Guia TP Intervalos

Agustin Muñoz Gonzalez

6/7/2020

Preparamos el entorno.

```
rm(list=ls())
library(ggplot2)
library(tidyr)
library(gganimate)
```

1. Implemente una función intervalo.mu.asin que tenga por input un conjunto de datos x 1 , . . . , x n , provenientes de una muestra de X, el nivel $1 - \alpha$ y devuelva el intervalo de confianza asintótico $1 - \alpha$ para $\mu = E(X)$.

Resolución:

```
intervalo.mu.asin=function(datos,nivel){
  mu=mean(datos)
  alpha=1-nivel
  z=qnorm(1-alpha/2)
  se=sd(datos)/sqrt(length(datos))
  error=z*se
  c(mu-error,mu+error)
}
```

Nivel de Cubrimiento empírico: El nivel de cubrimiento empírico (de un procedimiento) se define como la proporción de veces que los intervalos (construídos con el procedimiento) utilizando datos simulados contiene a μ (o θ), en cierta cantidad de N rep replicaciones.

2. **Simulación 1: bajo normalidad** Genere variables con distribución normal de media $\mu = 0$ y $\sigma = 0.1, 1, 10$. Calcule el cubrimiento empírico del intervalo de confianza asintótico de nivel $1 - \alpha$, definido en intervalo.mu.asin, para $\alpha = 0, 05, n = 5, n = 10, n = 30, n = 50, n = 100, n = 1000$, utilizando $N_{rep} = 1000$ replicaciones, y complete la siguiente tabla. En cada caso, calcule el promedio de las longitudes en las N rep = 1000 replicaciones e inlcuya el valor en la tabla (long). Comente los resultados observados.

Indique a que valor debe aproximarse el nivel de cubrimiento empírico. Comente los resultados obtenidos.

Resolución:

```
NCE_norm=function(n,sigma){
  Nrep=1000
  alpha=0.05
  mu=0
  proporciones=c()
  for(i in 1:Nrep){
    datos=rnorm(n,mean=mu,sd=sigma)
    intervalo=intervalo.mu.asin(datos,1-alpha)
```

```
    proporciones=c(proporciones,
                     ifelse(intervalo[1]<= mu & mu<= intervalo[2],1,0))
  }
  salida=mean(proporciones)
  names(salida)=paste('NCE sd=',sigma)
  salida
}
#######################################
enes=c(5,10,30,50,100,1000)
sigmas=c(0.1,1,10)
tabla_item2=matrix(NA,ncol=length(enes))
for(i in sigmas){
  tabla_item2=rbind(tabla_item2,sapply(enes, NCE_norm,i))
}
colnames(tabla_item2)=paste('n=',enes)
tabla_item2=tabla_item2[2:(length(sigmas)+1),]
rownames(tabla_item2)=paste('NCE sd=',sigmas)
tabla_item2
```

```
##              n= 5 n= 10 n= 30 n= 50 n= 100 n= 1000
## NCE sd= 0.1 0.891 0.908 0.937 0.937  0.951   0.942
## NCE sd= 1   0.892 0.914 0.949 0.952  0.951   0.956
## NCE sd= 10  0.883 0.925 0.942 0.938  0.957   0.955
```

Notamos que a mayor cantidad de datos en la muestra (i.e. a mayor n) la probabilidad de caer en el intervalo de confianza aumenta.

Como el intervalo de confianza es de nivel asintótico y como vimos que por el TCL la proba de que el parámetro caiga en el intervalo asintótico tiene que tender a $1-\alpha$, entonces uno esperaría que el NCE se aproxime a $1-\alpha$.

En nuestro caso es $1-\alpha = 0.95$ y precisamente observamos que el NCE se aproxima a ese valor.

3. **Simulación 3: Uniformes** Genere variables con distribución uniforme en el intervalo $[0,\theta]$, para $\theta = 3, 10, 100$. Calcule el cubrimiento empírico del intervalo de confianza asintótico de nivel $1-\alpha$, definido en intervalo.mu.asin, para $\alpha = 0,05$, $n = 5, 10, 30, 50, 100, 1000$, utilizando Nrep $= 1000$ replicaciones. En cada caso, calcule el promedio de las longitudes en las Nrep $= 1000$ replicaciones e informe los valores obtenidos. (long). Comente los resultados observados.

Resolución:

```
NCE_unif=function(n,theta){
  Nrep=1000
  alpha=0.05
  proporciones=c()
  for(i in 1:Nrep){
    datos=runif(n,0,theta)
    intervalo=intervalo.mu.asin(datos,1-alpha)
    proporciones=c(proporciones,
                     ifelse(intervalo[1]<= theta/2 & theta/2<= intervalo[2],1,0))
  }
  salida=mean(proporciones)
  names(salida)=paste('NCE sd=',theta)
  salida
}
################################################
enes=c(5,10,30,50,100,1000)
```

```
thetas=c(3,10,100)
tabla=matrix(NA,ncol=length(enes))
for(i in sigmas){
  tabla=rbind(tabla,sapply(enes, NCE_unif,i))
}
colnames(tabla)=paste('n=',enes)
tabla=tabla[2:(length(thetas)+1),]
rownames(tabla)=paste('NCE theta=',thetas)
tabla
```

```
##                  n= 5 n= 10 n= 30 n= 50 n= 100 n= 1000
## NCE theta= 3    0.869 0.924 0.948 0.944  0.941   0.939
## NCE theta= 10   0.882 0.910 0.932 0.930  0.946   0.959
## NCE theta= 100  0.877 0.914 0.947 0.949  0.951   0.949
```

Podemos ver que el NCE se aproxima a 0 para n cada vez mayores. Esto se debe a que theta no cae en el intervalo de confianza asintotico generado por intervalo.mu.asin, y esto es porque la justificación del procedimiento de intervalo.mu.asin viene de que asumimos que los datos tienen distribución normal. No es el caso de los datos de este ejercicio. OJO ESTO QUE DIJE ERA PORQUE NO HABIA DIVIDIDO A THETA POR 2!

NOTAR QUE INTERVALO.MU.ASIN DA UN INTERVALO DEL PARAMETRO THETA/2 (QUE ES LA ESPERANZA)

ASI QUE TENGO $A < \theta/2 < B$ ASI QUE TENGO QUE PREGUNTAR SI THETA/2 CAE EN EL INTERVALO! ARRELARLO! LISTO

4. Implemente una función intervalo.mu.exacto.normal que tenga por input un conjunto de datos x 1 , . . . , x n , provenientes de una muestra $\mathcal{N}(\mu, \sigma^2)$, el nivel $1-\alpha$ y devuelva el intervalo de confianza exacto de nivel $1-\alpha$ para μ.

Resolución:

Sabemos que $a(X_1,\dots,X_n) = \overline{\mu}_n - z_{\alpha/2} * \sqrt{\frac{\sigma_0^2}{n}}$ y $b(X_1,\dots,X_n) = \overline{\mu}_n + z_{\alpha/2} * \sqrt{\frac{\sigma_0^2}{n}}$ son los bordes del intervalo de confianza $1-\alpha$ cuando $X_i \sim \mathcal{N}(0,1)$ y conocemos σ_0.

Si en vez de trabajar con σ_0 trabajar con su estimador S pagamos el precio de cambiar la $Z \sim \mathcal{N}(0,1)$ por la t de student: $a(X_1,\dots,X_n) = \overline{\mu}_n - t_{n-1,\alpha/2} * \sqrt{\frac{\sigma_0^2}{n}}$ y $b(X_1,\dots,X_n) = \overline{\mu}_n + t_{n-1,\alpha/2} * \sqrt{\frac{\sigma_0^2}{n}}$ con $t_{n-1,\alpha/2} = qt(1-\alpha/2, n-1)$.

Como no conocemos σ usamos la t de student.

```
intervalo.mu.exacto.normal=function(datos,nivel){
  mu=mean(datos)
  alpha=1-nivel
  n=length(datos)
  t=qt(1-alpha/2,n-1)
  error=t*sd(datos)/sqrt(length(datos))
  c(mu-error,mu+error)
}
```

5. Repita el item 2. utilizando ahora la función intervalo.mu.exacto.normal y compare los resultados obtenidos (los de ahora con los del item 2.) Comente los resultados observados.

Resolución:

```
NCE_norm2=function(n,sigma){
  Nrep=1000
  alpha=0.05
```

```r
  mu=0
  proporciones=c()
  for(i in 1:Nrep){
    datos=rnorm(n,mean=mu,sd=sigma)
    intervalo=intervalo.mu.exacto.normal(datos,1-alpha)
    proporciones=c(proporciones,
                    ifelse(intervalo[1]<= mu & mu<= intervalo[2],1,0))
  }
  salida=mean(proporciones)
  names(salida)=paste('NCE sd=',sigma)
  salida
}
#######################################
enes=c(5,10,30,50,100,1000)
sigmas=c(0.1,1,10)
tabla_item5=matrix(NA,ncol=length(enes))
for(i in sigmas){
  tabla_item5=rbind(tabla_item5,sapply(enes, NCE_norm2,i))
}
colnames(tabla_item5)=paste('n=',enes)
tabla_item5=tabla_item5[2:(length(sigmas)+1),]
rownames(tabla_item5)=paste('NCE sd=',sigmas)
tabla_item5
```

```
##              n= 5 n= 10 n= 30 n= 50 n= 100 n= 1000
## NCE sd= 0.1 0.956 0.957 0.961 0.949  0.942   0.949
## NCE sd= 1   0.951 0.952 0.963 0.950  0.944   0.946
## NCE sd= 10  0.947 0.953 0.943 0.951  0.959   0.951
```

Comparemos ambas tablas.

```r
tabla_item2
```

```
##              n= 5 n= 10 n= 30 n= 50 n= 100 n= 1000
## NCE sd= 0.1 0.891 0.908 0.937 0.937  0.951   0.942
## NCE sd= 1   0.892 0.914 0.949 0.952  0.951   0.956
## NCE sd= 10  0.883 0.925 0.942 0.938  0.957   0.955
```

```r
tabla_item5
```

```
##              n= 5 n= 10 n= 30 n= 50 n= 100 n= 1000
## NCE sd= 0.1 0.956 0.957 0.961 0.949  0.942   0.949
## NCE sd= 1   0.951 0.952 0.963 0.950  0.944   0.946
## NCE sd= 10  0.947 0.953 0.943 0.951  0.959   0.951
```

EN REALIDAD LA COMPARACION ESTA MAL HECHA PORQUE GENERÉ NUEVOS DATOS EN CADA TABLA. EN TODO CASO SI QUIERO COMPARAR TENGO QUE USAR LA MISMA MUESTRA!

Se ve que los NCE correspondientes a los intervalos de confianza de la función intervalo.mu.exacto.normal son medio indep del tamaño de n, en cambio el asintotico te da mal para valores chicos de n.

6. **Diferencias de medias:** Supongamos que tenemos dos conjuntos de observaciones 2 provenientes de dos distribuciones $\mathcal{N}(\mu_X, \sigma_X)$ y $\mathcal{N}(\mu_Y, \sigma_Y)$, asumamos que $\sigma_X = \sigma_Y$.

El objetivo es decidir a partir de las observaciones si las medias son o no iguales.

a) Implemente una función dif.mu.interseccion que tenga por input el nivel $1 - \alpha$, un conjunto de datos x 1 , . . . , x n y un conjunto de datos y 1 , . . . , y m y devuelva un 1 si los intervalos exactos de nivel

$1 - \alpha$ para μ_X y para μ_Y se intersecan y un 0 si no se intersecan. Es decir, si los intervalos obtenidos en el item 4 para cada conjunto de datos se intersecan o no.

Resolución:

Usar

se_intersecan=1 if(intervalo_x[2]<intervalo_y[1] || intervalo_y[2]<intervalo_x[1]){ #NO SE INTERSECAN se_intersecan <- 0 } return(se_intersecan)

b) Implemente una función dif.mu.intervalo que tenga por input el nivel $1 - \alpha$, un conjunto de datos x 1 , . . . , x n y un conjunto de datos y 1 , . . . , y m y devuelva un intervalo exacto de nivel $1 - \alpha$ para $\mu_X - \mu_Y$ y un 1 si el 0 esta en el intervalo y un 0 si no.

c) Genere variables X con distribución normal de media $\mu_X = 1$ y $\sigma_X = 2$ y variables Y $\mu_Y = 2$ y $\sigma_Y = 2$. Calcule la proporción de veces que el método de intersección decide que ambas medias son iguales y calcule el cubrimiento empírico del intervalo de confianza exacto de nivel $1 - \alpha$ para la diferencia de medias (es decir la proporción de veces que el intervalo incluye al 0). Considere $\alpha = 0,05$, n = 5, n = 10, n = 30, n = 50, n = 100, n = 1000, utilizando N rep = 1000 replicaciones. Comente los resultados observados.

7. El objetivo de este ejercicio es comparar empíricamente la performance de diferentes intervalos. Por un lado, utilizaremos intervalos construídos estimando la varianza asintótica mediante (i) boostrap y (ii) propagación de errores; luego consideraremos (iii) intervalos bootstrap por percentil. Finalmente, contruiremos (iv) intervalos para una proporción, siendo que el parámetro de interés es una probabilidad. Estudiaremos el nivel de cubrimiento empírico y la longitud media de cada uno procedimientos mencionados. Generaremos datos asumiendo que provienen de una familia exponencial $X \sim \mathcal{E}(\lambda)$. Procuramos estimar $\theta = P(X > 1)$.

a) Obtenga una fórmla para el parámetro de interés en función de λ.

b) Genere datos utilizando $\lambda = 3$. Considere n = 100, 200, 500 y estudie el nivel de cubrmiento empírico en N rep = 1000 replicaciones de los intervalos propuestos, tomando N boot = 1000 muestras bootstrap.

usar lo uqe sigue + FOTOS

estim.se.f.bootstrap <- function(X){ Nboot <- 1000 fs <- rep(NA, Nboot) for(i in 1:Nboot){ X <- sample(x=X, size=Nboot, replace=T) fs[i] <- f(mean(X)) } #hist(fs) se <- sd(fs) return(se) } f <- function(phi){ return(exp(-1/phi)) }

inter <- intervalo.f.delta(X, 0.95, se) intervalo.f.delta <- function(X, nivel, se){ alpha <- 1 - nivel phi <- mean(X) f_phi <- f(phi) df_phi <- df(phi) z <- qnorm(p=1-alpha/2) a <- f_phi - z * abs(df_phi * se) b <- f_phi + z * abs(df_phi * se) return(c(a,b)) }

Bootstrap normal: usa el promedio para estimar la proba

a eso se refiere bootstrap normal. a usar que el estimador es asintoticamente normal y estimas su varianza/desvio

bootstrap percentil: usa quantiles para el estimador

bootstrap parametrico: hace plug in para el estimador, o sea implica que conozccas la distribucion.

i.e. la diferencia es el estimador, despues bootstrap es samplear indices en 1:n con reposicion y quedarte con datos[indices]. O sea es generar datos, agarrar samples aleatorias de tus datos.

dijo mariela 'en lugar de inventar datos con la empiricia, inventas con una exponencial!!!! y pones el parametro estimado del modelo. vamos a la modista!!!!! cambiamos la manera de inventar datos'

dijo ana 'el cambio es solo en eso, en como generas las muestras bootstrap usas la muestra original para la estimacion del aprametro y generar las muestras boot' otra forma es estimar el parametro con un metodo parametrico pero el intervalo de confianza hacerlo no parametrico (estimacion parametrica + bootstrap no parametrico) OJO QUE EL MODELO TIENE QUE SER EL CORRECTO. A UQE SE REFIEREN CON MODELO? A QUE ASUMIS QUE EL FENOMENO QUE ESTUDIAS SIGUE TAL DISTRIBUCION, O SEA CON MODELO SE REFIEREN A MODELO PARAMETRICO Y TIENE QUE VER CON QUE ESTA ASOCIADO A UNA DISTRIBUCION (QUE TIENE PARAMETROS, POR ESO PARAMETRICO)

o sea podes ir mezclando (siempre que tenga sentido)

NOTA: En la vida como que se usa mas boostrap parametrico que propagacion de errores