x)
s_value
```

```
## [1] 0.05388285
```

¿Qué significa esto? ¿El generador de números aleatorios esta mal?

```
datos <- as.data.frame(x)
ggplot(datos, aes(x = x)) + geom_histogram(alpha = 0.5)
```




```
s_boot <- function(x) {
  y <- sample(x, size = length(x), replace = TRUE)
  s(y)
}
```

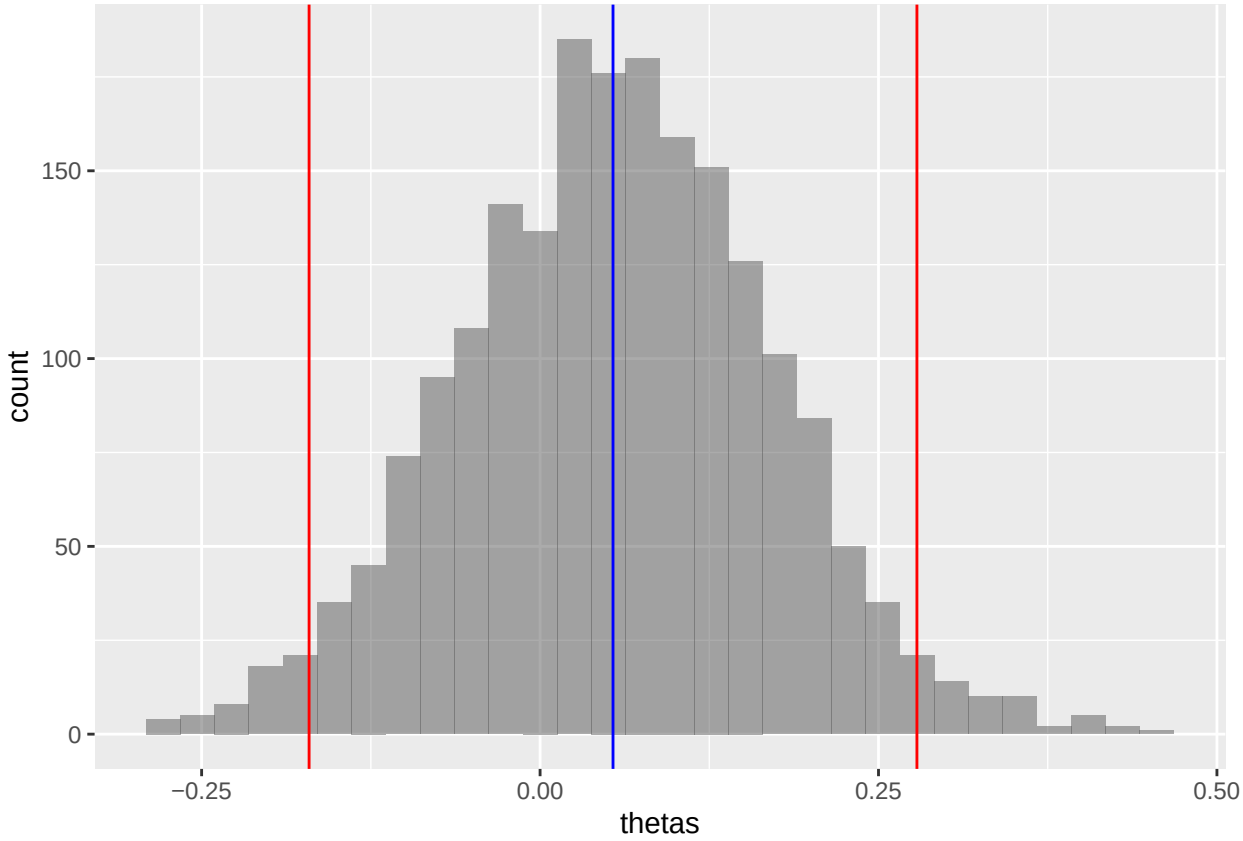
```
B <- 2000
thetas <- rerun(B,s_boot(x)) %>% flatten_dbl()
```

¿Cómo construimos los intervalos?

```
i_norm <- s(x) - 1.96*sd(thetas)
s_norm <- s(x) + 1.96*sd(thetas) #qnorm(0.975)
cat("(",i_norm,",",s_norm,")")
```

```
## ( -0.1706129 , 0.2783786 )
```

```
s_datos <-  as.data.frame(thetas)
ggplot(s_datos,aes(x=thetas)) + geom_histogram(alpha=0.5)+
  geom_vline(xintercept = s_value, col = 'blue') +
  geom_vline(xintercept = i_norm, col = 'red') +
  geom_vline(xintercept = s_norm, col = 'red')
```




Existen diferentes métodos para la creación de intervalos. Aquí mostraremos 3 de ellos

- Intervalo normal
- Inrtervalo t
- Intervalo Percentil

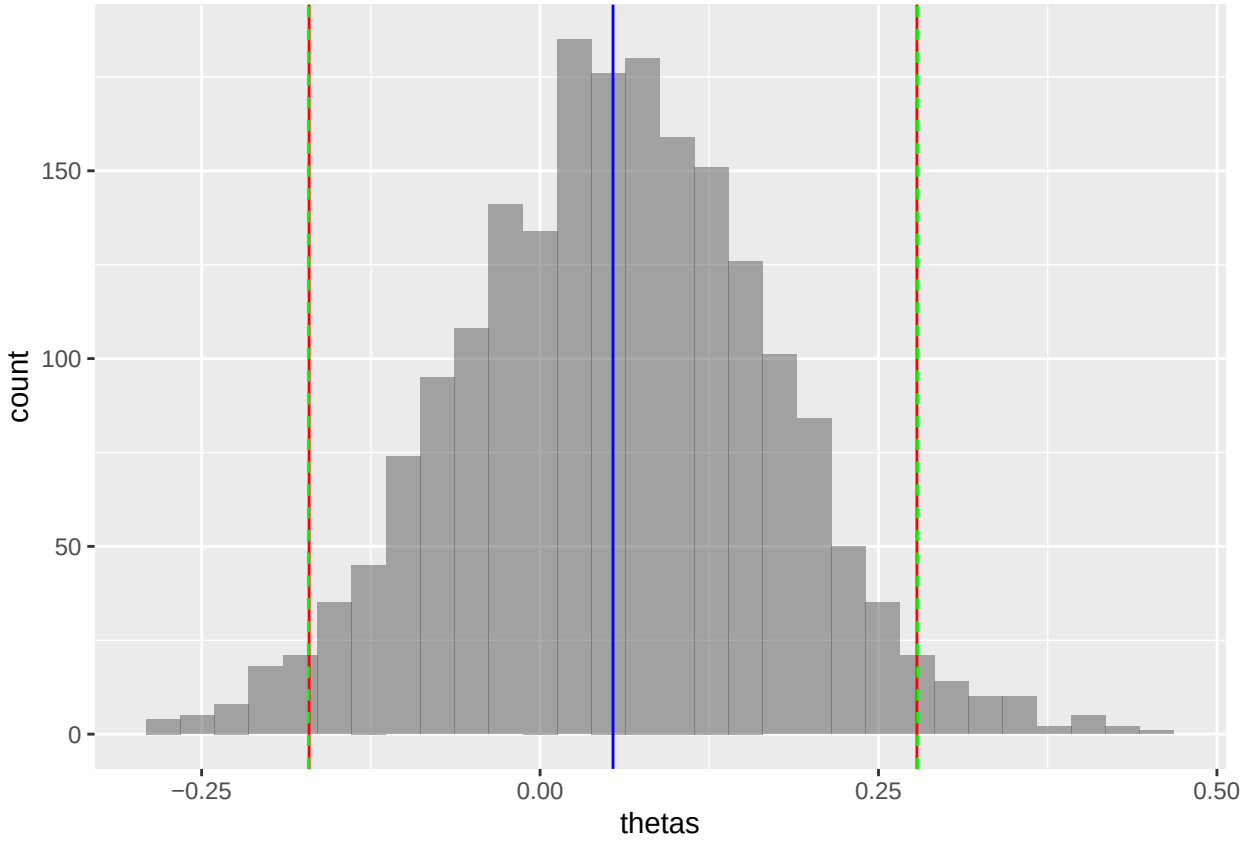
El primer intervalo que creamos corresponde al intervalo normal, es por que usamos el 1.96 el cual nos da el 95 de confianza. Como en teoría ya viste la costrucción de estos intervalos y su motivación, entonces podemo continuar y solo ver ejemplos.

Para el intervalo t solo utilizamos la distribución t

```
i_t <- s(x) - qt(0.975, length(x)-1)*sd(thetas)
s_t <- s(x) + qt(0.975, length(x)-1)*sd(thetas)
cat("(",i_t,",",s_t,")")
```

```
## ( -0.1711546 , 0.2789203 )
```

```
s_datos <-  as.data.frame(thetas)
ggplot(s_datos,aes(x=thetas)) + geom_histogram(alpha=0.5)+
  geom_vline(xintercept = s_value, col = 'blue') +
  geom_vline(xintercept = i_norm, col = 'red') +
  geom_vline(xintercept = s_norm, col = 'red') +
  geom_vline(xintercept = i_t, col = 'green', linetype = "dashed") +
  geom_vline(xintercept = s_t, col = 'green', linetype = "dashed")
```



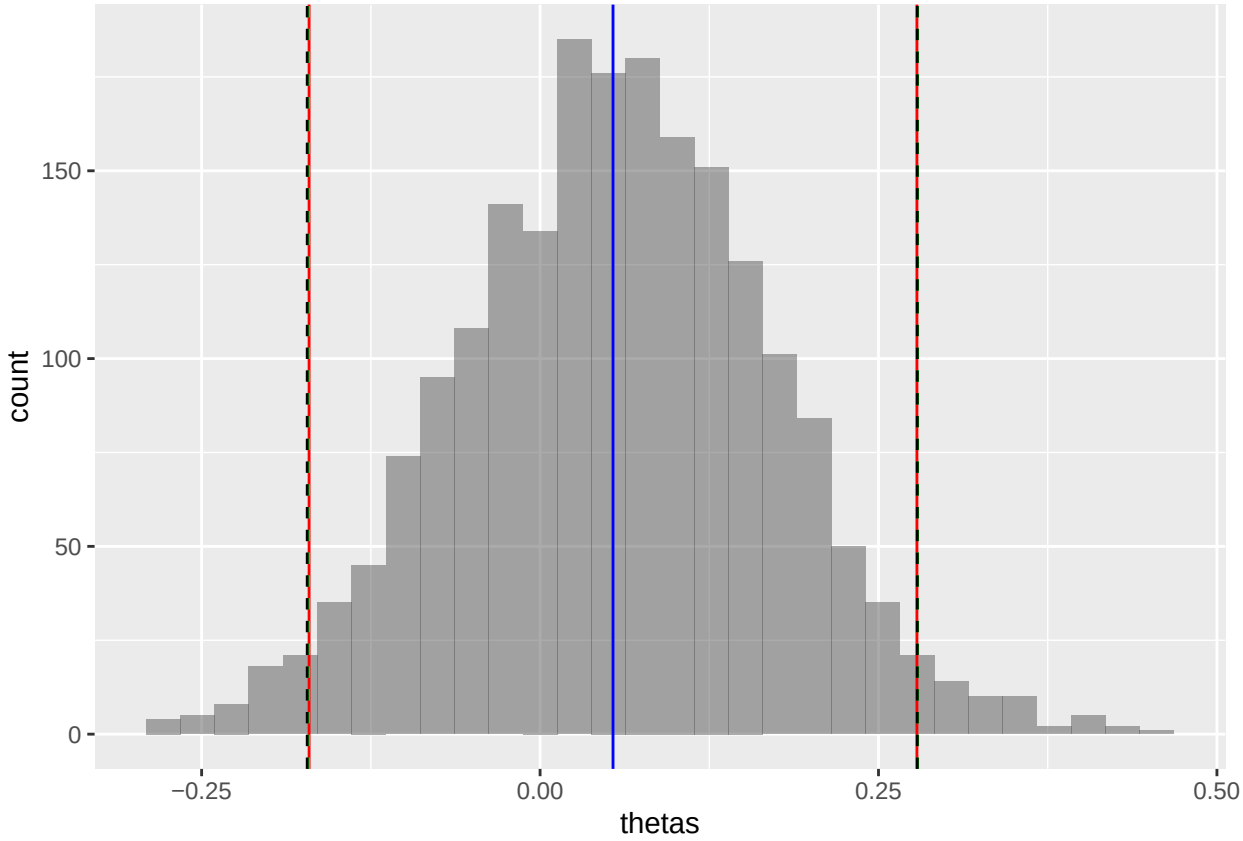

¿Qué puedes observar aquí?

Intervalos por percentil

```
s_per <- quantile(thetas, prob = 0.975)
i_per <- quantile(thetas, prob = 0.025)
cat("(",i_per,",",s_per,")")
```

```
## ( -0.1720097 , 0.2786523 )
```

```
s_datos <-  as.data.frame(thetas)
ggplot(s_datos,aes(x=thetas)) + geom_histogram(alpha=0.5)+
  geom_vline(xintercept = s_value, col = 'blue') +
  geom_vline(xintercept = i_norm, col = 'red') +
  geom_vline(xintercept = s_norm, col = 'red') +
  geom_vline(xintercept = i_t, col = 'green', linetype = "dashed") +
  geom_vline(xintercept = s_t, col = 'green', linetype = "dashed") +
  geom_vline(xintercept = i_per, col = 'black', linetype = "dashed") +
  geom_vline(xintercept = s_per, col = 'black', linetype = "dashed")
```

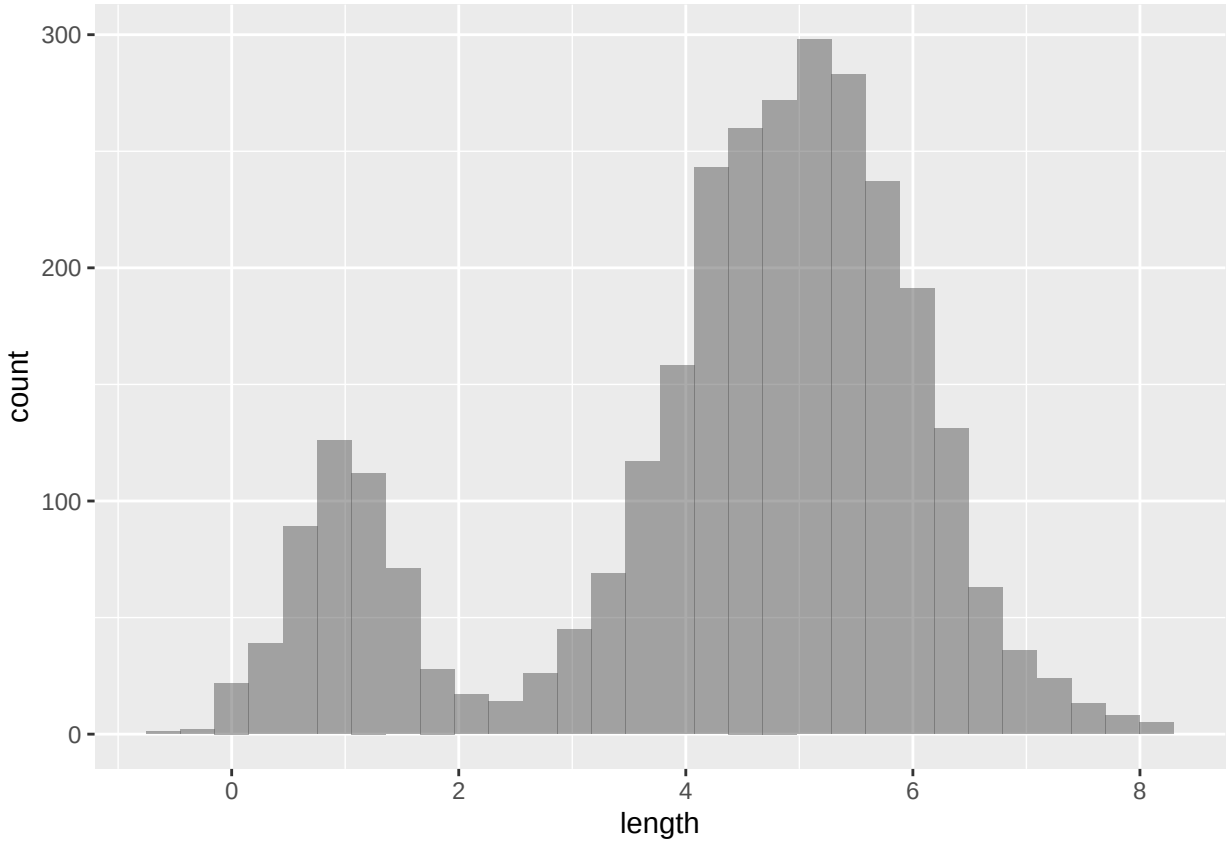



Veamos un ejemplo un poco más complejo.

```
set.seed(5865)
a <- data.frame(length = rnorm(500, 1, 0.5))
b <- data.frame(length = rnorm(2500, 5, 1))

a$pob <- 'a'
b$pob <- 'b'

datos <- as.data.frame(rbind(a, b))
ggplot(datos,aes(x = length)) + geom_histogram( alpha = 0.5)
```




calculamos la asimetría

```
s(datos$length)
```

```
## [1] -0.8517908
```

Hacemos el boostrap

```
B <- 2000
thetas <- rerun(B,s_boot(datos$length)) %>% flatten_dbl()
```

y ahora hacemos los intervalos de confianza

```
#Intervalos de confianza normales
i_norm <- s(datos$length) - 1.96*sd(thetas)
s_norm <- s(datos$length) + 1.96*sd(thetas)
#Intervalos de confianza t
i_t <- s(datos$length) - qt(0.975, length(datos$length)-1)*sd(thetas)
s_t <- s(datos$length) + qt(0.975, length(datos$length)-1)*sd(thetas)
#Intervalos percentil
s_per <- quantile(thetas, prob = 0.975)
i_per <- quantile(thetas, prob = 0.025)
cat("(",i_norm,",",s_norm,")\n")
```

```
## ( -0.9153148 , -0.7882669 )
```

```
cat("(",i_t,",",s_t,")\n")
```

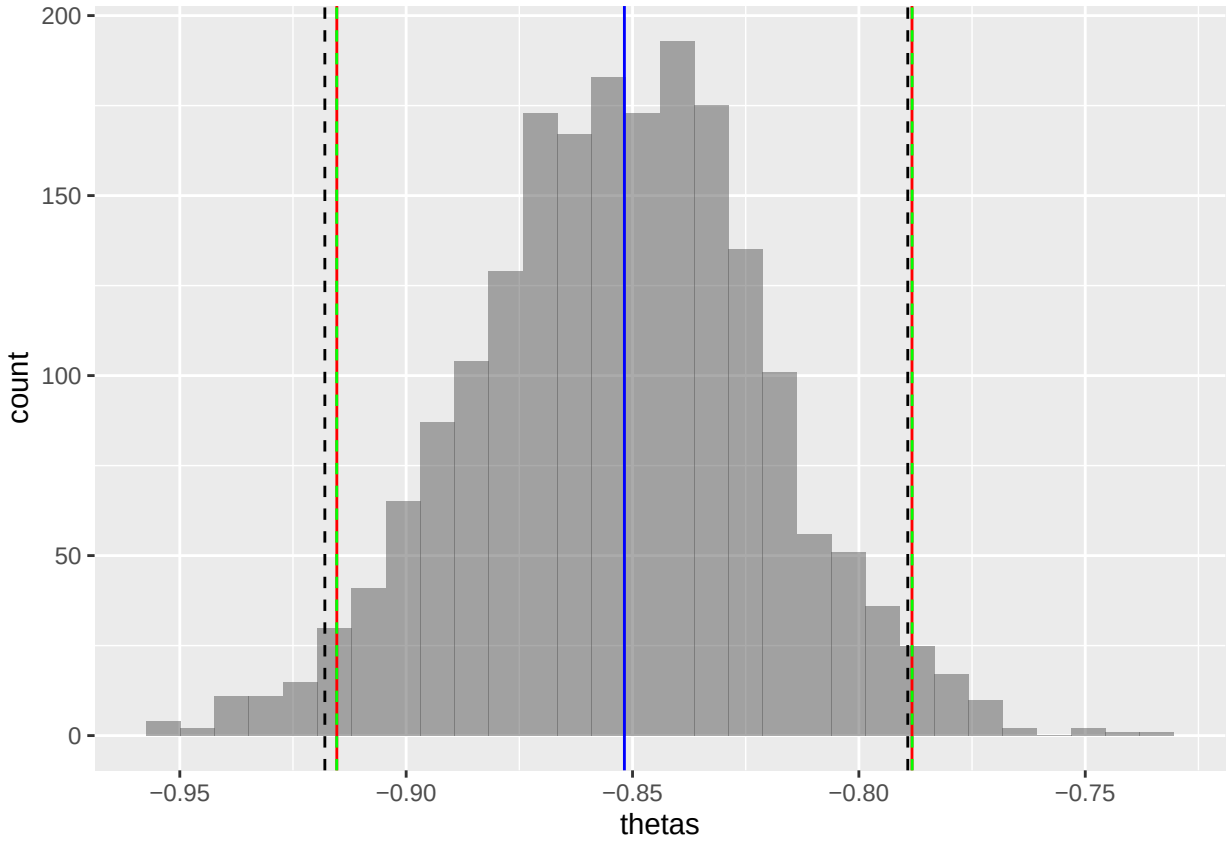
```
## ( -0.9153393 , -0.7882424 )
```

```r
cat("(",i_per,",",s_per,")")
```

```
## ( -0.9179572 , -0.7891956 )
```

Se puede observar que los intervalos normales y t son iguales, ¿Por qué?

```r
s_datos <-  as.data.frame(thetas)
ggplot(s_datos,aes(x=thetas)) + geom_histogram(alpha=0.5)+
  geom_vline(xintercept = s(datos$length), col = 'blue') +
  geom_vline(xintercept = i_norm, col = 'red') +
  geom_vline(xintercept = s_norm, col = 'red') +
  geom_vline(xintercept = i_t, col = 'green', linetype = "dashed") +
  geom_vline(xintercept = s_t, col = 'green', linetype = "dashed") +
  geom_vline(xintercept = i_per, col = 'black', linetype = "dashed") +
  geom_vline(xintercept = s_per, col = 'black', linetype = "dashed")
```




¿Qué interpretación le puedes dar?

En general podemos hacer esto para cualquier estadístico.